IBM Maintenance Library

3725/3726

Communication Controller and Expansion
Maintenance Information Manual (MIM) Part 1
Volume 2

SY33-2018-7

Eight Edition (June 1986)

This major revision obsoletes SY33-2018-6 and Technical Newsletter SN33-7126. Changes or additions to the text and illustrations are indicated by a vertical line to the left of the change. This edition reflects the 3725 Release 4 enhancements.

Changes are made periodically to the information herein. Any such changes will be reported in subsequent revisions or Technical Newsletters.

IBM has prepared this maintenance manual for the use of IBM customer engineers in the installation, maintenance, or repair of the specific machines indicated. IBM makes no representations that it is suitable for any other purpose.

Reference in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM program product in this publication is not intended to state or imply that only IBM's program product may be used. Any functionally equivalent program may be used instead.

Publications are not stocked at the addresses given below. Requests for IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form for readers' comments is provided at the back of this publication. If the form has been removed, comments may be addressed to either of the following:

- International Business Machines Corporation
  Department 6R1LG
  180 Kost Road
  Mechanicsburg
  PA 17055

  or

- IBM France
  Centre d'Etudes et Recherches
  Department 0762
  06610 La Gaude
  France

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Preface

This publication is intended for the customer engineer who maintains the IBM 3725 Models 1 and 2 Communication Controller and the IBM 3726 Communication Controller Expansion.

For the 3725 Model 1 and 3726, this manual should be used in conjunction with the 3725/3726 Maintenance Information Manual (MIM) Part 2 for locating and replacing failing field replaceable units within the communication controller and expansion.

For the 3725 Model 2, this manual should be used in conjunction with the 3725 Model 2 Maintenance Information Manual (MIM) Part 2.

Organization

The manual is divided into two volumes:

- Volume 1 contains introductory and how to fix information.
- Volume 2 contains detailed descriptions of 3725/3726 functional units as well as extended troubleshooting procedures for each unit.

Troubleshooting notes are added, where appropriate. These enable you to continue troubleshooting when card and FRU replacements have not removed the trouble, and to determine if a board, cable, or top connector is the failing part.

Prerequisite Publications

The reader should have an understanding of telecommunications and modems. The following manuals provide a training on the 3725/3726:

- 3725 Model 1 Communication Controller, Introduction, GA33-0010
- 3725 Model 2 Communication Controller, Introduction, GA33-0021

Corequisite Publications

The following manuals provide the procedures for operating the communication controller:

- 3725/3726 Communication Controller and Expansion, Diagnostic Descriptions, SY33-2027
- 3725 Communication Controller, Operator's Guide, GA33-0044
- 3725 Communication Controller, Problem Determination and Extended Services, GA33-0014

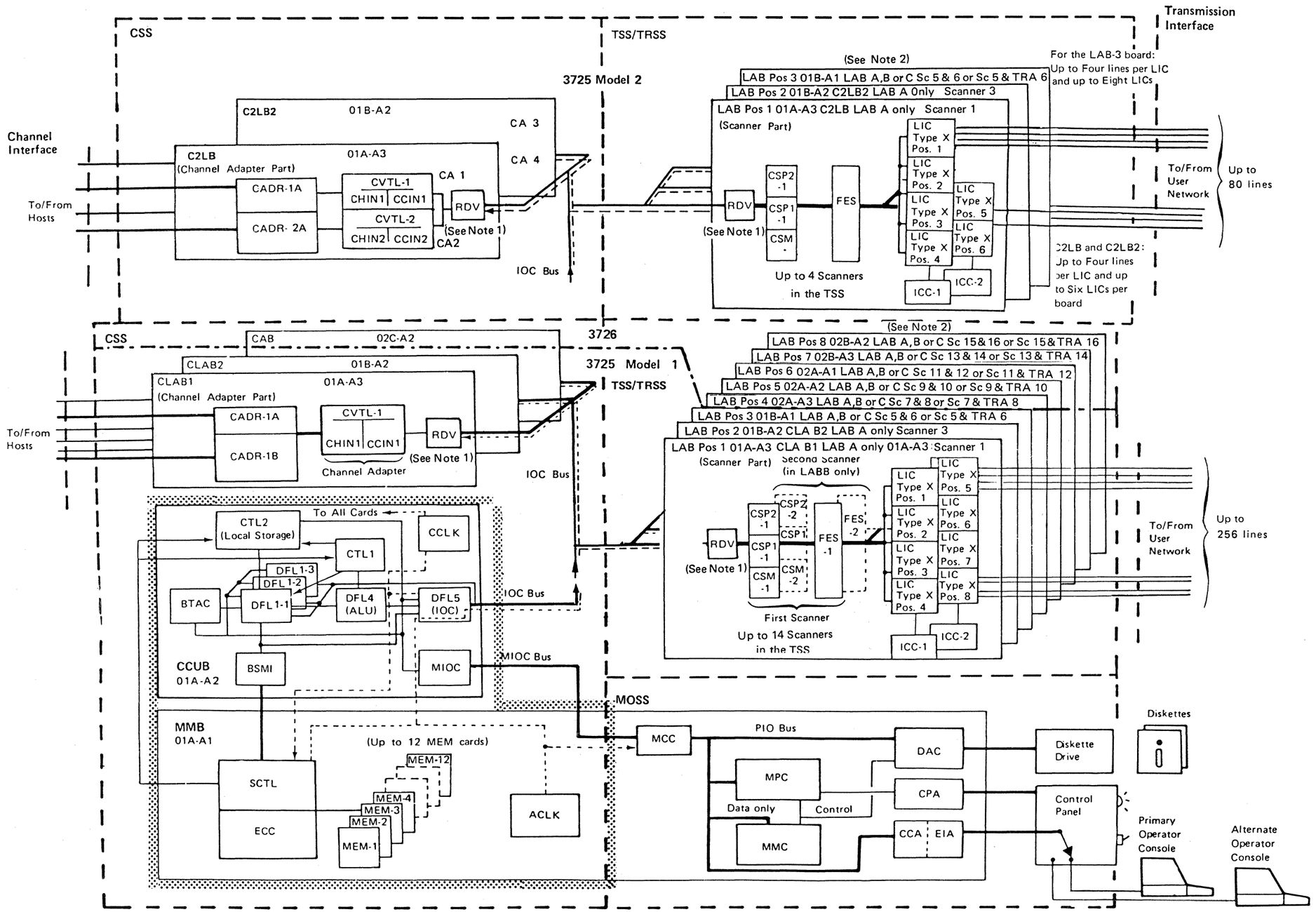
Contents

Chapter 10. Central Control Unit

Section 1. Unit Description

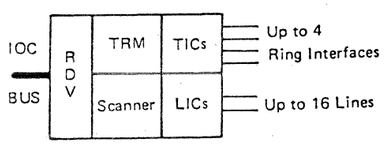
Contents

CCU in 3725 Data Flow



Note 1: As the CLAB and the CL2B boards are split into two parts in this figure, one for the channel adapter and one for the scanner, the redrive (RDV) function is shown twice for clarity. However, there is one RDV per CLAB or C2LB board.

Note 2: LAB Pos. 3 to 8 organization if LAB type C installed



Legend:

------ Clock signals

——— Data/control signals

CCU Data Flow

The CCU:

- Executes the machine instruction set to perform arithmetic or logical operations, exchange data between main storage and the work registers, and also between the local store and the work registers.
- Communicates with adapters through the IOC logic in PIO or AIO mode:

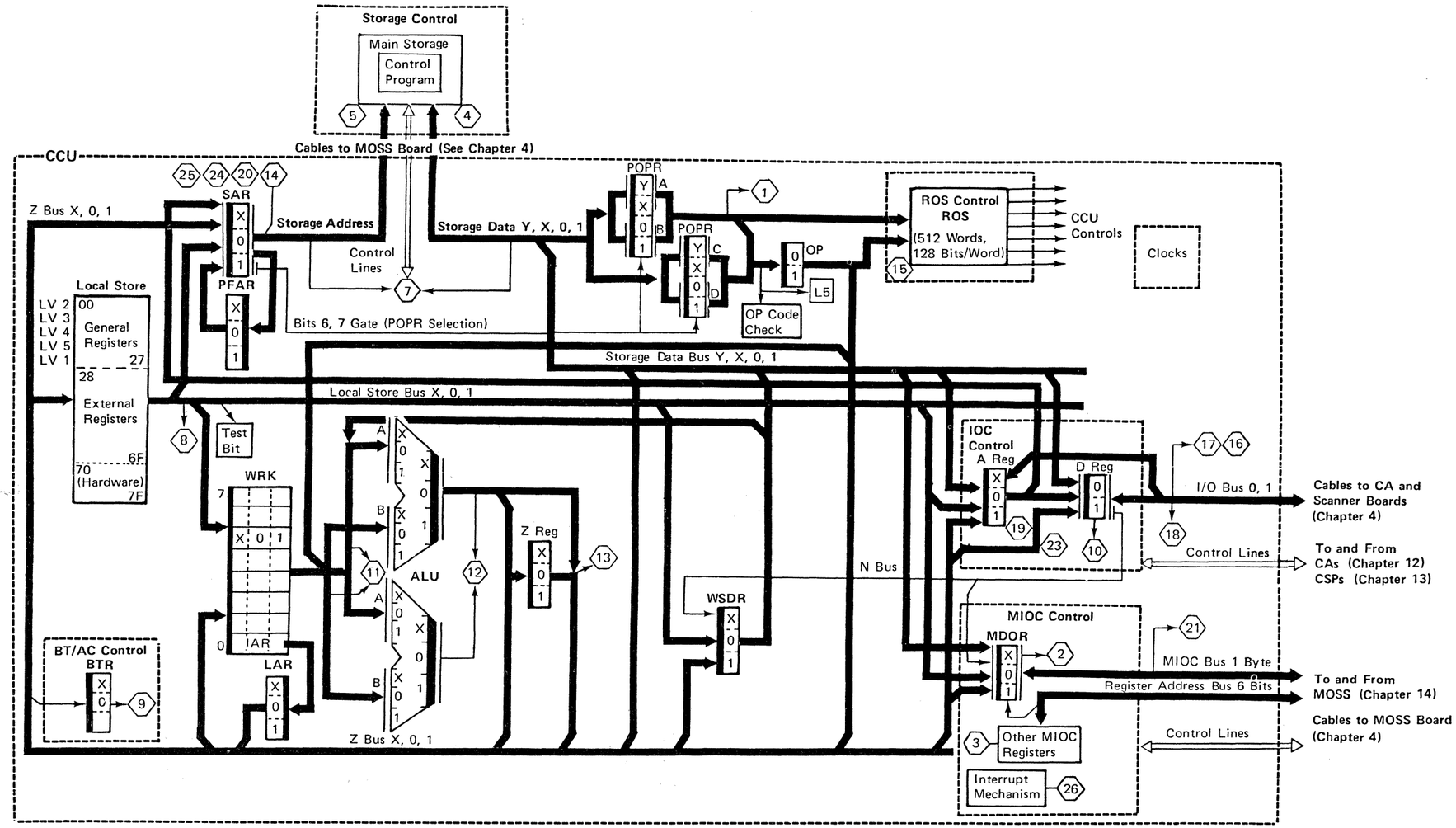
  PIO mode: The exchange operation is initiated by IOH/IOHI instructions in the CCU, or MIOH/MIOHI in the MOSS.

  AIO mode: The operation uses cycle steals for data exchange.
- Communicates with the MOSS through the MOSS logic. The operations performed can be direct, indirect, or use the CCU level scan sensitive design (LSSD) that enables MOSS to read or write any CCU discrete latch.

ERROR CHECKING CIRCUITS

An error in the 3725 sets one or more of the following conditions (identified with a hexagon on the data flow):

1	POPR parity check
2	MDOR parity error
3	MIOC parity error
4	Storage double-bit error
5	Storage control error
7	Storage address/data parity error
8	Local store parity error
9	Control error
10	D1 reg parity error
11	A/B bus parity error
12	ALU compare error
13	Z reg parity error
14	SAR parity error
15	ROS parity error
16	D2 reg parity error
17	IOC1 timeout
18	IOC1 bus in parity error
19	MOSS IOC OP error
20	Address exception error
21	MOSS OP check
23	IOC1 invalid CSCW
24	IOC1 storage protect
25	IOC1 address exception
26	Level 1 error re-entry



Legend:

⬡ Error Condition

CCU Circuit Description (Part 1 of 2)

In the following description, '*' indicates the registers that can be displayed on the 3727 console.

ARITHMETIC AND LOGIC UNIT (ALU)

There are two identical ALUs, working in parallel. Their results are constantly compared; if a discrepancy occurs, an ALU compare check is set.

The ALU operations performed are:

- Pass A bus data
- Pass B bus data
- A AND B
- A OR B
- A XOR B
- Zero
- Shift right
- Addition
- Subtraction

FUNCTION REGISTERS

These registers are located on the MIOC card.

Four function registers are used by MOSS to force the CCU in diagnostic mode or control the BT/AC function.

They are:

- Mode control register A
- Diagnostic mode control register
- Address compare control register
- BT level control register

For register bit descriptions, see page 14-051.

Seven function registers are used to request interrupts to the MOSS or the CCU, or to provide carry and zero indicator status. They are:

- MOSS to CCU status register
- CCU to MOSS status A - through F registers

For register bit descriptions, see page 14-040.

Four function registers are used by MOSS to control the LSSD circuits in the CCU. They are:

- String select register
- MOSS scan register
- Step register
- Clock step control register

For register bit descriptions, see page 14-071.

Three function registers are used by MOSS to control the indirect operations. They are:

- CCU ROS address register
- Maintenance data operand register
- CCU local store address register

For register operation, see page 14-060.

INSTRUCTION ADDRESS REGISTER (IAR *)

The IAR is work register 0. It stores the next sequential instruction address. At each instruction decode time, the IAR contents are loaded into the LAR.

IOC REGISTERS (A AND D)

The A reg is loaded with the contents of the R1 field of the IOH/IOHI instructions. In AIO operation it contains the cycle steal address.

The D reg is used to exchange data with the adapters.

LAGGING ADDRESS REGISTER (LAR *)

The LAR is a 'came from' register. It is loaded from the IAR at the beginning of each instruction (see page 10-230 for details).

MAINTENANCE DATA OPERAND REGISTER (MDOR)

The MDOR is a byte register used to exchange data between the CCU and the MOSS.

OPERATION REGISTER (OP REG *)

The OP register holds the first 2 bytes of the instruction being executed.

PRE-FETCH ADDRESS REGISTER (PFAR)

The PFAR holds the storage address of the instructions to be loaded into the POP registers. It is updated from the SAR each time a branch is taken.

PRE-FETCH OPERATION REGISTER (POPR)

Instructions are pre-fetched to increase the CCU performance.

Two to four instructions are loaded in sequence into POPRA through POPRD. The POPs are then transferred one by one to the operation register (OP reg) for decoding. If a branch is taken, the contents of the POPRs are emptied and POPR loading restarts from the new SAR address (branch address).

CCU READ-ONLY STORAGE (ROS)

The ROS consists of 512 words of 128 bits each. Decoding of the OP register causes the selection of a routine in the read only storage (ROS) to set hardware latches, to gate data information, and to select a data flow path for the execution of the instruction.

STORAGE ADDRESS REGISTER (SAR)

The SAR holds the current or last-used storage address. It is 3 bytes (X, 0, 1) long.

WORK REGISTERS (WKR *)

The eight work registers are all 22 bit registers. They are used with any interrupt level. The WKR contents update automatically the eight general registers of the working interrupt level. Thus, a branch to a higher interrupt level can be taken with no need to save the contents of the work registers.

WRITE STORAGE DATA REGISTER (WSDR)

The WSDR contains 24 data bits and three parity bits. It receives operation results from the ALU.

Z REGISTER

The Z register is a 22-bit register used to generate and check the parity of the Z bus.

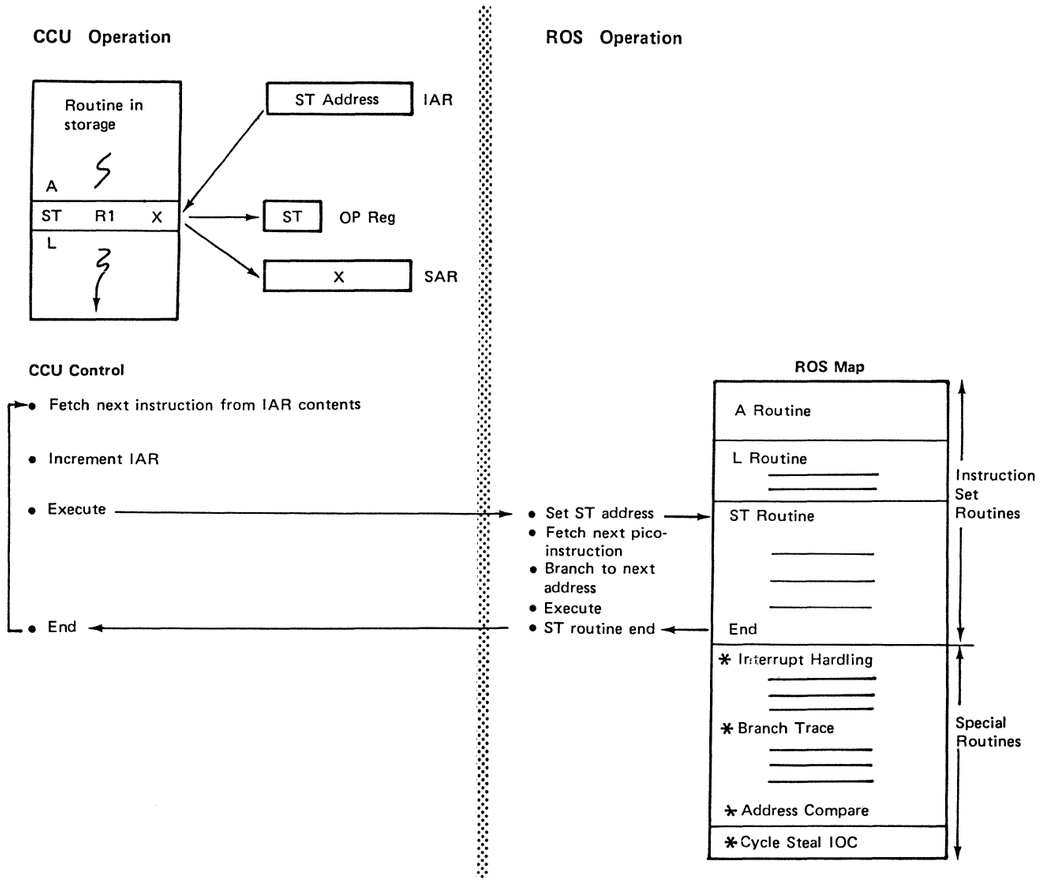
CCU Circuit Description (Part 2 of 2)

ROS OPERATION

The ROS operation within a CCU operation proceeds as follows:

Using its instruction set routines, the ROS performs the actual CCU instruction execution. Special routines control other operations such as interrupt handling and cycle stealing. Refer to the following figure for CCU/ROS operation.

CCU/ROS OPERATION



CCU Characteristics and Packaging

CONNECTORS

Connector	Cable	Signal Table Numbers
01A-A2A3	IOC	1 2 3
01A-A2B1	Storage	7 8 9 10 11
01A-A2V2	MIOC	12 13

For IOC and MIOC buses, see page 10-020.

TIMING

The CCU cycle is 200 ns.

The storage cycle is 400 ns for read operations (800 ns if a refresh cycle occurs) and 600 ns for write operations (1000 ns if a refresh cycle occurs).

The CCU is not synchronous with the storage, the adapters, or the MOSS.

INTERRUPTS

Level 1 All errors
Level 2 Scanner control
Level 3 CA control
Level 4 NCP/EP supervisor
Level 5 Message processing

CCU BOARD PACKAGING

Cable wires are described under "Signal Tables" on page 4-080 onward. Numbers in the following table correspond to circled numbers in the signal tables.

Cable	Signal Table Numbers
IOC	1 2 3
Storage	7 8 9 10 11
MIOC	12 13

(See page 4-060 for card location.)

CYCLE STEALS

High-priority cycle steal is used for scanners.

Low-priority cycle steal is used for CAs.

INSTRUCTION SET

The instruction set (listed on page 10-100) is the same as for the 3704/3705 except for two additional I/O instructions (IOH and IOHI).

CCU MAIN STORAGE

See details on page 10-050.

LOGIC PARTITIONING

Most of the cards perform complete CCU functions except for the DFL1, 2, and 3 cards, which share the CCU register bits. The CCU functions performed by each card are indicated in the following table:

Cards	Functions
DFL1-1 DFL1-2 DFL1-3 (See Note)	IAR, WKRs 1 to 7, SAR, MDOR, IOC address register, IOC data register, IOC D register parity check, OP register, POP register, POP control WSDR, error circuits
DFL4 (ALU)	ALU, ALU compare, IAR increment, SAR increment PFAR, LAR, Z register
DFL5 (IOC)	IOC registers, tag handling pointer address generation IOC error functions, MIOH flag
CTL1	Priority mechanism, MOSS OP control, IOC control, IPF control, CCU user indicator, Input instruction decode, Z bus and A/B bus source control, OP register latches, ROS address generation, ROS POP parity, B bus parity, Control of STG, WSDR, SAR, IAR, control STG to bus
CTL2	Local store, LS control, LS test, LS parity, SP/AE, SP/AE user keys, SP/AE RAM, output instruction decode, C/Z mechanism errors (for example, overflow, double-bit).
MIOC	Registers, direct read/write operations, LSSD mechanism, Selection, Read operation, LSAR, ROSAR, MIOC register, MIOC error detection, interrupt mechanism
BTAC	BTAC mechanism, high resolution timer, remote power off
CCLK (Clock)	CCU clocks: A, B for LSSD shifts, B, C operations Storage clock
BSMI	Buffers and drivers to MMB, CCU logic/storage logic converters

Note: DFL1-1, DFL1-2, and DFL1-3 have the same part number.

DFL1-1 (01A-A2M2)	DFL1-2 (01A-A2L2)	DFL1-3 (01A-A2K2)
Bits 5, 7, P bytes X, 0, 1	Bits 3, 4, 6 bytes X, 0, 1	Bits 0, 1, 2 bytes X, 0,1

Main Storage Description

DATA FLOW

Although the storage is packaged on the MMB, it is logically part of the CCU data flow.

The storage is divided into three parts:

1. Memory (MEM) Cards:

   Each card stores 64K words (256K bytes). On some early machines, depending on the EC level, MEM cards 1 through 8 may store 32K words (128K bytes) only. If this is the case, do not mix the 128K byte cards and 256K byte cards in the first eight MEM positions.

   Each word consists of 32 data bits and eight ECC bits.

   The maximum storage capacity requires 12 MEM cards for the 3725 Model 1, and four MEM cards for the 3725 Model 2.

2. Error Checking and Correction (ECC) Card:

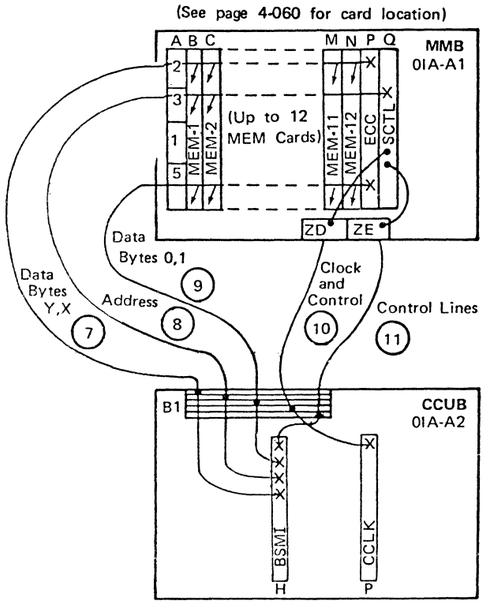
   The ECC card detects and corrects all single-bit errors, and some double-bit errors.

3. Storage Control (SCTL) Card:

   The SCTL card controls storage operations.

STORAGE BUSES AND PACKAGING

(See page 4-060 for card location)



STORAGE ADDRESSING

The SAR is used to address the storage. It is 25 bits wide (22 address bits, three parity bits). A storage position is always four bytes long.

The storage is structured on a word basis.

SAR

```
X.2
  3  --
  4  -- X.3 to X.6 are used to
  5  -- select MEM cards (4 bits)
  6  --
  7  -- see 1.0
---
0.0  -- 0.0 to 0.6 are used to
  1  -- select storage positions
  2  -- on MEM card to be
  3  -- refreshed
  4  --
  5  --
  6  --
  7

1.0  -- X.7 and 1.0 to 1.5 are
  1  -- used to select storage
  2  -- positions on the MEM
  3  -- card for data read and
  4  -- write
  5  --
  6  -- 1.6 and 1.7 select byte(s)
  7  -- in a storage word, as follows:
```

	0	7 0	7 0	7 0	7
Byte	Y	X	0	1	
1.6, 1.7	00	01	10	11	

ECC MECHANISM

The ECC card generates eight check bits, which are appended to the 32 data bits of a word to be written in storage. During read operations, the ECC card detects and corrects all single-bit errors and some double-bit errors.

Note: Cable wires are described on page 4-080 onward. Circled numbers relate cables to signal tables.

STORAGE OPERATIONS

Storage Word

0	7 0	7 0	7 0	7 0	7
Y	X	0	1	ECC	

|<-------Data Bytes---------->|<-Byte->|

The storage operations are read, write, ECC control, refresh, and read storage size.

Read: A 40-bit word is taken from a storage location. The ECC bits are checked and replaced with parity bits for CCU data flow operation.

Write: A 40-bit word is stored in a storage location.

A CCU instruction may need only one data byte changed out of the four data bytes of the storage word. For this reason, a write operation consists in:

1. Reading the storage word
2. Modifying the number of data bytes required
3. Writing back the new word

The ECC bits are computed for the new word.

ECC Control: (Output X'74') The SCTL can disable the ECC mechanism for initialization or diagnostic purposes:

ECC Disable Control (X'74' bit 2)
ECC Transparent Control (X'74' bit 3)
ECC Write Control (X'74' bit 4)

Activating the storage control output tag causes the SCTL to sample the three data bus bits as defined above.

Refresh: Reading a storage location does not change the contents of that location. However, the storage consists of dynamic array chips on which the cells must be refreshed every 2 ms to keep the data valid. 256 storage cycles (128 on some early machines) are required to refresh all storage positions.

The negative pulses of the refresh clock (from ACLK) start the refresh cycles on all MEM cards.

Read Storage Size: (Input X'70') When this operation is requested by the CCU, the SCTL places on the data bus (byte 0) the storage size actually implemented, for as long as the input X'70' line is active.

Double-Bit Recovery Cycle: When a double-bit error is detected by the ECC card, the contents of the ECC latches are frozen and, at completion of the current cycle, the storage enters a recovery cycle consisting of the following steps:

1. Invert write
2. Invert read
3. Write back
4. Read (after recovery)

The double-bit error is corrected provided one bit error is hardware, the other one software.

Storage Operation Timing

Note: See page 10-800 onward for scoping routines, timing diagrams, and details of read, write, output X'74', input X'70', and refresh operations.

READ

In one MEM cycle or two CCU cycles:

OBJECTIVE Get all necessary information from the CCU to address a 4-byte word.
OPERATION Set 'storage go' and 'address' lines
Fetch data. The data bus is disabled. The SCTL card reports to the CCU any address parity errors.
The 40-bit word is available on the data bus for correction by the ECC card. OPERATION The ECC card samples the 40-bits on the data bus. The ECC card makes a potential single-bit error correction, or detects a double-bit error. Storage grant is sent to the CCU if no double-bit error is detected.
36 bits are placed on the data bus for reading by the CCU. The ECC card reports any double-bit error detection, and starts the double-bit recovery cycle, if such an error was detected.

WRITE

In one MEM cycle or three CCU cycles:

OBJECTIVE Get all necessary information from the CCU to write 1 to 4 bytes into storage.
OPERATION Set 'storage go', 'byte select', and 'address' lines. Gate the bytes to be stored (from BSMI card), and the ECC half byte (from ECC card). The SCTL card reports any address parity error. The ECC card reports any parity error detected on incoming data bytes. Writing is aborted. If no error, read data is fetched and the data bus is disabled.
The 40-bit word fetched into MEM is placed on the data bus and sampled by the ECC card for correction. The ECC card makes a potential single-bit error correction, or detects a double-bit error.
The byte(s) to be stored are placed by the BSMI card on the data bus. The other byte(s) are merged on the data bus by the ECC card. The ECC byte is reset.
The ECC card generates the eight new ECC bits. Storage grant is sent to the CCU if no double-bit error was detected. Otherwise, a double-bit recovery cycle is started. The 40 new bits are placed on the data bus for writing into storage.

OUTPUT X'74' (ECC CONTROL)

In three CCU cycles:

OBJECTIVE Control the ECC input lines to change the ECC mode.
OPERATION The SCTL samples data bus byte X bits 2, 3, and 4, then conditions the ECC card, which stays in the same mode of operation until the next storage control output tag is executed. X.0 = Diagnostic only X.1 = Diagnostic only X.2 = ECC Disable X.3 = ECC Transparent X.4 = ECC Write Control (See Note) X.5 = Diagnostic only X.6 = Diagnostic only

Note: Bit X.4, when on, allows the ECC mode to be changed via bits X.2 and X.3 (see page 10-825).

INPUT X'70' (READ STORAGE SIZE)

In one or more CCU cycles:

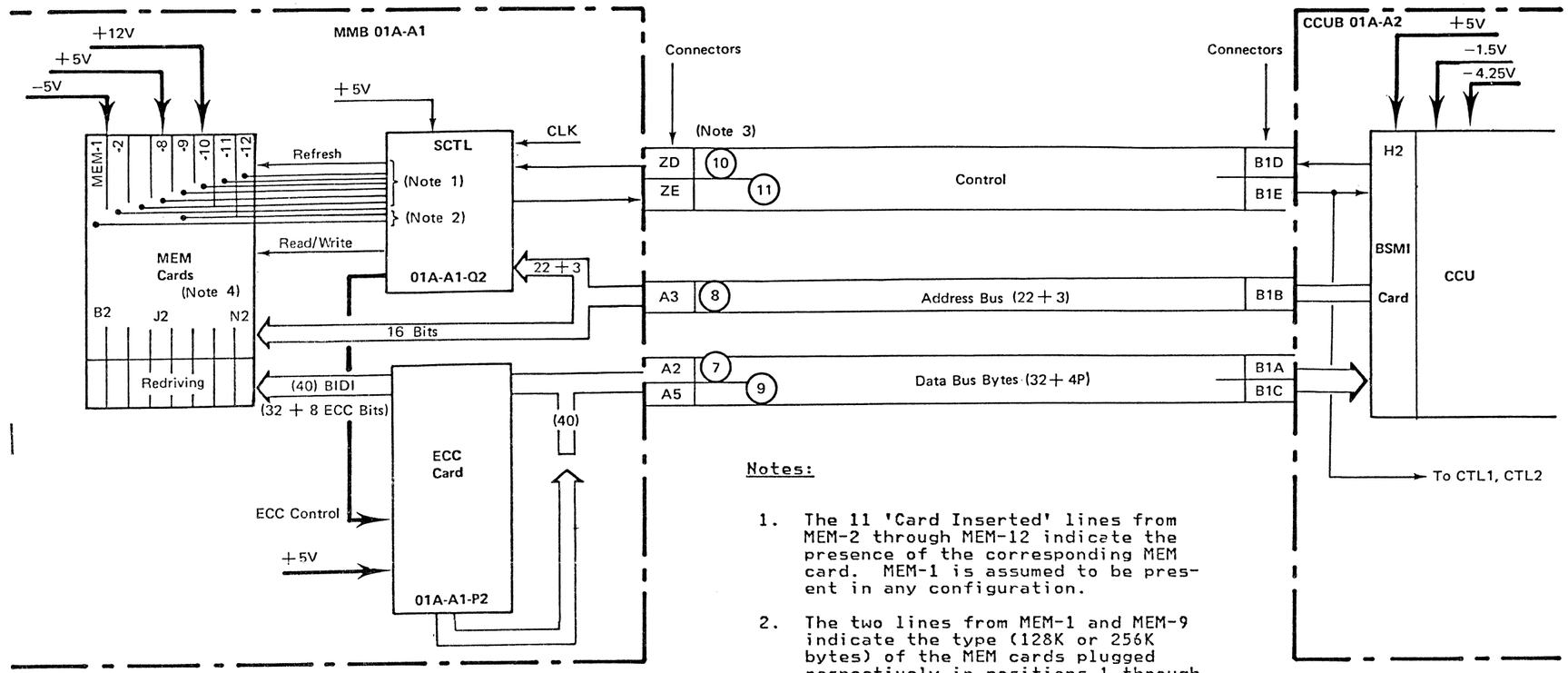
OBJECTIVE Get from SCTL card the size of the main storage.
OPERATION The SCTL, through the ECC card, places on data bus byte 0 (bits 3 to 7) the storage size implemented on the storage board. The SCTL also places information about the mixing, if any, of 128K and 256K MEM cards (bits 0 and 1).

REFRESH

In one MEM cycle or two CCU cycles:

OBJECTIVE Keep the main storage data valid
OPERATION The data bus and refresh address bits are disabled. The operation is controlled by the SCTL card, which generates the refresh address and appropriate timings.

Storage Data Flow and Partitioning



Notes:

1. The 11 'Card Inserted' lines from MEM-2 through MEM-12 indicate the presence of the corresponding MEM card. MEM-1 is assumed to be present in any configuration.

2. The two lines from MEM-1 and MEM-9 indicate the type (128K or 256K bytes) of the MEM cards plugged respectively in positions 1 through 8, and 9 through 12. Do not mix the two different types of cards.

   The storage size is obtained by combining the presence and type of all the MEM cards. This is carried out by Input X'70' (see page 10-230).

3. The circled numbers relate to the cables described in the signal tables on page 4-080 onwards.

4. Some early machines which do not have applied EC A04406 are limited to 1024K bytes of storage (eight 128K MEM cards in positions 1 through 8).

   Before incrementing storage above 1024K bytes on these machines, EC A04406 (SCTL and ECC cards upgrading) must be applied.

5. Before incrementing storage above 2048K bytes, EC A20965 and 342074 must be installed.

STORAGE CONFIGURATIONS

Card Location on MMB (01A-A1)	Field-Upgraded Machines		Factory-Upgraded Machines	
	MEM Cards	Storage (K Bytes)	MEM Cards	Storage (K Bytes)
B2	MEM-1		MEM-1*	
C2	MEM-2		MEM-2*	512
D2	MEM-3		MEM-3*	768
E2	MEM-4	512	MEM-4*	1024
F2	MEM-5		MEM-5*	1280
G2	MEM-6	768	MEM-6*	1536
H2	MEM-7		MEM-7*	1792
J2	MEM-8	1024 (Note 4)	MEM-8*	2048 (Note 5)
K2	MEM-9*	1280	MEM-9*	2304
L2	MEM-10*	1536	MEM-10*	2560
M2	MEM-11*	1792	MEM-11*	2816
N2	MEM-12*	2048 (Note 5)	MEM-12*	3072

Legend:
* : 256K MEM cards

MMB (CCU) Card Functions

Card	Functions
SCTL	Storage Internal timing - M clocks from CCLK Refresh timing from ACLK Data out gates to ECC, BSMI, storage Card inserted detection (input X'70') Address bit generation and checking Address/data parity checking (from CCU) Double-bit error reporting Recovery cycle when double-bit error
ECC	Single-bit error correction Double-bit error recovery
MEMx	32K words (128K bytes) of storage, or 64K words (256K bytes) of storage (see table at right)

Program Level Priorities and Interrupts (Part 1 of 2)

The 3725 control program is interrupt driven. It runs on five levels. Level 1 has the highest program priority, while level 5 has the lowest priority and forms the background level.

INTERRUPT MECHANISM

The interrupt mechanism (hardware) determines which level can be entered. The following conditions must occur for level 'n' to be entered:

- Program level 'n' must not be already active.
- The current instruction execution must be completed.
- No interrupt request of higher priority must be present.
- Program level 'n' must not be masked off.
- Cycle steal requests must not be present.

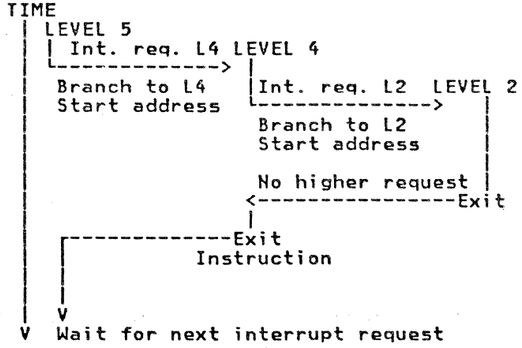
Just before a program level execution reaches its last instruction (Exit instruction):

- The interrupt that has just been serviced is reset.

Then, after the Exit instruction:

- The preceding interrupted program level resumes where it was stopped.

The following example illustrates the interrupt mechanism.



MULTIPLE INTERRUPT REQUESTS ON SAME LEVEL

Once entered, the program level must determine the cause of the interrupt. The priority between simultaneous requests for the same interrupt level is resolved by the program level just entered.

Input X'77': Obtains the level 2 and 3 interrupt requests that were set by the scanners or channel adapters.

Input X'7F': Obtains the level 2, 3, and 4 interrupt requests that were raised by the CCU, the MOSS, the program, or the operator.

Input X'7E': Obtains the level 1 interrupt requests that were set by the CCU.

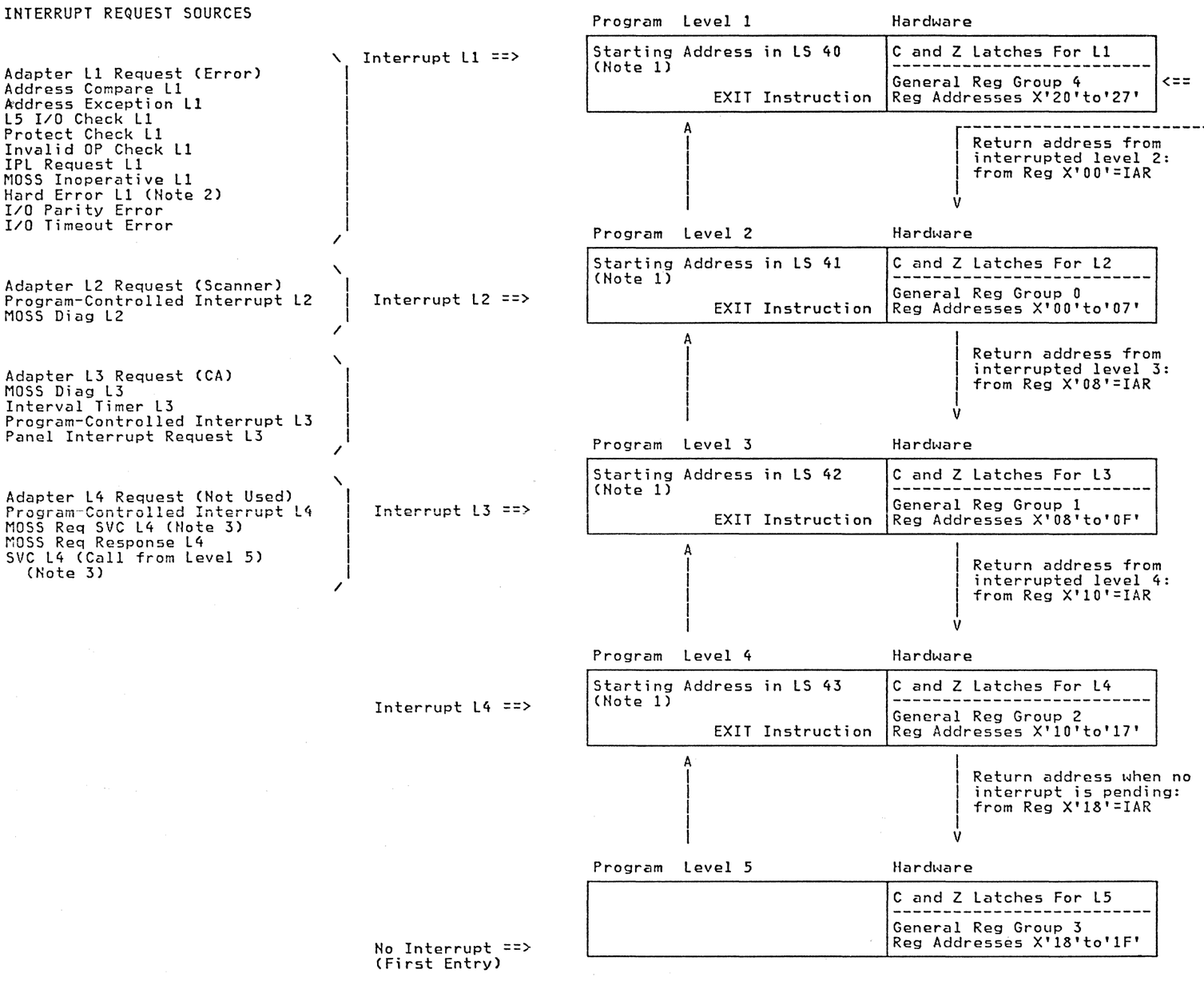
Input X'76': Obtains the level 1 interrupt requests that were set by errors in the adapters.

Input X'7D': Obtains the CCU hardware errors that caused the level 1 interrupt request.

SETTING/RESETTING INTERRUPT REQUESTS

Instructions X'7B' (set PCI level 2), X'7C' (set PCI level 3), and X'7D' (set PCI level 4) set the PCI interrupt requests. The output X'76' resets the IOC level 1 requests. Certain bits in output X'77' (miscellaneous control) reset the PCI requests and other CCU interrupt requests such as the interval timer level 3 request and the SVC level 4 request. If any bit is on in registers X'77', X'7E', or X'7F', a request for a particular program level is active; it must be reset or masked before the program can execute in a lower level.

CCU Program Level Priorities and Interrupts (Part 2 of 2)



Notes:

1. The interrupt level 1, 2, 3, and 4 starting addresses are set during IPL via output instructions X'40' through X'43'.
2. Hard errors stop the CCU unless it is in bypass mode.
3. SVC is supervisor call.

MIOC Control Logic

The CCU and the MOSS communicate via the MIOC control logic. This interface card enables the MOSS to:

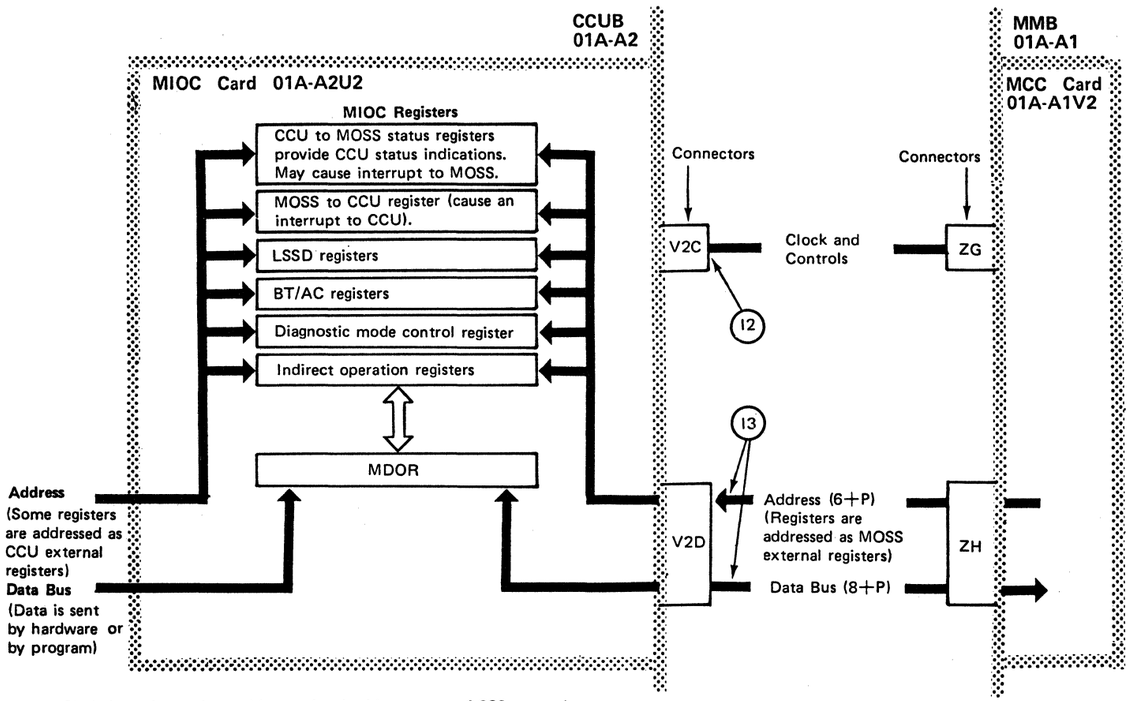
- Read or write any register or storage location in the CCU
- Set or reset all discrete CCU latches and read out their status
- Interrupt the CCU

Also, the MIOC allows the CCU to interrupt the MOSS.

To perform these functions the MIOC card executes:

- Direct operations
- Indirect operations
- Level scan sensitive design (LSSD) operations. For a description of LSSD, see Chapter 14.

MIOC DATA FLOW



Note: Circled numbers relate cables to signal table on pages 4-080 onwards.

DIRECT OPERATIONS

In direct operations, the MOSS reads or writes any MIOC register to:

- Check the CCU status (carry or zero indicators, interrupt level running)
- Place the CCU in diagnostic mode
- Prepare the branch trace and address compare circuits
- Control LSSD operations
- Prepare indirect operations

Direct operations usually take a CCU cycle, and require a CCU ROS cycle to run.

INDIRECT OPERATIONS

In indirect operations, the MOSS can read or write to any CCU circuits such as data flow registers, work registers, or CCU X'7x' registers.

Indirect operations take one or several CCU ROS cycles to run.

Indirect looped operations allow storage diagnostic operations (storage scan).

LSSD OPERATIONS

During LSSD operations, the MOSS can checkout or change the status of any discrete CCU latch. These operations can be performed after any CCU cycle, by stopping the CCU clock.

LSSD operations are diagnostic tools used during IML, or when running CCU diagnostics.

IOC CONTROL LOGIC

Data, address, and control information exchanges take place between the CCU work registers and the adapters attached to the I/O bus. They use the IOC control logic (packaged on the DFL5 card).

The IOC logic operates in two different ways, depending on whether the program initiates the operation (PIO), or an adapter initiates it (AIO). In both types of operation, the A (address) and D (data) registers of the IOC logic act as buffers between the CCU and the channel adapters or communication scanners.

The data bus carries interrupt requests from the adapters (levels 1, 2, and 3) when it is not busy with PIO or AIO operations.

For details of IOC operations, see Chapter 11.

Instruction Formats

Instruction	Format Code	Mnemonic	Operand Field Format
Adapter Input/Output	RR	IOH	R1,R2
Adapter Input/Output Immediate	RA	IOHI	R,A
Add Character Register	RR	ACR	R1(N1),R2(N2)
Add Halfword Register	RR	AHR	R1,R2
Add Register	RR	AR	R1,R2
Add Register Immediate	RI	ARI	R(N),I
AND Character Register	RR	NCR	R1(N1),R2(N2)
AND Halfword Register	RR	NHR	R1,R2
AND Register	RR	NR	R1,R2
AND Register Immediate	RI	NRI	R(N),I
Branch	RT	B	T
Branch and Link	RA	BAL	R,A
Branch and Link Register	RR	BALR	R1,R2
Branch on Bit	RT	BB	R(N,M),T
Branch on Count	RT	BCT	R(N),T
Branch on C Latch	RT	BCL	T
Branch on Z Latch	RT	BZL	T
CCU Register Input	RE	IN	R,E
CCU Register Output	RE	OUT	R,E
Compare Character Register	RR	CCR	R1(N1),R2(N2)
Compare Halfword Register	RR	CHR	R1,R2
Compare Register	RR	CR	R1,R2
Compare Register Immediate	RI	CRI	R(N),I
Exclusive OR Character Register	RR	XCR	R1(N1),R2(N2)
Exclusive OR Halfword Register	RR	XHR	R1,R2
Exclusive OR Register	RR	XR	R1,R2
Exclusive OR Register Immediate	RI	XRI	R(N),I
Exit	EXIT	EXIT	
Insert Character	RS	IC	R(N),D(B)
Insert Character and Count	RSA	ICT	R(N),B
Load	RS	L	R,D(B)
Load Address	RA	LA	R,A
Load Character Register	RR	LCR	R1(N1),R2(N2)
Load Character with Offset Reg	RR	LCOR	R1(N1),R2(N2)
Load Halfword	RS	LH	R,D(B)
Load Halfword Register	RR	LHR	R1,R2
Load Halfword with Offset Reg	RR	LHOR	R1,R2
Load Register	RR	LR	R1,R2
Load Register Immediate	RI	LRI	R(N),I
Load with Offset Register	RR	LOR	R1,R2
OR Character Register	RR	OCR	R1(N1),R2(N2)
OR Halfword Register	RR	OHR	R1,R2
OR Register	RR	OR	R1,R2
OR Register Immediate	RI	ORI	R(N),I
Store	RS	ST	R,D(B)
Store Character	RS	STC	R(N),D(B)
Store Character and Count	RSA	STCT	R(N),B
Store Halfword	RS	STH	R,D(B)
Subtract Character Register	RR	SCR	R1(N1),R2(N2)
Subtract Halfword Register	RR	SHR	R1,R2
Subtract Register	RR	SR	R1,R2
Subtract Register Immediate	RI	SRI	R(N),I
Test Register Under Mask	RI	TRM	R(N),I

F. Code	Mnemonic	Instruction	CZ Latch	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
EXIT	EXIT	Exit	n	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0
RA	BAL	Branch and Link	n	1	0	1	1	1	R			0	1	Add Byte X					
	IOHI	Adapter I/O Immediate	y	0	0	0	0	0	R			0	1	1	1	0	0	0	0
	LA	Load Address	n	1	0	1	1	1	R			0	0	Add Byte X					
RE	IN	CCU Register Input	n	0	E			0	R			E				1	1	0	0
	OUT	CCU Register Output	n	0	E			0	R			E				0	1	0	0
RI	ARI	Add Register Immediate	y	1	0	0	1	0	R	N	Immediate Data								
	CRI	Compare Reg. Immediate	y	1	0	1	1	0	R	N	Immediate Data								
	LRI	Load Register Immediate	y	1	0	0	0	0	R	N	Immediate Data								
	NRI	AND Register Immediate	y	1	1	1	0	0	R	N	Immediate Data								
	ORI	OR Register Immediate	y	1	1	0	1	0	R	N	Immediate Data								
	SRI	Subtract Reg. Immediate	y	1	0	1	0	0	R	N	Immediate Data								
	TRM	Test Register Under Mask	y	1	1	1	1	0	R	N	Immediate Data								
	XRI	XOR Register Immediate	y	1	1	0	0	0	R	N	Immediate Data								
RR	BALR	Branch and Link Register	n	0	R2			0	R1			0	1	0	0	0	0	0	0
	IOH	Adapter Input Output	y	0	R2			0	R1			0	1	0	1	0	0	0	0
	AR	Add Register	y	0	R2			0	R1			1	0	0	1	1	0	0	0
	CR	Compare Register	y	0	R2			0	R1			1	0	1	1	1	0	0	0
	LOR	Load with Offset	y	0	R2			0	R1			1	1	1	1	1	0	0	0
	LR	Load Register	y	0	R2			0	R1			1	0	0	0	1	0	0	0
	NR	AND Register	y	0	R2			0	R1			1	1	1	0	1	0	0	0
	OR	OR Register	y	0	R2			0	R1			1	1	0	1	1	0	0	0
	SR	Subtract Register	y	0	R2			0	R1			1	0	1	0	1	0	0	0
	XR	XOR Register	y	0	R2			0	R1			1	1	0	0	1	0	0	0
	ACR	Add Character Register	y	0	R2		N	0	R1		N	0	0	0	1	1	0	0	0
	CCR	Compare Character reg.	y	0	R2		N	0	R1		N	0	0	1	1	1	0	0	0
	LCOR	Load Char with Offset Reg.	y	0	R2		N	0	R1		N	0	1	1	1	1	0	0	0
	LCR	Load Character Register	y	0	R2		N	0	R1		N	0	0	0	0	1	0	0	0
	NCR	AND Character Register	y	0	R2		N	0	R1		N	0	1	1	0	1	0	0	0
	OCR	OR Character Register	y	0	R2		N	0	R1		N	0	1	0	1	1	0	0	0
	SCR	Subtract Character Reg.	y	0	R2		N	0	R1		N	0	0	1	0	1	0	0	0
	XCR	XOR Character Register	y	0	R2		N	0	R1		N	0	1	0	0	1	0	0	0
	AHR	Add Halfword Register	y	0	R2			0	R1			1	0	0	1	0	0	0	0
	CHR	Compare Halfword Register	y	0	R2			0	R1			1	0	1	1	0	0	0	0
	LHOR	Load Halfword with Offset	y	0	R2			0	R1			1	1	1	1	0	0	0	0
	LHR	Load Halfword Register	y	0	R2			0	R1			1	0	0	0	0	0	0	0
	NHR	AND Halfword Register	y	0	R2			0	R1			1	1	1	0	0	0	0	0
	OHR	OR Halfword Register	y	0	R2			0	R1			1	1	0	1	0	0	0	0
	SHR	Subtract Halfword Reg.	y	0	R2			0	R1			1	0	1	0	0	0	0	0
	XHR	XOR Halfword Register	y	0	R2			0	R1			1	1	0	0	0	0	0	0
RS	L	Load	y	0	B			0	R			0	Displ.					1	0
	ST	Store	n	0	B			0	R			1	Displ.					1	0
	LH	Load Halfword	y	0	B			0	R			0	Displ.						1
	STH	Store Halfword	n	0	B			0	R			1	Displ.						1
	IC	Insert Character	y	0	B			1	R			0	Displacement						
	STC	Store Character	n	0	B			1	R			1	Displacement						
RSA	ICT	Insert Char. and Count	n	0	B			0	R		N	0	0	0	1	0	0	0	0
	STCT	Store Char. and Count	n	0	B			0	R		N	0	0	1	1	0	0	0	0
RT	B	Branch	n	1	0	1	0	1	Displacement										
	BCL	Branch on C Latch	n	1	0	0	1	1	Displacement										
	BZL	Branch on Z Latch	n	1	0	0	0	1	Displacement										
	BB	Branch on Bit	n	1	1	M	M	1	R		N	M	Displ.						
	BCT	Branch on Count	n	1	0	1	1	1	R		N	1	Displ.						

Second halfword	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	1	1	1	1	1	1										

Add Byte 0	Add Byte 1
Contents depend on adapter	
Add Byte 0	Add Byte 1

Instruction Decoding

These tables may be used to decode the four-digit hexadecimal representation of a 3725 machine instruction.

Use the tables as follows:

1. Locate the first digit (D1) of the instruction in hex in the column of numbers on the left side of Table 1.
2. Locate the second digit of the instruction in the row of numbers at the top of Table 1.
3. Go to the intersection of the column and row represented by the two numbers. You will find either the mnemonic or a reference to Table 2, Table 3, or Table 4.

Tables 2 and 4 require that you locate digit three (D3) of the instruction only in the row of digits at the top of each chart. Follow the instructions for Table 1 to use Table 3, substituting digit three (D3) and digit four (D4).

Table 1

(D1) \ (D2)	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	Table 3								Table 2							
1																
2																
3																
4																
5																
6																
7																
8	LRI								BZL							
9	ARI								BCL							
A	SRI								B							
B	CRI								Table 4							
C	XRI															
D	ORI								BB							
E	NRI															
F	TRM															

Table 2

(D3) 0 1 2 3 4 5 6 7	8 9 A B C D E F
IC	STC

Table 3

(D3) \ (D4)	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	(Note 1)								LCR							
1	ICT								ACR							
2	(Note 1)								SCR							
3	STCT	LH	L	LH		LH	L	LH	CCR	LH	L	LH		LH	L	LH
4	BALR								XCR							
5	IOH								OCR							
6	(Note 1)								NCR							
7	(Note 2)	---	---	---	OUT				LCOR	---	---	---	IN	---	---	---
8	LHR								LR							
9	AHR								AR							
A	SHR	STH	ST	STH		STH	ST	ST	SR	STH	ST	STH		STH	ST	STH
B	CHR	H		H		H		H	CR	H		H		H		H
C	XHR								XR							
D	OHR								OR							
E	NHR								NR							
F	LHOR								LOR							

Notes:

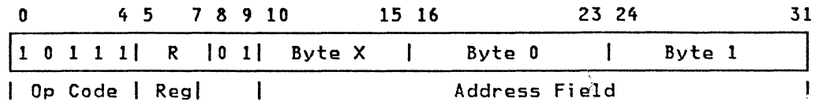
1. Invalid operations
2. If D2 is zero, the instruction is an Exit.
   If D2 is 1, 2, 3, 4, 5, 6, or 7, the instruction is an IOHI.

Table 4

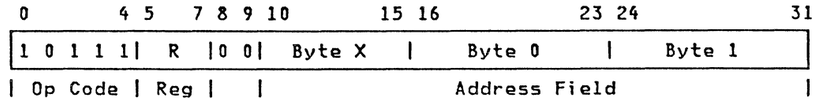
(D3) 0 1 2 3	4 5 6 7	8 9 A B C D E F
LA	BAL	BCT

Register and Immediate Address (RA) Instructions

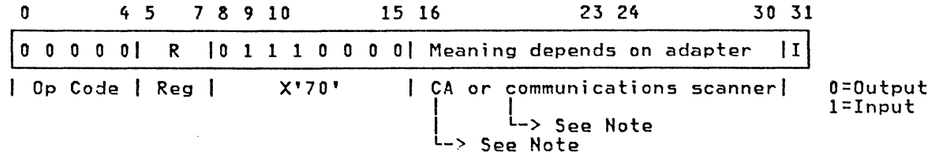
BALR INSTRUCTION



LA INSTRUCTION

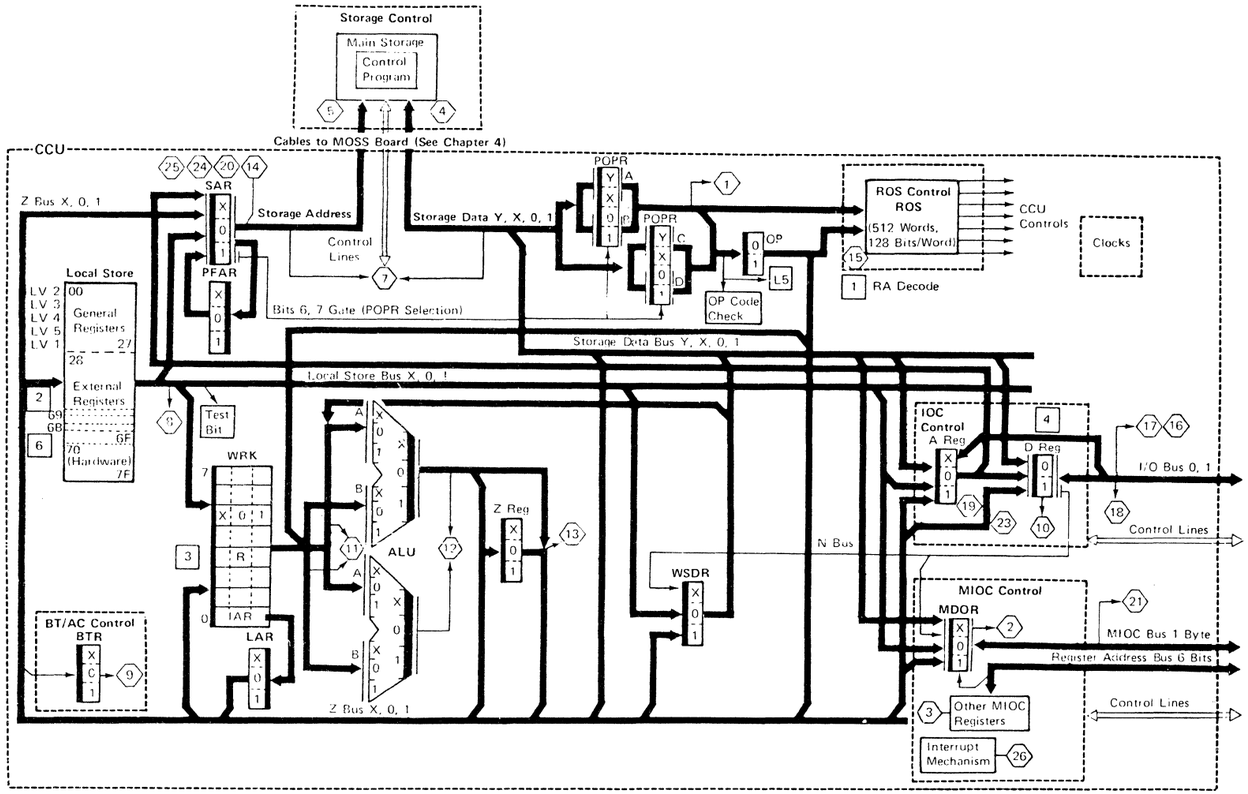


IOHI INSTRUCTION



Note: For more information on IOH, IOHI, and PIO operation, see:
"Address/Command Formats on IOC Bus", page 11-040
"PIO Operation", page 11-020
"Accessing Channel Adapter Registers", page 12-015

DATA FLOW FOR IOHI INSTRUCTION



Cycle Sequence Tables

BAL R=0 R≠0

1	1	A field --> IAR A field --> X'69', save branch address
	2	(IAR)+2 --> R, save link address
	3	(X'69') --> IAR, branch

LA R=0 R≠0

	1	A field --> R
1	1	(IAR)+2 --> IAR
		A field --> IAR, branch
	2	
	3	

IOHI

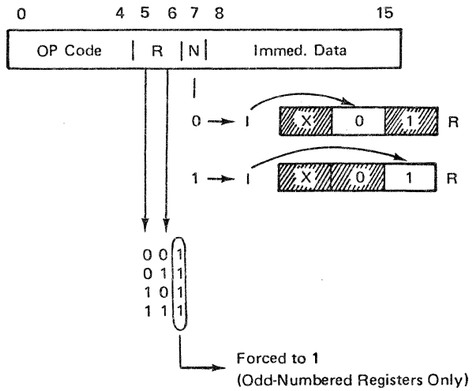
R≠0	
1	2nd Halfword --> X'69'
2	(IAR)+2 --> IAR, fullword instruction
3	(X'69') --> IOC A Reg, TD data
4 4	IOC A Reg --> IOC D Reg (TA data) (R) --> IOC A Reg (TD data)
5	(WSDR) --> LS6B (Save TD data if IOC busy)

If R=0 the IOHI acts as an exit instruction.

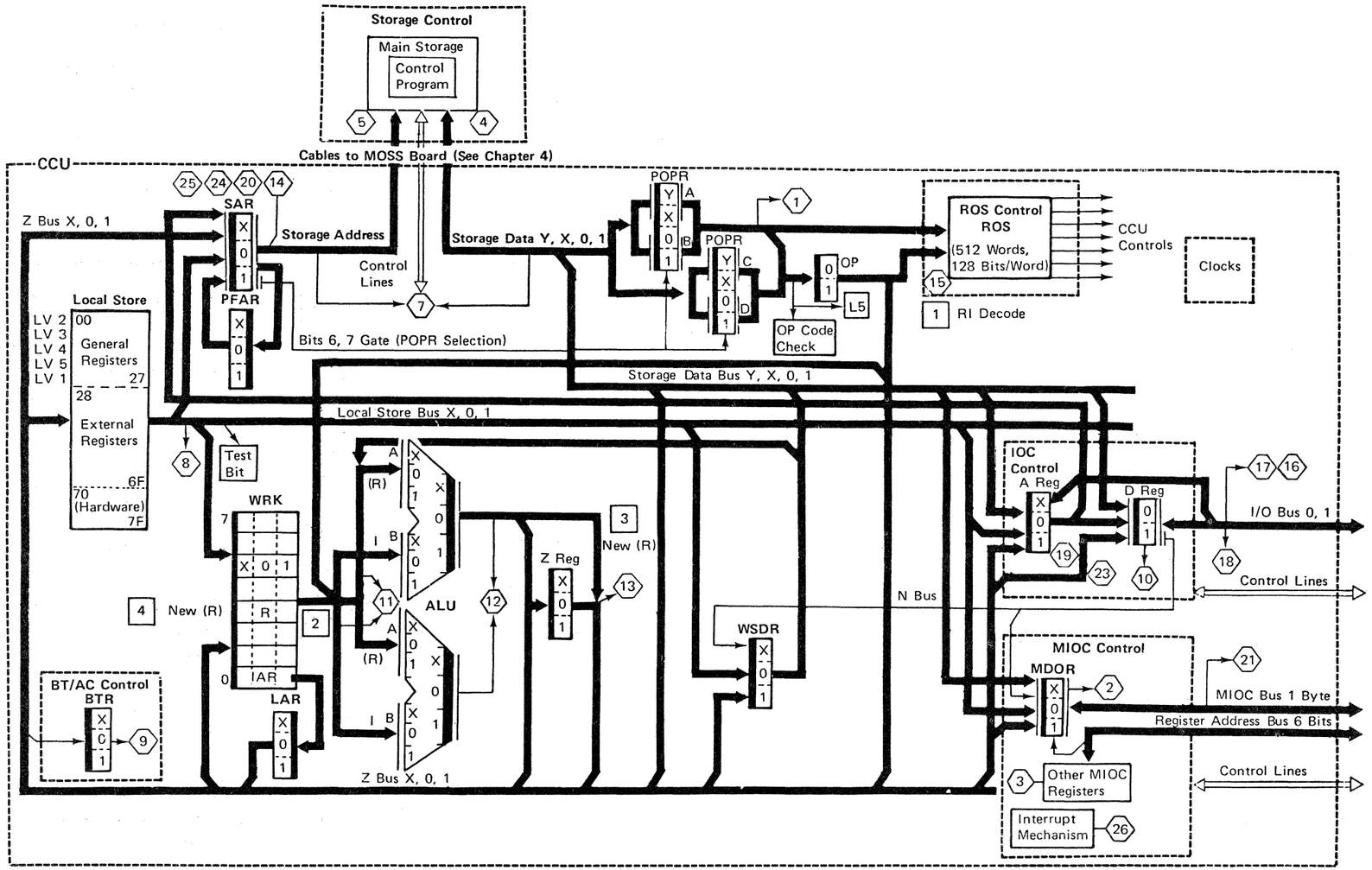
Operation

1. Decode RA.
2. Adapter information (2nd halfword)
3. (IAR)+2.
4. Adapter information (TA and TD data) to IOC control.
5. (R) to adapter (TD data).
6. (R) kept in X'6B'.

Register Immediate (RI) Instructions



DATA FLOW FOR RI INSTRUCTION



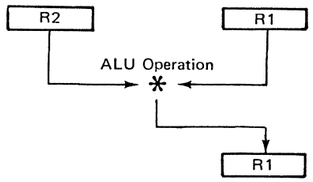
Operation

1. Decode RI instructions. Address selected odd WKR.
2. Gate (R) to ALU A. Gate I field to ALU B.
3. Build new (R) according to N value.
4. Transfer new (R) into selected WKR.

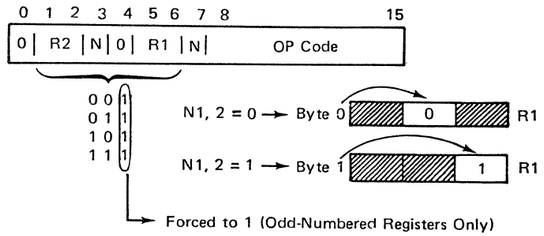
Note: The CCU takes one cycle to execute an RI instruction.

Register to Register (RR) Instructions

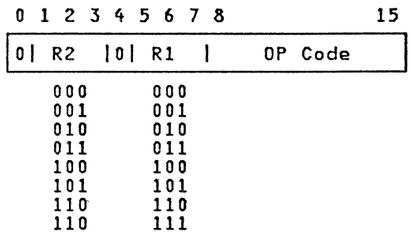
(Except BALR and IOH/IOHI)



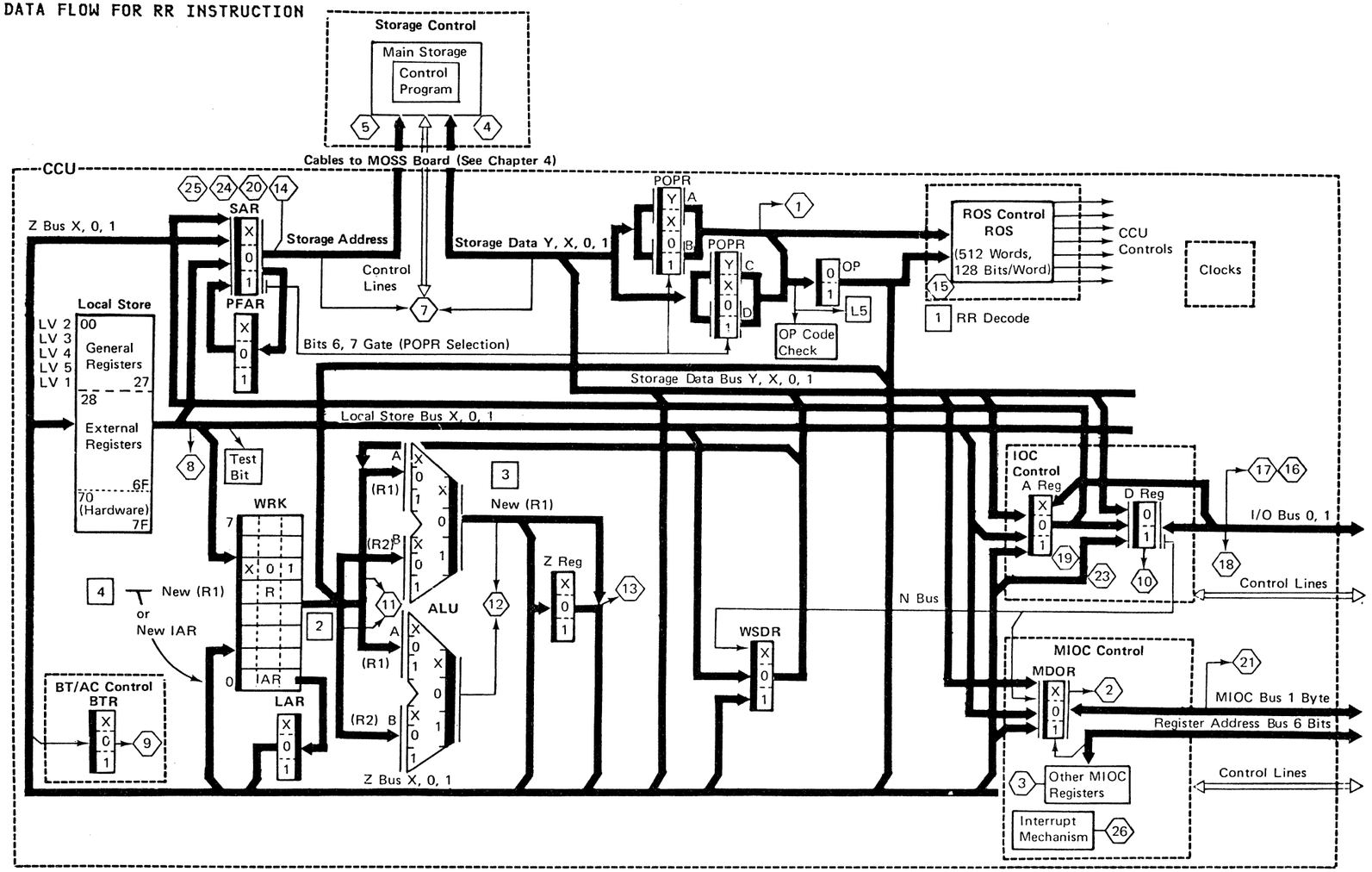
RR INSTRUCTIONS ON CHARACTER



RR INSTRUCTIONS ON HALFWORD AND WORD



DATA FLOW FOR RR INSTRUCTION

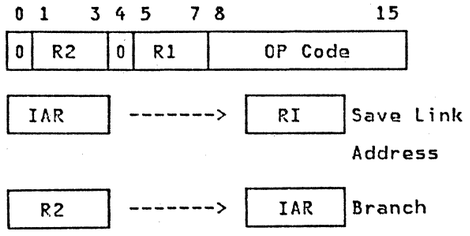


Operation

1. Decode RR instruction.
   Address selected R1, R2 WKR.
2. Gate (R1) to ALU A bytes X, 0, 1.
   Gate (R2) to ALU B bytes X, 0, 1.
3. Build new (R1) according to ALU operation
4. Transfer new (R1) into R1 WKR.
   If R1=0, the IAR is selected, and a branch occurs.

Exceptions to RR Instructions

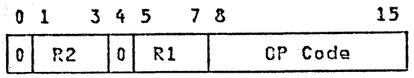
BRANCH AND LINK REGISTER



Cycle Sequence Table

R1≠0 R2≠0	R1=0 R2≠0	R1≠0 R2=0	R1=0 R2=0	
1				(IAR) --> Z Reg
1	1			(R2) --> IAR
		1	1	(IAR) --> SAR
		1		(IAR) --> R1
2				(Z Reg) --> R1

ADAPTER INPUT/OUTPUT (IOH)

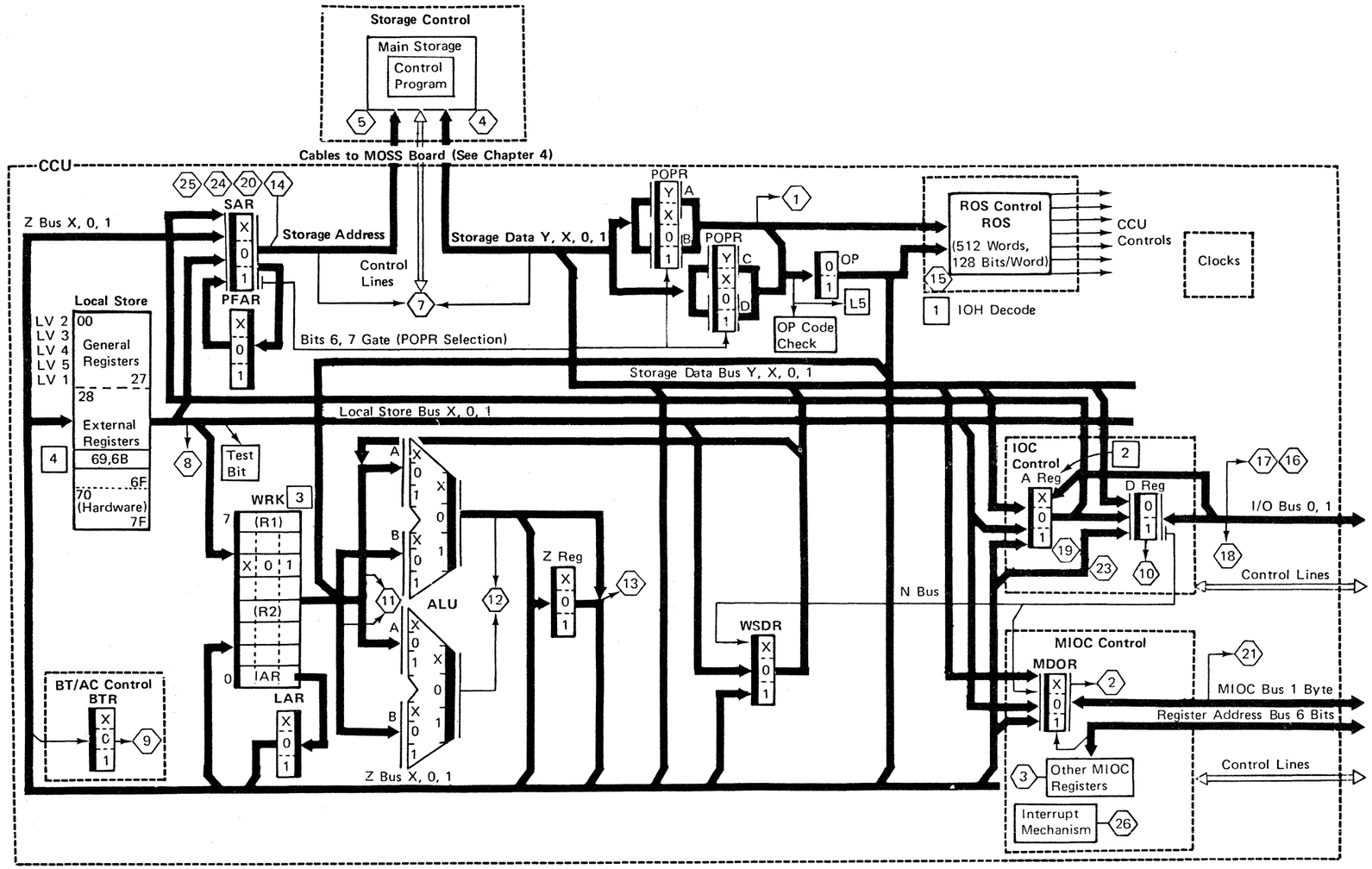


This instruction transfers the R1 contents to the selected adapter, or loads the adapter data into R1. The R2 contents provide the adapter address and the direction of data movement. See Chapter 12 for information on the channel adapter, and Chapter 13 for the communication scanner.

Cycle Sequence Table

R1≠0 R2=0	R1≠0 R2≠0	
1	1	Read R2 Read X'48'
2	2	LS (R2) --> IOC A Reg (TA data)
3	3	IOC A Reg --> IOC D Reg (TA data) LS (R1) --> IOC D Reg (TD data) LS (R1) --> WSDR (TD data)
4	4	(WSDR) -> X'6B' (Save TD data if IOC is busy)
5	5	(R2) --> WSDR (X'48') -> WSDR
6	6	(WSDR) -> X'69' (Save TD data if IOC is busy)

DATA FLOW FOR IOH INSTRUCTION



Operation

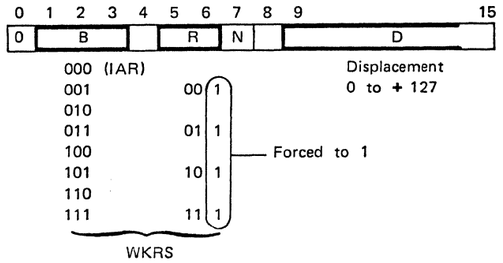
1. Decode IOH instruction.
2. Read adapter address in R2 for input or output operation.
3. Read out or write in R1.
4. Hold copies of (R1) in LS6B, (R2) in LS69.

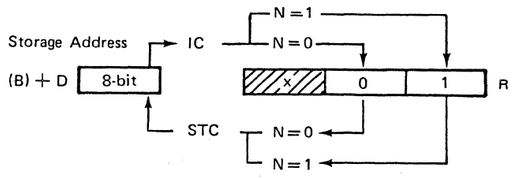
Notes:

1. If R2 designates the IAR, the adapter information is taken from X'48'.
2. If R1 designates the IAR, an invalid op check L1 is raised.

RS Instructions on Character

IC, STC





Cycle Sequence Table

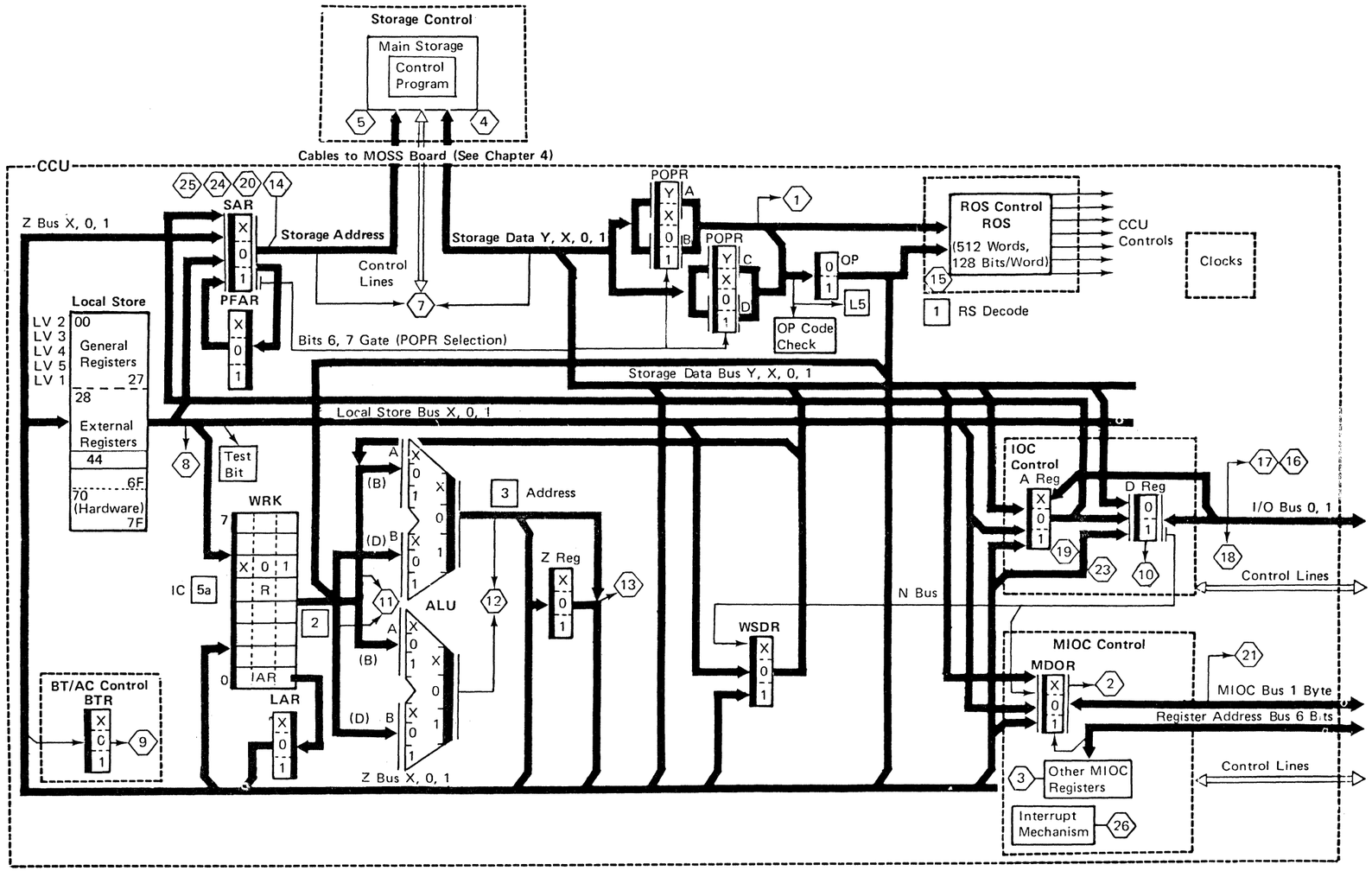
Store Character (STC)

B≠0	B=0	
	1	(X'44' --> WSDR
	2	(WSDR) + D --> SAR
1		(B) + D --> SAR
1		(R) --> WSDR
If SAR bit 1.6=0		
1	2	(R) byte N --> STG byte 0, or 1
If SAR bit 1.6=1		
1	2	(R) byte N --> STG byte y, or x

Insert Character (IC)

B≠0 N=0,1	B=0 N=0,1	
	1	(X'44') --> WSDR
1		(B) + D --> SAR
	2	(WSDR) + D --> SAR
2	3	STG y, x, 0, or 1 --> R byte N

DATA FLOW FOR RS INSTRUCTION ON CHARACTER



Operation

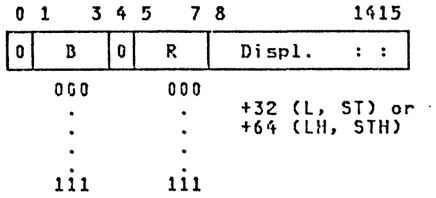
1. Decode RS instruction.
2. Gate (B) to ALU A and D to ALU B.
3. Add (B)+D --> SAR.
4. Address storage.
5a. IC, transfer storage data to selected R byte.
5b. STC, transfer data from selected R byte to storage.

Note: If B is 0, the IAR is not used. Instead, the contents of the external register X'44' is added to the D field. This permits direct addressing of the 128 bytes starting at the address contained in X'44' without having to load a base register.

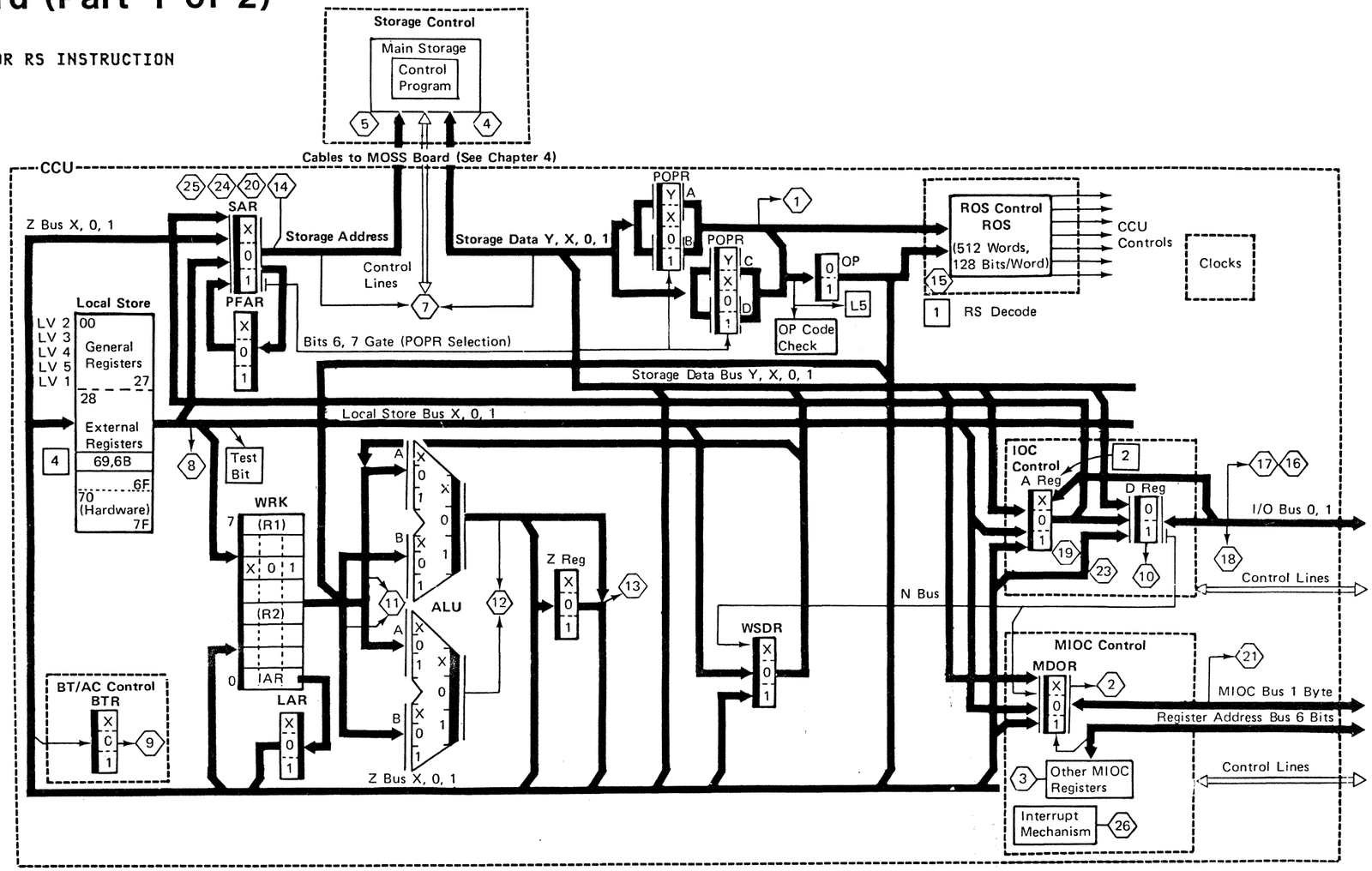
RS Instruction on Halfword and Word (Part 1 of 2)

L AND LH, ST AND STH

The CCU takes a number of cycles that depends on the value of R and B, and on the word or halfword addressing boundary in storage. For every RS instruction, a table indicates the sequence of cycles, and shows in which register(s) the cycle information result can be read.



DATA FLOW FOR RS INSTRUCTION



Operation

1. Decode RS instruction. Address B Reg (B"000).
2. Gate (B) to ALU A and D to ALU B.
3. Add (B)+D --> SAR, storage data address.
4. Address storage.
5a. L and LH, transfer storage data to R.
5b. ST and STH, transfer data from R to storage.

Notes:

1. If R=000, the IAR is selected. The operation results in a branch to the new address.
2. If B=000, the IAR is not used. Instead, the contents of a defined external register is added to the D field. This permits direct addressing of the 32 words starting at the address contained in X'46' (load and store operations), or the 64 halfwords starting at the address contained in X'45' (load halfword and store halfword operations) without having to load a base register.

RS Instruction on Halfword and Word (Part 2 of 2)

CYCLE SEQUENCE TABLES

Load (L)

R≠0 B≠0	R=0 B≠0	R≠0 B=0	R=0 B=0	
		0	0	(X'46') --> WSDR
		1	1	(WSDR)+D --> SAR
1	1			(B)+D --> SAR
If SAR bit 1.6=0				
2	2	2	2	Low-order bits of STG word --> Z Reg
3		3		(Z Reg) --> R
	3		3	(Z Reg) --> IAR (branch)
If SAR bit 1.6=1				
2	2	2	2	Low-order bits of 1st STG word --> Z Reg byte X
3	3	3	3	(Z Reg) --> R SAR+Z --> SAR
	4		4	High-order bits of 2nd STG word --> Z Reg bytes 0, 1
4		4		(Z Reg) merged with R, result --> R
	4		4	(Z Reg) merged with IAR, result --> IAR (branch)

Load Halfword (LH)

R≠0 B≠0	R=0 B≠0	R≠0 B=0	R=0 B=0	
		1	1	(X'45') --> WSDR
		2	2	(WSDR)+D --> SAR
1	1			(B)+D --> SAR
If SAR bit 1.6=0				
2		3		STG bytes 0, 1 --> R bytes 0, 1 and byte x=0
	2		3	STG bytes 0, 1 --> IAR, byte x=0
If SAR bit 1.6=1				
2		3		STG bytes y, x --> R bytes 0, 1 and byte x=0
	2		3	STG bytes y, x --> IAR, byte x=0

Store (ST)

R≠0 B≠0	R=0 B≠0	R≠0 B=0	R=0 B=0	
		0	0	(X'46') --> WSDR
		1	1	(WSDR)+D --> SAR
1	1			(B)+D --> SAR
			1	(X'68')=000 --> WSDR
		1		(R) --> WSDR
If SAR bit 1.6=0				
2	2	2	2	(WSDR) --> Low-order bits of STG word
If SAR bit 1.6=1				(WSDR) byte x -->
2	2	2		Low-order bits of 1st STG word
3	3	3	3	(SAR)+2 --> SAR
				(WSDR) bytes 0, 1 -->
				High-order bits of 2nd STG word
4	4	4	4	Operation End

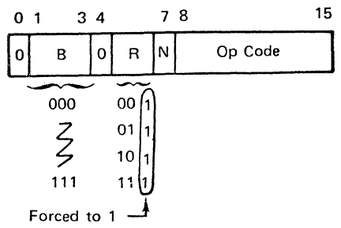
Store Halfword (STH)

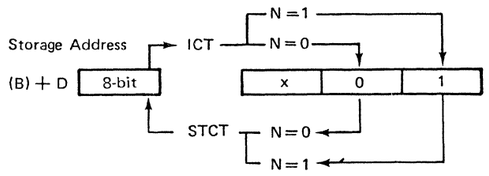
R≠0 B≠0	R=0 B≠0	R≠0 B=0	R=0 B=0	
		1	1	(X'45') --> WSDR
1	1			(B)+D --> SAR
		2	2	(WSDR)+D --> SAR
	1		2	(X'68')=000 --> WSDR
1		2		(R) --> WSDR
If SAR bit 1.6=0				
1	1	2	2	(WSDR) bytes 0, 1 --> STG bytes 0, 1
If SAR bit 1.6=1				
1	1	2	2	(WSDR) bytes 0, 1 --> STG bytes x, y

Register and Storage With Addition (RSA) Instructions

ICT AND STCT

The CCU takes three cycles to decode and execute an RSA instruction. Cycle sequence tables indicate in which register(s) the cycle information result can be read.





Cycle Sequence Tables

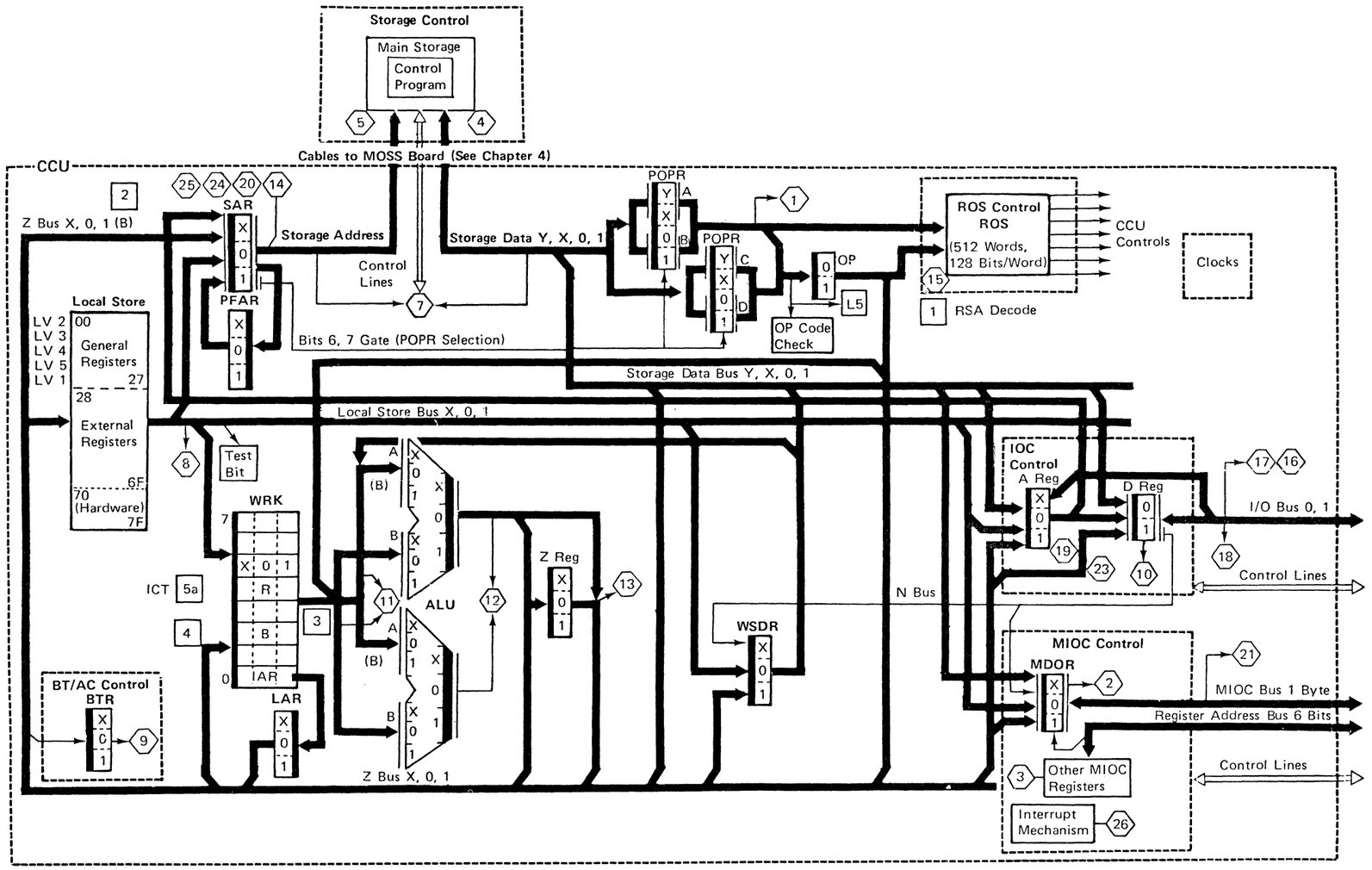
Insert Character and Count (ICT)

B≠0	B=0 is invalid
1	(B) --> SAR
2 2	(B) + 1 --> B (count) Read storage
3	Selected STG byte (Insert) --> R byte 0 (N=0), or --> R byte 1 (N=1)

Store Character and Count (STCT)

B≠0	B=0 is invalid
1	(B) --> SAR
2	(B) + 1 --> B (count)
3	R byte 0, or R byte 1 --> Selected STG byte (store)

DATA FLOW FOR RSA INSTRUCTION



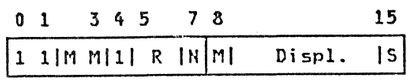
Operation

1. Decode instruction.
2. Send (B) to SAR.
3. Add 1 to (B), count.
4. Update B.
5a. ICT, transfer the selected storage data to R.
5b. STCT, transfer the selected R byte to storage.

Note: If B and R specify the same (odd) register, the contents of this register are incremented by 1 before the selected byte is transferred.

Branch (RT) Instructions

BRANCH ON BIT (BB)



This instruction results in a branch to a storage address, depending on the state of a bit tested in the general register specified by R.

The branch address is formed by adding or subtracting the displacement contained in the D field to or from the IAR, which contains the next sequential address. The maximum displacement is 63 halfwords. If S=0 in the instruction, the displacement is added to the IAR contents; if S=1, the displacement is subtracted from it.

The M field specifies which bit in byte 0 (N=0) or in byte 1 (N=1) of the general register R is to be tested. If the bit is zero, the IAR is not modified, the branch is not taken, and the instruction at the next sequential address is executed. If the bit is one, the instruction at the branch address is executed.

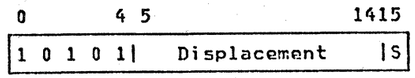
The C and Z condition latches remain unchanged.

	Bit to be tested in Byte 0								Bit to be tested in Byte 1							
	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
N=	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
M2=	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
M3=	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
M8=	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Cycle Sequence Table

S=0	S=1	
1		(IAR)+T --> Z Reg
	1	(IAR)-T --> Z Reg
1	1	Test bit in R (odd)
If Bit tested = 0, idle cycle		
2	2	
If Bit tested = 1, branch taken		
2	2	Z Reg --> SAR
2	2	Z Reg --> IAR

BRANCH (B)

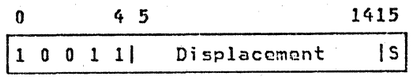


This instruction results in an unconditional branch to a storage address.

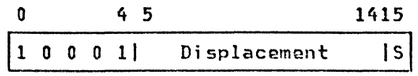
The branch address is formed by adding or subtracting the displacement in the I field to or from the IAR, which contains the next sequential address. The maximum displacement is 1023 halfwords. If S=0 in the instruction, the displacement is added to the IAR contents; if S=1, the displacement is subtracted from it.

The C and Z condition latches remain unchanged.

BRANCH ON C LATCH (BCL)



BRANCH ON Z LATCH (BZL)



These instructions result in a branch to a storage address according to the state of the C or Z condition latches.

The branch address is formed by adding or subtracting the displacement in the T field to or from the IAR, which contains the next sequential address. The maximum displacement is 1023 halfwords. If S=0 in the instruction, the displacement is added to the IAR contents; if S=1, the displacement is subtracted from it.

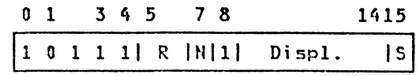
If the tested condition latch is not set, the IAR is not modified, the branch is not taken, and the instruction at the next sequential address is executed. If the tested condition latch is set, the instruction at the branch address is executed.

The C and Z condition latches remain unchanged.

Cycle Sequence Table

S=0	S=1	
1		(IAR)+T --> Z Reg
	1	(IAR)-T --> Z Reg
1	1	Test condition latch
If tested latch = 0, idle cycle		
2	2	(IAR) --> SAR
If tested latch = 1, branch is taken		
2	2	Z Reg --> SAR
2	2	Z Reg --> IAR

BRANCH ON COUNT (BCT)



The count contained in byte 0 (N=0), or byte 0 and 1 (N=1) of the general register specified by R is decremented by 1, and then tested. If the result is 0, the instruction at the next sequential address is executed; if it is not 0, the instruction at the branch address is executed. If the count is 0 before executing the instruction, the result is all 1s in byte 0 (N=0), or bytes 0 and 1 (N=1), the count value then being 256, or 65, 536 respectively.

The branch address is formed by adding or subtracting the displacement in the T field to or from the IAR, which contains the next sequential address. The maximum displacement is 63 halfwords. If S=0 in the instruction, the displacement is added to the IAR; if S=1, the displacement is subtracted from it.

The C and Z condition latches remain unchanged.

Cycle Sequence Table

S=0	S=1	
1		(IAR)+T --> Z Reg
	1	(IAR)-T --> Z Reg
1	1	Test count in R(odd)
If count = 0, No branch		
2	2	Idle cycle
3	3	(R) -1 --> R
If count is positive, branch is taken		
2	2	Z Reg --> SAR
2	2	Z Reg --> IAR
3	3	(R) -1 --> R

Exit Instruction

0	3 4	7 8	1112	15
0 0 0 0	0 0 0 0	0 1 1 1	0 0 0 0	

This instruction is used to exit from the active program level. The interrupt priority logic then determines which group of general registers to select as the active group for the next program operation. This instruction also resets the 'interrupt entered' latch for the program level that executes it.

If the Exit instruction is executed at program level 5, the level 4 supervisor call interrupt request (SVC L4) is set. The next instruction is then normally the instruction at the starting address for program level 4. However, if other interrupt requests of higher priority are present, the next instruction executed is the instruction at the starting address of the highest priority program level requesting an interrupt. If program level 4 is masked, program level 5 executes after setting an SVC L4 interrupt.

Resulting Condition Latches: Unchanged.

Cycle Sequence Table

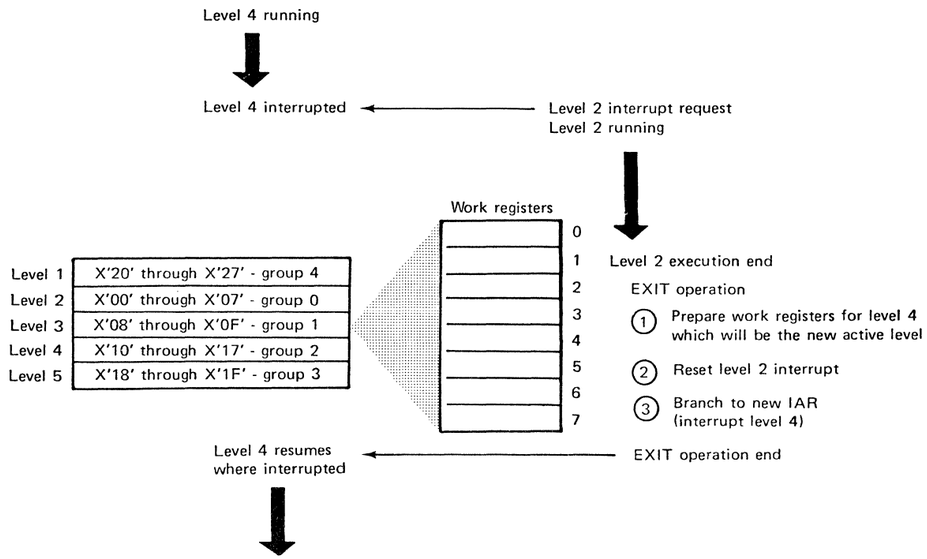
Level 1 to 4	Level 5	
1		General Reg 0 (1) --> WKR0 (IAR)
	1	No operation
2		General Reg 1 (1) --> WKR1
	2	No operation followed by int L4
3		General Reg 2 (1) --> WKR2
4		General Reg 3 (1) --> WKR3
5		General Reg 4 (1) --> WKR4
6		General Reg 5 (1) --> WKR5
7		General Reg 6 (1) --> WKR6
8		General Reg 7 (1) --> WKR7

(1) The new active interrupt level selects the general register group in local store.

Operation

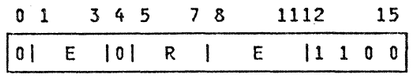
In the operation that follows, level 2 raises an interrupt request, which stops the running of level 3. Level 2 executes with no interruption until Exit is reached. Exit then executes, and level 3 resumes at the point where it was interrupted.

The interrupt mechanism is packaged on the MIOC card.



Register External (RE) Instructions (Part 1 of 2)

INPUT INSTRUCTION



This instruction loads the general register specified by the R field with the contents of the external register specified by the 7-bit E field. The general registers of program levels that are not running can be addressed as external registers.

The C and Z condition latches remain unchanged.

An input instruction can be executed at program levels 1, 2, 3, and 4 only. Any attempt to execute this instruction at program level 5 causes a level 1 input/output check (level 5 I/O error) to be set.

Note: If general register 0 (IAR) is specified as R, the operation results in a branch to the address formed in register 0.

The cycle sequence of the input instruction varies with the register that is specified by the E field. Instructions in the group A read the local store registers; those in group B read the hardware latches. Input X'70', X'7A', and X'74' work differently, and have their own cycle sequence.

	Register addresses
Group A	Local store: 00 to 6F, 71, 72 78, 7F, and 7C
Group B	Latches: 73, 75, 76, 77, 79 and 7D to 7F
Input '70'	Storage size bits
Input '74'	LAR contents
Input '7A'	High resolution timer bits

For the register bit meaning of the Input/Output X'7x' instruction, see pages 10-230 and 10-240.

Invalid Input Instructions

Any attempt to execute input X'28' through X'2F', X'49' through X'4F', X'60' through X'6F', and X'78' sets an error interrupt level 1. The instruction is not executed.

Cycle Sequence Tables

Group A				
R≠0 E≠IAR	R=0 E≠IAR	E=IAR	E=IAR	
1	1			(LS 'E' --> WSDR
2				(WSDR --> R
	2			(WSDR --> IAR branch
		1		(IAR --> R
			1	(IAR -->

E=00 (IAR) in program level 2
E=08 (IAR) in program level 3
E=10 (IAR) in program level 4
E=18 (IAR) in program level 5
E=20 (IAR) in program level 1

Cycle Sequence

Group B			
R≠0	R=0		
1	1	zeros	--> Z reg byte X
		latches	--> Z reg bytes 0, and 1
2		(Z reg)	--> R
	2	(Z reg)	--> IAR branch

Cycle Sequence

E='70'			
R≠0	R=0		
1	1	Cycle to prepare CCU circuits	
2	2	02	--> SAR (gate byte 0)
3		zeros	--> R bytes X, 1
3		STG size bits	--> R byte 0
	3	zeros	--> IAR/SAR bytes x, 1
	3	STG size bits	--> IAR/SAR byte 0 (branch on STG size)

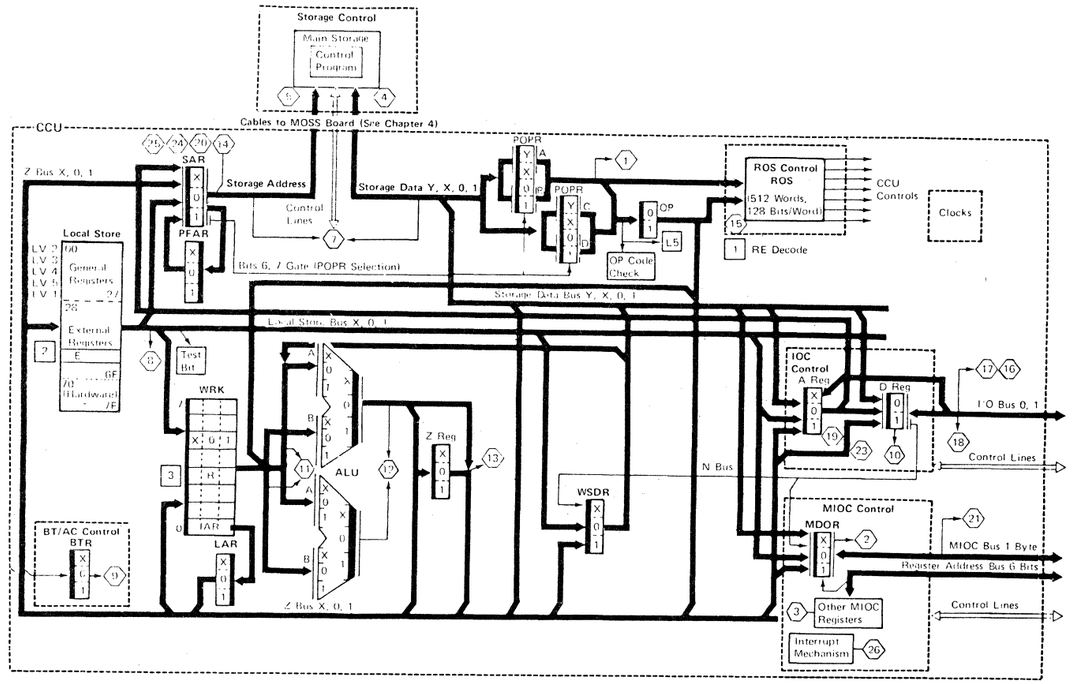
Cycle Sequence

E='7A'	
R≠0	R=0
1	1
2	
	2
	2

High resolution timer bits --> Z reg
(Z reg) --> R
(Z reg) --> SAR
(Z reg) --> IAR
(branch on timer value)

Note: Input X'74' requires only one cycle to place the LAR contents in the work register specified by R.

DATA FLOW FOR INPUT RE INSTRUCTION

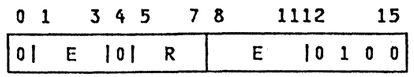


Operation

1. Decode RE instruction.
2. Read contents of local storage or hardware register specified by E.
3. Set the value read into R.

Register External (RE) Instructions (Part 2 of 2)

OUTPUT INSTRUCTION (RE)



This instruction places the contents of the general register specified by the R field in the external register specified by the 7-bit E field. See page 10-220 for the functions of the 128 external registers addressable via the output instruction. General registers of program levels that are not running can be addressed as external registers.

The C and Z condition latches remain unchanged.

An output instruction can be executed at program levels 1, 2, 3, or 4 only. An attempt to execute this instruction at program level 5 causes the level 1 input/output check (level 5 I/O error) to be set.

Note: If general register 0 (IAR) is specified by E, the operation results in a branch to the address formed in register 0.

The cycle sequence of the output instruction varies with the register that is specified by the E field. Instructions in groups A, B, and D are 1 cycle long. Cycle sequence tables follow for the longer output instructions.

	Register Addresses	Cycle
Group A	Local store: 00 to 67	1
Group B	Latches: 68,69,75,76,77 79,7A	1
Group C	Latches: 70, 7B, to 7F	3
Group D	Latches: 71, 72	1
Output'73'	Set SP/AE Keys	3
Output'74'	Control storage ECC	4
Output'78'	Force ALU check	2

Invalid Output Instructions

Any attempt to execute Output X'28' through X'2F', X'49' through X'4F', X'60' through X'6F', and X'75' sets an error interrupt level 1. The instruction is not executed.

Cycle Sequence Tables

R may specify the IAR (R=0); it does not change the number of cycles.

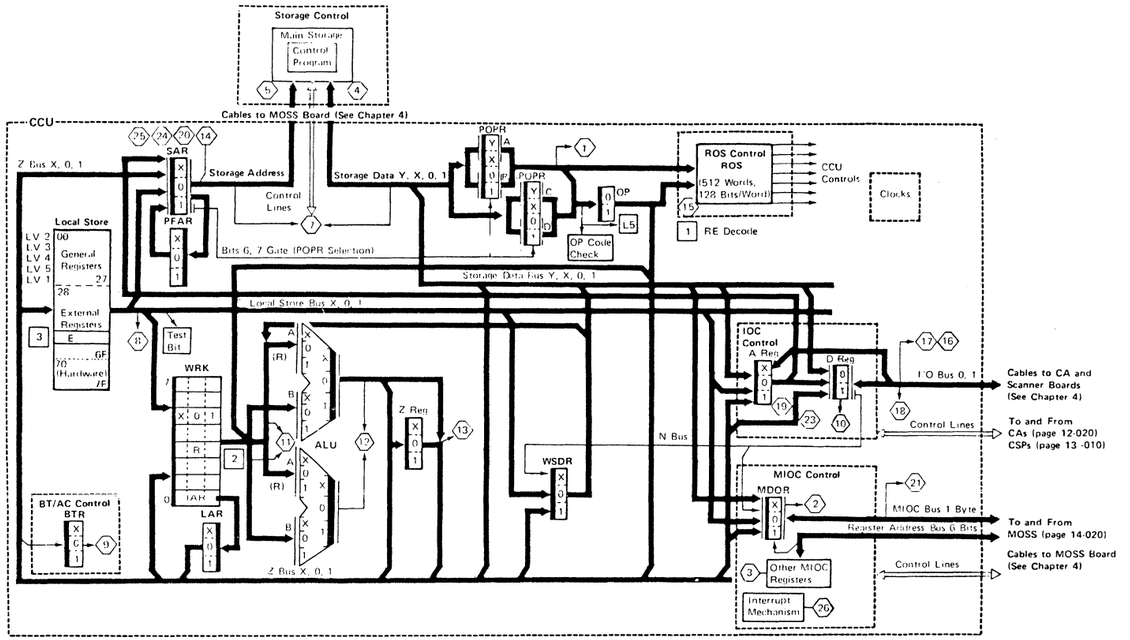
Group C	
1	(R) --> Z Bus (7E, 7F only)
2	Necessary idle cycles for setting
3	the interrupt mechanism, or program stop.

E='73'	Set SP/AE Keys
1	(R) --> Z Bus
2	SAR --> Storage key data reg
2	--> User key reg
3	Storage key data reg

E='74'	Control storage ECC
1	Prepare ALUA, ALUB
2	Zeros --> SAR
2	(LS'47') -->
3	Necessary idle cycles for timing reasons
4	

E='78'	Force ALU check
1	Force ALU compare check
1	Force parity error
2	Idle cycle to let hardstop
2	or interrupt level 1 propagate

DATA FLOW FOR OUTPUT RE INSTRUCTION



Operation

1. Decode RE instruction.
2. Read R.
3. Set (R) into the local storage or hardware register specified by E.

External Register Functions

The input/output registers are defined by the 'E' field of the CCU input/output instructions.

The local store array contains as many locations as can be addressed by the 'E' field of the input/output instructions.

For LSR locations '00-6F', the data obtained by an input instruction comes from the local store position designated by the 'E' field of the instruction. Similarly, the data set by an output instruction goes into the local store array location designated by the 'E' field.

However, for In X'70', X'73-7A', and X'7D-7F', the data are taken from various hardware registers and jumpers in the CCU. For Out X'70', and X'73-7F', the instruction sets/resets a control function or a hardware register in the CCU. In these cases, the local store array is not used. These points are shown in the following charts as 'HW' or control. Note also that Out X'71' and X'72' store data into LSR locations X'79' and X'7A' respectively.

As a result, the data obtained in these cases by displaying the LSR using is meaningless, and does not correspond to the input/output instruction operand location.

The information obtained by instruction In X'71' through In X'7F' can be displayed with the "I" function of display under "CCU". See 3725 Problem Determination and Extended Services.

Register X'70' cannot be displayed using the 'I' function. It can be displayed using the 'L' function, but always shows X'0000' and should be ignored. Use the CDF display to determine storage size.

Example 1:

In X'40' (interrupt start address level 1). The E field of this instruction corresponds to LSR 40.

Example 2:

In X'74' (lagging address register). The E field of this instruction designated the CCU lagging address register (HW) and not the LSR 74.

Example 3:

Out X'7B' (set PCI level 2). The E field of this instruction causes CCU level 2 interrupt to raise (control).

INPUT REGISTERS

E Field (Note 1)	Register Functions	LS Address
00 - 07	General register group 0 (interrupt level 2)	00 - 07
08 - 0F	General register group 1 (interrupt level 3)	08 - 0F
10 - 17	General register group 2 (interrupt level 4)	10 - 17
18 - 1F	General register group 3 (interrupt level 5)	18 - 1F
20 - 27	General register group 4 (interrupt level 1)	20 - 27
28 - 2F	Invalid register selection	28 - 2F
30	CS address pointer register for CA 1	30
31	CS address pointer register for CA 2	31
32	CS address pointer register for CA 3	32
33	CS address pointer register for CA 4	33
34	CS address pointer register for CA 5	34
35	CS address pointer register for CA 6	35
36 - 3E	Pointer registers 6, 7, ..., E	36 - 3E
3F	CS address pointer register for scanner	3F
40	Interrupt start address - Level 1	40
41	Interrupt start address - Level 2	41
42	Interrupt start address - Level 3	42
43	Interrupt start address - Level 4	43
44	Substitution register for operations on character	44
45	Substitution register for operations on halfword	45
46	Substitution register for operations on fullword	46
47	CCU storage control register on ECC	47
48	IOH TA substitution register	48
49 - 4F	Invalid register selection	49 - 4F
50 - 5F	Invalid register selection	50 - 5F
60 - 67	Invalid register selection	60 - 67
68	Zero register	68 (Note 2)
69	Holding register for IOH, IOHI, and BAL instruct.	69 (Note 2)
6A	Holding register for MIOH	6A (Note 2)
6B	Holding register for IOH	6B (Note 2)
6C - 6F	Not used	6C - 6F
70	Storage size installed	HW
71	Operator address/data entry register	71
72	Operator function select control	72
73	Insert storage protect/address exception key	HW
74	Lagging address register	HW
75	CCW for AIO operations	HW
76	Adapter level 1 interrupt requests	HW
77	Adapter level 2, 3, or 4 interrupt requests	HW
78	Invalid register selection	HW
79	Utility register	HW
7A	High resolution timer/utilization counter	HW
7B	Branch trace address pointer	7B
7C	Branch trace buffer count	7C
7D	CCU hardware check register	HW
7E	CCU level 1 interrupt requests	HW
7F	CCU level 2, 3, or 4 interrupt requests	HW

OUTPUT REGISTERS

E Field (Note 1)	Register Functions	LS Address
00 - 07	General register group 0 (interrupt level 2)	00 - 07
08 - 0F	General register group 1 (interrupt level 3)	08 - 0F
10 - 17	General register group 2 (interrupt level 4)	10 - 17
18 - 1F	General register group 3 (interrupt level 5)	18 - 1F
20 - 27	General register group 4 (interrupt level 1)	20 - 27
28 - 2F	Invalid register selection	28 - 2F
30	CS address pointer register for CA 1	30
31	CS address pointer register for CA 2	31
32	CS address pointer register for CA 3	32
33	CS address pointer register for CA 4	33
34	CS address pointer register for CA 5	34
35	CS address pointer register for CA 6	35
36 - 3E	Pointer registers 6, 7, ... E	36 - 3E
3F	CS address pointer register for scanner	3F
40	Interrupt start address - level 1	40
41	Interrupt start address - level 2	41
42	Interrupt start address - level 3	42
43	Interrupt start address - level 4	43
44	Substitution register for operations on character	44
45	Substitution register for operations on halfword	45
46	Substitution register for operations on fullword	46
47	CCU storage control register on ECC	47
48	IOH TA substitution register	48
49 - 4F	Invalid register selection	49 - 4F
50 - 5F	Invalid register selection	50 - 5F
60 - 67	Invalid register selection	60 -
68	Not used	68 (Note 2)
69	Not used	69 (Note 2)
6A	Not used	6A (Note 2)
6B	Not used	6B (Note 2)
6C - 6F	Not used	6C - 6F
70	Hardstop	Control
71	Program display register 1 (display A)	79
72	Program display register 2 (display B)	7A
73	Set storage protect/address exception keys	HW
74	Storage control register	HW
75	Invalid register selection	HW
76	Miscellaneous control 1	HW
77	Miscellaneous control 2	HW
78	Force ALU checks	Control
79	Utility register set	HW
7A	High resolution timer/utilization counter control	HW
7B	Set program-controlled interrupt - level 2	Control
7C	Set program-controlled interrupt - level 3	Control
7D	Set program-controlled interrupt - level 4	Control
7E	Set mask bits	HW
7F	Reset mask bits	HW

Notes:

1. Attempts to select an invalid register with an in or out instruction set an error interrupt level 1. The instruction is not executed.
2. Attempts to select LS68 to LS6B with an In or Out instruction set an error interrupt level 1. The instruction is not executed. Nevertheless, these LS registers are used internally by some instructions. For details, see page 10-150.

Input X'7X'- Register Bits (Part 1 of 2)

For notes, see following page.

	X'70' Storage Size Installed	X'71' Operator Address Data Entry	X'72' Operator Display Function Select	X'73' Read SP/AE Key
Byte X, Bit 2	Not used	Addr/data X.2	Not used	Not used
3	Not used	Addr/data X.3	Not used	Not used
4	Not used	Addr/data X.4	Not used	Not used
5	Not used	Addr/data X.5	Not used	Not used
6	Not used	Addr/data X.6	Not used	Not used
7	Not used	Addr/data X.7	Not used	Not used
Byte 0, Bit 0	1=MEM-1 to -8 are 256K	Addr/data 0.0	F. select 8	0
1	1=MEM-9 to -12 are 256K	Addr/data 0.1	F. select 9	0
2	Not used	Addr/data 0.2	F. select 10	0
3	1= 2048K bytes (Note 7)	Addr/data 0.3	F. select 11 (stg addr)	0
4	1= 1024K bytes (Note 7)	Addr/data 0.4	F. select 12 (reg addr)	0
5	1= 512K bytes	Addr/data 0.5	F. select 13	0
6	1= 256K bytes (Note 5)	Addr/data 0.6	F. select 14	0
7	1= 128K bytes (Note 6)	Addr/data 0.7	F. select 15	0
Byte 1, Bit 0	0	Addr/data 1.0	F. select 16	0
1	0	Addr/data 1.1	F. select 1	0
2	0	Addr/data 1.2	F. select 2	0
3	0	Addr/data 1.3	F. select 3	0
4	0	Addr/data 1.4	F. select 4	0
5	0	Addr/data 1.5	F. select 5	Key Value 0
6	0	Addr/data 1.6	F. select 6	Key Value 1
7	0	Addr/data 1.7	F. select 7	Key Value 2

	X'74' Lagging Address Register (LAR)	X'75' AIO CSCW	X'76' IOC Level 1 Interrupt Request	X'77' Adapter Level 2, 3 Int. Request
Byte X, Bit 2	LAR X.2	Not used	Not used	Not used
3	3	Not used	Not used	Not used
4	4	Not used	Not used	Not used
5	5	Not used	Not used	Not used
6	6	Not used	Not used	Not used
7	7	Not used	Not used	Not used
Byte 0, Bit 0	LAR 0.0	IOC CSCW bit 5 (Note 1)	IOC add except (Note 3)	0
1	1	IOC CSCW bit 11 (Note 2)	IOC STG protect (Note 3)	CSP L2
2	2	IOC CSCW bit 12 (Note 2)	IOC Invalid CCW (Note 3)	0
3	3	IOC CSCW bit 13 (Note 2)	0 (Note 3)	0
4	4	IOC CSCW bit 14 (Note 2)	IOC timeout	0
5	5	0	IOC bus in parity error	0
6	6	0	Adapter init operation	0
7	7	0	MOSS init operation	0
Byte 1, Bit 0	LAR 1,0	0	0	CA L3
1	1	0	0	0
2	2	0	0	0
3	3	0	0	0
4	4	0	0	0
5	5	0	0	0
6	6	0	0	0
7	7	0	0	0

	X'79' Utility	X'7A' High/Low Resolution Timer	X'7B' Branch/Trace Address Pointer	X'7C' Branch/Trace Buffer count
Byte X, Bit 2	Not used	Timer bit 0	B/T address X.2	Not used
3	Not used	Timer bit 1	3	Not used
4	Not used	Timer bit 2	4	Not used
5	Not used	Timer bit 3	5	Not used
6	Not used	Timer bit 4	6	Not used
7	Not used	Timer bit 5	7	Not used
Byte 0, Bit 0	0	Timer bit 6	B/T address 0.0	0
1	0	Timer bit 7	1	0
2	0	Timer bit 8	2	0
3	0	Timer bit 9	3	0
4	0	Timer bit 10	4	0
5	0	Timer bit 11	5	0
6	Prog L5-C card	Timer bit 12	6	0
7	Prog L5-Z card	Timer bit 13	7	0
Byte 1, Bit 0	Prog L2 interrupted (Note 4)	Timer bit 14	B/T address 1.0	0
1	Prog L3 interrupted (Note 4)	Timer bit 15	1	0
2	Prog L4 interrupted (Note 4)	Timer bit 16	2	0
3	Prog L5 interrupted (Note 4)	Timer bit 17	3	0
4	0	Timer bit 18	4	0
5	0	Timer bit 19	5	Ignored
6	0	Timer bit 20	6	Ignored
7	0	Timer bit 21	7	Ignored

	X'7D' CCU Hardware Errors	X'7E' Level 1 Interrupt Requests	X'7F' CCU Level 2, 3, 4 Interrupt Requests
Byte X, Bit 2	Not used	Not used	Not used
3	Not used	Not used	Not used
4	Not used	Not used	Not used
5	Not used	Not used	Not used
6	Not used	Not used	Not used
7	Not used	Not used	Not used
Byte 0, Bit 0	POP parity error	MOSS inoperative	PCI L2
1	MDOR parity error	Any CCU hard error	MOSS diag L2
2	MIOC parity error		MOSS diag L3
3	Storage 2-bit error	L5-I/O error	MOSS req SVC L4
4	Storage control error	Invalid operation	MOSS resp SVC L4
5	Ignored	IOC Adapt L1 request	0
6	STG add/data parity		User int req L3
7	LS parity error	IOC-L1 summary	PCI L4
Byte 1, Bit 0	0	Add compare L1	0
1	A/B bus parity error	Add except-inst fetch	0
2	D1 reg parity error	STG protect-inst fetch	0
3	ALU compare error	Add except-PGM exec	0
4	SAR parity error	STG protect-PGM exec	0
5	ROS parity error	Ignored	Internal timer L3
6	Z reg parity error	IPL L1	PCI L3
7	Ignored	Ignored	SVC L4

Input X'7X' - Register Bits (Part 2 of 2)

Notes:

1. Bit 0.0 = 0 indicates an AIO from channel adapter.
   Bit 0.0 = 1 indicates an AIO from scanner.

2. Bit 0.0 = 0 bits 0.1 to 0.4 show the CA pointer number:

0000 = CA#1
0001 = CA#2
0010 = CA#3
0011 = CA#4
0100 = CA#5
0101 = CA#6

   Bit 0.0 = 1 bits 0.1 to 0.4 show the line address group:

Bit Value	Line Address	LAB Type A	LAB Type B	LAB Name
0000	0-15	Scanner 1		CLAB1
0001	16-31			
0010	32-47	Scanner 3		CLAB2
0011	48-63	_		
0100	64-79	Scanner 5	Scanner 5	LAB
0101	80-95	_	Scanner 6	pos 3
0110	96-111	Scanner 7	Scanner 7	LAB
0111	112-127	_	Scanner 8	pos 4
1000	128-143	Scanner 9	Scanner 9	LAB
1001	144-159	_	Scanner 10	pos 5
1010	160-175	Scanner 11	Scanner 11	LAB
1011	176-191	_	Scanner 12	pos 6
1100	192-207	Scanner 13	Scanner 13	LAB
1101	208-223	_	Scanner 14	pos 7
1110	224-239	Scanner 15	Scanner 15	LAB
1111	240-255	_	Scanner 16	pos 8

3. If byte 0, bits 4 and 5 are zero, bits 0, 1, 2 and 3 are as indicated.

   If byte 0, bit 4 or 5 is one, bits 0, 1, 2, and 3 contain the IOC internal status at the time of error. This IOC internal status table (right) indicates the tag signal that is valid and the meaning of bit 4 (IOC timeout) and bit 5 (IOC bus in parity error).

4. Only one bit is on. It indicates which program level was interrupted by level 1.

5. The intermediate sizes (256K-byte increments) are given by combinations of bits 0.3 through 0.6.

6. Bit 0.7 is set on in case the installed storage size is not a multiple of 256K.

7. For 3072K byte 0 bit 3 and 4 are on.

Tag Conversion Table

Tag Encoding	Tags
08	VH
10	IRR
11	IRR, EOC
16	IRR, VB, M
18	IRR, VH
19	R/W

IOC Internal Status Table

IOC Status	Tags (See Table)	Meaning of 'IOC timeout'	Meaning of 'IOC Bus In Parity Error'
0	08	I/O tag is off	No
1	10	I/O tag raised (new AIO or IOH)	No
2	18,11	No response to TA tag or cycle steal grant	No
3	19	CSCW parity error ('3x')	CSCW parity error ('3x')
4	10	VH did not fall after TD fell	AIO data read or CH pointer register read
5	18,16,11	No response to TD for AIO data read IOH data read	No
6	18,16,11	No response to TD for AIO data write IOH data write	No
7	10	VH did not fall after TD fell (byte boundary transfer with storage)	No
8	08	I/O tag is off, VH must rise (EOC after CH pointer updating in local store)	CH pointer reg read transfer last transfer (EOC)
9	10	VH did not fall after CG fell	No
A	08	I/O tag is off, VH must rise (last AIO transfer data read is on byte boundary)	No
B	10	No	Loading of CCW
C	18,11	No response to TD for AIO pointer initialization	No
D	-	No	No
E	10	VH did not fall after TD fell for AIO pointer initialization	CH pointer register read
F	08	I/O tag is off, VH must rise (IOH end or AIO end after data exchange)	Last AIO data read (EOC or VB and M)

Output X'7X' Register Bits

	X'71' Program Display Register 1 (Note 3)	X'72' Program Display Register 2 (Note 4)	X'73' Storage Protect Address Exception	X'76' Miscellaneous Control
Byte X, Bit 2	Display reg 1.X.2	Display reg 2.X.2	SKA bit 0	Ignored
3	Display reg 1.X.3	Display reg 2.X.3	SKA bit 1	Ignored
4	Display reg 1.X.4	Display reg 2.X.4	SKA bit 2	Ignored
5	Display reg 1.X.5	Display reg 2.X.5	SKA bit 3	Ignored
6	Display reg 1.X.6	Display reg 2.X.6	SKA bit 4	Ignored
7	Display reg 1.X.7	Display reg 2.X.7	SKA bit 5	Ignored
Byte 0, Bit 0	Display reg 1.0.0	Display reg 2.0.0	SKA bit 6	Reset IOC errors
1	Display reg 1.0.1	Display reg 2.0.1	SKA bit 7	Ignored
2	Display reg 1.0.2	Display reg 2.0.2	SKA bit 8	Ignored
3	Display reg 1.0.3	Display reg 2.0.3	SKA bit 9, UKA Bit 0	CCU Pgm Req.
4	Display reg 1.0.4	Display reg 2.0.4	SKA bit 10, UKA Bit 1	CCU Pgm Resp
5	Display reg 1.0.5	Display reg 2.0.5	UKA Bit 2	Ignored
6	Display reg 1.0.6	Display reg 2.0.6	UKA Bit 3	Ignored
7	Display reg 1.0.7	Display reg 2.0.7	UKA Bit 4	Ignored
Byte 1, Bit 0	Display reg 1.1.0	Display reg 2.1.0	Ignored	Reserved
1	Display reg 1.1.1	Display reg 2.1.1	Enabled SP/AE	Ignored
2	Display reg 1.1.2	Display reg 2.1.2	Key definit. (Note 1)	Ignored
3	Display reg 1.1.3	Display reg 2.1.3	Key definit. (Note 1)	Ignored
4	Display reg 1.1.4	Display reg 2.1.4	Modify key value	Ignored
5	Display reg 1.1.5	Display reg 2.1.5	Key value 0	Ignored
6	Display reg 1.1.6	Display reg 2.1.6	Key value 1	Ignored
7	Display reg 1.1.7	Display reg 2.1.7	Key value 2	Ignored

	X'7E' Set Program Interrupt Mask	X'7F' Reset Program Interrupt Mask
Byte X, Bit 2	Ignored	Ignored
3	Ignored	Ignored
4	Ignored	Ignored
5	Ignored	Ignored
6	Ignored	Ignored
7	Ignored	Ignored
Byte 0, Bit 0	Ignored	Ignored
1	Ignored	Ignored
2	Ignored	Ignored
3	Ignored	Ignored
4	Ignored	Ignored
5	Ignored	Ignored
6	Ignored	Ignored
7	Ignored	Ignored
Byte 1, Bit 0	Ignored	Ignored
1	Adap prog L1 req	Adap prog L1 req
2	Program L2 req	Program L2 req
3	Program L3 req	Program L3 req
4	Program L4 req	Program L4 req
5	Prog L5 execution	Prog. L5 execution
6	Ignored	Ignored
7	Ignored	Ignored

	X'77' Miscellaneous Control	X'79' Utility	X'7A' High/Low Resolution Timer
Byte X, Bit 2	Ignored	Ignored	Ignored
3	Ignored	Ignored	Ignored
4	Ignored	Ignored	Ignored
5	Ignored	Ignored	Ignored
6	Ignored	Ignored	Ignored
7	Ignored	Ignored	Ignored
Byte 0, Bit 0	Reset IPL L1		(Note 2)
1	Reset CCU hard checks		(Note 2)
2	Reset MOSS panel int req L3	Set prog IPL req	(Note 2)
3	Reset MOSS diag req L3		Ignored
4	Reset MOSS SVC req L4	Remote POWER OFF	Ignored
5	Reset MOSS SVC req L4	Int prog L5, C+Z	Ignored
6		Prog L5, C card	Ignored
7	Reset PCI L2	Prog L5, Z card	Ignored
Byte 1, Bit 0	Reset MOSS inop L1	Ignored	Ignored
1	Reset interrupt timer L3	Ignored	Ignored
2	Reset PCI L3	Set AIO stop mode	Ignored
3	Reset MOSS L2 diag req	Reset AIO stop mode	Ignored
4	Reset address comp L1	Set bypass CCU check stop	Ignored
5	Reset soft checks	Set bypass CCU check stop	Ignored
6	Reset PCI L4	Scope Sync. pulse 1 (Note 5)	Ignored
7	Reset SVC L4	Scope Sync. pulse 2 (Note 5)	Ignored

Notes:

1.

Bits 1.2	Bits 1.3	Meanings
0	0	User key
0	1	Storage key
1	0	Exception key
1	1	Read-only key

2. Bit 0.0 = 0 display
   Bit 0.0 = 1 enables count

   Bit 0.1 = 0 means high resolution
   Bit 0.1 = 1 means low resolution

   Bit 0.2 = 0 selects timer
   Bit 0.2 = 1 selects utilization counter

3. In the 3704/3705, program display register 1 is called 'display A'

4. In the 3704/3705, program display register 2 is called 'display B'

5. Location for scoping: Top card connector (crossover) on the U2 (MIOC) Card of CCU board, pin Y06 for sync. pulse 1, and pin Y08 for sync. pulse 2.

   Voltage levels:
   - When active: 0V (pulse width is 225ns)
   - When not active: + 3.2V

   Scope sync. pulse(s) will be active during an output X'79' when the appropriate bit is on; see also page 3-040.

Storage Protect/Address Exception (SP/AE)

MAIN STORAGE PROTECTION STATES

With the storage protect/address exception mechanism, a main storage position can be placed in any of the following states:

- Write free
- Write and instruction fetch controlled
- Read only
- Write/read forbidden

The storage size changes with the customer needs. The figure below shows to which storage positions the storage protect/address exception states apply.

0000	Storage always present	Storage protection: • Write free • Write and instruction fetch controlled • Read only
512K		
768K	Optional storage present	
2048K	Optional storage Not present	Address exception: • Write/read forbidden
3072K	Optional storage Not present	• Write/read forbidden
4096K	Storage never present	

SP/AE KEY TYPES

The SP/AE mechanism needs four different keys to control all read/write operations in storage.

- Storage Protect Key (SPK)

  This key determines the key value for writing in a defined 2K-byte block of storage.

- User Key (UK)

  Every user is assigned a 3-bit register that holds the storage protect key it must use for writing in storage and fetching instructions for execution.

  There are 21 possible storage users, classified in two types:

  - The five CCU interrupt handlers (UKL1 to UKL5)
  - 16 other programs (UKP0 to UKPF)

- Read-Only Key (ROK)

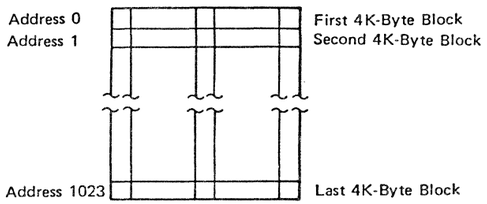
  This key indicates whether a 2K-byte block of storage is in the read-only state. For example, machine configuration data is placed in such a storage block.

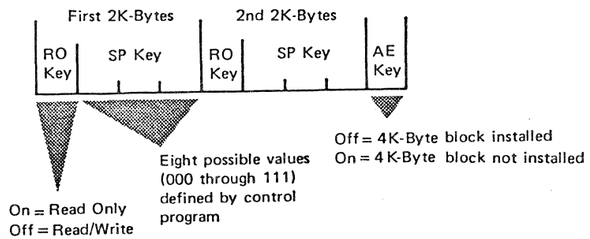
- Address Exception Key (AEK)

  This key indicates whether a 4K-byte block of storage is installed or not.

SP/AE STORAGE

The different keys are stored in a dedicated SP/AE storage located on the CTL2 card. Any entry to this storage (9 bits) holds the keys assigned to a 4K-byte block of main storage.





SP/AE INSTRUCTIONS

The CCU controls the SP/AE mechanism with the input X'73' and output X'73' instructions.

- The output X'73' allows the modification or initialization of the SP/AE storage. It enables the SP/AE mechanism.
- The input X'73' instruction reads out the last-set SPK from the SKDR or UKDR.

ERROR HANDLING

- Address Exception Key

  The user tries to address a main storage position not installed. The address exception bit is set on in the SP/AE storage.

  If the address exception was originated by the CCU, it causes a level 1 interrupt to the CCU. If the address exception was originated by the MOSS, it causes a level 1 interrupt to the MOSS with the AE bit on in the CCU to MOSS status register.

- Read-Only Key

  The user tries to write into a read-only main storage position. The read-only key bit is on in the SP/AE storage.

  This causes a level 1 interrupt to the CCU with the hardware check biton. Further input X'7E', X'7D', or X'76' instructions indicate the reason for storage protection error.

- Storage Protect Key

  The user tries to write into a main storage position, but the user key does not match with the storage protection key.

  This causes a level 1 interrupt to MOSS with hardware check bit on. Further input X'7E', X'7D', or X'76' instructions indicate the reason for storage protection error.

- User Key 000

  MOSS has the user key 000. It allows reading and writing in any main storage position that is present in the controller.

CCU Timers

Two timers are available in the CCU. They are the 100-ms interval timer and the high/low resolution timer.

100-MS INTERVAL TIMER

Every 100 ms, this timer requests a CCU level 3 interrupt. Output X'77', byte 1, bit 1 on is used to reset the timer interrupt. The 100-ms timer is used to:

- Count real time in storage
- Perform long timeouts
- Perform supervisory functions on a cyclic basis

HIGH/LOW RESOLUTION TIMER

Timer Operation

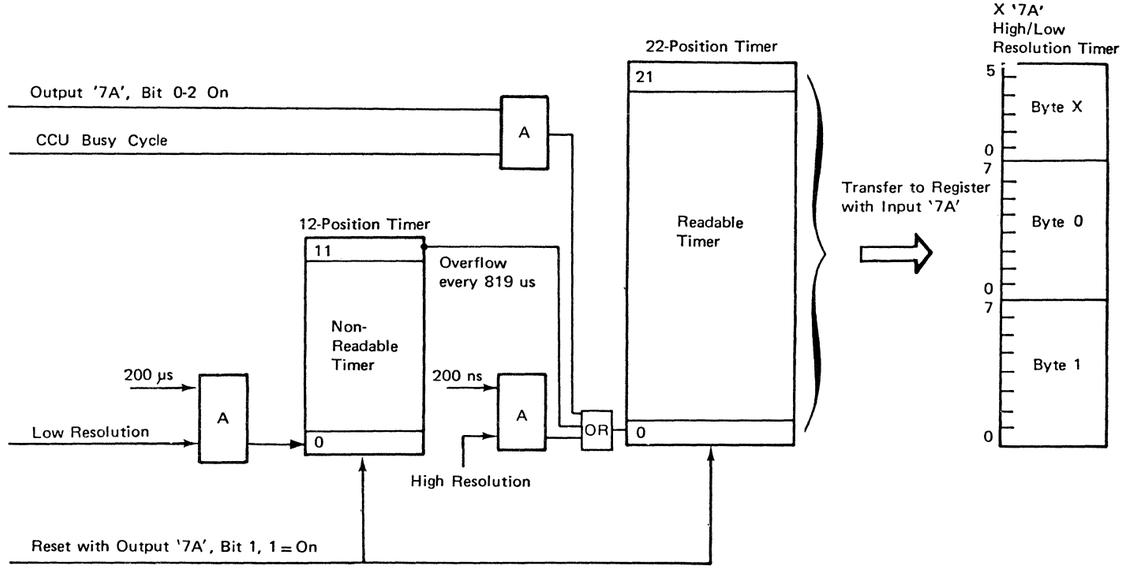
The high/low resolution timer does not raise any interrupts. It is driven by CCLK pulses (every 200 ns). Output X'7A' initializes the current timer value to X'00'; input X'7A' reads the current timer value.

The timer may work in high resolution, with a bit advance at every 200-ns clock pulse; the maximum time value is then 838 ms.

The timer may also work in low resolution, with a bit advance every 819 us (overflow from the internal non-readable timer). The maximum time value is then 57.2 minutes.

Cycle Counter Operation

Output X'7A' with byte 0, bit 2 on causes the timer to count 'CCU busy cycles'. The CCU is considered to be busy except when it is in the wait state, or during MIOH/MIOHI operations.



Chapter 10. Central Control Unit

Section 2. Troubleshooting Guidelines

DC Voltages and Tolerances at Board Pin Level

Vdc	Vmin	Vmax	Ripple (max)
-12.0	-10.92	-13.20	0.45V p-p
-8.5	-7.73	-9.35	0.25V p-p
-5.0	-4.55	-5.50	0.15V p-p
-4.3	-4.19	-4.48	0.07V p-p
-1.5	-1.48	-1.56	0.03V p-p
+5.0 (Note 1)	+4.55	+5.50	0.20V p-p
+5.0	+4.75	+5.25	0.13V p-p
+8.5	+7.73	+9.35	0.35V p-p
+12.0	+10.92	+13.20	0.40V p-p
+12.0 (Note 2)	+11.40	+13.20	0.40V p-p
+24.0	+21.00	+27.60	0.30V p-p

Notes:

1. 02-PS7 only
2. 01-PS4 only

CCU and Storage Troubleshooting Techniques (Part 1 of 2)

The purpose of this section is to help in isolating failures in the CCU and storage areas.

The following techniques are provided.

See figure on this page:

(A) CCU isolation: disconnect IOC bus (see page 10-800)

(B) CCU isolation: disconnect storage (see page 10-800)

See figure on facing page:

(C) CCU/storage clocking checks (see page 10-800)

(D) CCU storage refresh timing checks on board 01A-A1 (see page 10-810)

(E) Input X'70' scoping checks (see page 10-820)

(F) Storage control out tag: output X'74' (see page 10-825)

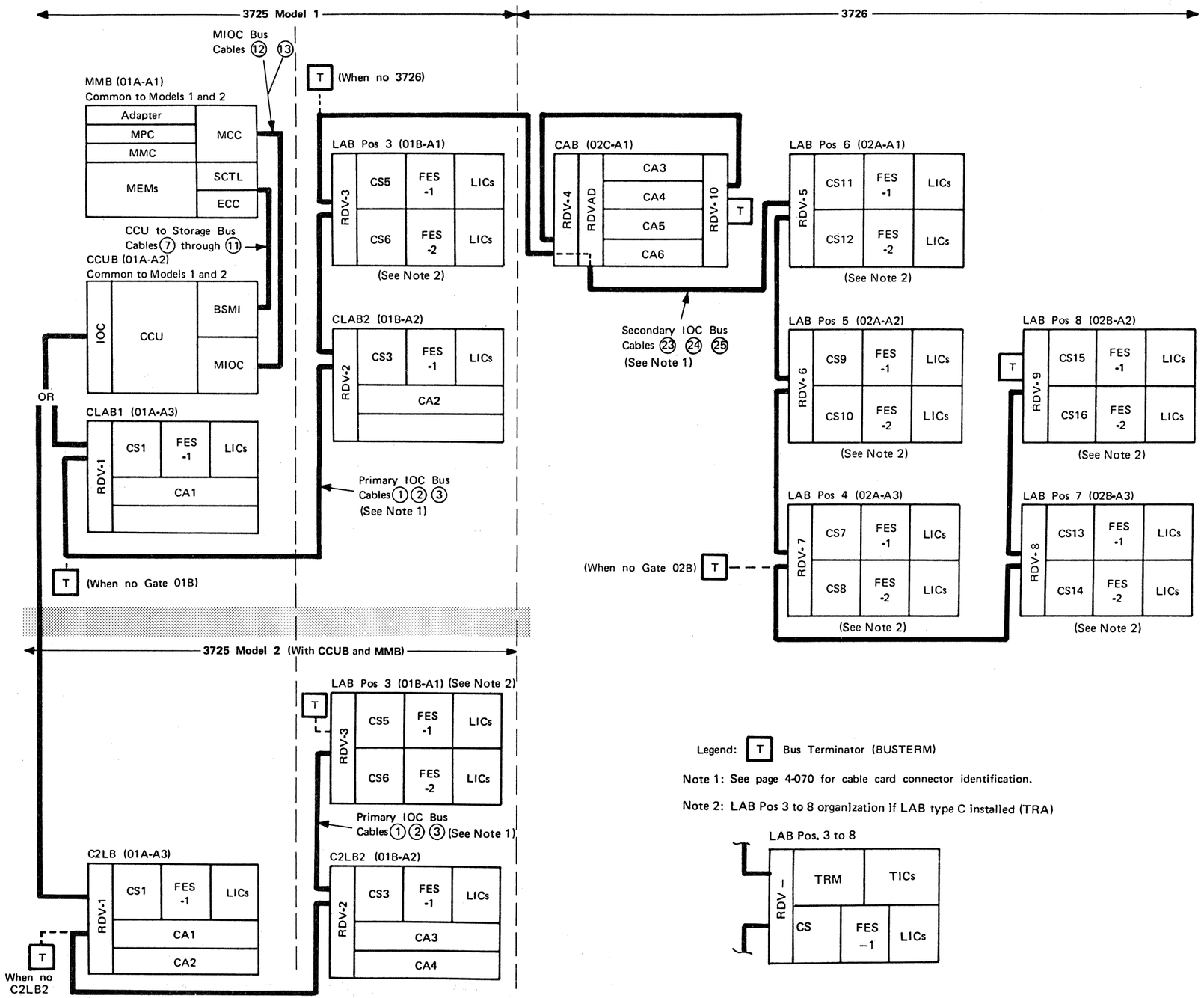
(G) CCU-to-storage bus scoping

- Running the routine (see page 10-830)
- Signal characteristics for the different modes (see page 10-831)
- Read storage timing on board 01A-A1 (see page 10-832)
- Read/modify/write storage timing on board 01A-A1 (see page 10-833)

(H) Storage signal routing (see page 10-840)

- Pin/net list (see page 10-841)

(I) Unexpected CCU interrupt processing: RAC 8FB/8FE (see page 10-850)

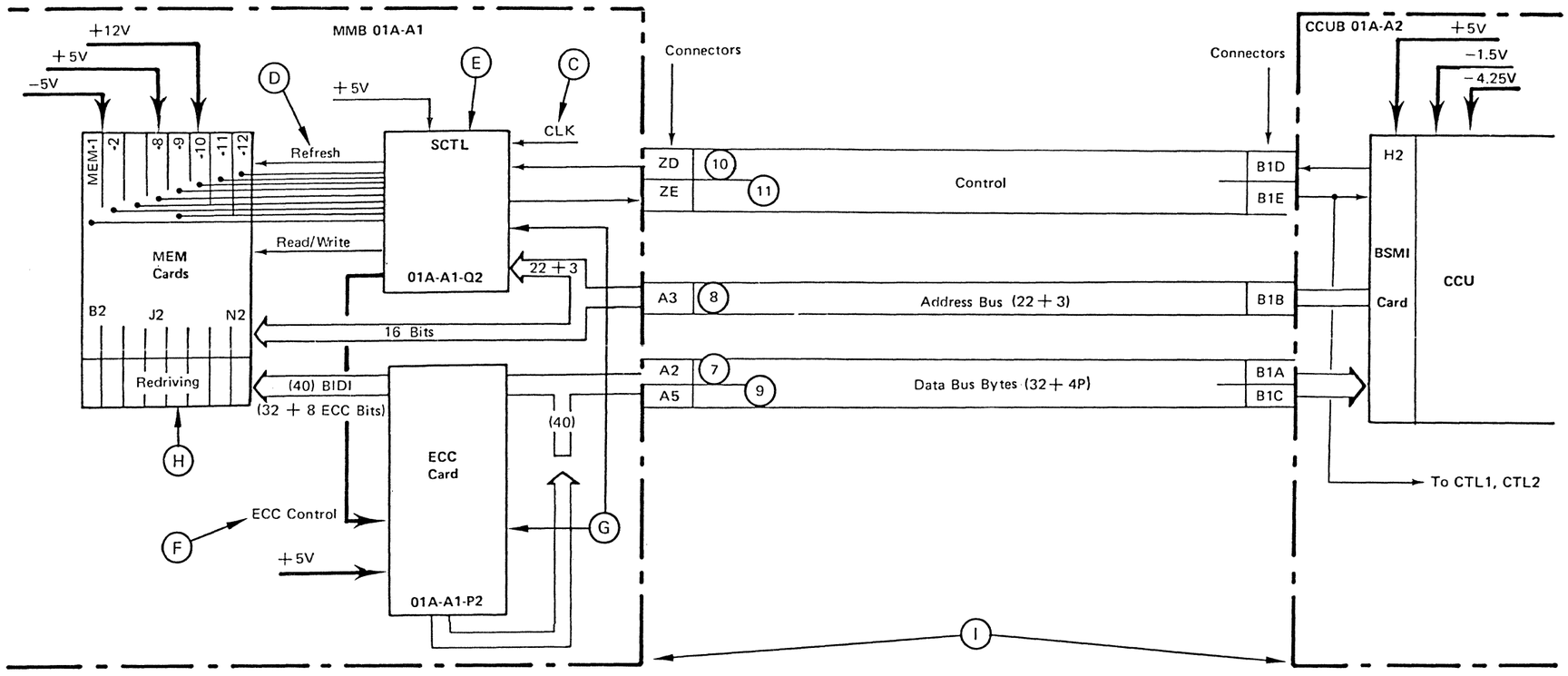


Legend: T Bus Terminator (BUSTERM)

Note 1: See page 4-070 for cable card connector identification.

Note 2: LAB Pos 3 to 8 organization if LAB type C installed (TRA)

CCU and Storage Troubleshooting Techniques (Part 2 of 2)



CCU Isolation and Clocking Checks (A) (B) (C)

CCU ISOLATION: DISCONNECT IOC BUS (A)

The CCU may be isolated from all its adapters by disconnecting the entire IOC bus as follows:

- Either unplug the CCU connector 01A-A2A3 (torque screwdriver required).
- Or unplug the IOC bus cables on gate connector 01A-J1 and also the top connectors of RDV-1 (01A-A3A2) (no tools required, but do not disconnect cable from CCU), and plug terminator card on connector 01A-J1.

WARNING: Special care must be taken when disrupting the IOC bus.

A CA that has been disconnected from the IOC bus is not necessarily disabled from the host channel side. It is therefore necessary to reset the CA(s) of each board that has been disconnected from the IOC bus by jumpering between:

01A-A3A2M12 and ground on CLAB-1 or C2LB for the 3725 Model 2

01B-A2A2M12 and ground on CLAB-2 or C2LB2 for the 3725 Model 2

02C-A1X2M12 and ground on CAB

The CCU diagnostics should now run error-free if there is no failure in the CCU and MMB boards.

CCU ISOLATION: DISCONNECT STORAGE (B)

The CCU storage may also be disconnected (see 3725/3726 MIM Part 2, 'CCU Replacement Procedure 3', Step 4). Proceed as follows:

1. Unplug the BSMI card at 01A-A2H2 and the SCTL card at 01A-A1Q2.
2. Install a jumper from 01A-A1Q2S07 to ground, and from 01A-A1Q2B12 to ground.

If the CCU is not failing, running the CCU diagnostics will raise the first error in routine AG01 (because this is the first routine to use the storage data and address buses). This routine tests the Z bus gating which includes gating storage data to the Z bus.

The storage disconnection may be useful in case of a permanent interruption to the MOSS, since it allows isolation of the interrupt between CCU and main storage.

Disconnecting the CCU Memory Cards

In order to determine if some errors are due to the storage itself (MEM cards) or to its interconnection with the CCU, you can unplug all the MEM cards and select a specific routine (refer to MIM Part 2, "CCU Replacement Procedure X"); see also Note.

Proceed as follows:

1. Unplug all the MEM cards, then:
2. If the CCU is not failing, the CCU diagnostic routine BJ23 (directly selected) will run error-free. In that case, a failure may be suspected on the MEM cards. The failing card may be isolated by plugging the MEM cards one by one while running BJ23 after each card plugging.

   If the CCU is failing, the CCU diagnostic routine BJ23 (directly selected) will detect an error and, in that case, the FRUs other than MEM cards may be suspected (they are given by the displayed RAC).

MEM card removal may be useful for isolating the failing card in the event of a permanent pollution of the CCU-to-storage interconnection by a MEM card.

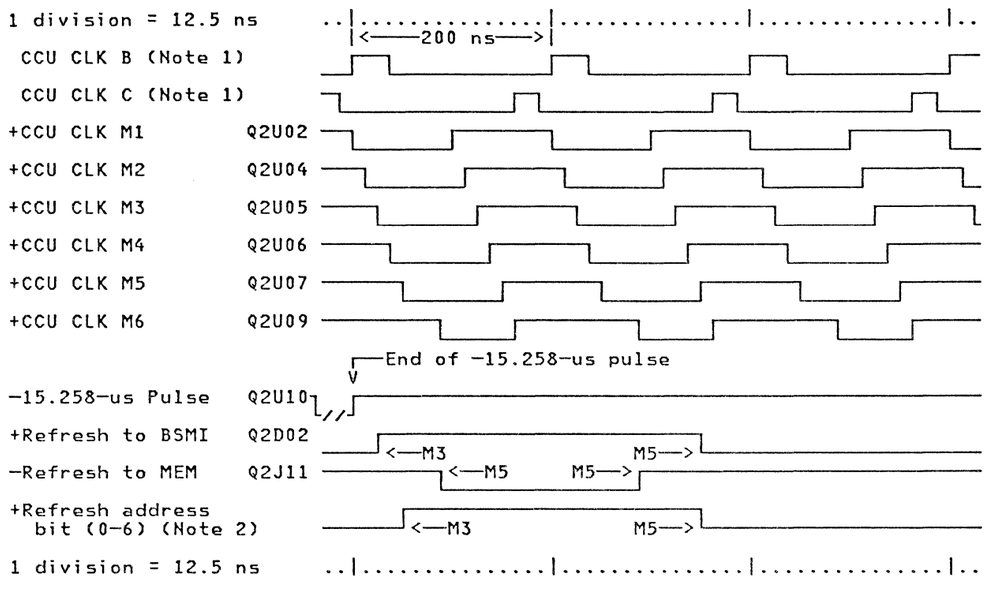
CCU/STORAGE CLOCKING CHECKS (C)

Six storage timings M1 through M6 generated by the CCLK card are sent to the SCTL card. Each of the timings M1 through M5 has an on/off duration of 100/100 ns. For M6, the duration is 125/75 ns. Delay between two successive M timings is 12.5 ns.

Note: When you unplug the MEM cards, be careful not to mix the two types of MEM cards (128 Kbytes or 256 Kbytes) that may be present on the machine.

CCU Storage Refresh Timing Checks (D)

The 3725 uses a capacitive type of storage which must be refreshed every 2 ms. The following text provides scoping points and timing charts for the storage refresh operation. The scoping is performed on board 01A-A1.



Legend:

+: between +3V and +5V
-: between 0V and 0.5V

Notes:

1. Not scopable.
2. There is one scopable point for each refresh address bit on the 01A-A1Q2 card, as follows:

Bit	Location
0	Q2D04
1	Q2D06
2	Q2D09
3	Q2D11
4	Q2G07
5	Q2G09
6	Q2P02

See the storage signal routing diagram on page 10-840 for the refresh bit scoping points on each storage card.

Input X'70' Scoping Checks (E)

The Input X'70' register gives the storage size and MEM card type installed (see page 10-230):

Storage Size (K bytes)	- Data Bus Byte 0 Bits 0	1	2	3	4	5	6	7
128	-	-	-	0	0	0	0	1
256	-	-	-	0	0	0	1	0
512	-	-	-	0	0	1	0	0
1024	-	-	-	0	1	0	0	0
2048	-	-	-	1	0	0	0	0
3072	-	-	-	1	1	0	0	0
MEM Positions	256K MEM cards							
MEM-1 through 8	1	-						
MEM-9 through 12	-	1						

When the SCTL gets the '-Input 70' line from the CCU (A1ZEB06 to A1Q2B13), the Input X'70' is generated as shown in the figure below. It shows the pin locations of the '-Card Location' (1) and '-Card Inserted' (2) signals and how they generate the 'Storage Data Bus Byte 0' (3) signals.

Card	(1)	(2)
1	B2J07	N/A
2	C2J07	Q2J10
3	D2J07	Q2J09
4	E2J07	Q2J07
5	F2J07	Q2D13
6	G2J07	Q2J05
7	H2J07	Q2M05
8	J2J07	Q2P13
9	K2J07	Q2D12
10	L2J07	Q2D10
11	M2J07	Q2D07
12	N2J07	Q2D05

==> SCTL logic ==>

(3)	Pins
0	Q2/P2X23 P2
1	Q2/P2X25 P2G10
2	Q2/P2X26 P2G12
3	Q2/P2X27 P2G13
4	Q2/P2X28 P2M02
5	Q2/P2X30 P2M03
6	Q2/P2X31 P2M04
7	Q2/P2X32 P2M05

Notes:

1. The '-Card Location' and '-Card Inserted' signals are permanently plus (approximately +5V) when the memory card is not installed, and permanently minus (approximately ground) when the memory card is installed. For all other signals, looping gives a 400-ns pulse every 3 ms.

2. For an Input X'70', no parity bit is generated for the data bus byte 0.

Storage Control Out Tag: Output X'74' (F)

The storage control out tag controls the ECC mechanism.

ECC MODES

The ECC mechanism may be set in the Disable, Enable, or Transparent mode.

- ECC Disable Mode

  This mode may be used to initialize the storage after a power on, or to read the real contents of storage.

- ECC Enable Mode

  This mode is the normal operational mode of the ECC mechanism. The ECC is enabled at power on time only after the entire storage has been scanned in the ECC Disable mode to align parity and ECC bits.

- ECC Transparent Mode

  This mode is for diagnostic purposes. It is used by the diagnostic routines to simulate single-bit and double-bit errors and to check that these errors are correctly handled by the ECC mechanism after it has been enabled.

Note: The ECC single-bit error correction and the ECC double-bit error detection circuits are activated only when the ECC is enabled.

DIAGNOSTIC FACILITIES

The following diagnostic facilities are available with the SCTL and ECC cards at EC A04406 and later. They are used by the diagnostics, or the CE when running the manual routine BI04 (see page 10-830).

Wrap-Up

This facility checks the address path and some control lines to the SCTL. The addresses and control lines are displayed on the data bus byte 0. This display is selected via the data bus byte X as shown in the following table. The following control lines are displayed:

Control Line	Byte 0
+ Single-bit error	Bit 0
- Input 70	Bit 1
+ Byte select 1	Bit 2
+ Byte select 0	Bit 3
+ Byte select X	Bit 4
+ Byte select Y	Bit 5
+ SP write inhibit	Bit 6
- Read (+ Write)	Bit 7

Force Hard Errors

This facility simulates single-bit or double-bit hard errors. One or two bit latches on the ECC card are forced to one or zero each time a memory card is read. These bits are X.7 and Y.7, as shown in the following table.

Force Hardware Checkers

This facility forces the three hardware checkers to their ON states, as shown in the following table. The "data/address parity" and "double-bit" checkers, when forced, are active without noticeable delay, while the "storage control" checker may become active as long as 6.6 ms after it has been forced.

Force ECC-Only

This facility allows "read" or "read/modify/write" operations with no memory card involved. The 36 latches (32 data plus 4 parity) are written and read by a normal write or read operation. The ECC card being transparent, the data bus shows during a read what was written during a write.

Output X'74' Command Summary

Out X'74' Command	Byte X 0 1 2 3 4 5 6 7	Action on Storage
ECC enable ECC disable ECC transparent	- - 0 0 . - - . - - 1 0 . - - . - - 1 0 . - - .	ECC enable mode ECC disable mode ECC transparent mode
Wrap-Up	1 0 - - . 0 0 . 1 0 - - . 0 1 . 1 0 - - . 1 0 . 1 0 - - . 1 1 .	Address byte X Address byte 0 Address byte 1 Control lines
Force Hard Errors	1 1 - - . 0 0 . 1 1 - - . 0 1 . 1 1 - - . 1 0 . 1 1 - - . 1 1 .	Force single-bit (Y.7 off) Force single-bit (Y.7 on) Force double-bit (Y/X.7 off) Force double-bit (Y/X.7 on)
Force Hardware Checkers	0 0 - - . 0 1 . 0 0 - - . 1 0 . 0 0 - - . 1 1 .	Force data/addr parity checker Force double-bit checker Force storage control checker
Force ECC only	0 1 - - . 0 0 .	Read or read/modify/write with no memory card involved
ECC enable	0 0 0 0 . 0 0 .	Normal operation

Notes:

1. Bit X.4 is a validation bit. It must be ON to update the latches that memorize the status.
2. Bit X.7 is not used.

CCU-to-Storage Bus Scoping (G) (Part 1 of 4)

The CCU manual intervention routine BI04 is designed to exercise the CCU to main storage data and control lines. This routine does not isolate an FRU, but allows looping with selected command, address, and data patterns for scoping. See pages 10-050 through 10-070 for a description of main storage.

Running the Routine

Proceed as follows:

1. Select the routine BI04. The message 'SELECT LINES TO SCOPE' is displayed.

2. Reply with

   'Rxx01aaaaaadddd' = Read/Modify/Write (Notes 1 and 5)

   'Rxx02aaaaaa' = Read (Notes 1, 2, and 5)

   'Rxx03' = Input X'70'

   'Rxx04yy' = Storage control out tag (Output X'74') (Notes 4 and 6)

   'Rxx05uu' = Storage control out tag (Output X'74') (Note 6)

   where xx is the request type:

   01 single run with error checking
   02 loop with error checking until an error is detected
   03 loop without error checking

   yy is the ECC sequence to be used:

   01 ECC disable
   02 ECC enable
   03 ECC transparent
   04 ECC disable, enable, disable
   05 ECC disable, transparent, enable, disable
   06 ECC disable, enable, transparent, disable

   aaaaaa is the storage address to be used (read and read/modify/write only)

   dddd is the write data to be used for read/modify/write.

   uu is the byte X value to be used with the output X'74' (see Note 6). The table on page 10-825 gives the possible byte X configuration meanings.

   The validity of the address is verified. The data pattern is set onto the data bus according to the position of the address X'aaaaaa' with respect to the word and halfword boundaries, as follows:

Boundary	Data is on bytes	Address Ending Digit
Word	Y and X	0-4-8-C
Halfword	0 and 1	2-6-A-E

   All other combinations cause the messages 'INVALID REQUEST' and 'PRESS SEND' to be displayed. Pressing SEND causes the 'SELECT LINES TO SCOPE' message to be redisplayed.

3. Looping continues until the ATTN key is pressed; the 'BREAK RECEIVED' message is then displayed. Typing G now returns to display the first request, C cancels the routine, and A aborts it.

Notes:

1. ECC is enabled for read and read/modify/write requests. The ECC can be put in disable or transparent mode via R010401 or R010403 requests.

2. Data read from storage with read request may be chosen by using a single read/modify/write prior to loop on read

3. If error checking is requested and an error is detected, RAC = 89F is displayed.

4. If the Storage Control Out Tag function is selected, there are two restrictions on the request type (xx) because error checking is not possible:

   a. xx = 01 - no error checking is done after a single run.

   b. xx = 02 is not available.

5. During a read/modify/write operation, both data and storage error conditions are checked by the program when xx = 01 or 02. During read operations, the data is not checked, only the storage error conditions.

   If the checkers are in error, the ERC = B1040701, and ERRBIT contains the value read by the Input X'7D' instruction (see page 10-230 for the bits of the Input X'7x' instructions).

6. Rxx04yy and Rxx05uu are mutually exclusive. The first one applies to ECC cards prior to EC A04406, the second one to ECC cards at EC A04406 or later. Using the wrong reply leads to an "invalid input" message.

CCU-to-Storage Bus Scoping (G) (Part 2 of 4)

Note: The information on this page does not apply to machines with an ECC card at EC A04406 or later.

SIGNAL CHARACTERISTICS FOR THE DIFFERENT MODES

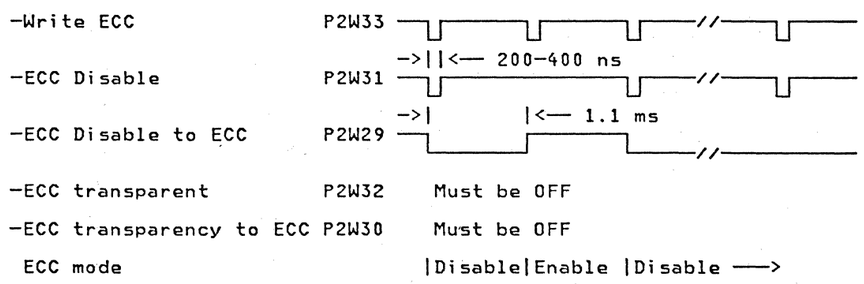
R030401, R030402, and R030403

Signal Name	Pin	Disable Mode (R030401)		Enable Mode (R030402)		Transp. Mode (R030403)	
	01A -A1	Period ms	Length ns	Period ms	Length ns	Period ms	Length ns
-Storage ctrl tag from BSMI	Q2U11	3	600	3	600	3	600
-Mem data bus byte X bit 2	P2G02	3	50-200	(Not used)		(Not used)	
-Mem data bus byte X bit 3	P2G03	(Not used)		(Not used)		3	50-200
-Mem data bus byte X bit 4	P2G04	3	50-200	3	50-200	3	50-200
-ECC disable	P2W31	3	400-800	(Not used)		(Not used)	
-ECC transparent	P2W32	(Not used)		(Not used)		3	400-800
-Write ECC	P2W33	3	400-800	3	400-800	3	400-800
-ECC disable to ECC	P2W29	Must be ON		Must be OFF		Must be OFF	
-ECC transparency to ECC	P2W30	Must be OFF		Must be OFF		Must be ON	

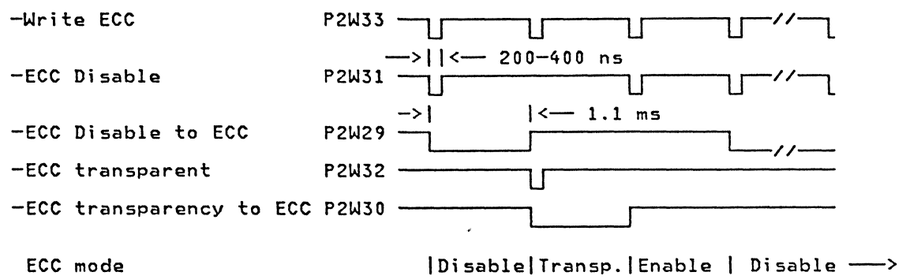
Note: For an Output X'74', no parity bit is generated for the data bus.

On other selections, the waveforms are as follows:

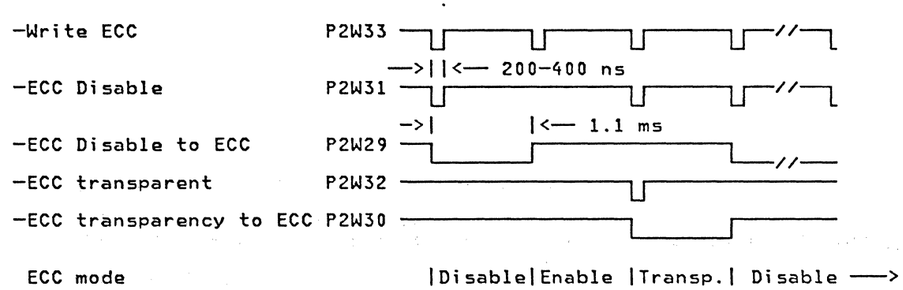
R030404



R030405

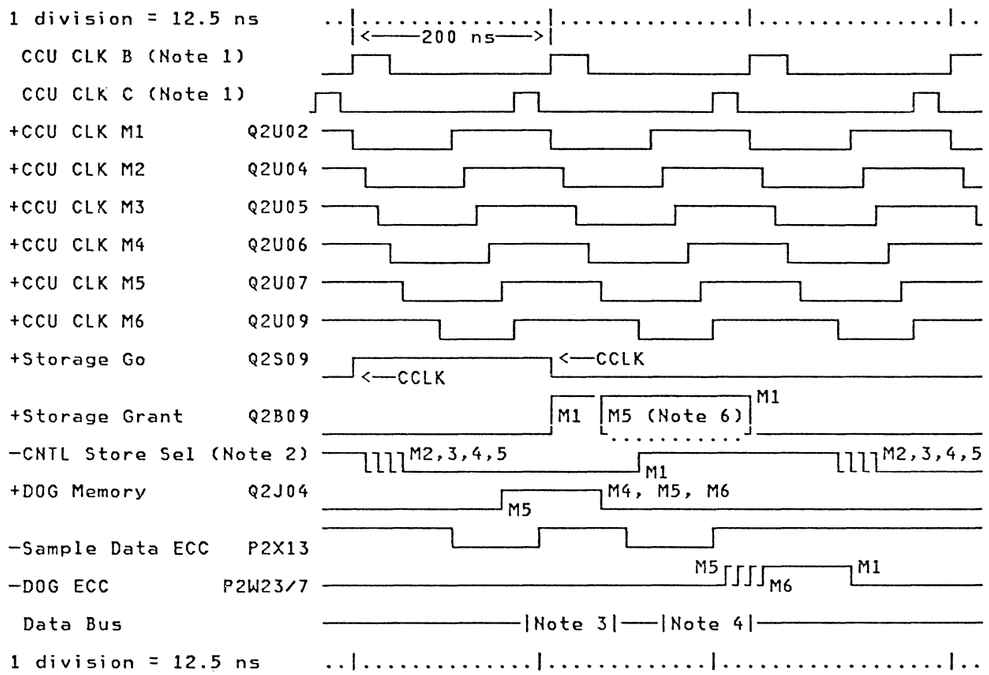


R030406



CCU-to-Storage Bus Scoping (G) (Part 3 of 4)

Read Storage Timing on Board 01A-A1



Legend:

+: between +3V and +5V
-: between 0V and 0.5V

Notes:

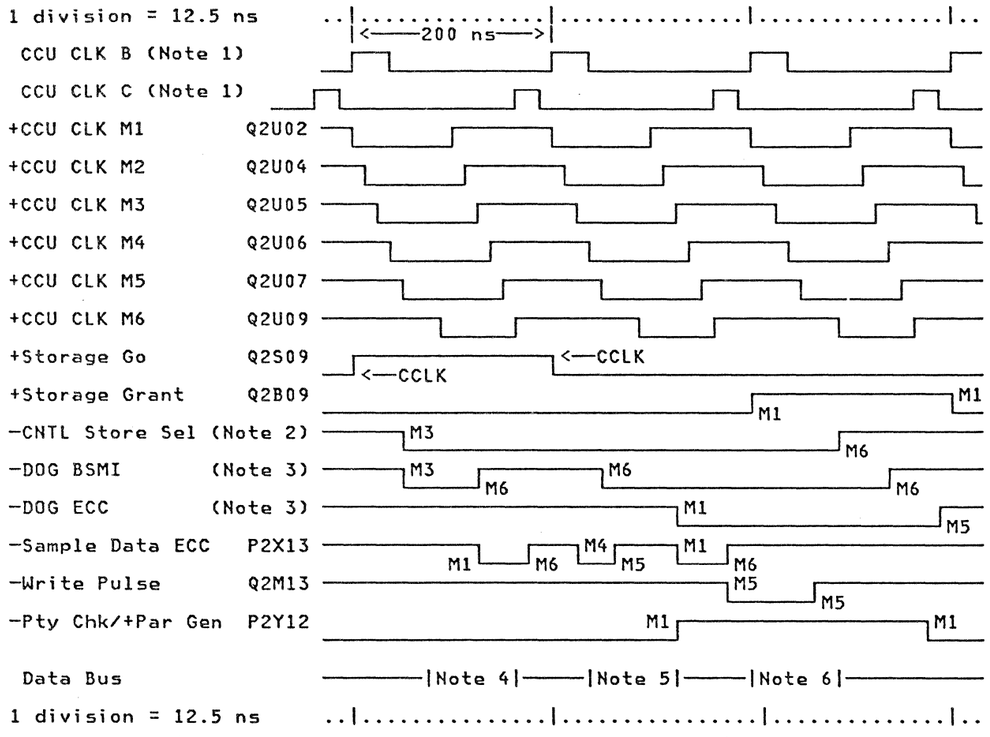
1. Not scopable.
2. There is a scopable point for each CNTL Store Sel (-Card Select) on the 01A-A1Q2 card, as follows:

Card	Location
0	Q2M10
1	Q2M12
2	Q2P06
3	Q2S13
4	Q2S12
5	Q2S10
6	Q2P11
7	Q2P12
8	Q2M02
9	Q2G12
10	Q2G10
11	Q2G08

   See the storage signal routing diagram on page 10-840 for other scoping points on this board.
3. Data from storage.
4. Data from ECC.
5. There may be a delay of up to 30 ns between the clock triggering the signal, and the signal itself.
6. If a double-bit error is detected in normal read.

CCU-to-Storage Bus Scoping (G)(Part 4 of 4)

Read/Modify/Write Storage Timing on Board 01A-A1



Legend:

+: between +3V and +5V
-: between 0V and 0.5V

Notes:

1. Not scopable.
2. There is a scopable point for each CNTL Store Sel (-Card Select) on the 01A-A1Q2 card, as follows:

Card	Location
0	Q2M10
1	Q2M12
2	Q2P06
3	Q2S13
4	Q2S12
5	Q2S10
6	Q2P11
7	Q2P12
8	Q2M02
9	Q2G12
10	Q2G10
11	Q2G08

   See the storage signal routing diagram on page 10-840 for other scoping points on this board.

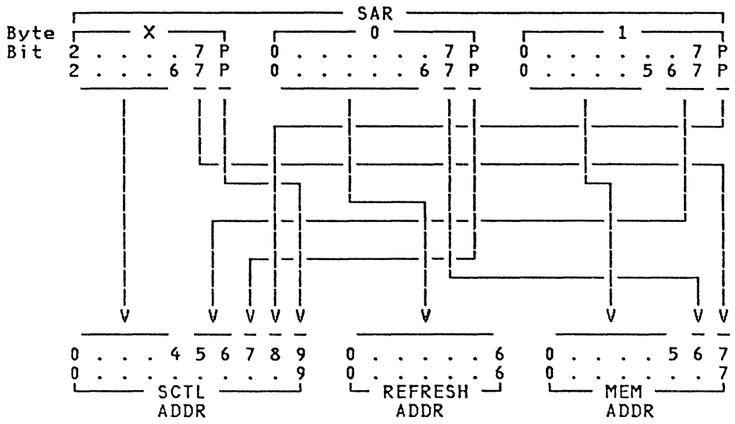
3. There is a scopable point for the data out gate (DOG) for each selected byte and parity, as follows:

Selected byte	DOG BSM pin	DOG ECC pin
Y	Q2B03	P2W23
X	Q2B04	P2W24
0	Q2B05	P2W25
1	Q2B07	P2W26
P	Q2B08	P2W27

   See the storage signal routing diagram on page 10-840 for other scoping points on this board.

4. Data from CCU to check parity.
5. Data from storage card (old bytes).
6. New data to be stored (40 bits) after ECC computation.
7. There may be a delay of up to 30 ns between the clock triggering signal, and the signal itself.

Storage Signal Routing (H) (Part 1 of 2)



All points are on the 01A-A1 board.

In the diagrams which follow, the symbols have the following meaning:

```
# = Board connector pin
I = Card input signal pin
O = Card output signal pin
```

Storage Signal Routing (H) (Part 2 of 2)

Pin/Net List

SIGNAL NAME	NET NAME	MEM-1 A	-2 B	-3 C	-4 D	-5 E	-6 F	-7 G	-8 H	-9 J	-10 K	-11 L	-12 M	N	ECC P	SCTL Q	U	X
BIT P1 (FROM ECC CARD	$EC+BITP1	------	------	------	------	------	------	------	------	------	------	------	------	------	O 2Y33	I 2Y33		
BYTE SELECT FROM BSMI	$EK+SBS0 X	------	------	------	------	------	------	------	------	------	------	------	# 6D04	------	------	I 5B03		
	$EK+SBS0 Y	------	------	------	------	------	------	------	------	------	------	------	# 6E02	------	------	I 5B02		
	$EK+SBS0 00	------	------	------	------	------	------	------	------	------	------	------	# 6C03	------	------	I 5B04		
	$EK+SBS0 01	------	------	------	------	------	------	------	------	------	------	------	# 6B02	------	------	I 5B05		
+CHK/RST P1 (FROM ECC CARD)	$EC+CHKP1	------	------	------	------	------	------	------	------	------	------	------	------	------	O 2Y22	I 2Y22		
-DOG BSMI PARITY BITS	$FA-DOGP0	------	------	------	------	------	------	------	------	------	------	------	------	# 6D04	------	O 2B08		
-DOG BYTE X BSMI	$FA-DOGX0	------	------	------	------	------	------	------	------	------	------	------	------	# 6E04	------	O 2E04		
-DOG BYTE Y BSMI	$FA-DOGY0	------	------	------	------	------	------	------	------	------	------	------	------	# 6E02	------	O 2B03		
-DOG BYTE 0 BSMI	$FA-DOG00	------	------	------	------	------	------	------	------	------	------	------	------	# 6C04	O 2Y04	O 2B05		
-DOG BYTE 1 BSMI	$FA-DOG10	------	------	------	------	------	------	------	------	------	------	------	------	# 6A02	O 2Y05	O 2B07		
-DOG ECC BYTE X	$FA-ECCX0	------	------	------	------	------	------	------	------	------	------	------	------	I 2H24	O 2Y06	O 2H24		
-DOG ECC BYTE Y	$FA-ECCY0	------	------	------	------	------	------	------	------	------	------	------	------	I 2H23	O 2Y07	O 2H23		
-DOG ECC BYTE 0	$FA-ECCC0	------	------	------	------	------	------	------	------	------	------	------	------	I 2H25	O 2Y09	O 2H25		
-DOG ECC BYTE 1	$FA-ECC10	------	------	------	------	------	------	------	------	------	------	------	------	I 2H26	O 2Y10	O 2H26		
-DOG ECC PARITY	$FA-ECCP0	------	------	------	------	------	------	------	------	------	------	------	------	I 2H27	O 2Y11	O 2H27		
+DOG MEMORY	$FA+DOG0	------	I 3D04	I 3D04	I 3D04	I 3D04	I 3D04	I 3D04	I 3D04	I 3D04	I 3D04	I 3D04	I 3D04	I 3D04	------	O 3D04		
ECC DISABLE	$EC+BITX2	------	------	------	------	------	------	------	------	------	------	------	------	------	O 2H31	I 2H31		
-ECC DISABLE(TO ECC CARD)	$FA-DIS0	------	------	------	------	------	------	------	------	------	------	------	------	------	I 2H29	O 2H29		
-ECC TRANSPARENCY TO ECC CARD	$FA-TRAN0	------	------	------	------	------	------	------	------	------	------	------	------	------	I 2H30	O 2H30		
ECC TRANSPARENT	$EC+BITX3	------	------	------	------	------	------	------	------	------	------	------	------	------	O 2H32	I 2H32		
+F (FROM ECC CARD)	$EC+F0	------	------	------	------	------	------	------	------	------	------	------	------	------	O 2Y23	I 2Y23		
+G (FROM ECC CARD)	$EC+G0	------	------	------	------	------	------	------	------	------	------	------	------	------	O 2Y24	I 2Y24		
+H (FROM ECC CARD)	$EC+H0	------	------	------	------	------	------	------	------	------	------	------	------	------	O 2Y25	I 2Y25		
-INPUT 70	$EK-IN70	------	------	------	------	------	------	------	------	------	------	------	------	# 6A04	------	I 2D13		
MEMA ADDRESS BIT FROM BSMI	$EK-MEMA000 #	3D07 I 4D05	I 4D05	I 4D05	I 4D05	I 4D05	I 4D05	I 4D05	I 4D05	I 4D05	I 4D05	I 4D05	I 4D05	I 4D05	------	I 4B04		
	$EK-MEMA001 #	3D06 I 3D06	I 3D06	I 3D06	I 3D06	I 3D06	I 3D06	I 3D06	I 3D06	I 3D06	I 3D06	I 3D06	I 3D06	I 3D06	------	I 4B05		
	$EK-MEMA002 #	3D05 I 4D04	I 4D04	I 4D04	I 4D04	I 4D04	I 4D04	I 4D04	I 4D04	I 4D04	I 4D04	I 4D04	I 4D04	I 4D04	------	I 4B04		
	$EK-MEMA003 #	3D04 I 3D02	I 3D02	I 3D02	I 3D02	I 3D02	I 3D02	I 3D02	I 3D02	I 3D02	I 3D02	I 3D02	I 3D02	I 3D02	------	I 3B04		
	$EK-MEMA004 #	3D03 I 4B04	I 4B04	I 4B04	I 4B04	I 4B04	I 4B04	I 4B04	I 4B04	I 4B04	I 4B04	I 4B04	I 4B04	I 4B04	------	I 3D12		
	$EK-MEMA005 #	3D04 I 3B04	I 3B04	I 3B04	I 3B04	I 3B04	I 3B04	I 3B04	I 3B04	I 3B04	I 3B04	I 3B04	I 3B04	I 3B04	------	I 3D06		
	$EK-MEMA006 #	3D03 I 4B03	I 4B03	I 4B03	I 4B03	I 4B03	I 4B03	I 4B03	I 4B03	I 4B03	I 4B03	I 4B03	I 4B03	I 4B03	------	I 4B03		
	$EK-MEMA007 #	3D02 I 3D12	I 3D12	I 3D12	I 3D12	I 3D12	I 3D12	I 3D12	I 3D12	I 3D12	I 3D12	I 3D12	I 3D12	I 3D12	------	I 3D06		
MEMORY CARD	$FA-MEM0100	------	0 3D07	------	------	------	------	------	------	------	------	------	------	------	------	I 3D13		
	$FA-MEM0101	------	------	0 3D07	------	------	------	------	------	------	------	------	------	------	------	I 3D10		
	$FA-MEM0102	------	------	------	0 3D07	------	------	------	------	------	------	------	------	------	------	I 3D07		
	$FA-MEM0103	------	------	------	------	0 3D07	------	------	------	------	------	------	------	------	------	I 2D13		
	$FA-MEM0104	------	------	------	------	------	0 3D07	------	------	------	------	------	------	------	------	I 3D06		
	$FA-MEM0105	------	------	------	------	------	------	0 3D07	------	------	------	------	------	------	------	I 4B05		
	$FA-MEM0106	------	------	------	------	------	------	------	0 3D07	------	------	------	------	------	------	I 4B06		
	$FA-MEM0107	------	------	------	------	------	------	------	------	0 3D07	------	------	------	------	------	I 4D13		
	$FA-MEM0108	------	------	------	------	------	------	------	------	------	0 3D07	------	------	------	------	I 2D12		
	$FA-MEM0109	------	------	------	------	------	------	------	------	------	------	0 3D07	------	------	------	I 2D07		
	$FA-MEM0110	------	------	------	------	------	------	------	------	------	------	------	0 3D07	------	------	I 2D06		
	$FA-MEM0111	------	------	------	------	------	------	------	------	------	------	------	------	0 3D07	------	I 2D05		
-MEMORY CARDS CNTL STORE SEL	$FA-SELO 00	------	I 4D12	------	------	------	------	------	------	------	------	------	------	------	------	O 4D12		
	$FA-SELO 01	------	------	I 4D12	------	------	------	------	------	------	------	------	------	------	------	O 4D11		
	$FA-SELO 02	------	------	------	I 4D12	------	------	------	------	------	------	------	------	------	------	O 5B10		
	$FA-SELO 03	------	------	------	------	I 4D12	------	------	------	------	------	------	------	------	------	O 5B13		
	$FA-SELO 04	------	------	------	------	------	I 4D12	------	------	------	------	------	------	------	------	O 5B15		
	$FA-SELO 05	------	------	------	------	------	------	I 4D12	------	------	------	------	------	------	------	O 4D06		
	$FA-SELO 06	------	------	------	------	------	------	------	I 4D12	------	------	------	------	------	------	O 4B12		
	$FA-SELO 07	------	------	------	------	------	------	------	------	I 4D12	------	------	------	------	------	O 4D10		
	$FA-SELO 08	------	------	------	------	------	------	------	------	------	I 4D12	------	------	------	------	O 4C05		
	$FA-SELO 09	------	------	------	------	------	------	------	------	------	------	I 4D12	------	------	------	O 3B12		
	$FA-SELO 10	------	------	------	------	------	------	------	------	------	------	------	I 4D12	------	------	O 3B10		
	$FA-SELO 11	------	------	------	------	------	------	------	------	------	------	------	------	I 4D12	------	O 3B06		
MEMORY DATA BUS BIT PA	$EC-PA0	------	0 2D13	0 2D13	0 2D13	0 2D13	0 2D13	0 2D13	0 2D13	0 2D13	0 2D13	0 2D13	0 2D13	0 2D13	I 5D10			
		------	------	------	------	------	------	------	------	------	------	------	------	------	O 5E09			
		------	------	------	------	------	------	------	------	------	------	------	------	------	O 2Y29	I 2Y29		
MEMORY DATA BUS BIT PB	$EC-PB0	------	0 3B03	0 3503	0 3B03	0 3B03	0 3B03	0 3B03	0 3B03	0 3503	0 3B03	0 3B03	0 3B03	0 3B03	I 5D11			
		------	------	------	------	------	------	------	------	------	------	------	------	------	O 5B10			
		------	------	------	------	------	------	------	------	------	------	------	------	------	O 2Y30	I 2Y30		
MEMORY DATA BUS BIT PC	$EC-PC0	------	0 5D13	0 5D13	0 5D13	0 5D13	0 5D13	0 5D13	0 5D13	0 5D13	0 5D13	0 5D13	0 5D13	0 5D13	I 5D12			
		------	------	------	------	------	------	------	------	------	------	------	------	------	O 5B12			
		------	------	------	------	------	------	------	------	------	------	------	------	------	O 2Y31	I 2Y31		
MEMORY DATA BUS BIT PD	$EC-PD0	------	0 5B13	0 5B13	0 5B13	0 5B13	0 5B13	0 5B13	0 5B13	0 5B13	0 5B13	0 5B13	0 5B13	0 5B13	I 5D13			
		------	------	------	------	------	------	------	------	------	------	------	------	------	O 5B13			
		------	------	------	------	------	------	------	------	------	------	------	------	------	O 2Y32	I 2Y32		
-MEMORY DATA BUS BYTE X	$EK-BYTX0 P #	2B08 0 2B07	0 2B07	0 2B07	0 2B07	0 2B07	0 2B07	0 2B07	0 2B07	0 2B07	0 2B07	0 2B07	0 2B07	0 2B07	I 5D06			
	$EK-BYTX000 #	2D13 0 2D12	0 2D12	0 2D12	0 2D12	0 2D12	0 2D12	0 2D12	0 2D12	0 2D12	0 2D12	0 2D12	0 2D12	0 2D12	O 2D12	I 2D12		
	$EK-BYTX001 #	2D12 0 2B10	0 2B10	0 2B10	0 2B10	0 2B10	0 2B10	0 2B10	0 2B10	0 2B10	0 2B10	0 2B10	0 2B10	0 2B10	O 2B10	I 2D13		
	$EK-BYTX002 #	2D13 0 2B12	0 2B12	0 2B12	0 2B12	0 2B12	0 2B12	0 2B12	0 2B12	0 2B12	0 2B12	0 2B12	0 2B12	0 2B12	O 2B12	I 3D02		
	$EK-BYTX003 #	2D11 0 2D11	0 2D11	0 2D11	0 2D11	0 2D11	0 2D11	0 2D11	0 2D11	0 2D11	0 2D11	0 2D11	0 2D11	0 2D11	O 2D11	I 3D04		
	$EK-BYTX004 #	2B10 0 2B09	0 2B09	0 2B09	0 2B09	0 2B09	0 2B09	0 2B09	0 2B09	0 2B09	0 2B09	0 2B09	0 2B09	0 2B09	O 2B09	I 3D05		
	$EK-BYTX005 #	2D10 0 2D10	0 2D10	0 2D10	0 2D10	0 2D10	0 2D10	0 2D10	0 2D10	0 2D10	0 2D10	0 2D10	0 2D10	0 2D10	O 2D10	I 3D06		
	$EK-BYTX006 #	2E09 0 2D08	0 2D08	0 2D08	0 2D08	0 2D08	0 2D08	0 2D08	0 2D08	0 2D08	0 2D08	0 2D08	0 2D08	0 2D08	I 5D07			
	$EK-BYTX007 #	2D09 0 2D09	0 2D09	0 2D09	0 2D09	0 2D09	0 2D09	0 2D09	0 2D09	0 2D09	0 2D09	0 2D09	0 2D09	0 2D09	I 3D09			
															O 3D08			
MEMORY DATA BUS BYTE Y	$EK-BYTY0 P #	2B02 0 2B02	0 2B02	0 2B02	0 2B02	0 2B02	0 2B02	0 2B02	0 2B02	0 2B02	0 2B02	0 2B02	0 2B02	0 2B02	I 5D05			
	$EK-BYTY000 #	2D07 0 2D07	0 2D07	0 2D07	0 2D07	0 2D07	0 2D07	0 2D07	0 2D07	0 2D07	0 2D07	0 2D07	0 2D07	0 2D07	O 2D07	I 2D02		
	$EK-BYTY001 #	2D06 0 2D05	0 2D05	0 2D05	0 2D05	0 2D05	0 2D05	0 2D05	0 2D05	0 2D05	0 2D05	0 2D05	0 2D05	0 2D05	O 2D05	I 2D04		
	$EK-BYTY002 #	2D06 0 2D06	0 2D06	0 2D06	0 2D06	0 2D06	0 2D06	0 2D06	0 2D06	0 2D06	0 2D06	0 2D06	0 2D06	0 2D06	O 2D06	I 2D03		
	$EK-BYTY003 #	2D05 0 2D04	0 2D04	0 2D04	0 2D04	0 2D04	0 2D04	0 2D04	0 2D04	0 2D04	0 2D04	0 2D04	0 2D04	0 2D04	O 2D04	I 2D06		
	$EK-BYTY004 #	2D05 0 2D05	0 2D05	0 2D05	0 2D05	0 2D05	0 2D05	0 2D05	0 2D05	0 2D05	0 2D05	0 2D05	0 2D05	0 2D05	O 2D05	I 2D07		
	$EK-BYTY005 #	2D04 0 2D03	0 2D03	0 2D03	0 2D03	0 2D03	0 2D03	0 2D03	0 2D03	0 2D03	0 2D03	0 2D03	0 2D03	0 2D03	O 2D03	I 2D09		
	$EK-BYTY006 #	2D03 0 2D04	0 2D04	0 2D04	0 2D04	0 2D04	0 2D04	0 2D04	0 2D04	0 2D04	0 2D04	0 2D04	0 2D04	0 2D04	I 2D10			
	$EK-BYTY007 #	2D02 0 2D02	0 2D02	0 2D02	0 2D02	0 2D02	0 2D02	0 2D02	0 2D02	0 2D02	0 2D02	0 2D02	0 2D02	0 2D02	I 2D11			
MEMORY DATA BUS BYTE 0	$EK-BYT00 P #	5B08 0 5B07	0 5B07	0 5B07	0 5B07	0 5B07	0 5B07	0 5B07	0 5B07	0 5B07	0 5B07	0 5B07	0 5B07	I 5B07				
	$EK-BYT0000 #	5D13 0 5D12	0 5D12	0 5D12	0 5D12	0 5D12	0 5D12	0 5D12	0 5D12	0 5D12	0 5D12	0 5D12	0 5D12	I 5D10				
	$EK-BYT0001 #	5D12 0 5D10	0 5D10	0 5D10	0 5D10	0 5D10	0 5D10	0 5D10	0 5D10	0 5D10	0 5D10	0 5D10	0 5D10	I 5D11				
	$EK-BYT0002 #	5D13 0 5D12	0 5D12	0 5D12	0 5D12	0 5D12	0 5D12	0 5D12	0 5D12	0 5D12	0 5D12	0 5D12	0 5D12	I 5D12	O 2Y04			
	$EK-BYT0003 #	5D11 0 5D11	0 5D11	0 5D11	0 5D11	0 5D11	0 5D11	0 5D11	0 5D11	0 5D11	0 5D11	0 5D11	0 5D11	I 5D13	O 2Y05			
	$EK-BYT0004 #	5B10 0 5B09	0 5B09	0 5B09	0 5B09	0 5B09	0 5B09	0 5B09	0 5B09	0 5B09	0 5B09	0 5B09	0 5B09	I 5B09	O 2Y06			
	$EK-BYT0005 #	5D10 0 5D10	0 5D10	0 5D10	0 5D10	0 5D10	0 5D10	0 5D10	0 5D10	0 5D10	0 5D10	0 5D10	0 5D10	I 5D10	O 2Y07			
	$EK-BYT0006 #	5B09 0 5B08	0 5B08	0 5B08	0 5B08	0 5B08	0 5B08	0 5B08	0 5B08	0 5B08	0 5B08	0 5B08	0 5B08	I 5B08	O 2Y09			
	$EK-BYT0007 #	5D09 0 5D09	0 5D09	0 5D09	0 5D09	0 5D09	0 5D09	0 5D09	0 5D09	0 5D09	0 5D09	0 5D09	0 5D09	I 5D09	O 2Y10			
															O 2Y11			
MEMORY DATA BUS BYTE 1	$EK-BYT10 P #	5B02 0 5B02	0 5B02	0 5B02	0 5B02	0 5B02	0 5B02	0 5B02	0 5B02	0 5B02	0 5B02	0 5B02	0 5B02	I 5B02				
	$EK-BYT1000 #	5D07 0 5D07	0 5D07	0 5D07	0 5D07	0 5D07	0 5D07	0 5D07	0 5D07	0 5D07	0 5D07	0 5D07	0 5D07	I 5D07				
	$EK-BYT1001 #	5D06 0 5B05	0 5B05	0 5B05	0 5B05	0 5B05	0 5B05	0 5B05	0 5D05	0 5D05	0 5B05	0 5B05	0 5B05	0 5B05				
	$EK-BYT1002 #	5D06 0 5D06	0 5D06	0 5D06	0 5D06	0 5D06	0 5D06	0 5D06	0 5D06	0 5D06	0 5D06	0 5D06	0 5D06	I 5D06				
	$EK-BYT1003 #	5305 0 5D04	0 5B04	0 5B04	0 5D04	0 5B04	0 5B04	0 5B04	0 5D04	0 5D04	0 5D04	0 5B04	0 5B04	I 5B04				
	$EK-BYT1004 #	5D05 0 5D05	0 5D05	0 5D05	0 5D05	0 5D05	0 5D05	0 5D05	0 5D05	0 5D05	0 5D05	0 5D05	0 5D05	I 5D05				
	$EK-BYT1005 #	5B04 0 5B03	0 5B03	0 5B03	0 5B03	0 5B03	0 5B03	0 5B03	0 5B03	0 5B03	0 5B03	0 5B03	0 5B03	I 5B03				
	$EK-BYT1006 #	5D03 0 5D04	0 5D04	0 5D04	0 5D04	0 5D04	0 5D04	0 5D04	0 5D04	0 5D04	0 5D04	0 5D04	0 5D04	I 5D04				
	$EK-BYT1007 #	5D02 0 5D02	0 5D02	0 5D02	0 5D02	0 5D02	0 5D02	0 5D02	0 5D02	0 5D02	0 5D02	0 5D02	0 5D02	I 5D02				
M1	$ED-CAR01	------	------	------	------	------	------	------	------	------	# 6A04	------	------	------	------	I 5D02		
M2	$ED-CAR02	------	------	------	------	------	------	------	------	------	# 6A02	------	------	------	------	I 5D04		
M3	$ED-CAR03	------	------	------	------	------	------	------	------	# 6E04	------	------	------	------	------	I 5D05		
M4	$ED-CAR04	------	------	------	------	------	------	------	------	# 6D02	------	------	------	------	------	I 5D06		
M5	$ED-CAR05	------	------	------	------	------	------	------	------	# 6C02	------	------	------	------	------	I 5D07		
M6	$ED-CAR06	------	------	------	------	------	------	------	------	# 6C04	------	------	------	------	------	I 5D09		
-PAR CHK/+PAR GEN TO ECC CARD	$FA-PAR0	------	------	------	------	------	------	------	------	------	------	------	------	------	I 2H28	O 2H28		
-RD/+WRT	$EK+SRW0	------	------	------	------	------	------	------	------	------	# 6B02	------	------	------	------	I 5D12		
-READ MODE	$FA-READ0	# 3D13	------	------	------	------	------	------	------	------	------	------	------	------	------	O 5D13		
+READ - WRITE TO MEMORY CARD	$FA-RW0	------	I 4B13	I 4B13	I 4B13	I 4B13	I 4B13	I 4B13	I 4B13	I 4B13	I 4B13	I 4B13	I 4B13	I 4B13	------	O 4D15		
+REFRESH ADDRESS BITS 0 TO 6	$EK-REFA000 #	3B13 I 3B12	I 3B12	I 3B12	I 3B12	I 3B12	I 3B12	I 3B12	I 3B12	I 3B12	I 3B12	I 3B12	I 3B12	I 3B12	------	O 2D04		
	$EK-REFA001 #	3D11 I 4D02	I 4D02	I 4D02	I 4D02	I 4D02	I 4D02	I 4D02	I 4D02	I 4D02	I 4D02	I 4D02	I 4D02	I 4D02	------	O 2D02		
	$EK-REFA002 #	3B10 I 3B10	I 3B10	I 3B10	I 3B10	I 3B10	I 3B10	I 3B10	I 3B10	I 3B10	I 3B10	I 3B10	I 3B10	I 3B10	------	O 2D09		
	$EK-REFA003 #	3D10 I 4D02	I 4B02	I 4B02	I 4B02	I 4B02	I 4B02	I 4B02	I 4D02	I 4D02	I 4B02	I 4B02	I 4B02	I 4B02	------	O 5D11		
	$EK-REFA004 #	3B09 I 3D09	I 3B09	I 3B09	I 3D09	I 3B09	I 3D09	I 3B09	I 3B09	I 3B09	I 3B09	I 3B09	I 3B09	I 3B09	------	O 5D07		
	$EK-REFA005 #	3D09 I 4B05	I 4B05	I 4B05	I 4B05	I 4B05	I 4B05	I 4B05	I 4B05	I 4B05	I 4B05	I 4B05	I 4B05	I 4B05	------	O 5D09		
	$EK-REFA006 #	3B05 I 3B07	I 3B07	I 3B07	I 3B07	I 3B07	I 3B07	I 3B07	I 3B07	I 3B07	I 3B07	I 3B07	I 3B07	I 3B07	------	O 4D03		
+REFRESH IN PROGRESS TO BSMI	$FA+REF0	------	------	------	------	------	------	------	------	------	------	------	------	# 6B04	------	O 2B02		
-REFRESH TO MEMORY CARD	$FA-REF0	------	I 3D11	I 3D11	I 3D11	I 3D11	I 3D11	I 3D11	I 3D11	I 3D11	I 3D11	I 3D11	I 3D11	I 3D11	------	O 3D11		
+SAMPLE DATA ECC	$FA+SAM0	------	------	------	------	------	------	------	------	------	------	------	------	------	I 2H22	O 2H22		
SCTL ADDRESS BIT FROM BSMI	$EK-SCTL000	------	------	------	------	------	------	------	------	------	------	------	# 6D02	------	------	I 4B09		
	$EK-SCTL001	------	------	------	------	------	------	------	------	------	------	------	# 6C04	------	------	I 4B08		
	$EK-SCTL002	------	------	------	------	------	------	------	------	------	------	------	# 6D04	------	------	I 3B13		
	$EK-SCTL003	------	------	------	------	------	------	------	------	------	------	------	# 6B02	------	------	I 3B05		
	$EK-SCTL004 #	3D12 I 3D09	I 3B09	I 3D09	I 3B09	I 3D09	I 3B09	I 3D09	I 3B09	I 3D09	I 3B09	I 3D09	# 6A02	I 3D09	------	I 3B03		
	$EK-SCTL005	------	------	------	------	------	------	------	------	------	------	------	------	------	------	I 4D07		
	$EK-SCTL006	------	------	------	------	------	------	------	------	# 6E02	------	------	------	------	------	I 4B09		
	$EK-SCTL007	------	------	------	------	------	------	------	------	# 6E02	------	------	------	------	------	I 4C07		
	$EK-SCTL008	------	------	------	------	------	------	------	------	# 6D04	------	------	------	------	------	I 4D10		
	$EK-SCTL009	------	------	------	------	------	------	------	------	# 6C02	------	------	------	------	------	I 3B02		
+SP WRITE INHIBIT FROM BSMI	$EK+WIN0	------	------	------	------	------	------	------	------	------	------	------	------	# 6B02	------	I 2B02		
-STG CNTL OUTPUT TAG FROM BSMI	$EK-STTG0	------	------	------	------	------	------	------	------	------	------	------	------	# 6A02	------	I 5D11		
+STG GO	$EK+STGG0	------	------	------	------	------	------	------	------	------	------	------	# 6B04	------	------	I 5B09		
+STG GRANT	$FA+STGG0	# 3B02	------	------	------	------	------	------	------	------	------	------	------	------	O 2B07			
+STOR DATA ADDR PAR ERROR TO C	$FB+STPE0	------	------	------	------	------	------	------	------	------	------	------	------	# 6C04	O 2B12			
+STORAGE CONTROL ERROR TO CCU	$FA+STCE0	------	------	------	------	------	------	------	------	------	------	------	------	# 6D02	O 5B07			
TIE DOWN	$LL-TIEDW	------	------	------	------	------	------	------	------	------	------	------	------	------	I 2Y28	O 2Y28		
TIE UP	$LL+TIEUP	------	------	------	------	------	------	------	------	------	------	------	------	------	I 2Y27	O 2Y27		
WRITE ECC	$EC+BITX4	------	------	------	------	------	------	------	------	------	------	------	------	------	O 2H33	I 2H33		
-15.259 US PULSE	$EK-TIME1	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	# 6C04	
															I 5D10		# 6C02	0 4J04
+2 BITS ERROR	$EC+2BIT0	------	------	------	------	------	------	------	------	------	------	------	------	------	O 2Y26	I 2Y26		
-2 BITS STG ERROR TO CCU	$FC-TCSE0	------	------	------	------	------	------	------	------	------	------	------	------	# 6E04	------	O 5B08		

Legend:

+ : between + 3V and + 5V

— : between 0V and 0.5V

Unexpected CCU Interrupt Processing (I)

An unexpected Level 1 or Level 4 interrupt occurred while running the CCU diagnostics.

- RAC 8FB indicates that one or more bits in Input X'7E' or X'76' were related to the IOC bus.
- RAC 8FE indicates that one or more bits in Input X'7D', X'7E', or X'76', or in MOSS status registers A, B, and C was set.

Proceed as follows:

1. Take a MOSS dump.
2. The address X'D80E' contains the address (2 bytes) which points to a group of 9 bytes containing the values of the Input X'7D', X'7E', and X'76' registers (2 bytes per register), and the CCU-to-MOSS status A, B, and C registers (1 byte per register).
3. Use the following tables to change the most probably-failing FRUs:

Input X'7D'

Byte	Bit	Suspected FRUs (in order)								
0	0	CTL1	CTL2	BSMI	SCTL	ECC	DFLN	MIOC	DFL4	BTAC
	1	MIOC	CTL2	CTL1	SCTL	ECC	DFLN	BSMI	DFL4	
	2	MIOC	BTAC	DFLN	SCTL	CTL2				
	3	MEMN	ECC	SCTL	CTL2	BSMI				
	4	SCTL	CTL2	DFL4	BSMI	CCLK	MEMN	DFL5	CTL1	
	5									
	6	BSMI	SCTL	CTL2	ECC	DFLN	DFL4	CTL1	MIOC	
	7	CTL2	DFLN	DFL4	DFL5	CTL1	BSMI	SCTL	MIOC	
1	0									
	1	DFL4	CTL1	DFLN	DFL5	CTL2	MIOC	BTAC		
	2	DFL5	CTL1	CTL2	DFLN	DFL4	ECC	SCTL	BSMI	MIOC
	3	DFL4	CTL1	CTL2	DFLN					
	4	DFL4	CTL1	DFLN	DFL5	CTL2				
	5	CTL1	CTL2	DFLN	DFL5	BTAC	DFL4	MIOC		
	6	DFL4	CTL2	CTL1	BSMI	ECC	SCTL	DFLN	BTAC	MEMN
		DFL5	MIOC							
	7	DFL5	CTL1	CTL2	DFLN	DFL4				

Input X'7E'

Byte	Bit	Suspected FRUs (in order)								
0	0	MIOC								
	1									
	2									
	3	MIOC	CTL1	DFL4	DFLN	CTL2				
	4	MIOC	CTL1	DFL4	DFLN	CTL2	SCTL	ECC	BTAC	
	5	DFL5	MIOC	CTL2	CTL1	BTAC	DFLN			
	6									
	7									
1	0	BTAC	MIOC	CTL2	DFLN					
	1	CTL1	CTL2	MIOC	BSMI	DFLN	SCTL			
	2	CTL1	CTL2	MIOC	BSMI	DFLN	SCTL			
	3	CTL1	CTL2	MIOC	BSMI	DFLN	DFL5			
	4	CTL1	CTL2	MIOC	BSMI	DFLN	SCTL			
	5	DFL5	MIOC	CTL2	CTL1	BTAC	DFLN			
	6	BTAC	MIOC	CTL2	DFLN					
	7									

Input X'76'

Byte	Bit	Suspected FRUs (in order)								
0	0	DFL5	DFLN	CTL1	CTL2					
	1	DFL5	DFLN	CTL1	CTL2					
	2	DFL5	DFLN							
	3									
	4	DFL5	DFLN							
	5	DFL5	DFLN							
	6	DFL5	DFLN							
	7	DFL5	MIOC	CTL2	CTL1	BTAC	DFLN			
1	0	DFL5	DFLN	CTL1	CTL2					
	1	DFL5	DFLN	CTL1	CTL2					
	2	DFL5	DFLN							
	3									
	4	DFL5	DFLN							
	5	DFL5	DFLN							
	6	DFL5	DFLN							
	7	DFL5	MIOC	CTL2	CTL1	BTAC	DFLN			

CCU-to-MOSS Status A Register (CMSA)

Bit	Suspected FRUs (in order)			
0	MIOC			
1	MIOC			
2	MIOC			
3				
4	MIOC			
5	CTL2	CCLK	CTL1	MIOC
6				
7	MIOC			

CCU-to-MOSS Status B Register (CMSB)

Bit	Suspected FRUs (in order)			
0				
1				
2	MIOC			
3	MIOC			
4				
5				
6	MIOC			
7	MIOC			

CCU-to-MOSS Status C Register (CMSC)

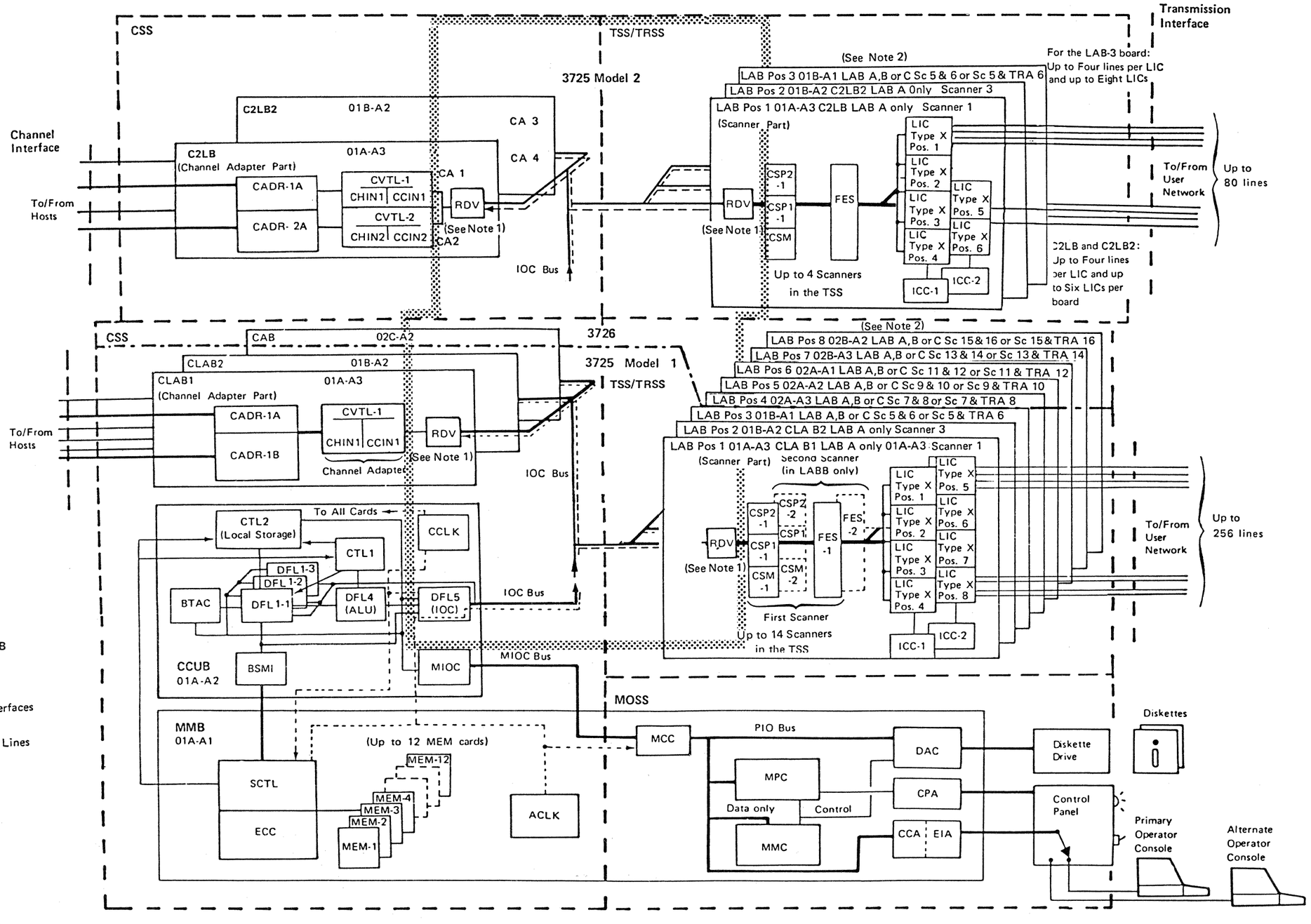
Bit	Suspected FRUs (in order)			
0	MIOC			
1	MIOC			
2	MIOC			
3	MIOC			
4	MIOC			
5	MIOC			
6	MIOC			
7				

Contents

Chapter 11. IOC Bus and Redrive

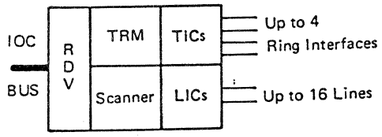
Section 1. Unit Description

IOC Bus and RDV in 3725 Data Flow



Note 1: As the CLAB and the CL2B boards are split into two parts in this figure, one for the channel adapter and one for the scanner, the redrive (RDV) function is shown twice for clarity. However, there is one RDV per CLAB or C2LB board.

Note 2: LAB Pos. 3 to 8 organization if LAB type C installed



Legend:

----- Clock signals

——— Data/control signals

IOC Bus Structure (Part 1 of 2)

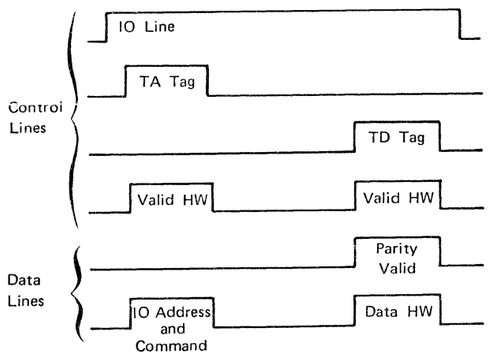
IOC CONTROL LOGIC

Data, address, and control information exchanges take place between the work registers and the adapters attached to the IOC bus. They use the IOC logic, which is packaged on the CCU DFL5 card.

The IOC logic operates differently depending on whether the program initiates an operation (PIO), or the adapter initiates it (AIO). In both operations, the A (address) and D (data) registers of the data flow controlled by the IOC logic act as buffers between the CCU and the adapters (CAs, CSPs, or TRAs).

The IOC bus carries interrupt requests from the adapters (level 1, 2, 3) when not busy with PIO or AIO operations.

PIO BUS PROTOCOL



IOC BUS LINE DESCRIPTIONS

The IOC bus consists of 18 bidirectional lines (2 data bytes plus 2 parity bits) and 16 tag and control lines as summarized below:

LINE FUNCTIONS		Abbr	CCU	RDV	CA	CSP TRA
Address/Command Tag (1)		TA	x--	-R-	--.	--.
Data Tag (1)		TD	x--	-R-	--.	--.
Interrupt Req. Removed (1)		IRR	.--	-R-	--x	--x
CS Req. High (1)		CSRH	.--	-R-	---	--x
CS Req. Low (1)		CSRL	.--	-R-	--x	
CS Grant High (1)		CSGH	x--	-R-	---	--.
CS Grant Low (1)		CSGL	x--	-R-	--.	
Input/Output (1)		I/O	x--	-R-	--.	--.
Halt (1)		HLT	x--	-R-	--.	--.
Reset (1)		RST	x--	-R-	--.	--.
Out (1)		R/W	x--	-R.		
Valid Byte (1)		VB	.--	-R-	--x	--x
Valid Halfword (1)		VH	.--	-R-	--x	--x
End of Chain (1)		EOC	.--	-R-	--x	--x
Modifier (1)		M	.--	-R-	--x	--x
Parity Valid (1)		PV	.--	-R-	--x	--x
Data Byte 0 (9)	OUT	DB0	x--	-R-	--.	--.
	IN	DB0	.--	-R-	--x	--x
Data Byte 1 (9)	OUT	DB1	x--	-R-	--.	--.
	IN	DB1	.--	-R-	--x	--x

Legend:

(): The contents of the parentheses indicate the number of wires in line function
x : Signal generated
R : Signal redriven
. : Indicates where the signal arrives

The following is a detailed description of each of the lines in the summary table above.

TA (Address/Command Tag)

The 'TA' line is activated by the CCU to indicate that the RDV or adapter address is in data byte 0 and the command is in data byte 1.

TD (Data Tag)

The 'TD' line is activated by the CCU to indicate that the data bus contains write data or that the CCU is ready to receive read data while the 'I/O' line is active. When the 'I/O' line is not active, the 'TD' line is activated by the CCU to indicate that it is permissible to change the state of any interrupt request on the data bus.

IRR (Interrupt Request Removed)

The 'IRR' line is activated by any adapter that has removed its interrupt request from the data bus. Each adapter should activate 'IRR' in response to 'I/O' line being activated and, of course, remove its interrupt request from the data bus. Each adapter should allow 'IRR' to change to the inactive level when the 'I/O' line is inactive. When all the adapters have allowed 'IRR' to drop, this common 'IRR' line going inactive indicates to the CCU that all adapters have placed their interrupt requests, if any, on the data bus and the CCU may now sample for interrupts.

CSRH (Cycle Steal Request High)

A scanner activates 'CSRH' whenever it wishes to start an AIO operation. A scanner keeps 'CSRH' active until it receives 'CSGH' (Cycle Steal Grant High).

CSRL (Cycle Steal Request Low)

A channel adapter activates CSRL whenever it wishes to start an AIO operation. A channel adapter keeps 'CSRL' active until it receives 'CSGL' (Cycle Steal Grant Low).

CSGH (Cycle Steal Grant High)

The CCU activates 'CSGH' in response to 'CSRH' for the purpose of selecting a scanner for an AIO operation, and receiving a 'CSCW' from the selected scanner. 'CSGH' will be deactivated by the CCU when 'valid halfword' is received from the scanner.

CSGL (Cycle Steal Grant Low)

The CCU activates 'CSGL' in response to 'CSRL' for the purpose of selecting a channel adapter for an AIO operation, and receiving a 'CSCW' from the selected channel adapter. 'CSGL' will be deactivated by the CCU when 'valid halfword' is received from the channel adapter.

I/O (Input/Output)

The 'I/O' line is activated by the CCU to indicate either that an I/O operation is about to start on the IOC bus, or that one is in progress. For any I/O operation, 'I/O' is the first line activated and the last one deactivated. When the 'I/O' line is active, all adapters should remove any interrupt requests on the data bus and not present any more until I/O is deactivated.

HLT (Halt)

The CCU activates the 'halt' line to indicate to the selected adapter that the CCU has detected an error condition associated with the current operation. The CCU will activate the 'halt' line after 'TA' or 'CSGH' or 'CSGL' has been deactivated. The CCU will deactivate the 'halt' line when it deactivates the 'I/O' line. The selected adapter will terminate the current operation and set a check bit active in its status register.

IOC Bus Structure (Part 2 of 2)

RST (Reset)

The CCU may activate the 'reset' line, at any time, to initialize all adapters. This initialization will cause all adapters to immediately terminate current operations, go to a disabled state, and prepare to respond to PIO commands.

R/W (Out)

The CCU activates the 'out' line, while the 'I/O' line is active, to indicate that the direction of information on the data bus is outbound from the CCU. The CCU deactivates the 'Out' line to indicate that the direction of information on the data bus is inbound to the CCU.

VB (Valid Byte)

A selected adapter will activate the 'valid byte' line during an AIO operation to indicate a byte transfer instead of a halfword transfer. The valid byte of information will be on data bus byte 1.

VH (Valid Halfword)

In some places the term 'valid' is used in place of 'valid halfword'. A selected adapter will activate the 'valid halfword' line in response to the activation of the 'TA', 'TD', or 'CSGH' or 'CSGL' line from the CCU. 'Valid halfword' line active indicates that the adapter has either placed information the data bus or has received information from the data bus. It also indicates that the CCU may deactivate its control line.
All adapters will activate 'valid halfword' when the CCU deactivates the 'I/O' line, and deactivate 'valid halfword' when the CCU activates the 'I/O' line. The CCU will proceed with an IO operation after all adapters have deactivated 'valid halfword'.

EOC (End of Chain)

A selected adapter will activate the 'end of chain' line instead of 'valid halfword' for the last halfword transfer of an AIO operation. 'End of chain' active indicates that the AIO operation should be concluded and that the adapter has either placed information on the data bus or has received information from the data bus.

M (Modifier)

A selected adapter will activate the 'modifier' line with 'valid byte' line for the last byte transfer of a AIO operation. 'Modifier' line active at this time indicates that the AIO operation should be concluded.

PV (Parity Valid)

A selected adapter will activate 'parity valid' to indicate to the CCU that it wishes to have parity checking of data inbound to the CCU. If 'parity valid' is active and bad parity is detected by the CCU, the CCU will activate the 'halt' line. If 'parity valid' is deactivated and bad parity is detected by the CCU, the parity is corrected, the data is stored, and a status bit is set.

Data Bus Bytes 0 and 1

The data bus is a halfword wide with 18 bidirectional lines. Each of the two bytes (0 and 1) contains 8 bits plus a parity bit (0-7,P). Information is transferred between the CCU and the adapters in either direction when the 'I/O' line is active. Three bits of the bus are used for an additional function when 'I/O' line is deactivated as follows:

Byte 0 bit 1 = Lvl 2 interrupt req to CCU.
Byte 0 bit 5 = Lvl 1 interrupt req to CCU.
Byte 1 bit 0 = Lvl 3 interrupt req to CCU.

The adapters activate their interrupt requests to the CCU using these paths, but only when the 'I/O' line is inactive. The CCU will sample the data bus for interrupts after activating the 'TD' line, while the 'I/O' line is inactive.

PIO Operation (Part 1 of 3)

PIO OPERATION SEQUENCE

A PIO operation to control a channel or a scanner operation may be started either by the control program in the CCU, or by the microcode in the MOSS.

PIO Initiated by the CCU

During such an operation two halfwords (address and command, and data) are exchanged with a selected adapter.

A PIO operation proceeds in four steps:

1. IOH or IOHI instruction decode
2. IOC initialization
3. Adapter addressing and selection
4. Data transfer:
   Write = from CCU
   Read = to CCU

At step 1:

- Instruction code 50 = IOH with R1 and R2.
- Instruction code 70 = IOHI with R1 and the second halfword of the instruction whose contents go into the D register.

PIO Initiated by the MOSS

MIOH/MIOHI instructions are equivalent to IOH/IOHI instructions (see details on page 10-100), except that:

1. The MOSS initiates the operation.
2. R2 contents are found in the LS address given by the LSAR at TA time.
3. The MDOR receives or sends the TD time data instead of R1.

PIO Operation (Part 2 of 3)

TABLE 1. PIO OPERATION SEQUENCE (INITIALIZATION)

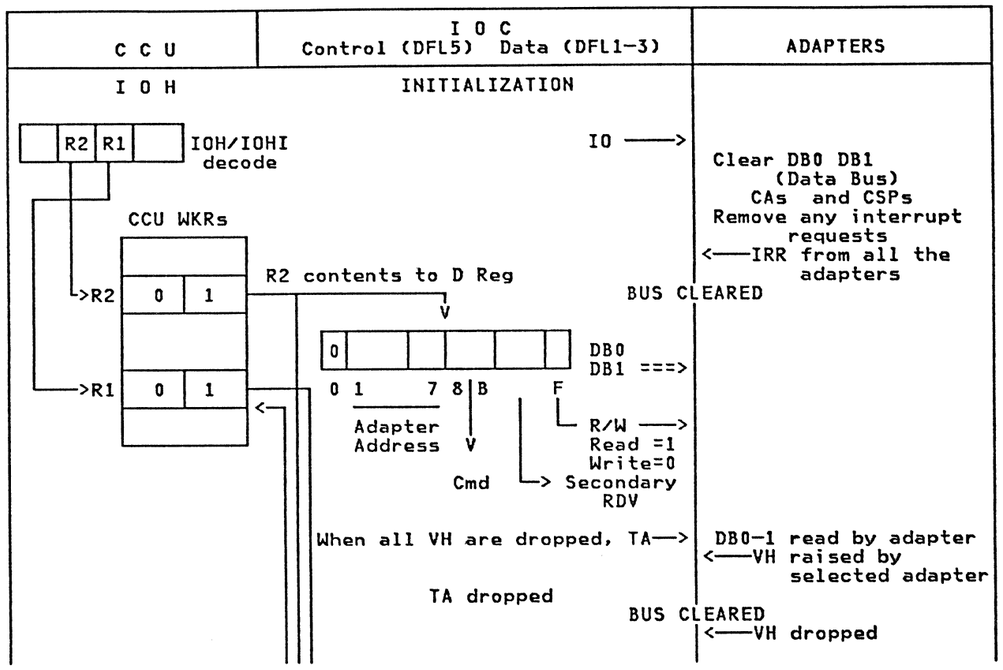
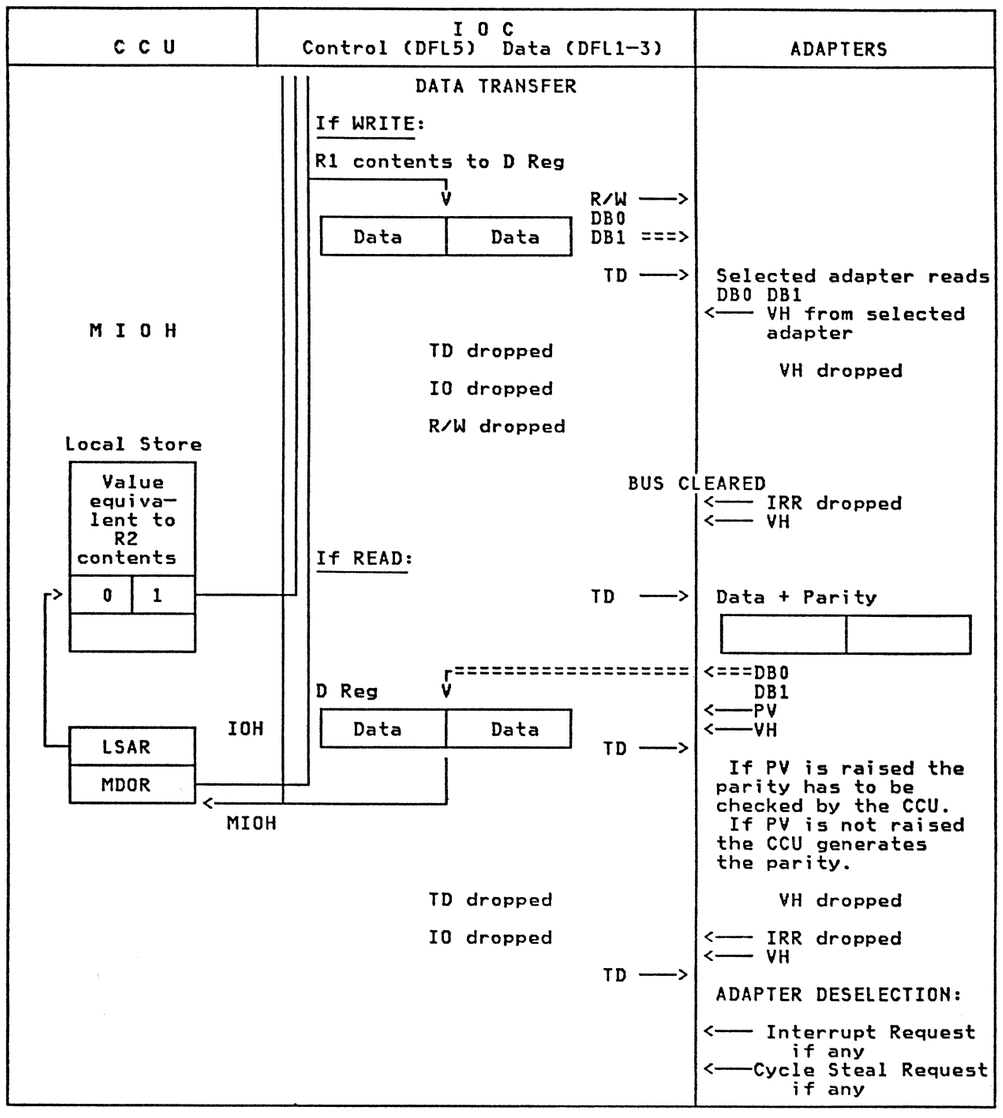
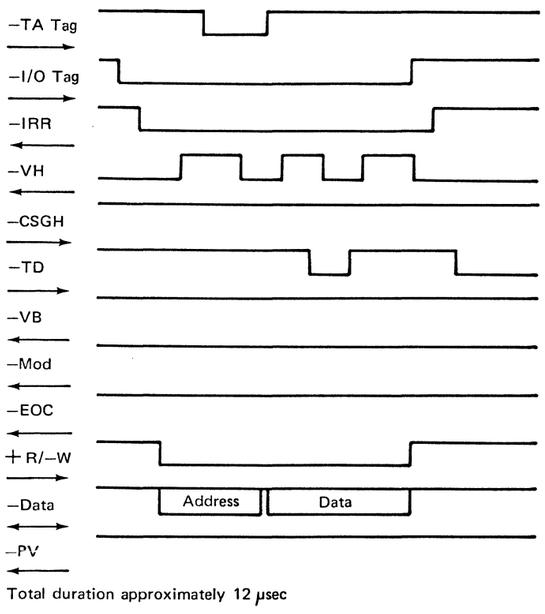


TABLE 2. PIO OPERATION SEQUENCE (DATA TRANSFER)



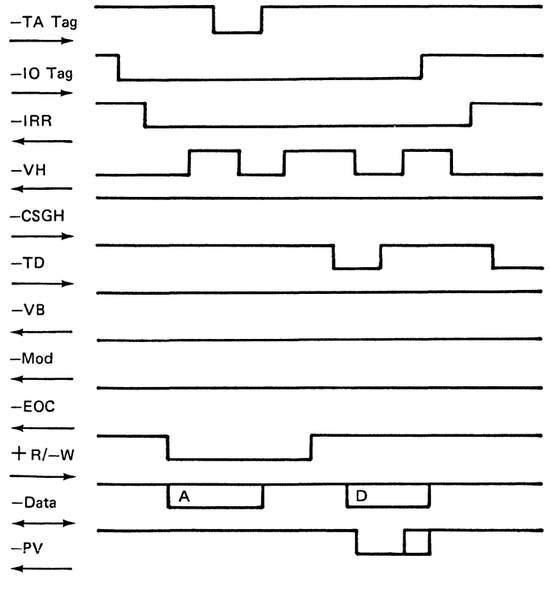
PIO Operation (Part 3 of 3)

PIO WRITE TIMING



Total duration approximately 12 μsec

PIO READ TIMING



AIO Operation (Part 1 of 4)

AIO OPERATION (CYCLE STEALING)

During such an operation several units of data are exchanged with a selected adapter. The maximum data transfer is 256 bytes.

A selected scanner provides the storage addresses at which the data bytes are to be stored. For this purpose, a pointer shared by all scanners is used. This information is first placed in the adapter registers by the control program using the IOH/IOHI instructions in PIO mode.

A selected channel adapter uses its dedicated pointer that has first been loaded by the control program.

TABLE 3. AIO OPERATION SEQUENCE (INITIALIZATION)

C C U	I O C Control (DFL5) Data (DFL1-3)	ADAPTERS
Adapter registers preset to prepare subsequent AIO operation. IOH/IOHI decode and execution. ——>	PIO mode ——>	Registers Byte count Storage add in Read Storage add in Write AIO mode
	INITIALIZATION AIO operation starts IO ——>	<—— CSR Adapters remove any interrupt requests. The objective is to clear DB0 DB1 for any interrupt request. <—— IRR from all adapters VH dropped
	BUS CLEARED CSG ——>	Adapters are cabled in priority order, and the requesting adapter with the highest priority keeps the "Grant". Example: Adapter 1 — No request Adapter 2 — Request Adapter 3 — Request CSG is chained and only adapter 2 proceeds with the request (see Table 4). <—— CSR dropped

AIO Operation (Part 2 of 4)

TABLE 4. AIO OPERATION SEQUENCE (CSCW TRANSFER)

C C U	I O C Control (DFL5) Data (DFL1-3)	ADAPTERS
	CYCLE STEAL CONTROL WORD TRANSFER (CSCW)	
		DB0 <=== DB1 <—— VH <—— PV if CSP AIO. If CA AIO, PV may be off but the parity must be correct and the CCU checks it.
	Is parity correct ⊥ Yes no = HALT ——> D Reg contains CSCW 000 00 0 2 34 | 67 8 A B F V CA = 0 CSP/TRA= 1 V V Type of operation V Pointer register number that holds the storage data address (CA) or Scanner identification (CSP) 1 = only byte 1 in this control word is valid. 0 = both bytes of this control word are valid.	
	CSG dropped	VH dropped (see Table 5)

TABLE 5. AIO OPERATION SEQUENCE (STORAGE ADDRESS TRANSFER)

C C U	I O C Control (DFL5) Data (DFL1-3)	ADAPTERS
	STORAGE ADDRESS TRANSFER	DIRECT MODE
		<—— PV <—— VH
	TD dropped	TD ——> <=== Address byte X VH dropped
		<—— VH
	TD dropped	TD ——> <=== Address bytes 0, 1 VH dropped
	V A Reg V TD dropped S A R <— X 0 1 <— INCREMENTER V A V Main Storage L S (CTL2)	INDIRECT MODE No address data: only POINTER NUMBER from control word (CSCW) for CA or from dedicated pointer = 3F for CSP or TRA
	For data transfer in Write mode, see Table 6.	For data transfer in Read mode, see Table 7.

AIO Operation (Part 3 of 4)

TABLE 6. AIO OPERATION SEQUENCE (DATA TRANSFER IN WRITE)

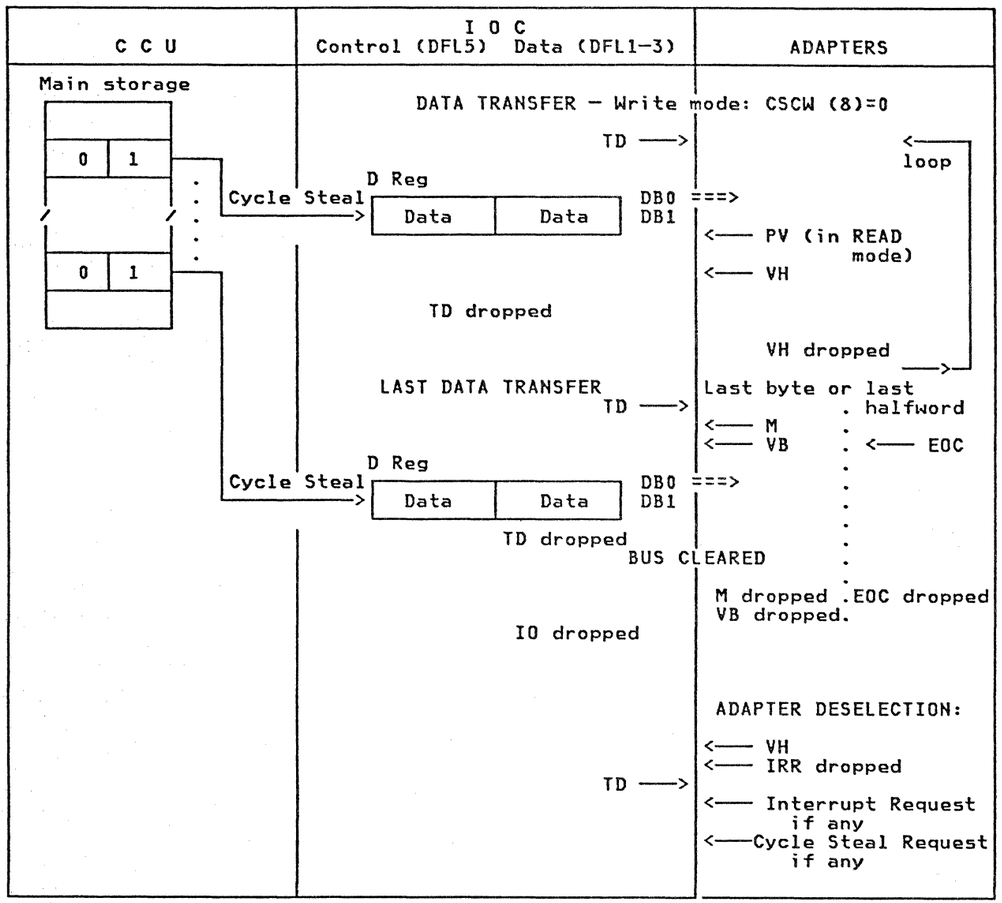
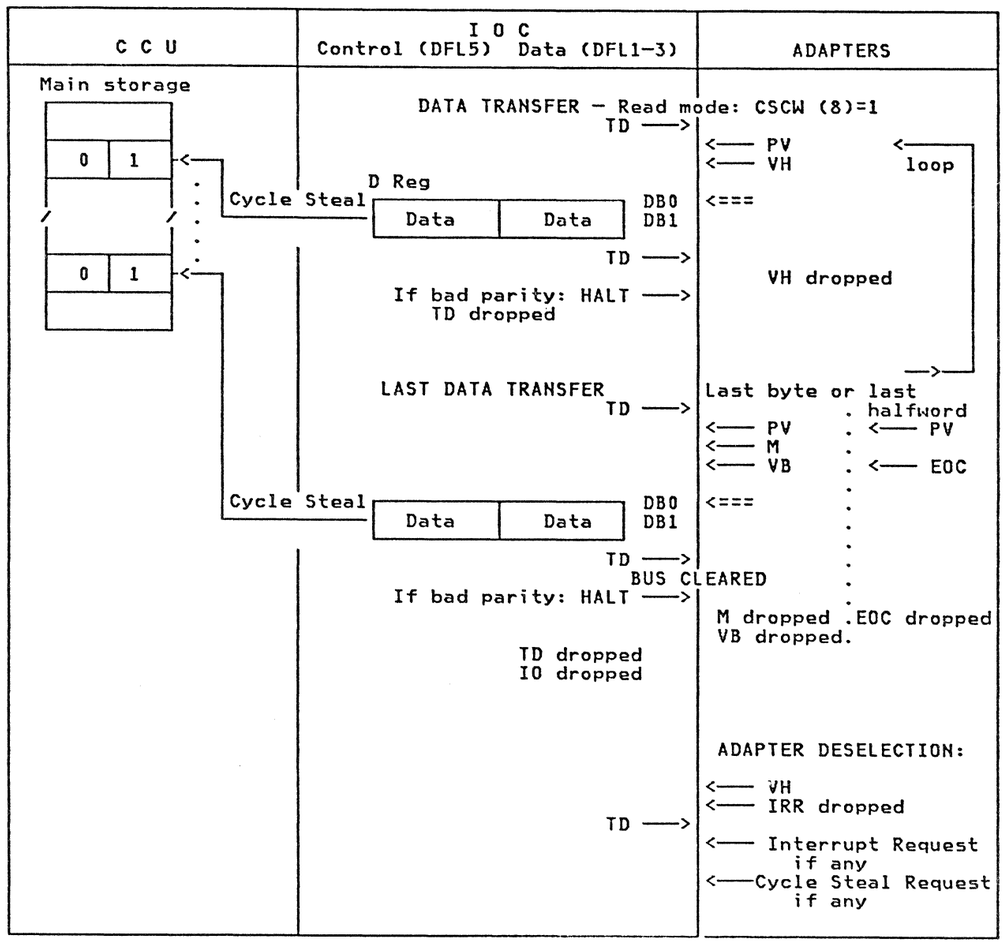
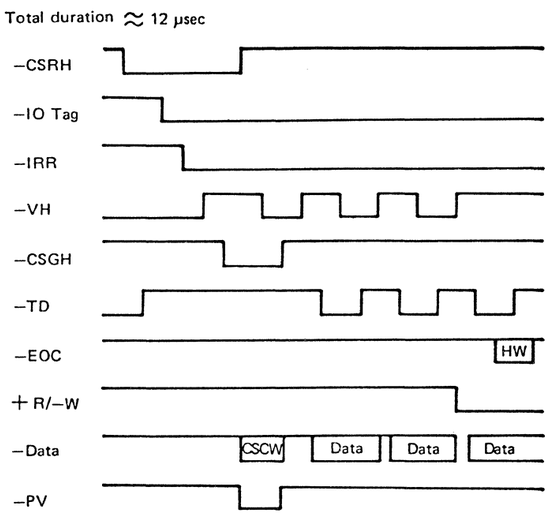


TABLE 7. AIO OPERATION SEQUENCE (DATA TRANSFER IN READ)

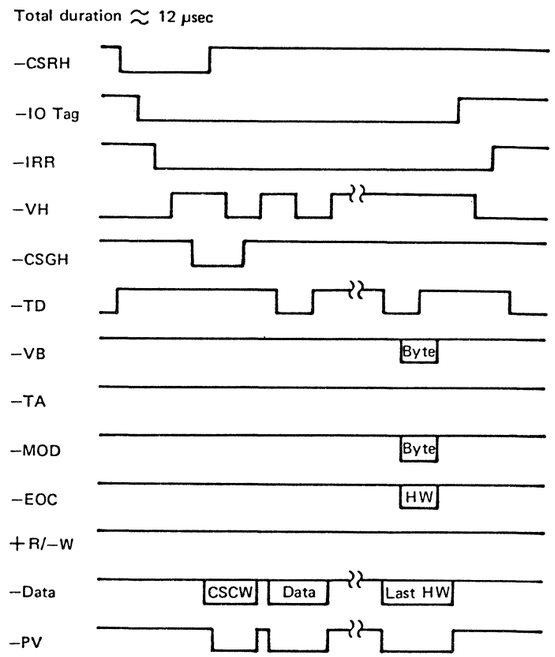


AIO Operation (Part 4 of 4)

AIO WRITE TIMING



AIO READ TIMING



Address/Command Formats on IOC Bus

IOC BUS: ADDRESS/COMMAND FORMAT

At TA Time

Byte-->	DB0		DB1		
Type	0123	4567	0123	4567	Comments
RDV	0100	000.	CCCC	....	Format
	....	....	0...	....	Broadcast
	....	....	1...	....	Specific
	....	....	0000	...1	Poll in read
	....	....	.001	...1	Read error
	....	....	.000	...0	Write error
	....	....	.001	...0	Disable drivers
	....	....	.010	...0	Enable
	....	....	.101	...0	Reset
	....	....	1010	...1	Diag read
	....	....	.110	...0	Diag write
C2LB	....	0000	....	000.	3725 Mod 2
CLAB1	....	0000	....	000.	3725 Mod 1
C2LB2	....	0000	....	001.	3725 Mod 2
CLAB2	....	0000	....	001.	3725 Mod 1
LAB pos 3	....	0000	....	010.	3725 Mod 1
LAB pos 3	....	0000	....	010.	3725 Mod 2
CAB	....	0000	....	011.	3726
Frame RDV	....	0001	....	000.	3726
LAB pos 4	....	0001	....	011.	3726
LAB pos 5	....	0001	....	100.	3726
LAB pos 6	....	0001	....	101.	3726
LAB pos 7	....	0001	....	110.	3726
LAB pos 8	....	0001	....	111.	3726

	DB0		DB1		
	0123	4567	0123	4567	Comments
CA	0000	1000	....	....	Format
	....	....	....	...0	Write(OUT)
	....	....	....	...1	Read (IN)
	....	....	....	NNN.	CA address
IN/OUT 0	....	....	0000	....	
IN/OUT 1	....	....	0001	....	
IN/OUT 2	....	....	0010	....	
IN/OUT 3	....	....	0011	....	
IN/OUT 4	....	....	0100	....	
IN/OUT 5	....	....	0101	....	
IN/OUT 6	....	....	0110	....	
IN/OUT 7	....	....	0111	....	
IN/OUT C	....	....	1100	....	
IN/OUT D	....	....	1101	....	
IN/OUT E	....	....	1110	....	
IN/OUT F	....	....	1111	....	

	DB0		DB1		
	0123	4567	0123	4567	Comments
CSP	00..	0...	CCCC	....	Format
	....	....	0000	0..0	Start line
	..11	....	0000	0..1	Get line ID
	....	....	0001	0..1	Get error status
	....	....	0001	0..0	Start line initial
	....	....	0010	0..0	Set LNVT high
	....	....	0011	0..0	Set LNVT low
	....	....	0100	0..0	Scanner reset
	....	....	....	0...	NCP Cmd
	....	....	....	1...	MOSS Cmd
	....	....	....	.0..	n/a
	....	....	....	..0.	Normal mode
	....	....	....	..1.	Character mode
	....	....	....	...0	Write
	....	....	....	...1	Read
CSP/1st	..01	....	....	....	Any LAB type
CSP/2nd	..10	....	....	....	LAB type B
CSP-1	..01	.000	....	....	C2LB 3725
CSP-1	..01	.000	....	....	CLAB1 3725
CSP-3	..01	.001	....	....	CLAB2 3725
CSP-3	..01	.001	....	....	C2LB2 3725
CSP-5	..01	.010	....	....	LAB-3 3725
CSP-6	..10	.010	....	....	LAB-3 3725
CSP-7	..01	.011	....	....	LAB-4 3726
CSP 8	..10	.011	....	....	LAB-4 3726
CSP-9	..01	.100	....	....	LAB-5 3726
CSP 10	..10	.100	....	....	LAB-5 3726
CSP-11	..01	.101	....	....	LAB-6 3726
CSP 12	..10	.101	....	....	LAB-6 3726
CSP-13	..01	.110	....	....	LAB-7 3726
CSP 14	..10	.110	....	....	LAB-7 3726
CSP-15	..01	.111	....	....	LAB-8 3726
CSP 16	..10	.111	....	....	LAB-8 3726

Note: CSP-2 and CSP-4 do not exist.
CSP = communication scanner processor

	DB0		DB1		
	0123	4567	0123	4567	Comments
TRA	0...	....	CCCC	..0.	Format
	..11	....	0000	0..1	Get LID
	.100	1...	0000	11.1	Get cmd completn
	.100	1...	0000	.1.0	Start/Stop Mask/Unmsk
	.100	1...	0001	.1..	TRM Ctrl
	.100	1...	0010	.1..	TIC Ctrl
	.100	1...	0011	.1..	LID base
	.100	1...	0100	.1..	IR/BR reg
	.100	1...	0101	.1..	Diag reg
	.100	1...	0110	.1..	Buffer reg
	.100	1...	.111	.1..	Ext buffer
	.100	1...	1000	.1.1	Read LID
	.100	1...	1000	.1.0	Prog Reset
	.100	1...	1001	.1.1	L2 stat(1)
	.100	1...	1010	.1.1	L2 stat(2)
	.100	1...	1011	.1.1	L2 stat(3)
	.100	1...	1100	.1.1	L2 stat(4)
	.100	1...	1101	.1.1	L1 status
	.100	1...	1110	.1.1	MOSS stat
	.100	1...	1111	.1.1	Read CSCW
	.100	1...	00..	.0..	TIC Data
	.100	1...	01..	.0..	TIC Data†
	.100	1...	10..	.0..	TIC Addr
	.100	1...	10..	.0..	TIC Intrpt
	....	....	..00	.0..	TIC 1
	....	....	..01	.0..	TIC 2
	....	....	..10	.0..	TIC 3
	....	....	..11	.0..	TIC 4
	....	....	....	0...	CCU Cmd
	....	....	....	1...	MOSS Cmd
	....	....	....	.0..	Cmd to TIC
	....	....	....	.1..	Cmd to TRM
	....	....	....	...0	Write
	....	....	....	...1	Read
TRA-6	....	.010	....	....	LAB-3 3725
TRA-8	....	.011	....	....	LAB-4 3726
TRA-10	....	.100	....	....	LAB-5 3726
TRA-12	....	.101	....	....	LAB-6 3726
TRA-14	....	.110	....	....	LAB-7 3726
TRA-16	....	.111	....	....	LAB-8 3726

Note: TRA = Token Ring Adapter
TIC = Token Ring Interface Coupler

AIO - CSCW Contents

DB0		DB1		
0123	4567	0123	4567	(1)
0123	4567	89AB	CDEF	(1) Comments
0000	0...	...P	PPP.	Format
....	.0..	....	....	CA
....	.1..	....	....	TSS
....	..0.	....	....	Not DI
....	..1.	....	....	DI
....	...0	....	....	Short (D or DI)
....	...1	....	....	Long (D or DI)
....	....	0...	....	Write
....	....	1...	....	Read
....	...0	.0..	....	Indirect
....	....	.1..	....	Direct
....	....	..0.	....	NCP
....	....	..1.	....	MOSS
....	....	...P	PPP.	CS Pointer number or scanner identification (CSP)
....	....	....	...0	H (2 byte CSCW)
....	....	....	...1	L (1 byte CSCW)

(1) 8 to F when in a register;
0 to 7 byte 1 when in bus.

IOC BUS: ADDRESS/COMMAND DESCRIPTION

At TD Time

DB0		DB1		
0123	4567	0123	4567	Comments
DDDD	DDDD	DDDD	DDDD	Data or:
HHHH	HHHH	...L	LLLL	High level command Line Interface Addr. or:
0000 AAAA	0000 AAAA	XXXX BBBB	XXXX BBBB	Address byte X (*) Address byte 0 and Address byte 1 (*)

(*) of line vector table (LNVT).

Scanner, TRA and Line Addressing

The scanner and line addressing consists of three elements:

1. The line attachment board address
2. The line group address (within a board)
3. The line interface address (within a group)

LINE ATTACHMENT BOARD ADDRESS

The LAB address is a 3-bit field that is decoded to address one of the eight possible line boards (CLAB1 or C2LB, CLAB2 or C2LB2, and LAB position 3 through LAB position 8). It is contained in bits 5-7 of the second halfword of the instruction (R2 for an IOH, or the immediate field for an IOHI).

R2 or Immediate Field (DB0 and DB1 at TA):

Bit	Function
0	0
1-4	Line group
5-7	LAB address
8-11	Operation code
12	0 = control program command 1 = MOSS command
13	0
14	0 = Normal mode 1 = Character mode
15	O = Output I = Input

Note: Bits 13 and 14 have different meaning for TRA.

13: 0 = TIC Cmd
    1 = TRM Cmd
14: Always 0 for TRM

Bits 1-4: Give the line group address as follows:

0010 : First line group (16 lines) in a C2LB, CLAB or LAB type A, or first scanner in a LAB type B
0100 : Second line group (16 lines) in a C2LB, CLAB or LAB type A, or second scanner in a LAB type B or C
0110 : All scanners and/or TRAs
1001 : Token Rings Group (4)

Bits 5-7: Give the LAB address as follows:

000 : C2LB
000 : CLAB1 (LAB-1)
001 : C2LB2
001 : CLAB2 (LAB-2)
010 : LAB-3
011 : LAB-4
100 : LAB-5
101 : LAB-6
110 : LAB-7
111 : LAB-8

Bits 8-11: See page 11-040, DB1 bits 0 through 3.

On the machine, the boards are personalized with their redrive address (see page 11-070). Physically, addresses are set by printed circuit, or by jumpers on the pin side of the board in location YB (see page 4-270).

Jumpers must be installed when replacing boards (LAB boards only).

LINE GROUP

The line group is a 4-bit field with three possible formats that are used to select one of the two groups of 16 lines within a LAB, or to broadcast commands to all the scanners. It is contained in bits 1-4 of the second halfword of the instruction. For a LAB type B, the line group also specifies the first or second scanner.

Bits 1-4: Give the line group address as follows:

0010 : First line group (16 lines) in a C2LB, CLAB or LAB type A, or first scanner in a LAB type B
0100 : Second line group (16 lines) in a C2LB, CLAB or LAB type A, or second scanner in a LAB type B
0110 : All scanners and/or TRAs
1001 : Token Ring Group (4)

Combination 0110 is used by the 'get line identification' instruction. It is interpreted as a broadcast invitation to the scanner or TRA holding the highest priority interrupt to send the line identification to the CCU.

LINE INTERFACE ADDRESS

3725/3726

The line interface address is a 5-bit field that is decoded to address one of the 32 line interfaces (16 lines, each with a receive and a transmit interface). It is contained in bits 11-15 of the addressed register of the instruction (R1 for an IOH).

R1 (DB0 and DB1 at TD):

Bit	Function
0	0
1-7	Command code
8-10	0 0 0
11-15	Line interface address

The line interface address varies from 00 through 31 (X'00' through X'1F'). Bit 15 is on for a receive line interface and off for a transmit line interface. For a half-duplex line, bit 15 is off.

3725 Model 2

For the C2LB and C2LB2 boards:

The line interface address is a 5-bit field that is decoded to address one of the 24 line interfaces (12 lines, each with a receive and a transmit interface). It is contained in bits 11-15 of the addressed register of the instruction (R1 for an IOH). See table R1 (DB0 DB1 at TD) above.

SCANNER AND TRA ADDRESSING

The following table summarizes the scanner/TRA addressing. The bits are taken from byte 0 of R2 or immediate field of the instruction.

Board	Line Group 01234	LAB Addr 567	Hex	Scanner or TRA
CLAB1 or C2LB	00010 00100	000 000	10 20	S1 S1
CLAB2 or C2LB2	00010 00100	001 001	11 21	S3 S3
LAB pos 3	00010 00100 01001	010 010 010	12 22 4A	S5 S5 or S6 TRA 6
LAB pos 4	00010 00100 01001	011 011 011	13 23 4B	S7 S7 or S8 TRA 8
LAB pos 5	00010 00100 01001	100 100 100	14 24 4C	S9 S9 or S10 TRA 10
LAB pos 6	00010 00100 01001	101 101 101	15 25 4D	S11 S11 or S12 TRA 12
LAB pos 7	00010 00100 01001	110 110 110	16 26 4E	S13 S13 or S14 TRA 14
LAB pos 8	00010 00100 01001	111 111 111	17 27 4F	S15 S15 or S16 TRA 16

SCANNER/TRA PRE-AUTO SELECTION

In order to give priority to the adapters supporting the high-speed lines, an automatic selection function is provided for handling:

- CCU level 2 interrupt requests
- CCU cycle steal requests

This function is provided by the RDV, CSP and TRA cards, and is organized in two steps: preselection and autoselection. The scanner microcode attributes the priority level (low only) to the scanner at IML time.

CCU Level 2 Interrupt Requests

Preselection: When a adapter requires service from the CCU for a specific line, it issues a level 2 service request. This request is placed on the IOC bus. Several adapters may raise service requests at the same time.

The level 2 preselection interconnects all the adapters and continuously determines which adapter has the highest priority among the level 2 service requests. The corresponding adapter is said to be 'preselected'.

Autoselection: To service the level 2 interrupts, the control program issues a 'get line ID' instruction, which is decoded by all the adapters. Only the adapter that has been preselected will answer to the 'get line ID' instruction.

If several adapters issue level 2 service requests with the same priority at the same time, the selected adapter is the first implemented on the IOC bus. As soon as the get line identification operation ends, preselection resumes.

Cycle Steal Requests

Preselection: The preselection for the cycle steal mechanism interconnects all the scanners and determines which scanner among those which have issued a cycle steal request must be served first by the CCU.

A scanner can simultaneously issue a level 2 service request and a cycle steal request but the same priority is used for both requests.

Autoselection: The CCU answers a cycle steal request by sending the cycle steal grant signal to the scanners. Only the

Channel Adapter Addressing

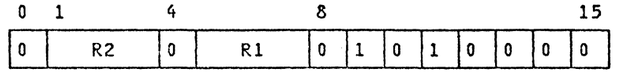
Although up to six CAs can be installed, the control program can perform CA inputs and outputs on only one CA at a time. In order to do so, it must first select that CA.

Note: CAs may also be selected with the CA autoselection mechanism explained in Chapter 12.

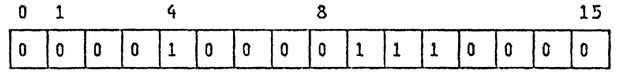
The IOH and IOHI instructions are used to access the CA registers. These two instructions do not contain an explicitly defined CA address but are performed on the selected CA. In order to perform CA inputs and outputs, the first IOH or IOHI issued must be a CA output X'7' with the applicable select bits set. The following is a description of the CA output X'7' using an IOH or IOHI.

INPUT/OUTPUT (IOH)

IOH Halfword Instructions



R2 must be loaded as follows:

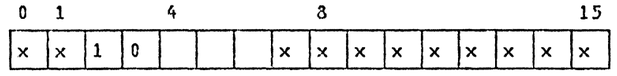


Bit 4= 1, IOH indicates a channel adapter

Bits 8 to 11= 0111, indicate a CA X'7'

Bit 15= 0, indicates an output

R1 must be loaded as follows:



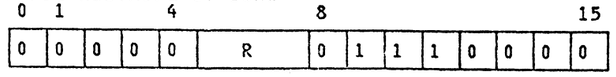
Bit 2= 1, indicates select CA addressed by bits 4 through 6

Bits 4, 5, 6= 000 select CA#1
001 select CA#2
010 select CA#3
011 select CA#4
100 select CA#5
101 select CA#6

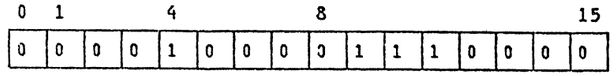
Note: Refer to the section in Chapter 12 labeled CA input/ouptut registers for a dafinition of the bits that may be 0 or 1 indicated by an X.

INPUT/OUTPUT IMMEDIATE (IOHI)

First Halfword of IOHI



Second Halfword



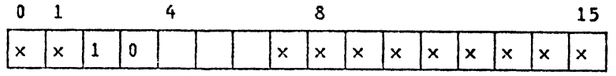
Second halfword must be as follows:

Bit 4= 1, indicates a channel adapter IOHI

Bits 8 to 11= 0111, indicate a CA X'7'

Bit 15= 0, indicates an output

R must be loaded as follows:



The bit definition of R of an IOHI is the same as the bit definition of R1 of an IOH.

Redrive Cards

The redrive (RDV) cards connect the IOC bus to the channel and line boards.

FUNCTIONS

The main purpose of the redrive card is to repower the IOC bus signals at the entry to the boards. In addition, it performs the following logical functions:

- Handles CCU level 1 interrupts
- Propagates the 'priority cycle steal grant' signal
- Generates from the adapter clock (ACLK) the timing for the adapters
- Handles the preselection and autoselection functions for the scanners

The redrive also performs error-checking functions:

- Isolates the board from the IOC bus
- Detects parity errors and tag sequence errors on the IOC bus
- Executes diagnostic commands

REDRIVE CARD IMPLEMENTATION

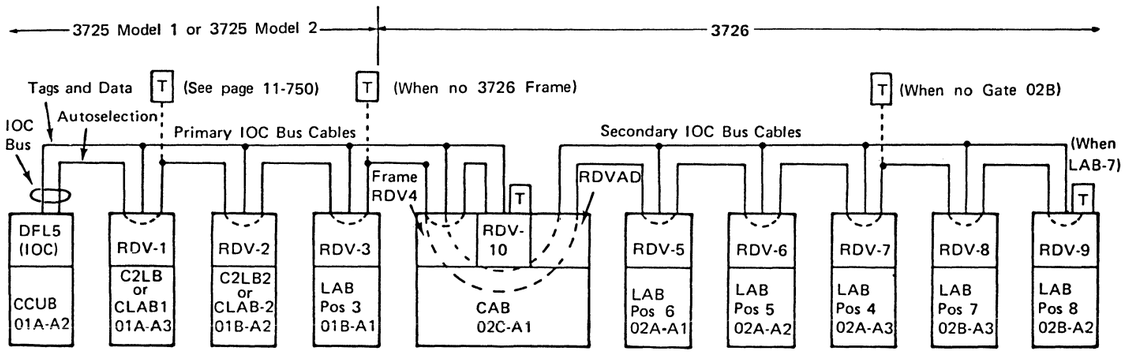
Each adapter board except the CAB has one redrive card. The CAB has two additional redrive cards used as follows:

- One frame RDV card is used as frame redrive to connect the IOC bus to the 3726 frame.
- One adapter RDVAD card provides impedance matching for connecting LABs 4 through 8 to the IOC bus.

IOC BUS CONNECTION

The following figure shows the connections of the IOC bus signals:

- All tags and data wires are connected in parallel.
- The signals used for the scanner and channel adapter autoselection are connected in series.



Legend: T Bus Terminator (BUSTERM)

Note: For IOC wire continuity, refer to pages 4-090 and 4-091.

Redrive Addressing

The redrive address consists of three elements:

1. The RDV group address
2. The primary redrive address
3. The secondary redrive address

R2 or Immediate Field (DB0 and DB1 at TA):

Bit	Function
0	0
1-4	RDV group address
5-7	Primary redrive address
8-11	Command code
12-14	Secondary redrive address
15	Input/output or read/write

RDV GROUP ADDRESS

The RDV group address is a 4-bit field that is always 1000. It is contained in bits 1-4 of the second halfword of the instruction sent at TA time on the IOC bus.

PRIMARY REDRIVE ADDRESS

The primary redrive address is a 3-bit field that is decoded to address the first or second frame (CAB excepted). It is contained in byte 0, bits 5-7 of the second halfword of the instruction sent at TA time on the IOC bus (R2 for IOH, the immediate field for IOHI).

Bits 5-7: These are coded as follows to give the primary redrive address:

000 : first redrive group (CLAB1, C2LB, CLAB2, C2LB2, LAB position 3, and CAB)
001 : second redrive group (other boards)

SECONDARY REDRIVE ADDRESS

The secondary redrive address is a 3-bit field that is decoded to address one redrive card within a frame. It is contained in byte 1, bits 4-6 of the second halfword of the instruction sent at TA time on the IOC bus.

REDRIVE ADDRESSES AND COMMANDS

Address and command format on the IOC bus at TA time.

R2 or Imm Field--->	0.....7		8.....15		
Byte --->	DB0		DB1		
Type	0123	4567	0123	4567	Comments
RDV	0100	000.	CCCC	....	Format
	....	....	0...	....	Broadcast
	....	....	1...	....	Specific
	....	....	0000	...1	Poll in read
	....	....	.001	...1	Read error
	....	....	.000	...0	Write error
	....	....	.001	...0	Disable drivers
	....	....	.010	...0	Enable
	....	....	.101	...0	Reset
	....	....	1010	...1	Diag read
	....	....	.110	...0	Diag write
C2LB	....	...0	....	000.	3725 Mod 2
CLAB1	....	...0	....	000.	3725 Mod 1
C2LB2	....	...0	....	001.	3725 Mod 2
CLAB2	....	...0	....	001.	3725 Mod 1
LAB pos 3	....	...0	....	010.	3725 Mod 1
LAB pos 3	....	...0	....	010.	3725 Mod 2
CAB	....	...0	....	011.	3726
Frame RDV	....	...1	....	000.	3726
LAB pos 4	....	...1	....	011.	3726
LAB pos 5	....	...1	....	100.	3726
LAB pos 6	....	...1	....	101.	3726
LAB pos 7	....	...1	....	110.	3726
LAB pos 8	....	...1	....	111.	3726

Board Position	RDV Number	Redrive Address Primary Field (DB0) 5 6 7	Secondary Field (DB1) 4 5 6
C2LB	1	0 0 0	0 0 0
CLAB1	1	0 0 0	0 0 0
C2LB2	2	0 0 0	0 0 1
CLAB2	2	0 0 0	0 0 1
LAB pos 3	3	0 0 0	0 1 0
CAB	10	0 0 0	0 1 1
Frame	4	0 0 1	0 0 0
LAB pos 4	7	0 0 1	0 1 1
LAB pos 5	6	0 0 1	1 0 0
LAB pos 6	5	0 0 1	1 0 1
LAB pos 7	8	0 0 1	1 1 0
LAB pos 8	9	0 0 1	1 1 1

REDRIVE ADDRESSING ON BOARDS

On the boards, primary and secondary redrive addresses are set by printed circuits or by jumpers on the pin side of the board in location YB as shown in the following table.

Board Position	B08 to B07	C08 to C07	D08 to D07	E08 to E07	Address Set By Printed Circuit	Jumper
DB0/DB1	7	4	5	6		
C2LB or CLAB1 01A-A3D1					X	
C2LB2 or CLAB2 01B-A2D1				X	X	
LAB pos 3 01B-A1D1			X			X
CAB 02C-A1D1			X	X	X	
LAB pos 4 02A-A3D1	X		X	X		X
LAB pos 5 02A-A2D1	X	X				X
LAB pos 6 02A-A1D1	X	X		X		X
LAB pos 7 02B-A3D1	X	X	X			X
LAB pos 8 02B-A2D1	X	X	X	X		X

Notes:

1. Bits 5 and 6 of the primary redrive address field are forced to zero.
2. See jumper information on page 4-270, when one LAB is to be replaced.

Redrive Functions

The redrive card:

- (A) Repowers the IOC bus signals for the connected channel adapter or communication scanner
- (B) Decodes the addresses and commands received from the CCU over the IOC bus
- (C) Switches the diagnostic and error information to the IOC bus when the redrive command and address are decoded
- (D) Handles the priority of the cycle steal requests and level 1 and 2 interrupt requests
- (E) Generates the clocking signals to the channel adapters and scanners
- (F) Isolates the channel adapter or scanner from the IOC bus when the disable redrive command is received
- (G) Detects errors on bad parity and tag sequence
- (H) Executes the diagnostic commands by writing and reading the error and diagnostic registers

REDRIVE ERROR REGISTER

The error register of the redrive card is an 8-bit register that records the errors detected on the IOC bus. The bits are individually set when the corresponding error occurs:

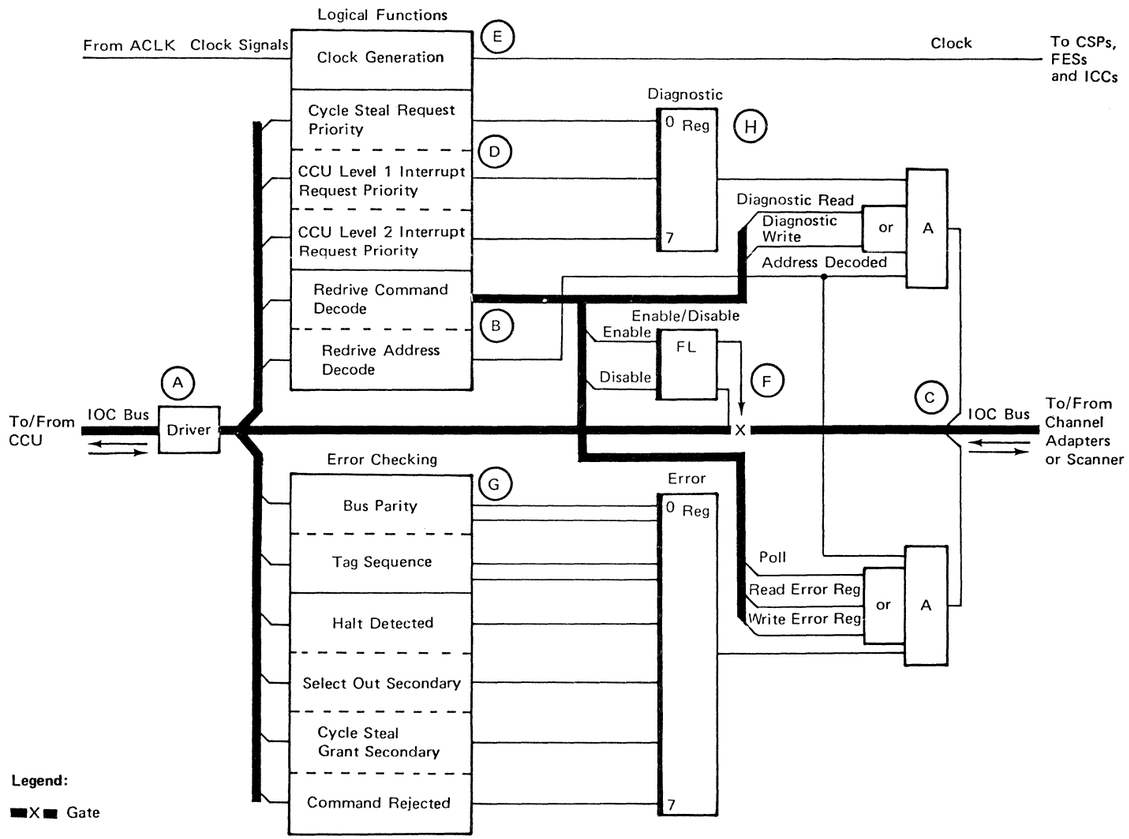
Bit	Error Cause
0	IOC bus parity error (inbound)
1	IOC bus parity error (outbound)
2	IOC bus tag check (outbound)
3	IOC bus tag check (inbound)
4	Halt detected
5	Select out secondary
6	Cycle steal grant secondary
7	Command rejected

Bits 5 and 6 are set on when a timeout occurs on a cycle steal. This information is used for diagnostics and for recovery.

RESET

When the reset tag of the IOC bus is activated (at IPL time, for example), all latches and registers are reset on the redrive cards. All redrives are reset at the same time. Individual reset is obtained through the reset command (see "Redrive Commands" on page 11-100).

RDV



Redrive State Definitions

ENABLED

When enabled, the redrive completes the logical communication path between itself and the adapters on the same board. In the case of the frame redrive, the logical communication path between the redrive and the secondary IOC bus is completed. An IOH instruction containing an enable command must be executed in the CCU to enter the enabled state. There is no physical change to the machine. The redrive card will still respond to redrive IOH commands. Any adapters on the board with this redrive can then detect and respond to adapter IOH commands.

DISABLED

When disabled, the redrive blocks the logical communications path between itself and the adapters on the same board. In the case of the frame redrive, the communications path between itself and the secondary IOC bus is blocked. An IOH instruction containing a disable command must be executed in CCU to enter the disabled state. A power on reset will also cause the redrives to enter the disabled state. There are no physical changes to the machine. The redrive will still respond to redrive IOH commands. None of the adapters on the same board with this redrive can detect or respond to any adapter IOH commands.

BOARD DISABLED BY JUMPER

A physical jumper is installed from ground (D08) to pin D11 of a redrive card. This has the same effect as the disable command. In addition, it prevents the activation of the valid halfword tag and the deactivation of the interrupt request removed tag by this redrive. This jumper is the only physical change to the machine. The redrive does not respond to any IOH commands. None of the adapters on the same board can detect or respond to any adapter IOH commands.

DEACTIVATED

The configuration data file (CDF) screen is altered to indicate that a redrive is not present in the machine. This is done by replacing the asterisk (*) next to the RDV number with a space on the LAB/CAB CDF screen. The IFT diagnostics do not attempt to run with any redrive that does not indicate that the redrive is present. There are no physical changes to the machine.

DISCONNECTED

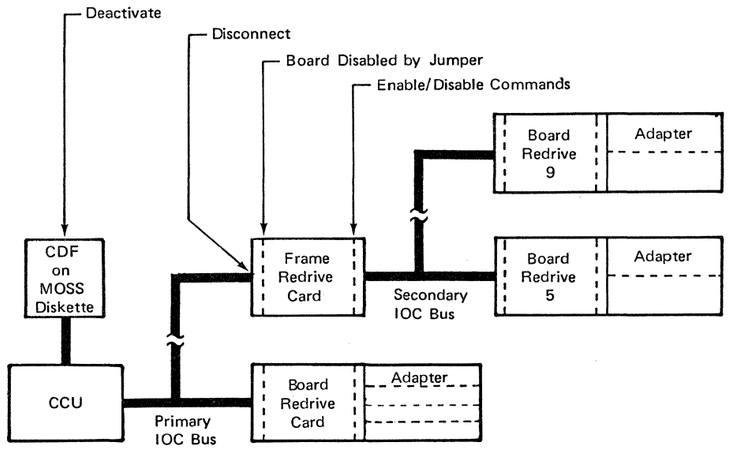
The IOC bus cables are physically unplugged from the top of the redrive card and the continuity jumper plugs are installed in the cable connectors Y and Z. The redrive may still be plugged into the board, but the redrive and any adapters on that same board have no connection to the IOC bus.

SUMMARY TABLE

	Console implemented	IOH command implemented	Physically implemented	Redrive response to IOH command	Path enabled between redrive and adapters	Adapter response to IOH command	Redrive physically isolated from IOC bus
Enabled	N	Y	N	Y	Y	Y	N
Disabled	N	Y	N	Y	N	N	N
Board Disabled by Jumper	N	N	Y	N	N	N	N
Deactivated	Y	N	N	Y	Y*	Y*	N
Disconnected	N	N	Y	N	N	N	Y

Y = Yes, N = No
* If IOH sent to redrive

DATA FLOW



Redrive Commands (Part 1 of 2)

COMMAND LIST

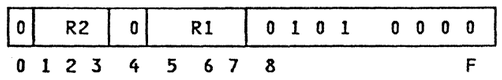
The following commands are used to control the redrives during tests and diagnostics:

- Poll
- Read error register
- Write error register
- Disable drivers
- Enable drivers
- Reset redrive
- Diagnostic read
- Diagnostic write

The CCU I/O instructions may be IOH or IOHI. In the following descriptions, the output instructions are IOH and the input instructions are IOHI.

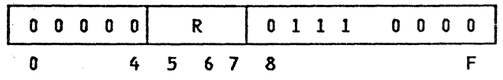
COMMAND FORMAT

IOH:



The IOH transfers the contents of register R1 to a redrive or from a redrive to R1 via the IOC bus. The redrive is specified by the contents of register R2. The contents of R1 and R2 depends on the instruction.

IOHI:



The IOHI places information coming from (or going to) a redrive into register R. The redrive is specified by the contents of the immediate data field of the instruction. The contents of register R and the immediate field depends on the instruction.

POLL (X'0')

This poll command is a broadcast command and its primary and secondary address fields are ignored. The poll command should only be used after an adapter L1 interrupt request has been detected by the CCU, otherwise an IOC bus timeout will result.

If an attached adapter indicates an L1 interrupt request to the CCU, the CCU sends a poll command in order to get the address of the redrive card with the L1 request. Only one redrive card will respond to the poll regardless of multiple L1 interrupt requests. Response priority is determined by the physical location of the redrive card on the IOC bus cables.

The data returned to the poll (board redrive response to poll or BRR) appears in register R as below:

Immediate Field (DB0 DB1 TA):

Bit	Function
0	0
1-4	1000 (RDV address)
5-7	Primary address
8	0 (broadcasting)
9-11	000 (command code)
12-14	Secondary address
15	1 (input)

Bit 8: This is set off to indicate that the command is transmitted to all redrives. The primary and secondary addresses are not taken into account.

Register R (DB0 DB1 TD):

Bit	Function
0	1
1	Enable/disable latch
2-4	Primary address
5-7	Secondary address
8-15	Redrive error register contents

READ ERROR REGISTER (X'1'OR X'9')

This command is used to read the contents of the error register of one specific redrive or of all redrives. The selected redrive or the redrive in error responds with an error status, which is loaded in register R.

Immediate Field (DB0 DB1 TA):

Bit	Function
0	0
1-4	1000 (RDV address)
5-7	Primary address
8	Not broadcast/broadcast
9-11	001 (command code)
12-14	Secondary address
15	1 (input)

Bit 8: This is set on for selecting one specific redrive via the primary and secondary addresses. It is set off to indicate that the command is transmitted to all redrives.

Register R (DB0 DB1 TD):

Bit	Function
0	1
1	Enable/disable latch
2-4	Primary address
5-7	Secondary address
8-15	Redrive error register contents

WRITE ERROR REGISTER (X'0' OR X'8')

This command is used to set bits in the error register of either one specific redrive or of all redrives. The error register is loaded with the contents of byte 0 of register R1. Byte 1 is not used.

R2 (DB0 DB1 TA):

Bit	Function
0	0
1-4	1000 (RDV address)
5-7	Primary address
8	Not broadcast/broadcast
9-11	000 (command code)
12-14	Secondary address
15	0 (output)

Bit 8: This is set on for selecting one specific redrive via the primary and secondary addresses. It is set off to indicate that the command is transmitted to all redrives.

Redrive Commands (Part 2 of 2)

DISABLE DRIVER (X'1'OR X'9')

This command inhibits the inputs to the redrives from the connected channel adapters or scanners, and from all the dependent redrives. However, communicating with the disabled redrives, and sending information to the dependent cards, remain possible. Register R1 is not used.

R2 (DB0 DB1 TA):

Bit	Function
0	0
1-4	1000 (RDV address)
5-7	Primary address
8	Not broadcast/broadcast
9-11	001 (command code)
12-14	Secondary address
15	0 (output)

Bit 8: This is set on for selecting one specific redrive via the primary and secondary addresses. It is set off to indicate that the command is transmitted to all redrives.

ENABLE DRIVER (X'2' OR X'A')

This command is used to enable the inputs to the addressed redrives from the connected channel adapters or scanners, and from all the dependent redrives. Register R1 is not used.

R2 (DB0 DB1 TA):

Bit	Function
0	0
1-4	1000 (RDV address)
5-7	Primary address
8	Not broadcast/broadcast
9-11	010 (command code)
12-14	Secondary address
15	0 (output)

Bit 8: This is set on for selecting one specific redrive via the primary and secondary addresses. It is set off to indicate that the command is transmitted to all redrives.

RESET REDRIVE (X'5'OR X'D')

This command resets the latches either of one specific redrive or of all redrives. The enable/disable latch is not reset.

R2 (DB0 DB1 TA):

Bit	Function
0	0
1-4	1000 (RDV address)
5-7	Primary address
8	Not broadcast/broadcast
9-11	101 (command code)
12-14	Secondary address
15	0 (output)

Bit 8: This is set on for selecting one specific redrive via the primary and secondary addresses. It is set off to indicate that the command is transmitted to all redrives.

DIAGNOSTIC WRITE (X'6' OR X'E')

This command tests the logical circuits that handle the priorities in either one specific redrive or of all redrives. The contents of byte 1 of register R1 are placed in the diagnostic register.

R2 (DB0 DB1 TA):

Bit	Function
0	0
1-4	1000 (RDV address)
5-7	Primary address
8	Not broadcast/broadcast
9-11	110 (command code)
12-14	Secondary address
15	0 (output)

Bit 8: This is set on for selecting one specific redrive via the primary and secondary addresses. It is set off to indicate that the command is transmitted to all redrives.

Register R (DB0 DB1 TO)

Bit	Function
0-7	Unused
8	Cycle steal grant high in
9	Cycle steal grant low in
10	Cycle steal request high in
11	Cycle steal request low in
12	Cycle steal request priority down
13	Cycle steal request priority up
14	Level 1 remember latch
15	Redrive level 1 pending

DIAGNOSTIC READ (X'A')

This command is used as a complement of the diagnostic write command to obtain the status of specific tags and latches of a specific redrive card. The contents of the diagnostic register are loaded in register R.

Immediate Field (DB0 DB1 TA):

Bit	Function
0	0
1-4	1000 (RDV address)
5-7	Primary address
8	1 (not broadcast)
9-11	010 (command code)
12-14	Secondary address
15	1 (input)

Register R (DB0 DB1 TD):

Bit	Function
0-7	Diagnostic register
8	Cycle steal grant high out
9	Cycle steal grant low out
10	Cycle steal grant high secondary
11	Cycle steal grant low secondary
12	Allow poll response secondary in
13	Allow poll response out
14	Select out primary
15	Select out secondary

Chapter 11. IOC Bus and Redrive

Section 2. Troubleshooting Guidelines

DC Voltages and Tolerances at Board Pin Level

Vdc	Vmin	Vmax	Ripple (max)
-12.0	-10.92	-13.20	0.45V p-p
-8.5	-7.73	-9.35	0.25V p-p
-5.0	-4.55	-5.50	0.15V p-p
-4.3	-4.19	-4.48	0.07V p-p
-1.5	-1.48	-1.56	0.03V p-p
+5.0 (Note 1)	+4.55	+5.50	0.20V p-p
+5.0	+4.75	+5.25	0.13V p-p
+8.5	+7.73	+9.35	0.35V p-p
+12.0	+10.92	+13.20	0.40V p-p
+12.0 (Note 2)	+11.40	+13.20	0.40V p-p
+24.0	+21.00	+27.60	0.30V p-p

Notes:

1. 02-PS7 only
2. 01-PS4 only

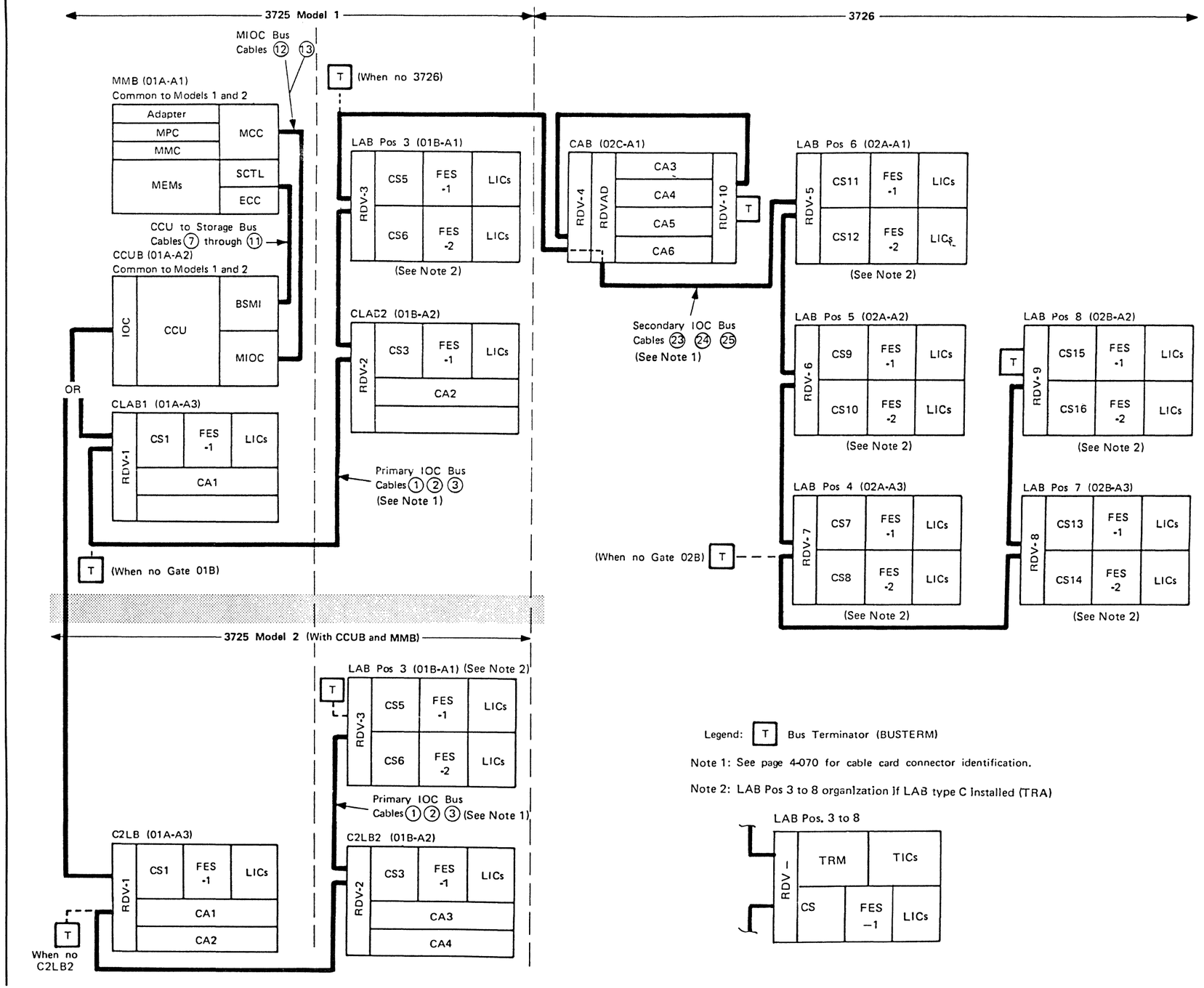
Troubleshooting Techniques

The purpose of this section is to help in isolating failures between RDVs, adapters, CCU, IOC bus cables, and terminators. The techniques provided are:

- RDV disconnection (board isolation) (See page 11-803)
- IOC Bus shortening (gate isolation) (See page 11-803)
- RDV Clocking checks (See page 11-804)
- Manual intervention scoping routines (See page 11-805)

Note: For the 3725 Model 2, only one redrive and the attached adapter can fail. Troubleshooting is therefore limited to the C2LB board.

Warning: Special care must be taken when disrupting or changing configurations on the IOC bus (See page 11-800).



Legend: T Bus Terminator (BUSTERM)

Note 1: See page 4-070 for cable card connector identification.

Note 2: LAB Pos 3 to 8 organization if LAB type C installed (TRA)

Warning Notes

CDF Change Warning

'Create CDF' should not be used when a failure is suspected in the system. It would destroy the file. To update the CDF, the 'CDF Update' facility should be used instead. This is particularly important when the RDV/IOC Bus area is suspected to be failing.

It is also important to keep one Service Diskette with its original CDF unchanged so that it can be used as a reference.

Channel Adapter Reset Warning

1. Channel adapters normally receive their power on reset signal on the IOC bus from the CCU via the RDV card.
2. If the path mentioned in (1) above is broken, the channel adapter cannot be reset.
3. A channel adapter that is not reset will cause problems on the host channel interface.
4. A power on reset can be forced by inserting the following jumpers:

   Model 1

	Jumper		
Board	From	To	Autoselect Cable
CLAB-1	01A-A3A2M12	Ground	N/A
CLAB-2	01B-A2A2M12	Ground	01B-A2YH
CAB	02C-A1X2M12	Ground	02C-A1YD

   Model 2

	Jumper		
Board	From	To	Autoselect Cable
C2LB	01A-A3A2M12	Ground	N/A
C2LB2	01B-A2A2M12	Ground	01B-A2YH

5. A channel adapter which has its normal power on reset path broken will cause autoselect problems on a functioning channel adapter. This problem can be eliminated by unplugging 02C-A1YD if the path is broken to the CAB board, or unplugging 02C-A1YD and 01B-A2YH if the path is broken to the CLAB2 board. If the path is broken to the CLAB1 board, none of the other channel adapters will function with autoselect. For the 3725 Model 2, the same considerations apply for the C2LB and C2LB2 board as those for the CLAB1 on CLAB2 board.

IOC Bus Terminator Power Warning

A blinking hex display of D2D will occur if the IOC bus terminator does not have its 5 volt power cable properly plugged onto the back of a board when the Power On Reset or Function Start switch is operated.

A blinking hex display of B1C will occur if IOC bus terminator voltage is plugged into any CLAB/CAB board position other than 0xx-xxB2E14.

Secondary Bus Enabling Warning

When using the manual intervention MB scoping routine, RDV-4 or the frame RDV must be enabled before addressing RDV 5 through 9 of the secondary bus.

Bus Configuration and RDV States

IOC BUS CONFIGURATION

The IOC bus is in reality composed of two buses as follows:

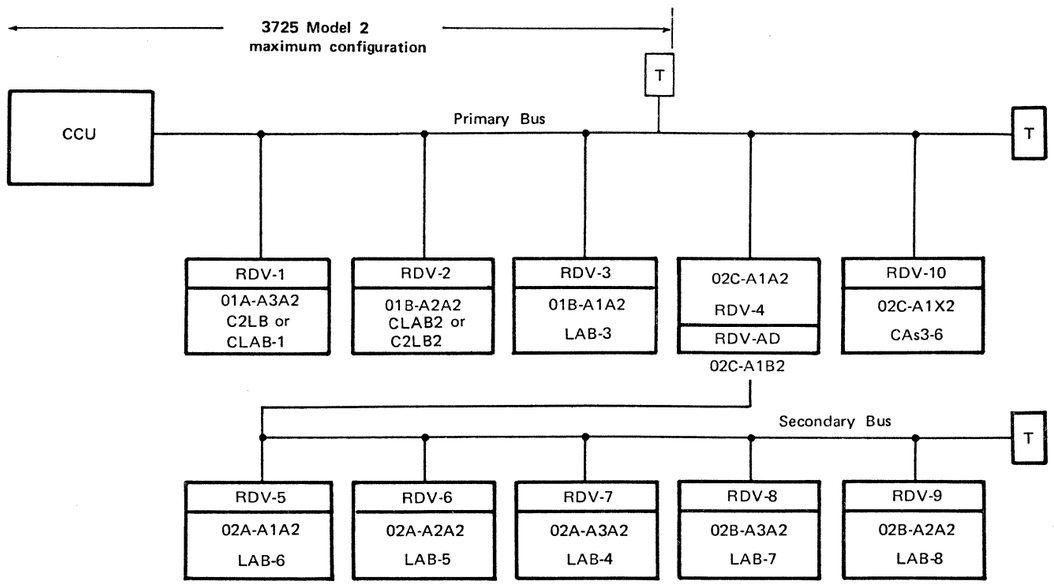
- The primary bus is driven by the DFL5 and DFL1-1, 1-2, and 1-3 cards of the CCU. The RDVs are connected to it in the following sequence:

  RDV - 1 for CLAB-1 or C2LB for the 3725 Model 2
  2 for CLAB-2 or C2LB2 for the 3725 Model 2
  3 for LAB-3
  4 which drives the secondary bus
  10 for CAB

- The secondary bus is driven by the RDV-4/RDV-AD of the primary bus, and installed in the CAB. The RDVs are installed in the following sequence:

  RDV - 7 for LAB-4
  6 for LAB-5
  5 for LAB-6
  8 for LAB-7
  9 for LAB-8

This is the installation sequence, due to the need for cooling from the bottom to the top of gate 02A. The physical sequence on the bus is shown in the following figure:



REDRIVE STATES

Each RDV may be in one of the following states (see "Redrive State Definitions" on page 11-090 for more details):

1. Enabled

   This is the case of normal operation. The attached adapters are logically connected to the RDV. An RDV is enabled via an 'Enable' command, which can address either a specific RDV or all the RDVs (broadcast command).

2. Disabled

   The attached channel adapters cannot communicate with the IOC Bus. Their inputs to the RDV are inhibited. However, the communication between the RDV and the CCU is not inhibited. An RDV is disabled via a 'Disable' command, which can address either a specific RDV or all RDVs (broadcast command).

   For example, when doing a Power On/Reset, all RDVs are disabled.

3. Board Disabled (Jumper Required)

   This requires a jumper from pin D11 of the board RDV to ground. The RDV does not reply to any command when addressed.

   For example a 'Create CDF' command will not consider that RDV as present. However, the IOC bus chain is not interrupted.

4. Disconnected (Top Connectors Unplugged)

   When unplugging top connectors, the following cable pins must be jumpered (see page 4-090).

   4B02 to D02
   4B03 to D03
   5B02 to D02
   5B03 to D03

A disconnected RDV must be deactivated from the CDF and must have jumpers installed in cables in order for the IOC bus IFTs to run without error. For the last redrive of the primary and secondary IOC bus, the jumpers are not required.

IOC Bus IFT Diagnostics

The IOC Bus is tested by four sections I, JA, JB, and JC.

SECTION I

Section I tests successively all RDVs, first with the RDVs disabled from their adapter, and then with them enabled. It is not possible to run section I for a specific RDV.

Note: To run section I, the first RDV at least must be connected to the 3725.

SECTION J

Section J tests the path up to the adapter. The associated RDV is of course enabled but the others are disabled.

For example RDV-4, which drives the secondary bus, is enabled when testing RDV-5 to RDV-9 or their associated adapters.

SECTION JA

Section JA can be run for the CSPs on a given RDV as follows:

Request : Diag ==> JA Adp^{J} ==> RDV^{J} Line ==>

Note: the adapter and line selection entries are used for troubleshooting only.

SECTION JB

Section JB is run for testing the channel adapters. The disconnect procedure used by routine IA01 correlates the errors between the first and second RDVs of the Primary or Secondary buses.

JA AND JB FAILING

If both are failing, the whole primary bus is considered to be failing. The suspected FRUs are DFL5, DFL1-1, DFL1-2, and DFL1-3, and the terminator with its powering.

The same correlation is done for the two RDVs closest to RDV-4 for the secondary bus (RDV-6 and RDV-7 or RDV-5 and RDV-6). If both are failing, the whole secondary bus is considered failing. The suspected FRUs are RDV-4, RDV-AD, and the terminator with its powering.

SECTION JC

Section JC tests the IOC bus to the TRM, with the ability to write into and read from the TRM.
The capability to generate a Level 2 interrupt is also tested. The RDV for a specific TRM is enabled during the test, all others are disabled.
It is not possible to run Section JC for a specific TRM.

IOC Bus Isolation

Two different methods are used to isolate failures between the RDVs, the RDV attached adapters, CCU, IOC bus cables, and the terminators; these methods can be used either separately or successively.

RDV DISCONNECTION

This method is used with RAC 0A0 repair procedure. It consists in unplugging the top connectors of a RDV (do not forget to connect the jumpers) in order to disconnect a complete board: CLAB-1, CLAB-2, LAB-A, LAB-B, or CAB for the 3725 Model 1; C2LB, C2LB2, LAB-A, LAB-B for the 3725 Model 2. Observe all the warnings given on page 11-800.

To run section I which tests all RDVs, the CDF must be updated with the new configuration. To do this, the RDV which has been unplugged must be deactivated in the CDF (CDF update).

To run section J, the adapters attached to the unplugged RDV must also be deactivated in the CDF. It is possible to run section JA for a specific RDV without updating the CDF. This allows the checking of the connection of each TSS. Via successive runs of section JA, all connected TSSs can be checked without manipulating the CDF.

However, the channel adapter connection to the RDV is not tested.

Section JB of the IOC diagnostics handles all CAs.
See page 11-800, "Channel Adapter Reset Warning", paragraph 5, for auto select cable disconnection.

It is possible to test a specific channel adapter using the channel adapter diagnostics, section L. It is also possible to test the path from the CCU to a TSS via a specific TSS request which loads the TSS diagnostics into CSP storage.

IOC BUS SHORTENING

Depending on the system configuration the IOC bus can be shortened in several independent steps (pages 4-090 and 4-091). Observe all the warnings given on page 11-800.

1. Secondary bus disconnection, which can be done in two ways:

   Disconnect RDV-4 (jumper 02C-A1A2D11 to D08)
   or
   Unplug RDV-4 top connectors

   In both cases RDVs 4 through 10 must be deactivated from the CDF to run the diagnostic IFT I.

2. Secondary bus shortening by disconnecting gate 02B:

   Move bus terminator from: 02B-J1
   to : 02A-J1
   Put power jumper to : 02A-A3 B2E14 --> black wire = ground
   B3E01 --> yellow wire = +5V
   (see page 4-290)

   Warning:

   RDV 8 and 9 must be deactivated from the CDF. If running the scanner diagnostics, scanners 13 through 16 must also be deactivated from the CDF.

3. Primary bus shortening by disconnecting CAB (02C-A1).

   Disconnect one end of cable P/N 6081182 and one end of cable P/N 4712960 from 01B-J1A2. Disconnect cable P/N 4712959 from the pin side of 01B-A2YK.

   Move bus terminator from: 02C-J1
   to : 01B-J1

   Install loose end of cable P/N 6081389 into 01B-J1A2.

   Warning:

   All RDVs except RDVs 1, 2, and 3 must be deactivated from the CDF. If running the scanner or channel adapter diagnostics, scanners 7 through 16 and channel adapters 3 through 6 must also be deactivated from the CDF.

4. Primary bus shortening by disconnecting gate 01B:

   Move bus terminator from: 01B-J1
   to : 01A-J1

   Remove the BUSTERM power cable P/N 6081389 from 01B-J1A2 and 01B-A2B2E14, and remove cable 21 from position 01A-A1YE (see page 4-070). Replug cable P/N 6081389 to 01A-J1A2 and 01A-A3B3E14 (see page 4-290).

Warning:

All RDVs except RDV-1 must be deactivated from the CDF. If running the scanner or channel adapter diagnostics, scanners 3 through 16 and channel adapters 2 through 6 must also be deactivated from the CDF.

RDV Clocking Checks

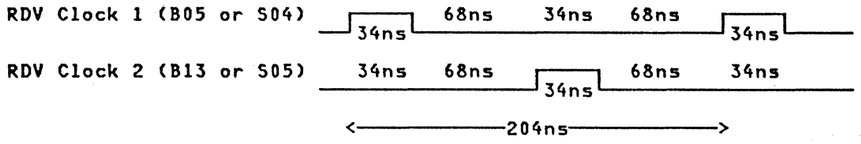
The RDV card receives the high speed clock signals (29.4912 MHz) from the ACLK card as shown in the diagrams on pages 4-100 and 5-051.

Also RDV-10 of the CAB receives the high speed clock signals from RDV-4 via a twisted pair, as follows:

Signal name	Wire color	RDV-4	RDV-10
-29.4912 MHz Oscillator	Red	02C-A1A2D07	02C-A1X2B04
+29.4912 MHz Oscillator	Black	02C-A1A2D05	02C-A1X2B02

The RDV card also generates the CLK1 through CLK4 signals as shown on pages 5-053 and 5-054.

RDV Clock 1 and RDV Clock 2 may be scoped as follows:



PIO Scoping Routine

The purpose of the PIO scoping routine is to allow scoping of the PIO tags and data bus for the 3725, and runs on level 4 in the CCU.

The following adapters may be exercised using the normal PIO:

- Redrive cards
- Scanners
- Channel adapters
- Token-ring subsystem

RUNNING THE ROUTINE

1. Start the routine by calling manual intervention routine MB01.
2. The first prompting message requests the operator to enter 'CA' to test the channel adapter autoselection; to scope the PIO, ignore this message and simply press SEND.
3. The second prompting message requests the operator to enter 'CE' to test the channel adapter cycle steal; to scope the PIO, ignore this message and simply press SEND.
4. When prompted, enter the parameters in the following format (see Notes 2 and 3 below):

   'Rooaaaadddd'

   where:

   oo = the selected option

   aaaa = the selected address (TA value)

   dddd = the selected data (TD value)

5. After all entries have been made, see timing charts on page 11-812 for scoping basic PIO operations.

Notes:

1. The address on the bus at TA time is used to address an adapter; to avoid timeouts on the bus, this address must be correct.
2. The redrive for a CSP, channel adapter, or TRA must be enabled before a PIO routine can address or be run for that CSP, channel adapter, or TRA. See page 11-808 for examples.
3. When using the manual intervention scoping routine MB01, RDV-4 (frame RDV) must be enabled before addressing RDV-5 through RDV-9 of the secondary bus.

Option

The option 'oo' may be one of the following values:

01 The requested PIO is executed once only. The '01' option may be repeated as often as required.

02 The requested PIO is executed and loops on the PIO until an error is found. The error is reported once only via an RAC 675. If you wish to continue, type 'G' (for Go); the routine then loops indefinitely until the 'BREAK' key is pressed.

03 This option is identical to option 02 above, except that if an error is found, it is not reported.

04 This option is reserved for the CSP or TRA, and is used to scope level 1 and level 2 interrupt requests to the CCU (adapter level 1 and adapter level 2).

TA and TD

The remaining parameters are 'aaaa' (TA) and 'dddd' (TD). A preliminary check is done on the TA value; a null value is not allowed. In addition, for option 04, the routine checks that TA is X'3001' for the get line ID command, and X'nn41' for the get error status command on a given CSP. For TA format details, see the next section.

Inputting the Values

Each request must always contain 10 hexadecimal digits. If, by error, fewer than 10 digits are entered, or if a wrong option ('oo') is requested, the following error message is displayed:

INVALID REQUEST: xxxxxxxxxx PRESS SEND

where xxxxxxxxxx are the digits entered in error.

After pressing SEND, the prompt re-appears, and the request may be re-entered.

PIO Scoping Address and Commands

ADDRESS (TA) FORMATS

The address format depends on the type of adapter addressed. Refer to pages 11-040, 11-100, and 11-101 for additional information.

Redrive TA Format

Note: a redrive does not need to be enabled for a PIO routine to address or to be run for that redrive.

The redrive TA has the following format:

0	1 0 0 0	P P P	X X X X	S S S	I/O
	Group Address	PRA	Command	SRA	
0	1 4	5 7	8 11	12 14	15

RDV	0	Group Address	PRA	Command	SRA	I/O	TA
1	0	1 0 0 0	0 0 0	X X X X	0 0 0	1/0	40X0/1
2	0	1 0 0 0	0 0 0	X X X X	0 0 1	1/0	40X2/3
3	0	1 0 0 0	0 0 0	X X X X	0 1 0	1/0	40X4/5
4	0	1 0 0 0	0 0 1	X X X X	0 0 0	1/0	41X0/1
10	0	1 0 0 0	0 0 0	X X X X	0 1 1	1/0	40X6/7
5	0	1 0 0 0	0 0 1	X X X X	1 0 1	1/0	41XA/B
6	0	1 0 0 0	0 0 1	X X X X	1 0 0	1/0	41X8/9
7	0	1 0 0 0	0 0 1	X X X X	0 1 1	1/0	41X6/7
8	0	1 0 0 0	0 0 1	X X X X	1 1 0	1/0	41XC/D
9	0	1 0 0 0	0 0 1	X X X X	1 1 1	1/0	41XE/F
		Byte 0		Byte 1			

Notes:

1. Bit 0 is always 0.
2. Bits 1 through 4 are always binary '1000'
3. Bits 5 through 7 comprise the primary redrive address (PRA), and are always binary '000' for RDVs 1 through 3 and 10, and binary '001' for RDVs 4 through 9.
4. 'XXXX' is the command, and is described below.
5. Bits 12 through 14 comprise the secondary redrive address, (SRA) and depend on the RDV.
6. Bit 15 is the I/O bit; this bit is 0 for output and 1 for input.

COMMAND FORMAT

The command occupies bits 8 through 11 of the TA. It has the following format (the I/O bit is also shown for convenience):

Command	Bit 8*	9	10	11	I/O Bit
Poll	0	0	0	0	1
Read error reg.	0/1	0	0	1	1
Write error reg.	0/1	0	0	0	0
Disable drivers	0/1	0	0	1	0
Enable drivers	0/1	0	1	0	0
Reset redrive	0/1	1	0	1	0
Diagnostic write	0/1	1	1	0	0
Diagnostic read	1	1	0	1	1

* bit 8 is the specific bit. If the bit is off, the command is broadcast to all redrives; if on, it is sent to a specific redrive. See page 11-808 for examples.

Error Register in the Redrive

To write data into the RDV error registers, 'dddd' must be set as follows:

xx00 where xx is one byte of data (00 is not used)

To read data from an RDV error register, the data is placed into register in X'76' as follows:

aadd where aa is the RDV address
dd is the error register content

Redrive TD Format

For a diagnostic write, only byte 1 of the TD field is used (dddd = 00xx).

Note: Bit 2 and/or bit 3 of byte 1 should not be set on as they will cause an unexpected error in MB01 from which recovery is only possible via a MOSS re-IML.

PIO Scoping, Redrive Cards

REDRIVE CARD SCOPING

To write a data pattern into the RDV error register and then read it back, the following patterns may be used (this example is for RDV1):

R014080FF00 First request: write 'FF' into the error register of RDV1.

R0240910000 Second request and loop until an error is found: read back the error register of RDV1.

Alternatively:

R0340910000 Second request and loop indefinitely.

REDRIVE CARD SCOPING (VIA BOARD RDV-4)

This example is the same as example 1, except that as the RDV card being scoped is in the 3726, RDV4 (the frame redrive) must be enabled as well (this example is for RDV5):

R0141A00000 First request: enable RDV4.

R01418AFF00 Second request: write 'FF' into the error register of RDV5.

R02419B0000 Third request and loop until an error is found: read back the error register of RDV5.

Alternatively:

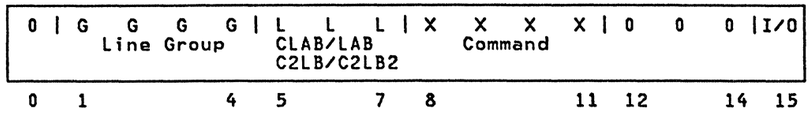
R03419B0000 Third request and loop indefinitely.

PIO Scoping Scanners

CSP TA FORMAT

The redrive for a CSP must be enabled before a PIO routine can address or be run for that CSP. In addition, RDV-4 must be enabled before RDVs 5 through 9 can be addressed.

The CSP TA has the following format:



CSP	0	Line Group				CLAB/LAB C2LB/C2LB2			Command				Bits 12-14			I/O	TA
1	0	0	0	1	0	0	0	0	X	X	X	X	0	0	0	1/0	10X0/1
1	0	0	1	0	0	0	0	0	X	X	X	X	0	0	0	1/0	20X0/1
3	0	0	0	1	0	0	0	1	X	X	X	X	0	0	0	1/0	11X0/1
3	0	0	1	0	0	0	0	1	X	X	X	X	0	0	0	1/0	21X0/1
5	0	0	0	1	0	0	1	0	X	X	X	X	0	0	0	1/0	12X0/1
5/6	0	0	1	0	0	0	1	0	X	X	X	X	0	0	0	1/0	22X0/1
7	0	0	0	1	0	0	1	1	X	X	X	X	0	0	0	1/0	13X0/1
7/8	0	0	1	0	0	0	1	1	X	X	X	X	0	0	0	1/0	23X0/1
9	0	0	0	1	0	1	0	0	X	X	X	X	0	0	0	1/0	14X0/1
9/10	0	0	1	0	0	1	0	0	X	X	X	X	0	0	0	1/0	24X0/1
11	0	0	0	1	0	1	0	1	X	X	X	X	0	0	0	1/0	15X0/1
11/12	0	0	1	0	0	1	0	1	X	X	X	X	0	0	0	1/0	25X0/1
13	0	0	0	1	0	1	1	0	X	X	X	X	0	0	0	1/0	16X0/1
13/14	0	0	1	0	0	1	1	0	X	X	X	X	0	0	0	1/0	26X0/1
15	0	0	0	1	0	1	1	1	X	X	X	X	0	0	0	1/0	17X0/1
15/16	0	0	1	0	0	1	1	1	X	X	X	X	0	0	0	1/0	27X0/1

Notes:

1. Bit 0 is always 0.
2. Bits 1 through 4 select the line group (16 lines). Binary '0010' selects the first group of 16 lines on a CLAB or LAB board (corresponds to the first CSP on the board); binary '0100' selects the second group of 16 lines (corresponds to the second group of 16 lines if only one scanner is installed on the board, or to the second CSP if the board is a LAB type B).
3. Bits 5 through 7 comprise the LAB address.
4. Bits 8 through 11 comprise the command. See Notes 7, 8, and 9 for the commands that can be exercised.
5. Bits 12 through 14 are always 0 in this application.
6. Bit 15 is the I/O bit; this bit is 0 for output and 1 for input.
7. To clear level 2 interrupt requests from the CSP to the CCU, use the broadcast 'Get Line ID' command; the corresponding TA format is X'3001'.
8. To clear level 1 interrupt requests from the CSP to the CCU, use a read type of PIO with a command of binary '0100'.
9. The RDV for the selected adapter must be enabled prior to testing.

CCU L1 AND L2 INTERRUPTS FROM ADAPTER SCOPING

WARNING: to perform the following function on an CSP, reset it by touching board pin A2M05 to ground.

To scope the up and down levels of level 1 and level 2 interrupt requests from a CSP to the CCU, the following patterns may be used:

R0140A00000 First request: enable RDV1.

R011020FFFF Second request: ask CSP 1 to send a level 2 interrupt request. At this point, the interrupt request level 2 line should go to the up level.

R0430010000 Third request (Get Line ID): pressing SEND should cause the interrupt request level 2 to be cleared; the line goes to the down level.

R0110810000 Fourth request: ask CSP 1 to send a level 1 interrupt request. At this point, the interrupt request level 1 line should go to the up level.

R0410410000 Fifth request (Get Error Status): pressing SEND should cause the interrupt request level 1 to be cleared; the line goes to the down level.

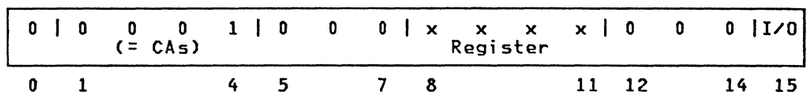
PIO Scoping, Channel Adapters

CHANNEL ADAPTER TA FORMAT

WARNING the ENBL/DSBL switches on the control panel must all be set to DSBL.

Note: The redrive for a channel adapter must be enabled before a PIO routine can address or be run for that channel adapter. For the 3725 Model 2, only CA1, CA2 CA3, and CA4 can be installed, and no TPS feature is available. Therefore, the Enbl/Dsbl switches can be used only for interface A.

The CA TA has the following format:



Notes:

1. Bit 0 is always 0.
2. Bits 1 through 4 are always binary '0001' to select the channel adapters.
3. Bits 5 through 7 are always 0.
4. 'xxxx' is the CA register that is to be read or written. For a channel selection, 'xxxx' is '0111' (see page 12-020 onward) for channel adapter register bit descriptions.
5. Bits 12 through 14 are always 0.
6. Bit 15 is the I/O bit; this bit is 0 for output (Out X'x') and 1 for input (In X'x').
7. The CA is selected by an Out X'7' command with the address contained in TD as follows:

CA	TD
1	X'2000'
2	X'2200'
3	X'2400'
4	X'2600'
5	X'2800'
6	X'2A00'

   This selection must be performed before any other register access.
8. The redrive for the tested channel adapter must be enabled before selecting the channel adapter.

To perform an Out X'x' or an In X'x' after the channel has been selected, use the TA value given above. The local storage and register contents are then given by the TD value.

Data loaded into CA registers may be checked using CCU function 10 (ten).

Note: Do not forget to restart the CCU before leaving the CCU function.

CHANNEL ADAPTER SCOPING

To write a data pattern of X'FFFF' into local store X'4' (Out X'4') of a channel adapter and then read it back, the following patterns may be used (this example is for channel adapter 1):

```
R0140A00000     First request: enable RDV1.
R0108702000     Second request: select channel adapter 1.
R010840FFFF     Third request: write X'FFFF' into local store X'4' of CA1.
R0208410000     Fourth request: loop on In X'4'.
```

PCI LEVEL 3 INTERRUPT FOR A GIVEN CHANNEL ADAPTER SCOPING

To set up a PCI level 3 interrupt on channel adapter 1 for scoping, the following patterns may be used:

```
R0140A00000     First request: enable RDV1.
R0108702100     Second request: select and reset channel adapter 1.
R0108702000     Third request: select channel adapter 1.
R0208700040     Fourth request: Set the PCI at level 3 for channel adapter 1 and loop until
                an error is found.
```

PIO Scoping, Error Reporting

ERROR REPORTING

The following RAC-ERC codes are displayed when an error is found by the scoping routine:

RAC	ERC	Meaning
670	-	This is an ABEND. A Level 1, 2, or 3 interrupt occurred, but the cause cannot be identified. Action: Retry or run CCU diagnostics.
675	77F9	A Level 2 interrupt occurred and has been identified. However this level 2 interrupt cannot be reset. Action: Terminate the routine and retry.
675	7FF0	Error occurred on a PIO for a single request execution (Option 1). Action: Compare the error bits with the error field in the table at right. Type G to continue.
675	7FF1	Error occurred on a PIO with loop requested. Action: Compare the error bits with the error field in the table at right. Type G to continue.
675	7FF9	A Level 1 interrupt occurred and has been identified. However this level 1 interrupt cannot be reset. Action: Terminate the routine and retry.

REGISTER IN X'76'

When an error occurs with RAC = 675 and ERC = 7FF0/7FF1, the error bit field displayed is the contents of In X'76'. This register latches the error conditions found during PIO execution. The following combinations may occur:

Error field	Suspected IOC Line*	Meaning
X'0400'	Data bus bits	IOC Bus parity check
X'0800'	VH	I/O tag is off
X'1800'	IRR	I/O tag raised on a new IOH
X'2800'	IRR, VH, EOC	No response to TA
X'4800'	IRR	VH did not fall after TD dropped
X'5800'	IRR, VH, EOC, VB, M	No response to TD for PIO read
X'6800'	IRR, VH, EOC, VB, M	No response to TD for PIO read
X'F800'	VH	I/O tag is off, VH must rise (PIO end after a data exchange)

Where:

Data bus bits = bytes 0 and 1, bits 0 through 7 and parity
EOC = end of chain
IRR = interrupt request removed
M = modifier
VB = valid byte
VH = valid halfword

PIO Scoping TRAs

TRA TA FORMAT

The redrive for a TRA must be enabled before a PIO routine can address or be run for that TRA. In addition, RDV-4 must be enabled before RDVs 5 through 9 can be addressed.

The CSP TA has the following format:

0	G	G	G	G	L	L	L	X	X	X	X	0	0	0	I/O
	Group Address				LAB			Command							
0	1			4	5		7	8			11	12		14	15

TRA	0	Group Address				LAB			Command				Bits 12-14			I/O
6	0	1	0	0	1	0	1	0	X	X	X	X	X	X	0	1/0
8	0	1	0	0	1	0	1	1	X	X	X	X	X	X	0	1/0
10	0	1	0	0	1	1	0	0	X	X	X	X	X	X	0	1/0
12	0	1	0	0	1	1	0	1	X	X	X	X	X	X	0	1/0
14	0	1	0	0	1	1	1	0	X	X	X	X	X	X	0	1/0
16	0	1	0	0	1	1	1	1	X	X	X	X	X	X	0	1/0

Notes:

1. Bit 0 is always 0.
2. Bits 1-4 select the Group Address. Binary '1001' selects the token-ring subsystem. The specific TRM is selected by the LAB address bits.
3. Bits 5 through 7 comprise the LAB address.
4. Bits 8 through 11 comprise the command. See Notes 7, 8, 9, and page 11-040 for the commands that can be exercised.
5. Bits 12-14:
   Bit 12 determines MOSS or CCU. 1 = MOSS IOH 0 = CCU IOH
   Bit 13 determines TRM or TIC. 1 = TRM 0 = TIC
   Bit 14 is always 0 for this application.
6. Bit 15 is the I/O bit; this bit is 0 for output and 1 for input.
7. To clear level 2 interrupt requests from the TRA to the CCU, use the broadcast 'get line ID' command; the corresponding TA format is X'3001'.
8. To clear level 1 interrupt requests from the TRA to the CCU, use a read type of PIO with a command of binary '1101' (Get line ID Error Status).
9. The RDV for the selected adapter must be enabled prior to testing.

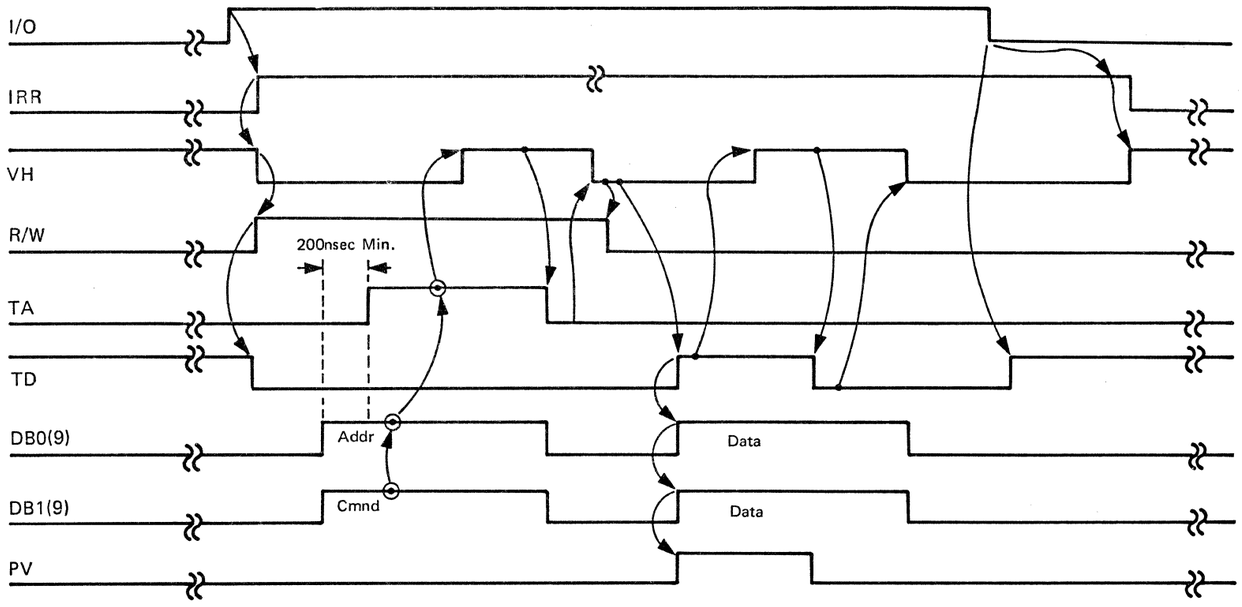
TRM CARD SCOPING (VIA BOARD RDV-3)

This example allows scoping of the IOC Bus, via redrive 3, to the TRM in Adapter position 6.

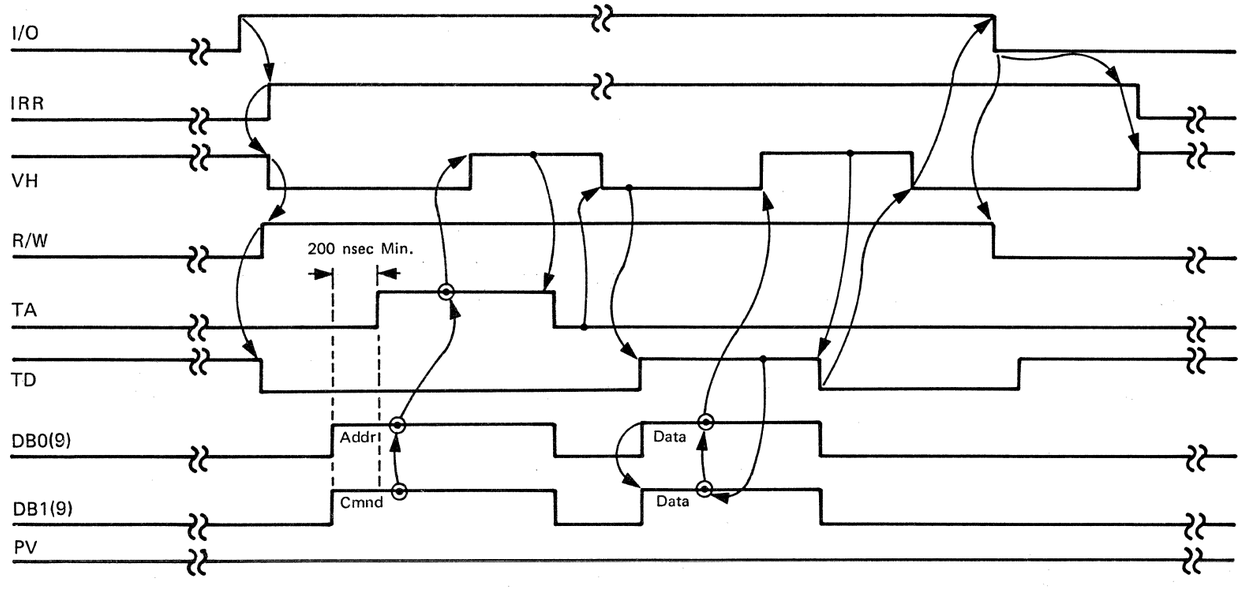
R0140A40000	First request: enable Redrive 3.
R014A150000	Second request: Get TRM control registar to allow communication with the TRM.
R014A34FFFF	Third request: Write FFFF (all bits) into the LID base register.
R024A350000	Fourth request: Loop until error is found. Read back data from LID base register.
	or
R034A350000	Fourth request: Loop indefinitely, do not stop on error. Read back data from LID base register.

PIO Timings

READ TIMING



WRITE TIMING



IOC Bus Signal Scoping

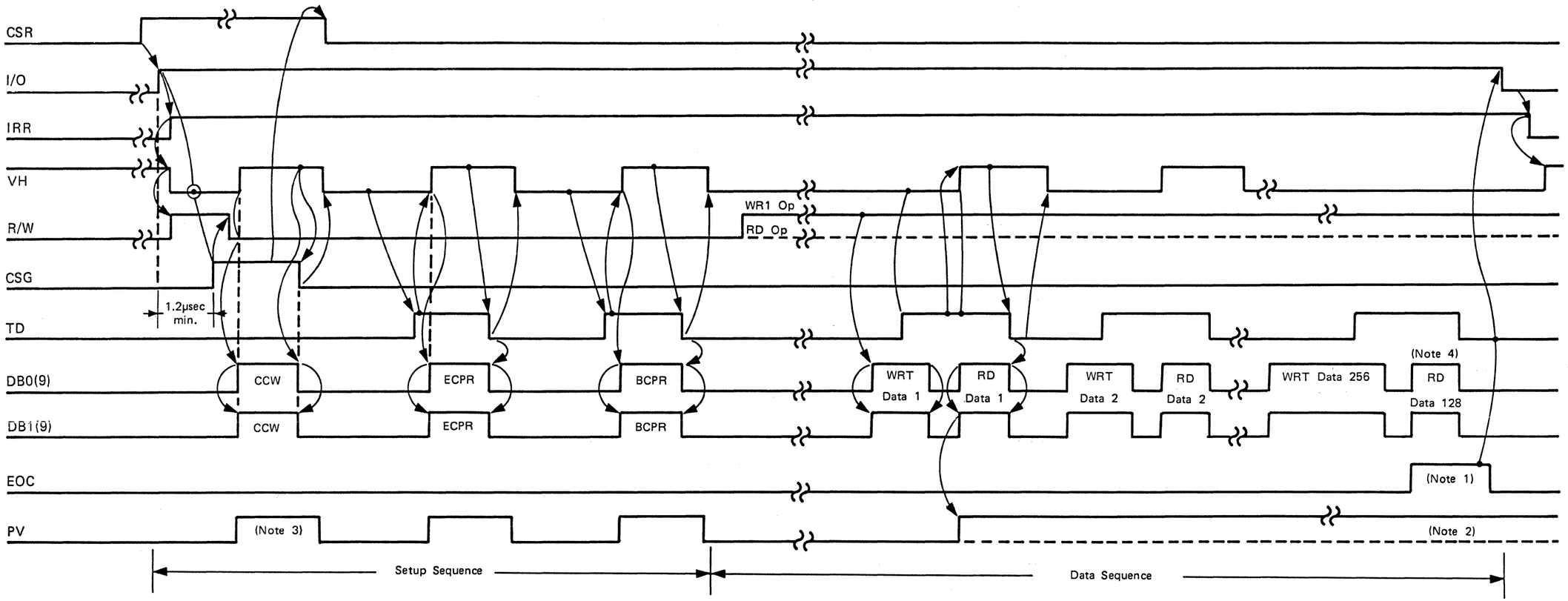
Scoping the signals of the IOC bus can be performed:

- On the BUSTERM cards (see page 4-092 for pin assignment)
- On the CCU board at the IOC cable connector 01A-A2A3 (see page 4-080 for pin assignment).

Notes :

1. ⊙ Indicates the checking of that signal level before proceeding.
2. These diagrams shown in positive logic. Actual logic levels are active when negative.

Cycle Steal Timing



Notes :

1. EOC in place of VH indicates last transfer.
2. Parity valid active only for RD data and RD control data during setup sequence.
3. The parity valid tag may be off, but the CCW always has good parity.
4. The data exchange is always of a single type only (read or write) during any given AIO operation.

IOC Bus Adapter Interrupts

Three interrupt levels 1, 2, and 3 are raised by the IOC bus adapters. The interrupts are placed on the IOC data bus by the adapter when the I/O tag is inactive.

Data Bus Byte 0 bit 1 = L2 interrupt request to CCU
Data Bus Byte 0 bit 5 = L1 interrupt request to CCU
Data Bus Byte 1 bit 0 = L3 interrupt request to CCU

The interrupt requests are removed from the data bus by the RDV logic when the adapters activate the 'interrupt request removed' (IRR) tag.

LEVEL 1 INTERRUPT REQUEST

This level is raised by the IOC logic (DFL-5 card in the CCU), the CSPs and the CAs. The RDV does not raise a level 1 interrupt request. The interrupt request is displayed in CCU registers X'7E' and X'76' (see page 10-230). The IOC1-L1 Summary bit in X'7E' indicates that there is an 'IOC level 1 interrupt request' displayed in X'76'.

When a CSP raises a level 1 interrupt request, it is latched in the RDV and sent to the CCU. Polling the RDVs gives the address and error register of the first RDV on the daisy chain which raises the interrupt request.

The CCU action for a CSP error is described on pages 13-330 and 13-331.

The interrupt request for the CSP is reset by a Get Error instruction.

To test the rise and fall of an interrupt request level 1 (A2U06 of the RDV) to the CCU, use the PIO scoping routine MB01 with option 04. This option also shows that the adapter is replying to a request coming from the CCU. See page 11-808 for a sample procedure.

INTERRUPT LEVEL 2 TEST

Level 2 interrupt requests are initiated by the CSP microcode through the CSP external register X'05' bit 3, which enters the CSP and RDV autoselection mechanism.

The CCU sends a Get Line ID broadcast read command to get the CSP address. The interrupt condition XR05 bit is reset by the microcode.

To test the rise and fall of interrupt request level 2 (A2J02 of RDV) to the CCU, use the PIO scoping routine MB01 with option 04. This option also shows that the adapter is replying to a request coming from the CCU. See page 11-808 for a sample procedure.

LEVEL 3 INTERRUPT REQUEST

A level 3 interrupt request is initiated by the CA and is sent to the CA autoselection mechanism.

The CCU must send an Out X'7' instruction to enable the autoselect mechanism and an In X'F' instruction to get the CA address. The interrupt request is reset by an Out X'7' instruction. See page 11-809 for a sample procedure.

SCANNER INTERRUPT REQUEST TO THE MOSS

Each scanner can interrupt the MOSS on level 4 by setting bit 0 of XR05 in the CSP. This interrupt is dotted for all the scanners on the IOC bus and goes to the MCC in the MOSS via the CCU board. The signal routing is as follows:

	CSP-2	RDV	CCU 01A-A2 Connectors*	MOSS 01A-A1 Connector	MCC
-Interrupt to MOSS -Scanner Interrupt to MOSS	B10	A2U09 A2Y30	A3B10 V2C02	V6A02	V2B07

* Not scopable

If permanent scanner interrupt at level 4 to the MOSS is suspected, it may be checked by using the signal routing diagrams in the cable charts starting on page 4-080 and the pin/net list (YZ pages). Missing interrupts from a given scanner are checked by diagnostics JA and the TSS diagnostics. If the level 4 interrupt to the MOSS does not occur, the MOSS DCM displays RAC 039.

Contents

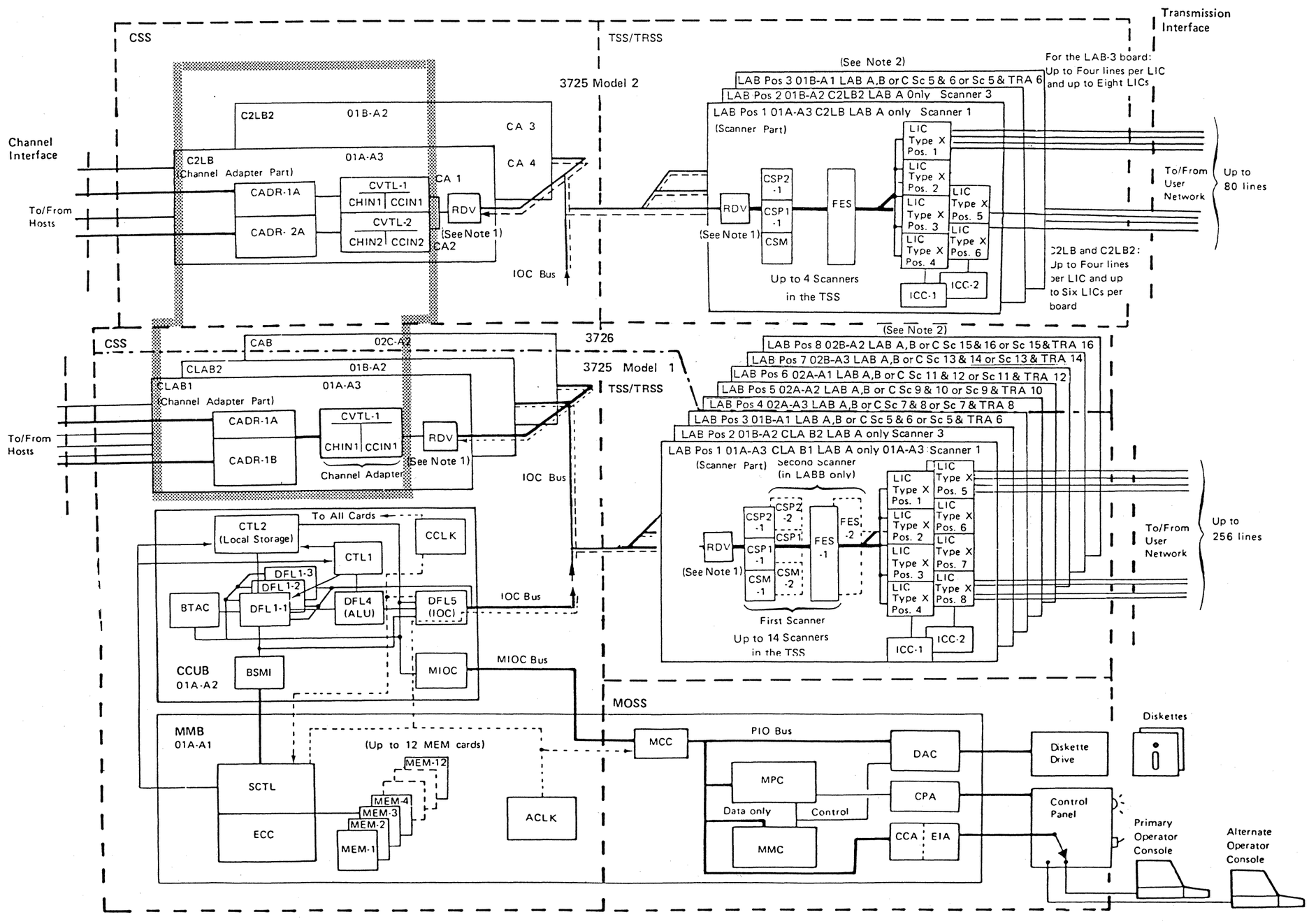
Chapter 12. Channel Adapter

Section 1. Unit Description

Contents

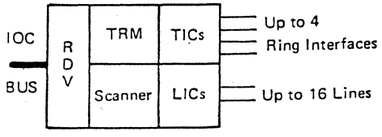
Contents

Channel Adapter in 3725 Data Flow



Note 1: As the CLAB and the CL2B boards are split into two parts in this figure, one for the channel adapter and one for the scanner, the redrive (RDV) function is shown twice for clarity. However, there is one RDV per CLAB or C2LB board.

Note 2: LAB Pos. 3 to 8 organization if LAB type C installed



Legend:

------ Clock signals

——— Data/control signals

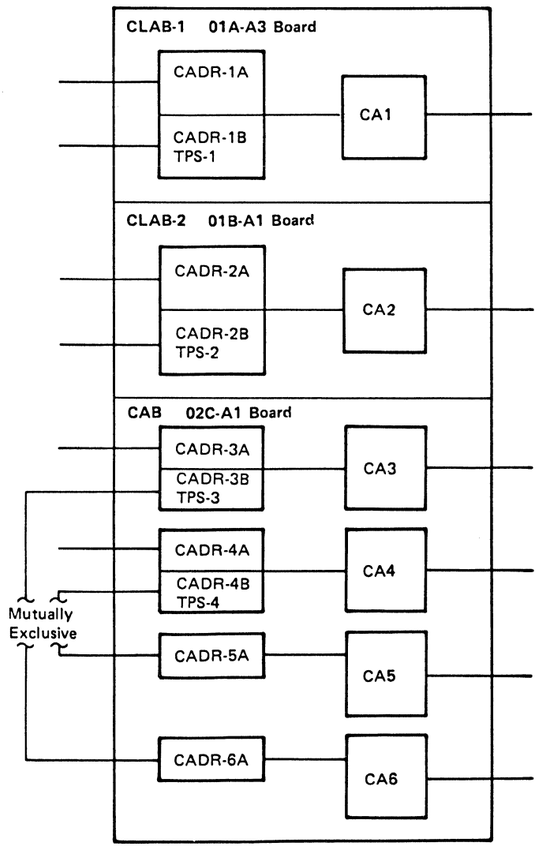
Introduction

The 3725 channel adapter (CA) allows the 3725 Communication Controller to be attached to the selector, block multiplex, or byte multiplex channels of one or more host processors (for example, IBM System/370 or IBM 3030, 4300, or 3081 Processors).

3725/3726

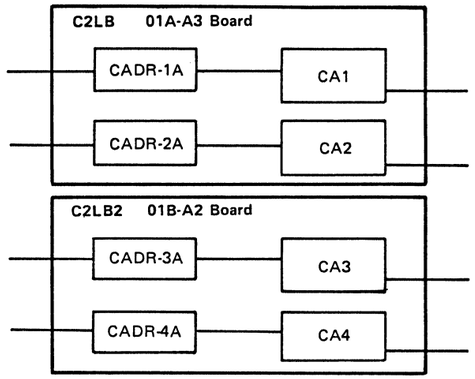
As shown in the following figure, up to six channel adapters may be installed in a 3725. In addition, CA#1 through CA#4 may have a two-processor switch (TPS) installed. The TPS allows a single channel adapter to connect to two separate host channels or to two channels of a single host.
The following restrictions apply to TPS installation:

1. If CA#5 is installed, CA#4 cannot have a TPS.
2. If CA#6 is installed, CA#3 cannot have a TPS.



3725 MODEL 2

For the Model 2, up to four channel adapters may be installed and the TPS feature is not available.



CARD DESCRIPTIONS

Each channel adapter contains four logic cards with a fifth card added for a TPS. The logic cards are defined below.

Channel-to-CCU Interconnection (CCIN) Card

This card contains logic that enables the channel adapter to communicate with the CCU via the IOC bus through a redrive card. The CCIN card interacts with the CHIN card to pass channel data, status, and sense information to or from the CCU.

Channel Interface (CHIN) Card

This card contains timing, tag controls, and data buffering logic that enables the CADR card to communicate over the channel interface.

Channel Adapter Driver Receiver (CADR) Card

This card contains channel interface driver and receiver logic.

Card Vendor Transistor Logic

This card contains supplemental CCIN, CHIN, and CADR timing, and control logic.

SOCKET LOCATION

3725/3726

The card socket locations for each channel adapter are shown in the following table.

Card Name	CA#1 01A-A3	CA#2 01B-A1	CA#3 02C-A1	CA#4 02C-A1	CA#5 02C-A1	CA#6 02C-A1
CCIN	U2	E2	T2	N2	J2	E2
CHIN	V2	F2	U2	P2	K2	F2
CVTL	T2	D2	S2	M2	H2	D2
CADR-A	X2	B2	W2	R2	L2	G2
CADR-B (TPS)	W2	C2	V2	Q2	N/A	N/A

3725 Model 2

The card socket locations for each channel adapter are shown in the following table.

Card Name	CA#1 01A-A3	CA#2 01A-A3	CA#3 01B-A2	CA#4 01B-A2
CCIN	U2	R2	E2	H2
CHIN	V2	S2	F2	J2
CVTL	T2	Q2	D2	G2
CADR	X2	W2	B2	C2

HOST UNIT CONTROL WORD (UCW) DEFINITION

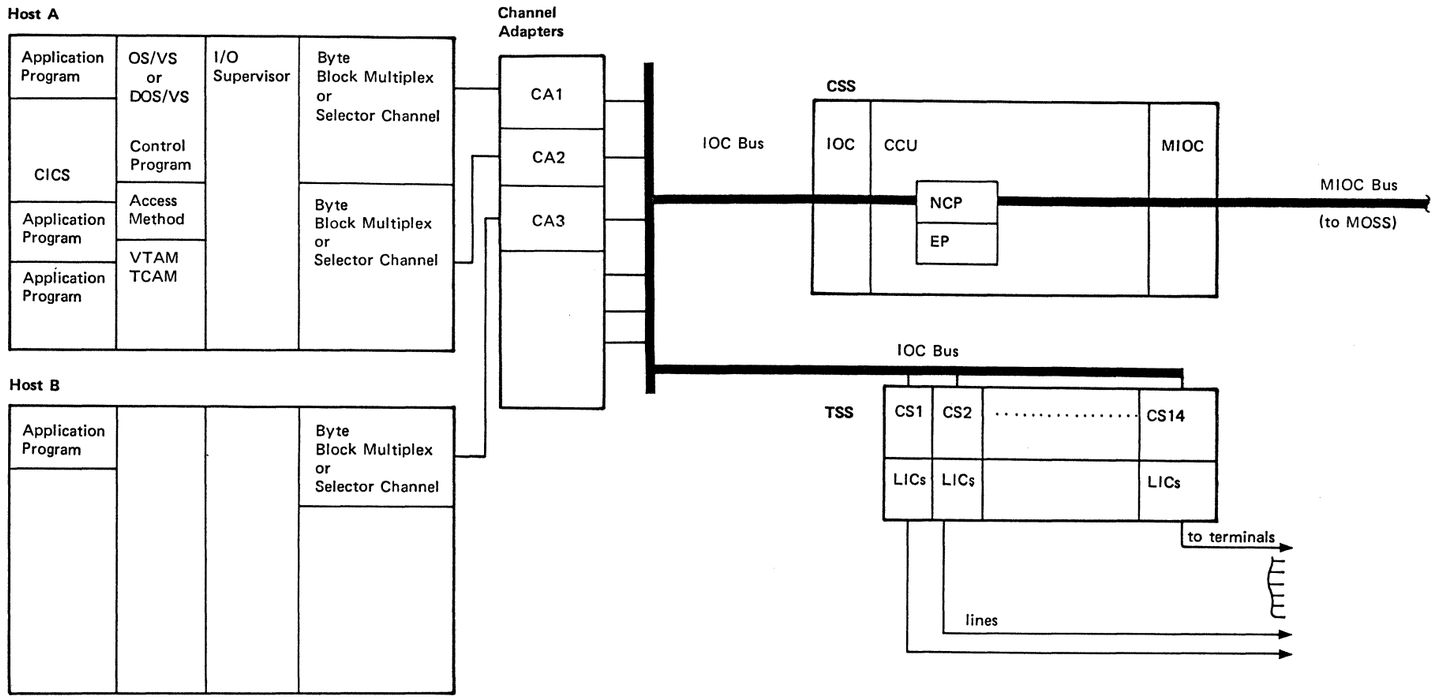
The host system UCW requirements for the controller vary according to type of control program and features. The UCW requirements are determined as follows:

- One UCW is required for each unique NSC address.
- An additional UCW is required for each emulated subchannel address. For example:
  - A controller with EP and two emulation subchannels (ESCs) needs three UCWs.
  - A controller running PEP with two ESCs needs three UCWs.
- All UCWs must be unshared and unfolded.

Channel Adapter Operating Environment

As shown on the following figure, the channel adapter is used by the 3725 control program (NCP or EP) to communicate with an application program in the host(s) through various telecommunication access methods.

Physically, a channel interface data and control bus attaches the channel adapter to the host channels, and an IOC bus attaches the channel adapter to the CCU.



MODES OF OPERATION

The channel adapter can run in either native subchannel (NSC) mode or in emulation subchannel (ESC) mode. The CA operates in the following modes:

- NSC with NCP
- ESC with EP
- Both simultaneously with PEP

Note: With EP, the NSC is used for loading and dumping.

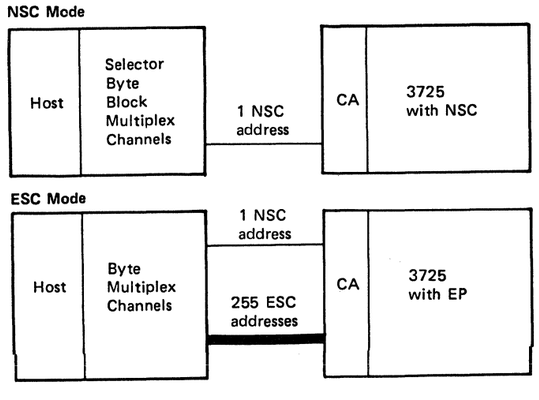
NSC Mode

The NSC mode is supported for all types of host channels (byte multiplex, selector, or block multiplex), and allows the servicing of any number of lines up to 256 using only one host subchannel address. The line address decoding is handled entirely by NCP.

Note: Initial program load (IPL) must always be done in NSC mode.

ESC Mode

The ESC mode is supported for byte multiplex channels only, and allows the controller to emulate the 2701 Data Adapter Unit, 2703 Transmission Control, and 2704 Transmission Control using existing host programs and subchannel addresses. This mode requires the emulation program (EP) in the 3725, and a separate subchannel address for each line.



Data Transfer Methods(Part 1 of 2)

Two methods are used to transfer data between the channel adapter and the host channel:

- Program-initiated operation (PIO)
- Adapter-initiated operation (AIO)

CONTROLLING THE CHANNEL ADAPTER

The channel adapter is controlled by instructions issued by the control program. These instructions are of two types only: adapter input/output (IOH) and adapter input/output immediate (IOHI). They examine or set the channel adapter registers, load or read buffers, and initiate cycle stealing.

Note: Throughout this chapter, the channel IOH and IOHI instructions are referred to as input register X'N' and output register X'N'.

CHANNEL ADAPTER STATES

The channel adapter may be in one of the following states:

Ready State

In the ready state, the channel adapter accepts instructions, but it is not in one of the three active states (initial selection, data transfer, or status transfer).

Initial Selection State

The channel adapter enters the initial selection state when an initial selection is started by the host processor. The channel adapter continuously monitors its channel interface for one of its assigned addresses. When it detects one of these addresses, the channel adapter enters the initial selection state, and proceeds with the operation.

Data Transfer State

The channel adapter enters the data transfer state when the control program initiates a data transfer sequence. Data is transferred across the interface from the host channel to the channel adapter, or from the channel adapter to the host, by hardware.

Status Transfer State

The channel adapter enters the status transfer state when the control program initiates a status transfer sequence.

Disabled State

The channel adapter is in disabled state when it does not reply to selection signals coming from the host.

CHANNEL ADAPTER DEVICE ADDRESSES

Channel adapter device addresses are required on two separate occasions:

1. At initial selection, the channel adapter must be able to recognize its own address and respond to it.
2. After raising 'request in', the channel adapter must present a valid device address to the channel before it can transfer data or status information.

CHANNEL ADAPTER INTERRUPT REQUESTS

The CCU is an interrupt-driven processor, and almost all processing for the channel adapter is done in response to channel adapter interrupts. The channel adapter can raise interrupt requests at level 1 and at level 3.

Level 1

Level 1 interrupt requests are caused by check or error conditions.

Level 3

Level 3 interrupt requests are caused by two different conditions:

1. Initial selection interrupt requests are raised when the channel adapter receives an address and a command across the channel interface.
2. Data/status interrupt requests are raised when the channel adapter requires data or status service.

ACCESSING CHANNEL ADAPTER REGISTERS

The channel adapter registers are accessed by one of the following methods:

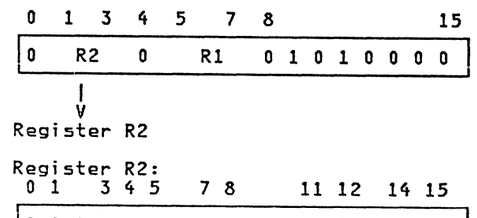
- By the operator selecting the channel adapter status when using the CCU function, as described in the 3725 PD Guide and Extended Services or
- By the input/output instruction (IOH), as described below
- By the input/output immediate instruction (IOHI), as described below

The CCU IOH/IOHI instructions are used to transfer the contents of one of the general registers to a selected channel adapter register, or conversely. The following is a description of the two instructions.

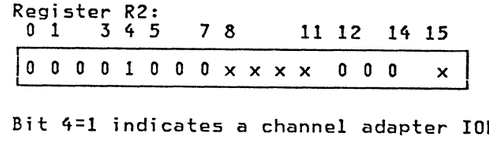
INPUT/OUTPUT (IOH)

This halfword instruction transfers the contents of the register specified by R1 to the channel adapter, or places information coming from the channel adapter into the register specified by R1. The channel adapter register, and the direction of data movement, are specified by the contents of R2.

IOH halfword instruction:



Register R2:



Bit 4=1 indicates a channel adapter IOH.

Bits 8-11 indicate the channel adapter register to be used, as follows:

Bits 8-11	Reg No.	Register Function
0000	X'0'	Initial selection control/ reset initial selection
0001	X'1'	Initial selection address and command
0010	X'2'	Data/status control
0011	X'3'	ESC address and status byte
0100	X'4'	Data buffer
0101	X'5'	Data buffer
0110	X'6'	NSC status control
0111	X'7'	Channel adapter condition/ control
1011	X'B'	ESC test I/O address and status
1100	X'C'	Cycle steal mode control
1101	X'D'	* Channel adapter level 1 interrupt check
1101	X'E'	* Channel adapter level 1 interrupt requests
1111	X'F'	* Channel adapter level 3 interrupt requests

* Input register only

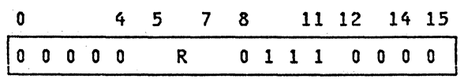
Bit 15 indicates input or output:
1 = input
0 = output

Data Transfer Methods (Part 2 of 2)

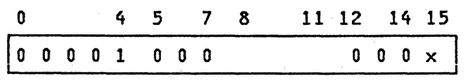
INPUT/OUTPUT IMMEDIATE (IOHI)

This fullword instruction transfers the contents of the register specified by R to the channel adapter, or places information coming from the channel adapter into the register specified by R. The channel adapter register and the direction of data movement are both specified by the contents of the second halfword.

First halfword of IOHI:



Second halfword of IOHI:



Bit 4=1 indicates a channel adapter IOHI.

Bits 8-11 indicate the channel adapter register to be used (refer to IOH description for bit definition).

Bit 15 indicates input or output
1 = input
0 = output

CA BROADCAST COMMANDS

IOH CA commands Input X'7', X'E', and X'F' and Output X'7' are recognized and executed by every CA on the IOC bus, whether selected or not. These commands are called "broadcast commands". As each CA executes the broadcast command, the 'common valid feed auto' line is activated. Since this line is DOT-ANDed, the line goes active only when all CAs have executed and responded to the broadcast command.

During the IOH CA commands Input X'7' and X'E', the CAs DOT-OR bits onto the IOC data bus. The parity bit is not generated. The parity valid (PV) tag is not raised because parity may be bad. The 'common valid feed auto' line signals that all bits have been placed on the IOC bus, and the selected CA validates the data to the CCU.

Local Store and Register (Part 1 of 3)

The channel adapter contains a number of registers that are accessible via IOH/IOHI instructions, or by selecting the channel adapter status when using the CCU functions.

The following table lists these registers.

Input	Output	Register Function
X'0'		Initial selection control
	X'0'	Reset initial selection
X'1'	X'1'	Initial selection address and command
X'2'	X'2'	Data/status control
X'3'	X'3'	ESC address and status byte
X'4'	X'4'	Data buffer
X'5'	X'5	Data buffer
X'6'	X'6'	NSC status/control
X'7'		Channel adapter condition
	X'7'	Channel adapter control
X'B'	X'B'	ESC test I/O address and status
X'C'	X'C'	Cycle steal mode control
X'D'		CA level 1 interrupt check
X'E'		CA level 1 interrupt requests
X'F'		CA level 3 interrupt requests

Input X'0' : Initial Selection Control

Byte 0		Byte 1	
Bit	Name	Bit	Name
0	Initial selection interrupt	0	(Not used)
1	Interface disconnect	1	(Not used)
2	Selective reset	2	(Not used)
3	Channel bus out check	3	(Not used)
4	Emulation subchannel operation	4	(Not used)
5	Stacked initial status	5	(Not used)
6	Status byte cleared	6	(Not used)
7	System reset	7	(Not used)

Output X'0': This command resets all the initial selection hardware latches as well as the initial selection interrupt request on level 3. Register bit settings are not relevant.

For Reset System Reset, output X'7' with byte 1, bit 3=1 is used.

Input X'1' : Initial Selection Address and Command

Byte 0		Byte 1	
Bit	Name	Bit	Name
0	Address byte bit 0	0	I/O command byte bit 0
1	Address byte bit 1	1	I/O command byte bit 1
2	Address byte bit 2	2	I/O command byte bit 2
3	Address byte bit 3	3	I/O command byte bit 3
4	Address byte bit 4	4	I/O command byte bit 4
5	Address byte bit 5	5	I/O command byte bit 5
6	Address byte bit 6	6	I/O command byte bit 6
7	Address byte bit 7	7	I/O command byte bit 7

Output X'1' : Initial Selection Address and Command

(Diagnostic Control Purposes Only).

Byte 0		Byte 1	
Bit	Name	Bit	Name
0	Address byte bit 0	0	I/O command byte bit 0
1	Address byte bit 1	1	I/O command byte bit 1
2	Address byte bit 2	2	I/O command byte bit 2
3	Address byte bit 3	3	I/O command byte bit 3
4	Address byte bit 4	4	I/O command byte bit 4
5	Address byte bit 5	5	I/O command byte bit 5
6	Address byte bit 6	6	I/O command byte bit 6
7	Address byte bit 7	7	I/O command byte bit 7

Input X'2' : Data/Status Control

Byte 0		Byte 1	
Bit	Name	Bit	Name
0	Outbound data transfer sequence	0	Channel bus out check
1	Inbound data transfer sequence	1	Selective reset
2	Status transfer sequence	2	Suppress out
3	Emulation Subchannel = 1	3	Stacked ending status
4	Channel end presented	4	Priority outbound service
5	Channel stop/interface disconnect	5	)
6	Suppress out monitor interrupt	6	)--Residual byte count
7	Program requested interrupt	7	)

Output X'2' : Data/Status Control

Byte 0		Byte 1	
Bit	Name	Bit	Name
0	Set/reset outbound data transfer sequence	0	Set monitor for circle B
1	Set/reset inbound data transfer sequence	1	(Not used)
2	Set/reset status transfer sequence	2	Set monitor for 2848 ETX
3	Set/reset ESC operation	3	Set suppressible status
4	Set/reset PIO mode	4	Priority outbound service
5	Reset initial selection interrupt	5	)
6	Reset data/status interrupt	6	)--Request byte count
7	(Not used)	7	)

Input and Output X'3' : ESC Address and Status Byte

Byte 0		Byte 1	
Bit	Name	Bit	Name
0	ESC: Address byte bit 0	0	ESC: Attention
1	ESC: Address byte bit 1	1	ESC: Status modifier
2	ESC: Address byte bit 2	2	ESC: Control unit end
3	ESC: Address byte bit 3	3	ESC: Busy
4	ESC: Address byte bit 4	4	ESC: Channel end
5	ESC: Address byte bit 5	5	ESC: Device end
6	ESC: Address byte bit 6	6	ESC: Unit check
7	ESC: Address byte bit 7	7	ESC: Unit exception

Local Store and Register Contents(Part 2 of 3)

Input and Output X'4' : Data Buffer

Byte 0		Byte 1	
Bit	Name	Bit	Name
0	Data buffer byte 1/5 bit 0	0	Data buffer byte 2/6 bit 0
1	Data buffer byte 1/5 bit 1	1	Data buffer byte 2/6 bit 1
2	Data buffer byte 1/5 bit 2	2	Data buffer byte 2/6 bit 2
3	Data buffer byte 1/5 bit 3	3	Data buffer byte 2/6 bit 3
4	Data buffer byte 1/5 bit 4	4	Data buffer byte 2/6 bit 4
5	Data buffer byte 1/5 bit 5	5	Data buffer byte 2/6 bit 5
6	Data buffer byte 1/5 bit 6	6	Data buffer byte 2/6 bit 6
7	Data buffer byte 1/5 bit 7	7	Data buffer byte 2/6 bit 7

Input and Output X'5' : Data Buffer

Byte 0		Byte 1	
Bit	Name	Bit	Name
0	Data buffer byte 3/7 bit 0	0	Data buffer byte 4/8 bit 0
1	Data buffer byte 3/7 bit 1	1	Data buffer byte 4/8 bit 1
2	Data buffer byte 3/7 bit 2	2	Data buffer byte 4/8 bit 2
3	Data buffer byte 3/7 bit 3	3	Data buffer byte 4/8 bit 3
4	Data buffer byte 3/7 bit 4	4	Data buffer byte 4/8 bit 4
5	Data buffer byte 3/7 bit 5	5	Data buffer byte 4/8 bit 5
6	Data buffer byte 3/7 bit 6	6	Data buffer byte 4/8 bit 6
7	Data buffer byte 3/7 bit 7	7	Data buffer byte 4/8 bit 7

Input X'6' : NSC Status/Control

Byte 0		Byte 1	
Bit	Name	Bit	Name
0	CA switched to interface B	0	NSC: Attention
1	CA switched to interface A	1	NSC: Status modifier
2	(Not used)	2	NSC: Control unit end
3	(Not used)	3	NSC: Busy
4	(Not used)	4	NSC: Channel end
5	(Not used)	5	NSC: Device end
6	(Not used)	6	NSC: Unit check
7	(Not used)	7	NSC: Unit exception

Output X'6' : NSC Status/Control

Byte 0		Byte 1	
Bit	Name	Bit	Name
0	Set force A busy	0	Set NSC: Attention
1	Set force B busy	1	Set NSC: Status modifier
2	Force error	2	Set NSC: Control unit end
3	Diagnostic storage mode	3	Set NSC: Busy
4	(Not used)	4	Set NSC: Channel end
5	Check the checkers	5	Set NSC: Device end
6	A/B data buffer diagnostic mode	6	Set NSC: Unit check
7	Reset to neutral state	7	Set NSC: Unit exception

Input X'7' : Channel Adapter Condition

Byte 0		Byte 1	
Bit	Name	Bit	Name
0	CA5 enabled	0	CA1 interface A enabled
1	(Not used)	1	CA1 interface B enabled
2	CA6 enabled	2	CA2 interface A enabled
3	(Not used)	3	CA2 interface B enabled
4	(Not used)	4	CA3 interface A enabled
5	NSC address active	5	CA3 interface B enabled
6	PIO mode	6	CA4 interface A enabled
7	(Not used)	7	CA4 interface B enabled

Output X'7' : Channel Adapter Control

Byte 0		Byte 1	
Bit	Name	Bit	Name
0	Enable autoselection	0	Set suppress out monitor
1	Disable autoselection	1	Set pgm requested interrupt
2	Select CA addressed by bits 4-6	2	Reset CA interrupt level 1 checks
3	Execute output on CA addressed by bits 4-6	3	Reset system reset/ NSC address active
4	)	4	Set allow channel interface enable (A and B)
5	)--Channel adapter address bits 0-2	5	Set ESC operational
6	)	6	Set ESC command free
7	Channel adapter reset	7	Set allow channel interface disable (A and B)

Input and Output X'B' : ESC Test I/O Address and Status

Byte 0		Byte 1	
Bit	Name	Bit	Name
0	ESC TIO: Address bit 0	0	ESC TIO: Attention
1	ESC TIO: Address bit 1	1	ESC TIO: Status modifier
2	ESC TIO: Address bit 2	2	ESC TIO: Control unit end
3	ESC TIO: Address bit 3	3	ESC TIO: Busy
4	ESC TIO: Address bit 4	4	ESC TIO: Channel end
5	ESC TIO: Address bit 5	5	ESC TIO: Device end
6	ESC TIO: Address bit 6	6	ESC TIO: Unit check
7	ESC TIO: Address bit 7	7	ESC TIO: Unit exception

Input X'C' : Cycle Steal Mode Control

Byte 0		Byte 1	
Bit	Name	Bit	Name
0	SYN monitor latch	0	)
1	DLE temporary latch	1	)
2	USASCII monitor control latch	2	)
3	EBCDIC monitor control latch	3	)--Residual byte count
4	(Not used)	4	)
5	(Not used)	5	)
6	(Not used)	6	)
7	(Not used)	7	)

Local Store and Register Contents (Part 3 of 3)

Output X'C' : Cycle Steal Mode Control

Byte 0		Byte 1	
Bit	Name	Bit	Name
0	SYN monitor control latch	0	)
1	DLE remember control latch	1	)
2	USASCII monitor control latch	2	)
3	EBCDIC monitor control latch	3	)--Request byte count
4	(Not used)	4	)
5	(Not used)	5	)
6	(Not used)	6	)
7	(Not used)	7	)

Input X'D' : Channel Adapter Level 1 Interrupt Check

Byte 0		Byte 1	
Bit	Name	Bit	Name
0	IOC bus parity error	0	Output exception check
1	Internal bus parity error	1	PIO halt remember latch
2	CCIN card check	2	Cycle steal halt remember latch
3	(Not used)	3	Bus in check interface A
4	CHIN card check	4	Ground fault error
5	Address compare error	5	Bus in check interface B
6	Initiate service latch ungated	6	CADR card check interface A
7	(Not used)	7	CADR card check interface B

Input X'E' : Channel Adapter Level 1 Interrupt Requests

Byte 0		Byte 1	
Bit	Name	Bit	Name
0	CA5 L1 interrupt request	0	CA1 L1 interrupt request
1	(Not used)	1	(Not used)
2	CA6 L1 interrupt request	2	CA2 L1 interrupt request
3	Any CA L1 interrupt request	3	(Not used)
4	)	4	CA3 L1 interrupt request
5	)--Channel adapter address	5	(Not used)
6	) bits 0-2	6	CA4 L1 interrupt request
7	(Not used)	7	(Not used)

Input X'F' : Channel Adapter Level 3 Interrupt Requests

Byte 0		Byte 1	
Bit	Name	Bit	Name
0	(Not used)	0	(Not used)
1	TPS installed	1	(Not used)
2	Selected CA initial selection L3 interrupt request	2	(Not used)
3	Selected CA data/status L3 interrupt request	3	(Not used)
4	)	4	(Not used)
5	)--Channel adapter address	5	(Not used)
6	) bits 0-2	6	(Not used)
7	(Not used)	7	(Not used)

Hardware Status

Byte 0	
Bit	Name
0	(Not used)
1	Status modifier
2	Control unit end
3	Busy
4	Channel end
5	(Not used)
6	Unit check
7	(Not used)

Cycle Steal Control Word (CSCW)

Bit	Name
0	(Not used)
1	(Not used)
2	(Not used)
3	(Not used)
4	(Not used)
5	CA=0
6	(Not used)
7	(Not used)
8	Read/write
9	(Not used)
10	(Not used)
11	)
12	) Cycle steal register address cabled
13	) at board level during manufacturing
14	)
15	(Not used)

Registers X'0' and X'1'

INPUT X'0' (INITIAL SELECTION CONTROL)

The input X'0' register bits are set by the channel adapter hardware. They contain information that identifies the event that set the channel adapter initial selection level 3 interrupt. This type of interrupt request may be set by:

- The completion of an initial selection sequence
- The detection of a system reset sequence

During a normal initial selection sequence, the initial selection interrupt bit (byte 0, bit 0) is set. The remaining bits give supplementary information, or indicate certain reset and error conditions. The bits of this register have the following meaning:

Byte	Bit	Meaning
0	0	Initial selection interrupt
	1	Interface disconnect
	2	Selective reset
	3	Channel bus out check
	4	Emulation subchannel operation
	5	Stacked initial status
	6	Status byte cleared
	7	System reset
1	0–7	(Not used)

Byte 0, Bit 0: Initial Selection Interrupt

This bit, when on, indicates that normal initial selection sequence has occurred. The I/O device address and the I/O command byte may be determined by an input X'1'. If this bit is on, indicating a normal initial selection sequence, all other bits of this register should be off, with the possible exception of the ESC selection bit (byte 0, bit 4).

If byte 0, bit 0 is off, it indicates that the interrupt request was due to the detection of an unusual condition as defined by the remaining bits of the register.

Byte 0, Bit 1: Interface Disconnect

This bit, when on, indicates that the channel adapter level 3 interrupt request was caused by the detection of an interface disconnect sequence (halt I/O) during initial selection. The addressed subchannel can be determined via an input X'1'.

Byte 0, Bit 2: Selective Reset

This bit, when on, indicates that the channel adapter level 3 interrupt request was caused by the detection of a selective reset sequence during initial selection. The addressed subchannel can be determined via an input X'1'.

Byte 0, Bit 3: Channel Bus Out Check

This bit, when on, indicates that the channel adapter level 3 interrupt request was caused by the detection of bad (even) parity on the I/O channel interface bus out when the channel I/O command byte was presented during initial selection. The addressed subchannel can be determined via an input X'1'.

Byte 0, Bit 4: Emulation Subchannel Operation

This bit, when on, indicates that the channel adapter initial selection address and command register (X'1') contains an ESC address.

If the bit is off, register X'1' contains the NSC address.

Byte 0, Bit 5: Stacked Initial Status

This bit, when on, indicates that the channel adapter level 3 interrupt request was caused by the completion of an initial selection sequence in which the initial status byte presented to the channel has been stacked. The addressed subchannel and I/O command can be determined by an input X'1'.

The initial status byte stacked indication can occur in the following situations:

ESC (BYTE 0, BIT 4 ON): The ESC status byte, prepared by the control program for a given subchannel, was presented to the host in response to an ESC test I/O command issued to the same subchannel. However, the status was stacked (not accepted) by the host.

NSC (BYTE 0, BIT 4 OFF): The NSC status byte, prepared by the program, was presented to the host during an initial selection sequence, but was stacked (not accepted) by the host. As a result, a channel adapter initial selection level 3 interrupt request is set.

Byte 0, Bit 6: Status Byte Cleared

The interpretation of this bit depends on the state of byte 0, bit 4.

NSC (BYTE 0, BIT 4 OFF): The NSC status byte, prepared by the program, has been transferred to the host during an initial selection sequence. Thus, the NSC status byte has been cleared, resulting in a channel adapter initial selection level 3 interrupt request.

ESC (BYTE 0, BIT 4 ON): The channel adapter initial selection level 3 interrupt request results from the completion of an initial selection sequence in which a Test I/O to an ESC subchannel was serviced, and TIO status had previously been set up for that subchannel. The subchannel that was serviced, and the status that was presented, can be determined by an input X'B'.

Byte 0, Bit 7: System Reset

This bit, when on, indicates that a system reset sequence has occurred on the channel, causing the channel adapter to reset. All other bits obtained by an input X'0' are automatically zero.

Note: Since a system reset may occur at any time, all indications of previous channel sequences that have not yet been serviced are lost.

OUTPUT X'0' (RESET INITIAL SELECTION)

An output X'0' resets all the initial selection hardware latches and also the channel adapter level 3 interrupt request resulting from an initial selection sequence. As this instruction performs a function, the bit settings of the register are not used.

Note: An output X'0' does not reset a system reset condition, nor the resulting channel adapter level 3 interrupt request.

INPUT X'1' (INITIAL SELECTION ADDRESS AND COMMAND)

The input X'1' bits are set by the channel adapter hardware with the address and the command received from the channel.

The bits of the register have the following meaning:

Byte	Bits	Meaning
0	0–7	Address byte bits 0–7
1	0–7	I/O command byte bits 0–7

OUTPUT X'1' (INITIAL SELECTION ADDRESS AND COMMAND)

An output X'1' allows the program to set register X'1' with an initial selection address and command for diagnostic purposes only.

Register X'2'(Part 1 of 3)

INPUT X'2' (DATA/STATUS CONTROL)

The Input X'2' register is used to identify the event(s) that caused a channel adapter data/status level 3 interrupt.

Note: If a system reset sequence occurs before an input X'2', the bits that define the cause of the interrupt request, and the interrupt itself, are reset.

The bits of this register have the following meaning:

Byte	Bit	Meaning
0	0	Outbound data transfer sequence
	1	Inbound data transfer sequence
	2	Status transfer sequence
	3	Emulation subchannel = 1
	4	Channel end presented
	5	Channel stop/interface disconnect
	6	Suppress out monitor interrupt
	7	Program requested interrupt
1	0	Channel bus out check
	1	Selective reset
	2	Suppress out
	3	Stacked ending status
	4	Priority outbound service
	5	Residual byte count
	6	Residual byte count
	7	Residual byte count

Byte 0, Bit 0: Outbound Data Transfer Sequence

This bit, when on, indicates that the channel adapter data/status level 3 interrupt was caused by the ending of an outbound (3725 to host) data transfer sequence. If this bit is on, byte 0, bit 1 should be off. On a block multiplex channel, byte 0, bit 2 may also be on if the block is the last data block.

In PIO mode, byte 1, bits 5-7 indicate the residual byte count.

ESC: If the transfer is over an emulator subchannel (byte 0, bit 3 on), an input X'3' should be executed to determine the address of the subchannel.

Byte 0, Bit 1: Inbound Data Transfer Sequence

This bit, when on, indicates that the channel adapter data/status level 3 interrupt was caused by the ending of an inbound (host to 3725) data transfer sequence. If this bit is on, byte 0, bits 0 and 2 should both be off.

In PIO mode, byte 1, bits 5-7 indicate the residual byte count. The data bytes transferred from the host processor may be obtained by executing Input X'4' and X'5'.

ESC: If the transfer is over an emulator subchannel (byte 0, bit 3 on), an input X'3' should be executed to determine the address of the subchannel.

Byte 0, Bit 2: Status Transfer Sequence

This bit, when on, indicates that the channel adapter data/status level 3 interrupt was caused by ending of a status transfer sequence. If this bit is on, byte 0, bits 0 and 1 should both be off.

ESC: If the transfer is over an emulator subchannel (byte 0, bit 3 on), an input X'3' should be executed to determine the address of the subchannel over which the transfer occurred, and to obtain the status that was presented.

Byte 0, Bit 3: Emulation Subchannel = 1

ESC (BYTE 0, BIT 3 ON): The transfer sequence defined by byte 0, bits 0-2 was performed on the ESC subchannel using the ESC address, and the status in the case of a status transfer. Address and status are taken from the ESC address and status register (X'3').

NSC (BYTE 0, BIT 3 OFF): The transfer sequence defined by byte 0, bits 0-2 was performed on the native subchannel using the assigned hardware address. For a status transfer, the status was taken from the NSC status register (X'6').

Byte 0, Bit 4: Channel End Presented

This bit applies only to a native subchannel working with a block multiplex or selector channel. For all other combinations, it is not used.

Note: When this bit is on, byte 0, bit 1, or byte 0, bits 0 and 2 are also on.

When the bit is on, it indicates that the hardware has presented, or has tried to present, a channel end status to the host during a data transfer. A level 3 interrupt request is set up at the same time.

The hardware 'channel end' is presented when the CCU program sets up an outbound (3725 to host) data transfer sequence and a status transfer sequence at the same time (output X'2', byte 0, bits 0 and 2 both on), and the data transfer to the host occurs normally.

The hardware 'channel end' is also presented during any data transfer sequence if the host issued a channel stop.

Byte 0, Bit 5: Channel Stop/Interface Disconnect

This bit, when on, indicates that during a data transfer sequence, a channel stop or interface disconnect (halt I/O) sequence has occurred. The CCU program cannot distinguish between these. The transfer sequence is ended. The residual byte count is greater than zero, and indicates the number of bytes that were not transferred.

The residual byte count is contained in register X'2', byte 1, bits 5-7 for PIO operations, and in register X'C', byte 1, bits 0-7 for AIO operations.

The bit, when on, may also indicate that an interface disconnection (halt I/O) occurred during a status transfer sequence.

Byte 0, Bit 6: Suppress Out Monitor Interrupt

This bit, when on, indicates that the channel adapter data/status level 3 interrupt was set because the 'suppress out' tag line was found to be inactive. The program signals the channel adapter to monitor for the inactive condition of 'suppress out' by executing an output X'7' with the set suppress out monitor interrupt bit (byte 0, bit 0) set to 1.

Byte 0, Bit 7: Program Requested Interrupt

This bit, when on, indicates that the channel adapter data/status level 3 interrupt was set because the program requested an interrupt at level 3 by executing an output X'7' with byte 1, bit 1 set to 1. An output X'2' should be executed to reset this bit and the resulting channel adapter data/status level 3 interrupt request.

Byte 1, Bit 0: Channel Bus Out Check

This bit, when on, indicates that during an inbound (host to 3725) data transfer sequence, a bad (even) parity condition was detected on bus out during the transfer of a data byte. The transfer sequence is terminated. The byte that was transferred with bad parity may be located by examining the residual byte count. The residual byte count is contained in register X'2', byte 1, bits 5-7 for PIO operations, and in register X'C', byte 1, bits 0-7 for AIO operations.

Byte 1, Bit 1: Selective Reset

This bit, when on, indicates that a selective reset sequence occurred during the transfer.

Byte 1, Bit 2: Suppress Out

This bit, when on, indicates that the 'suppress out' tag line on the channel was active at the time that the input X'2' was executed.

Byte 1, Bit 3: Stacked Ending Status

This bit, when on, indicates that during a final status transfer sequence, the status byte was not accepted by the channel, and was stacked.

ESC: The status byte that was presented can be examined by executing an input X'3'.

Register X'2' (Part 2 of 3)

Byte 1, Bit 4: Priority Outbound Service

This bit, when on, indicates to the autoselect mechanism that the channel adapter with the priority outbound level 3 interrupt has the highest priority and will be selected with an input X'F'.

Byte 1, Bits 5-7: Residual Byte Count

These bits apply only to PIO operations. They contain the residual byte count for an inbound or outbound data transfer sequence. The residual byte count is the number of bytes that were not successfully transferred.

Note: For AIO operations, the residual byte count can be obtained by executing an input X'C'.

OUTPUT X'2' (DATA/STATUS CONTROL)

Output X'2' is used to control the operation of the channel adapter. An output X'2' also resets the program requested interrupt and suppress out monitor bits. It should be issued only if a level 3 channel adapter initial selection interrupt has been set.

The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	Set/reset outbound data transfer sequence
	1	Set/reset inbound data transfer sequence
	2	Set/reset status transfer sequence
	3	Set/reset ESC operation
	4	Set/reset PIO mode
	5	Reset initial selection interrupt
	6	Reset data/status interrupt
	7	(Not used)
1	0	Set monitor for circle B
	1	(Not used)
	2	Set monitor for 2848 ETX
	3	Set suppressible status
	4	Priority outbound service
	5	Request byte count
	6	Request byte count
	7	Request byte count

Byte 0, Bit 0: Set/Reset Outbound Data Transfer Sequence

This bit, when on, sets the outbound data transfer bit and causes the channel adapter to initiate an outbound (3725 to host) data transfer sequence.

If the bit is off, it resets the channel adapter outbound data transfer sequence bit.

NSC: If the transfer is over the native subchannel (byte 0, bit 3 off), it is initiated using the assigned NSC address.

ESC: If the transfer is over an emulator subchannel (byte 0, bit 3 on), it is initiated using the address contained in register X'3'.

Byte 0, Bit 1: Set/Reset Inbound Data Transfer Sequence

This bit, when on, sets the inbound data transfer bit and causes the channel adapter to initiate an inbound (host to 3725) data transfer sequence.

If this bit is off, it resets the channel adapter inbound data transfer sequence bit.

In PIO mode, byte 1, bits 5-7 indicate the residual byte count. The data bytes transferred from the host may be obtained by executing input X'4'and input X'5'.

ESC: If the transfer is over an emulator subchannel (byte 0, bit 3 on), an input X'3' will indicate the address of the subchannel.

Byte 0, Bit 2: Set/Reset Status Transfer Sequence

This bit, when on, sets the status transfer bit and causes the CA to initiate a status transfer sequence. If the bit is off, the CA status transfer sequence bit is reset.

NSC: If the transfer is over the native subchannel (byte 0, bit 3=0) the transfer is initiated using the assigned NSC address and the NSC status byte (from register X'6'). The NSC status remains available until it is accepted by the channel. Byte 0, bits 0 and 2 may be on together; the CA hardware then presents a CE status to the channel.

ESC: If the transfer is over an emulator subchannel (byte 0, bit 3 on), an input X'3' gives the address of the subchannel over which the transfer occurred and the status that was presented. When the status is presented to the channel, the 'ESC TIO status available' latch is reset.

If the status is stacked, the ESC address and the status that was stacked are moved by hardware to the ESC TIO address and status byte register X'B' at the same time, the ESC TIO status available latch is set. When the status is presented to the channel, the 'ESC TIO status available' latch is reset. If the status is stacked, the ESC address and the status that was stacked are moved by hardware to the ESC TIO address and status byte register (X'B'). At the same time, the 'ESC TIO status available' latch is set.

Byte 0, Bit 3: Set/Reset ESC Operation

ESC (BYTE 0, BIT 3 ON): The transfer sequence defined by byte 0, bits 0-2 is initiated on the ESC subchannel using the ESC address and the status, in the case of a status transfer. Address and status are taken from the ESC address and status register (X'3').

NSC (BYTE 0, BIT 3 OFF): The transfer sequence defined by byte 0, bits 0-2 is initiated on the native subchannel using the assigned hardware address. In the case of a status transfer, the status is taken from the NSC status register (X'6').

Byte 0, Bit 4: Set/Reset PIO Mode

PIO MODE (BYTE 0, BIT 4 ON): The data transfer sequence defined by byte 0, bit 0 or 1 is executed in PIO mode with program intervention required every 4 bytes. The request byte count is taken from byte 1, bits 5-7 while byte 1, bits 0 and 2 enable certain special control functions. The transfer byte count in register X'C' is not used for PIO operations.

AIO MODE (BYTE 0, BIT 4 OFF): The data transfer sequence defined by byte 0, bit 0 or 1 is executed in AIO mode using the cycle steal mechanism. Byte 1, bits 5-7 are not used. The request byte count and the special control functions are taken from register X'C'.

Byte 0, Bit 5: Reset Initial Selection Interrupt

This bit, when on, causes the channel adapter to reset all the initial selection hardware latches and also the channel adapter level 3 interrupt request resulting from an initial selection sequence. This bit does not reset a system reset condition, nor the resulting channel adapter level 3 interrupt request.

Byte 0, Bit 6: Reset Data/Status Interrupt

This bit, when on, causes the channel adapter to reset the following bits in the data/status control register (input X'2'):

- Channel stop/interface disconnect (byte 0, bit 5)
- Channel bus out check (byte 1, bit 0)
- Selective reset (byte 1, bit 1)
- Stacked ending status (byte 1, bit 3)

The channel adapter data/status level 3 interrupt is also reset. In addition, if one of the transfer bits (byte 0, bits 0-2) is on, the channel adapter hardware also raises the 'request in' channel interface tag line in order to initiate the transfer sequence.

Register X'2'(Part 3 of 3)

Byte 1, Bit 0: Set Monitor for Circle B

This bit turns on the monitoring for the circle B character on inbound data transfer only. If the bit is off, the monitoring is reset.

Byte 1, Bit 2: Set Monitor for 2848 ETX

This bit turns on the monitoring for the 2848 'ETX' character on inbound data transfer only. If the bit is off, the monitoring is reset.

Byte 1, Bit 3: Set Suppressible Status

This bit is set after a status has been stacked on the channel to inhibit a status transfer to the host as long as 'suppress out' is active.

Byte 1, Bit 4: Priority Outbound Service

This bit applies only to outbound (3725 to host) operations. When on, the bit forces an outbound data transfer sequence to the autoselection logic.

When the priority outbound sequence is completed, the channel adapter with the priority outbound level 3 will have the highest selection priority.

Byte 1, Bits 5-7: Request Byte Count

These bits apply to PIO operations only. For AIO operation, they are ignored. They are used to indicate the number of bytes to be transferred to or from the host. A minimum of 4 bytes can be transferred at one time.

Registers X'3', X'4', X'5'

INPUT/OUTPUT X'3' (ESC ADDRESS AND STATUS BYTE)

Register X'3' contains the following information:

- When transferring data to or from the channel, byte 0 contains the address to be used by the ESC. Byte 1 is not used.
- When transferring status information to the channel, byte 0 contains the address to be used by the ESC. Byte 1 contains the status.

An input/output X'3' should be issued only if a level 3 channel adapter data/status interrupt has been set, and prior to informing the channel adapter that an ESC data/status transfer is required.

The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0-7	ESC address byte bits 0-7
1	0	ESC: Attention
	1	ESC: Status modifier
	2	ESC: Control unit end
	3	ESC: Busy
	4	ESC: Channel end
	5	ESC: Device end
	6	ESC: Unit check
	7	ESC: Unit exception

INPUT/OUTPUT X'4'AND X'5' (DATA BUFFER)

Registers X'4' and X'5' hold data during data transfers in either direction between the channel adapter and the host channel. The way in which these registers are used depends on whether the buffers are being loaded in PIO or AIO mode.

The bits of the register have the following meaning:

Register X'4' (Data Buffer Bytes 1 and 2, or 5 and 6)

Byte	Bit	Meaning
0	0-7	Data buffer byte 1/5 bits 0-7
1	0-7	Data buffer byte 2/6 bits 0-7

Register X'5' (Data Buffer Bytes 3 and 4, or 7 and 8)

Byte	Bit	Meaning
0	0-7	Data buffer byte 3/7 bits 0-7
1	0-7	Data buffer byte 4/8 bits 0-7

Note: Bytes 5-8 are used for diagnostic purposes only

Program-Initiated Operation (PIO)

Note: To transfer data in PIO mode, bit 4 of register X'2' must be set to 1.

Four single-byte buffers are used for data transfer, 2 bytes (1 and 2) being contained in register X'4', and two bytes (3 and 4) in register X'5'.

INBOUND OPERATION: During an inbound operation (host to 3725: byte 0, bit 1 of register X'2' is on), the four buffers are loaded by the channel adapter hardware. When the 4 bytes have been loaded, a data/status level 3 interrupt occurs. The data can then be accessed by using input X'4' and X'5'.

Notes:

1. After power on, the buffers must be loaded via the output X'4' and X'5' commands to prevent parity errors when reading.
2. Before an input X'4' and X'5', the residual byte count in register X'2' should be examined:

   Buffer byte 1 contains valid data if the residual count is less that the requested transfer count (issue input X'4').

   Buffer byte 2 contains valid data if the residual count is at least 2 less than the requested transfer count (issue input X'4')

   Buffer byte 3 contains valid data if the residual count is at least 3 less that the requested transfer count (issue input X'4' and X'5').

   Buffer byte 4 contains valid data only if the residual count is 0 and a 4-byte transfer was requested (issue input X'4' and X'5').

OUTBOUND OPERATION: During an outbound operation (3725 to host), the four buffers are loaded by the program using the output X'4' and X'5' instructions. Byte 0, bit 0 in the data/status control register (X'2') must then be set to start the transfer. When the four data bytes have been transferred across the channel, the hardware causes a data/status level 3 interrupt to ask for four more bytes to be loaded into the buffers.

Note: To ensure data integrity, the request byte count contained in byte 1, bits 5-7 of register X'2' must be consistent with the number of data bytes loaded into the data buffers.

Adapter-Initiated Operation (AIO)

In AIO mode, two 8-byte buffers are used alternately for data transfer.

They are called the A and B data buffers. Loading and unloading the buffers is done entirely by hardware. Program intervention is never required during normal operation. The two buffers are switched between the CCU and the channel adapter:

- During a host write operation, one data buffer is loaded by the channel adapter hardware while the other is emptied by the CA/CCU cycle steal mechanism.
- During a host read operation, one data buffer is loaded by the CA/CCU cycle steal mechanism while the other is emptied by the channel adapter hardware.

The A or B data buffers can also be accessed without cycle stealing. The A or B data buffers must first be selected via the A or B data buffer diagnostic bit in the NSC status/control register (output X'3', byte 0, bit 6). If the bit is off, the A data buffer is selected. If it is on, the B data buffer is selected. The input/output X'4/5' can then access bytes 1-4 directly, and bytes 5-8 by first setting on the diagnostic storage mode bit in the NSC status/control register (output X'6', byte 0, bit 6). The following table explains this process:

Instruction	Diag Storage Mode Bit Off	Diag Storage Mode Bit On
IOH/IOHI X'4'	Bytes 1 and 2	Bytes 5 and 6
IOH/IOHI X'5'	Bytes 3 and 4	Bytes 7 and 8

Register X'6'

INPUT X'6' (NSC STATUS/CONTROL)

Register X'6' contains the current NSC status byte.

The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	CA switched to interface B
	1	CA switched to interface A
	2-7	(Not used)
1	0	NSC: Attention
	1	NSC: Status modifier
	2	NSC: Control unit end
	3	NSC: Busy
	4	NSC: Channel end
	5	NSC: Device end
	6	NSC: Unit check
	7	NSC: Unit exception

Byte 0, Bit 0: CA Switched to Interface B

This bit, when on, indicates that the channel adapter is switched to interface B. Byte 0, bit 1 cannot be on at the same time.

Note: The execution of an output X'6' with byte 0, bit 1 set to 1 also makes this bit active.

Byte 0, Bit 1: CA Switched to Interface A

This bit, when on, indicates that the channel adapter is switched to interface A. Byte 0, bit 0 cannot be on at the same time.

Note: The execution of an output X'6' with byte 0, bit 0 set to 1 will also make this bit active.

Byte 1, Bits 0-7: NSC Status Byte

These are bits that were set into the NSC status register with an output X'6', when a hardware status was stacked. The bits have the usual meaning of the device status byte.

OUTPUT X'6' (NSC STATUS/CONTROL)

Register X'6' is used to set the current status of the NSC. This status is gated over the channel interface during NSC status transfer sequences. It is also used to set certain conditions in the adapter.

An output X'6' should be executed before signaling to the channel adapter that an NSC final status transfer sequence is required, if the status byte has not been previously given to the channel adapter. When the status byte has been given previously to the channel adapter but has been stacked by the channel, it need not been given again.

An output X'6' should only be executed when an initial selection, data/status, or program-controlled interrupt is set. After the NSC final status transfer sequence occurs, the status byte provided by this output is presented to the channel.

An output X'6' should also be used when presenting an asynchronous (attention) status, or when presenting the final status byte ending a channel I/O command on the NSC.

The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	Set force A busy
	1	Set force B busy
	2	Force error
	3	Diagnostic storage mode
	4	(Not used)
	5	Check the checkers
	6	A/B data buffer diagnostic mode
	7	Reset to neutral state
1	0	Set NSC: Attention
	1	Set NSC: Status modifier
	2	Set NSC: Control unit end
	3	Set NSC: Busy
	4	Set NSC: Channel end
	5	Set NSC: Device end
	6	Set NSC: Unit check
	7	Set NSC: Unit exception

Byte 0, Bit 0: Set Force A Busy

When on, this bit forces interface A of a TPS to the busy state, and sets interface B into the long-term allegiance state.

Byte 0, Bit 1: Set Force B Busy

When on, this bit forces interface B of a TPS to the busy state, and sets interface A into the long-term allegiance state.

Byte 0, Bit 2: Force Error

When on, this bit forces errors on the channel interface and driver/receiver cards for checking purposes.

It should be used only after the channel adapter has been disabled.

Byte 0, Bit 3: Diagnostic Storage Mode

When on, this bit sets the diagnostic storage mode latch. This allows the input/output instructions X'4' and X'5' to access bytes 5 to 8 of the data buffer, instead of bytes 1 to 4. Turning the bit off resets the diagnostic storage mode latch and the X'4' and X'5' instructions revert to accessing data bytes 1 to 4.

Byte 0, Bit 5: Check the Checkers

This bit is used along with force error (byte 0, bit 2) to verify the functioning of the channel adapter checking circuits.

Byte 0, Bit 6: A/B Data Buffer Diagnostic Mode

This bit is used to select either buffer A or buffer B. If bit=1, buffer B is selected. If bit = 0, buffer A is selected.

Byte 0, Bit 7: Reset to Neutral State

When on, this bit returns the channel adapter to the neutral state from the long-term allegiance state.

Byte 1, Bits 0-7: Set NSC Status Byte

The bits are used as for the device status byte.

Register X'7'(Part 1 of 2)

INPUT X'7' (CHANNEL ADAPTER CONDITION)

Register X'7' contains information mainly concerning the enable/disable status of the channel adapter interfaces for all the CAs.

CAs 1 to 4 can be equipped with TPSs, and therefore have interfaces A and B. CAs 5 and 6 cannot be equipped with TPSs, and therefore have only interface A.

Byte	Bit	Meaning
0	0	CA5 enabled
	1	(Not used)
	2	CA6 enabled
	3	(Not used)
	4	(Not used)
	5	NSC address active
	6	PIO mode
	7	(Not used)
1	0	CA1 interface A enabled
	1	CA1 interface B enabled
	2	CA2 interface A enabled
	3	CA2 interface B enabled
	4	CA3 interface A enabled
	5	CA3 interface B enabled
	6	CA4 interface A enabled
	7	CA4 interface B enabled

Byte 0, Bit 0: CA5 Enabled

This bit is set on by the channel hardware when CA5 is enabled. It is reset when CA5 is disabled.

Byte 0, Bit 2: CA6 Enabled

This bit is set on by the channel hardware when CA6 is enabled. It is reset when CA6 is disabled.

Byte 0, Bit 5: NSC Address Active

This bit is set by hardware:

- When NSC is initially selected by accepting a command
- When an output X'2' with the bit status transfer sequence (byte 0, bit 2) set is executed.

The bit remains active until 'device end' status is accepted for this command.

The bit is reset when NSC 'device end' status is accepted by the host channel on a NSC final status transfer.

Byte 0, Bit 6: PIO Mode

When on, this bit indicates that the last output X'2' set PIO data transfer mode, and all data transfers will be carried out in PIO mode until this bit is reset. When off, this bit indicates AIO mode.

Byte 1, Bit 0: CA1 Interface A Enabled

This bit is set on by the channel hardware when interface A of CA1 is enabled and reset when it is disabled.

Byte 1, Bit 1: CA1 Interface B Enabled

This bit is set on by the channel hardware when interface B of CA1 (equipped with a TPS) is enabled.

Byte 1, Bit 2: CA2 Interface A Enabled

This bit is set on by the channel hardware when interface A of CA2 is enabled and reset when it is disabled.

Byte 1, Bit 3: CA2 Interface B Enabled

This bit is set on by the channel hardware when interface B of CA2 (equipped with a TPS) is enabled.

Byte 1, Bit 4: CA3 Interface A Enabled

This bit is set on by the channel hardware when interface A of CA3 is enabled and reset when it is disabled.

Byte 1, Bit 5: CA3 Interface B Enabled

This bit is set on by the channel hardware when interface B of CA3 (equipped with a TPS) is enabled.

Byte 1, Bit 6: CA4 Interface A Enabled

This bit is set on by the channel hardware when interface A of CA4 is enabled and reset when it is disabled.

Byte 1, Bit 7: CA4 Interface B Enabled

This bit is set on by the channel hardware when interface B of CA4 (equipped with a TPS) is enabled.

OUTPUT X'7' (CHANNEL ADAPTER CONTROL)

Output X'7' is recognized by all channel adapters. The main function of the register is to select one of the six channel adapters:

- Either for the duration of the instruction
- Or until the channel adapter selection is changed, either by autoselection, or by another X'7'.

An output X'7' is also used to control channel adapter operations by setting/resetting control latches. The register is as follows:

Byte	Bit	Meaning
0	0	Enable autoselection
	1	Disable autoselection
	2	Select CA addressed by bits 4-6
	3	Execute output on CA addressed by bits 4-6
	4	Channel adapter address bit 0
	5	Channel adapter address bit 1
	6	Channel adapter address bit 2
	7	Channel adapter reset
1	0	Set suppress out monitor
	1	Set pgm requested interrupt
	2	Reset CA interrupt level 1 checks
	3	Reset system reset/NSC address active
	4	Set allow channel interface enable (A and B)
	5	Set ESC operational
	6	Set ESC command free
	7	Set allow channel interface disable (A and B)

Byte 0, Bit 0: Enable Autoselection

When on, this bit enables the autoselection mechanism for all channel adapters.

Byte 0, Bit 1: Disable Autoselection

When on, this bit disables the autoselection mechanism for all channel adapters.

Byte 0, Bit 2: Select CA Addressed by Bits 4-6

When on, this bit causes the addressed channel adapter to be selected for all subsequent channel operations. The CA selected before execution of the output X'7' with byte 0, bit 2 on, remains selected after the Output X'7' is completed (see note).

Register X'7'(Part 2 of 2)

Byte 0, Bit 3: Execute Output on CA Addressed by Bits 4-6

When on, this bit allows a particular channel adapter (addressed by byte 0, bits 4-6) to be temporarily selected for the purpose of changing any of the bits in its register (see note).

Note: If one of the installed CAs has already been selected, either by autoselection or by an Output X'7', bits 2 and 3 may both be off. If no CA has been selected, either bit 2 or bit 3 must be on, and a valid CA address must be contained in bits 4-6.

Byte 0, Bits 4-6: Channel Adapter Address Bits

These bits form the address of the channel adapters selected by byte 0, bits 2 or 3, either temporarily, or for subsequent input or output X'X'. The three bits are decoded as follows:

Byte 0 Bit 4 5 6	Channel Adapter
0 0 0	1
0 0 1	2
0 1 0	3
0 1 1	4
1 0 0	5
1 0 1	6

Note: If an attempt is made to communicate with uninstalled channel adapter, the CCU hardware times out and sets the 'PIO halt remember' latch in register X'D'. All channel adapters raise a level 1 interrupt request.

Byte 0, Bit 7: Channel Adapter Reset

This bit causes a power on reset of the channel adapters. It should be set only when the channel interface is disabled, or if channel adapters are hung up at the interface.

Byte 1, Bit 0: Set Suppress Out Monitor

When on, this bit causes the channel adapter to monitor for the inactive state of the tag line 'suppress out'. When this condition is detected, the channel adapter raises a level 3 interrupt request, and sets 'suppress out monitor interrupt' (X'2' byte 0, bit 6), which may be read by an Input X'2'.

Byte 1, Bit 1: Set Program Requested Interrupt

When on, this bit causes a channel adapter data/status interrupt request, and sets 'pgm requested interrupt' (X'2', byte 0, bit 7), unless one of the following conditions is present:

- A data/status transfer sequence has been initiated.
- The host is initiating an initial selection sequence on the channel.
- Chaining is indicated.
- A TPS is presenting 'device end' in response to a 'busy' status.

When the condition ends, the level 3 data/status interrupt request is set.

Byte 1, Bit 2: Reset CA Interrupt Level 1 Checks

When on, this bit causes the channel adapter to reset all CA level 1 check latches, which in turn causes the level 1 request to drop.

Byte 1, Bit 3: Reset System Reset/NSC Address Active

When on, this bit causes the channel adapter to reset 'system reset' (X'0' byte 0 bit 7) and 'NSC active' (X'7' byte 1 bit 5). If the channel adapter level 3 interrupt request is due to the detection of a system reset sequence, this bit must be used to reset 'system reset' and the level 3 interrupt request.

Byte 1, Bit 4: Set Allow Channel Interface Enable (A and B)

When on, this bit causes the channel adapter to set the 'allow channel interface enable' latch for both interfaces A and B. When the 'enable interface A' and/or 'enable interface B' signals are sent to the CA from the control panel, the appropriate interface(s) become enabled when all channel interface conditions are satisfied.

Note: After power-on reset, the control program must execute an X'7' with this bit on, before the enable/disable switches on the control panel have any effect.

Byte 1, Bit 5: Set ESC Operational

When on, this bit causes the channel adapter to make ESC addresses operational.

Byte 1, Bit 6: Set ESC Command Free

When on, this bit causes the channel adapter to reset the 'ESC command active' latch. This latch is set when the channel adapter hardware detects an initial selection sequence to an ESC address. When the latch is reset, it indicates that the ESC part of the channel adapter is free of commands. Because the channel adapter cannot disable the interface before the ESC is free of commands, this bit must be set by the program whenever it detects the channel adapter to be free of ESC commands. If the latch was set by an initial selection sequence, the channel adapter does not disable the interface.

Byte 1, Bit 7: Set Allow Channel Interface Disable (A and B)

When on, this bit causes the channel adapter to set the 'allow channel interface disable' latch for both interfaces. This latch overrides the channel adapter enable/disable signals when:

- The channel adapter is free of commands.
- No chaining is specified.
- No initial selection is in progress.

Bits 7 and 4 are mutually exclusive.

Registers X'B' and X'C'

INPUT/OUTPUT X'B' (ESC TEST I/O ADDRESS AND STATUS)

The X'B' register contains the test I/O address in byte 0, and the test I/O status in byte 1. The register is loaded by either output X'B' or stacking a final status on the ESC.

The register layout is as follows:

Byte	Bit	Meaning
0	0-7	ESC TIO address byte bits 0-7
1	0	ESC TIO: Attention
	1	ESC TIO: Status modifier
	2	ESC TIO: Control unit end
	3	ESC TIO: Busy
	4	ESC TIO: Channel end
	5	ESC TIO: Device end
	6	ESC TIO: Unit check
	7	ESC TIO: Unit exception

INPUT X'C' (CYCLE STEAL MODE CONTROL)

The X'C' is used in AIO mode. It contains the various CS controls for the AIO operation.

The register layout is as follows:

Byte	Bit	Meaning
0	0	SYN monitor control latch
	1	DLE remember control latch
	2	USASCII monitor control latch
	3	EBCDIC monitor control latch
	4	(Not used)
	5	(Not used)
	6	(Not used)
	7	(Not used)
1	0-7	Residual byte count

Byte 0, Bit 0: SYN Monitor Latch

This bit is only used in host to controller BSC operation. It indicates that four SYN characters have been detected in the data stream.

Byte 0, Bit 1: DLE Remember Control Latch

This bit is used only in host to controller BSC operation. It indicates the state of the 'DLE remember' latch. This latch is set by the channel adapter hardware each time a DLE character is detected in the data stream, and is reset by the following character if it is NOT a DLE character, otherwise the latch stays set. If the last character in the operation is the DLE character, the latch stays set, and the following instruction will find this bit on.

Byte 0, Bit 2: USASCII Monitor Control Latch

This bit is used only in host to controller BSC operation. It indicates that the channel adapter checked for USASCII characters during the last transfer sequence.

Byte 0, Bit 3: EBCDIC Monitor Control Latch

This bit is used only in host to controller BSC operation. It indicates that the channel adapter checked for EBCDIC control characters during the last transfer sequence.

Byte 1, Bits 0-7: Residual Byte Count

This byte contains the number of byte transfers NOT transferred during the last transfer sequence, and represents the difference between the number requested and the number effected. Normally it should be zero.

OUTPUT X'C' (CYCLE STEAL MODE CONTROL)

The output X'C' register is used in AIO mode. It contains various cycle steal controls in byte 0, and the residual byte count in byte 1.

The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	SYN monitor control latch
	1	DLE remember control latch
	2	USASCII monitor control latch
	3	EBCDIC monitor control latch
	4	(Not used)
	5	(Not used)
	6	(Not used)
	7	(Not used)
1	0-7	Request byte count

Byte 0, Bit 0: SYN Monitor Control Latch

This bit is used only on BSC inbound (host to 3725) operations. When on, it sets the 'SYN monitor' latch. It causes the hardware to monitor the data coming from the host for SYN characters. If four consecutive SYN characters are detected in the incoming data stream, the data transfer is terminated, and a channel adapter data/status level 3 interrupt is requested. The 'SYN monitor control' latch is reset when any non-SYN character is detected.

When the bit is off, the 'SYN monitor control' latch is reset, and SYN monitoring is stopped.

Byte 0, Bit 1: DLE Remember Control Latch

This bit is used only on BSC inbound (host to 3725) operations and is used to restore the state of the 'DLE remember control' latch at the start of a new transfer sequence. This is to continue the test for the start of transparent mode. (See "Control Character Recognition", that follows, for use of this bit.)

Byte 0, Bit 3: EBCDIC Monitor Control Latch

This bit is used only on BSC inbound (host to 3725) operations. When on, it indicates that monitoring for certain EBCDIC control characters is to be carried out by the channel adapter hardware. (See "Control Character Recognition", that follows, for use of this bit.)

Byte 1, Bit 0-7: Request Byte Count

This byte contains the count of the number of bytes that are to be transferred to or from the host.

Register X'D'

INPUT X'D' (CA LEVEL 1 INTERRUPT CHECK)

The input X'D' register bits are set by hardware with the various checks that can cause a level 1 interrupt.

The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	IOC bus parity error
	1	Internal bus parity error
	2	CCIN card check
	3	(Not used)
	4	CHIN card check
	5	Address compare error
	6	Initiate service latch ungated
	7	(Not used)
1	0	Output exception check
	1	PIO halt remember latch
	2	Cycle steal halt remember latch
	3	Bus-in check interface A
	4	Ground fault error
	5	Bus-in check interface B
	6	CADR card check interface A
	7	CADR card check interface B

Byte 0, Bit 0: IOC Bus Parity Error

This bit, when on, indicates that a bad parity has been detected on the IOC bus between the CCU and the channel adapter. If the error was detected on data transferred from the CCU to the channel adapter, this bit is set on.

This bit does not cause a level 1 interrupt request.

Byte 0, Bit 1: Internal Bus Parity Error

This bit, when on, indicates that a bad parity has been detected between the CHIN card and the CCIN card.

Byte 0, Bit 2: CCIN Card Check

This bit, when on, indicates that a hardware failure has been detected on the CCIN card. Four different hardware failures may cause this check:

- IOC bus parity error in inbound (host to 3725) operations
- Internal bus parity error on outbound (3725 to host) operations
- I/O command decoder failure

Note: If an invalid CA register is accessed (X'8', X'9', or X'A'), the I/O decoder will fail and cause this check. In this case, there is a program check.

- Byte counter failure

Byte 0, Bit 4: CHIN Card Check

This bit, when on, indicates that a hardware failure has been detected on the CHIN card.

Byte 0, Bit 5: Address Compare Error

This bit, when on, indicates that the program has addressed an emulator subchannel that is outside the plugged address range.

Byte 0, Bit 6: Initiate Service Latch Ungated

This bit indicates that an initial selection has been made, or is in the process of being made, on this channel adapter.

Byte 1, Bit 0: Output Exception Check

This bit, when on, indicates that the channel adapter hardware has detected an invalid output. Outputs, with the single exception of output X'7', are not allowed during a data/status transfer.

Notes:

1. An output X'B' will cause an output exception check unless it is performed during an initial select interrupt.
2. This bit does not cause a level 1 interrupt.

Byte 1, Bit 1: PIO Halt Remember Latch

This bit, when on, indicates that the CCU has detected an error during an input/output operation, and has activated the 'halt' signal on the interface.

This bit does not cause a level 1 interrupt request.

Byte 1, Bit 2: Cycle Steal Halt Remember Latch

This bit, when on, indicates that the CCU has detected an error during cycle stealing, and has activated the 'halt' signal on the interface.

This bit does not cause a level 1 interrupt.

Byte 1, Bit 3: Bus-In Check Interface A

This bit, when on, indicates that a hardware failure has occurred on the channel adapter internal bus path during a data or address transfer to the host. The condition was detected on interface A.

Byte 1, Bit 4: Ground Fault Error

This bit is set if the CA detects that a bus in line or tag in line (except for 'select in') is at ground potential when the line is attempting a logical 1 voltage. The CCU attempts to reset the error using output X'7' with 'reset CA interrupt level 1' (byte 1, bit 2), and tests if a reset took place using input X'D' instructions.

Byte 1, Bit 5: Bus-In Check Interface B

This bit, when on, indicates that a hardware failure has occurred on the channel adapter internal bus path during a data or address transfer to the host. The condition was detected on interface B.

Byte 1, Bit 6: CADR Card Check Interface A

1. A hardware failure has been detected on the interface A driver/receiver card.
2. A hardware failure has been detected in the bus-in or tag-in interface drivers.

Byte 1, Bit 7: CADR Card Check Interface B

1. A hardware failure has been detected on the interface B driver/receiver card caused by a parity error on the CHIN card during an inbound (host to 3725) data transfer. This parity error is detected when the bus out check is inactive, but the channel interface card detects bad parity on data transferred from the bus out register.
2. A hardware failure was detected in the I/O interface drivers.

Note: Byte 0, bit 0 and byte 1, bits 1-2 do not cause a channel adapter level 1 interrupt request to the CCU. Instead, the channel adapter forces an IOC bus error, the IOC bus hardware in the CCU requests a level 1, indicated by CCU input instruction X'7E' byte 0, bit 7 set on with a further definition of the IOC bus error in CCU register X'76'. The channel adapter input register X'D' can then be accessed to determine the cause of the interrupt.

Registers X'E' and X'F'

INPUT X'E' (CA LEVEL 1 INTERRUPT REQUESTS)

This command indicates which CAs have a level 1 interrupt request pending. This register holds information on all the CAs.

The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	CA5 LI interrupt request
	1	(Not used)
	2	CA6 L1 interrupt request
	3	Any CA L1 interrupt request
	4	Channel adapter address bit 0
	5	Channel adapter address bit 1
	6	Channel adapter address bit 2
	7	(Not used)
1	0	CA1 L1 interrupt request
	1	(Not used)
	2	CA2 L1 interrupt request
	3	(Not used)
	4	CA3 L1 interrupt request
	5	(Not used)
	6	CA4 L1 interrupt request
	7	(Not used)

Byte 0, Bits 4-6: Channel Adapter Address Bits

These bits identify the currently selected channel adapter, as for output X'7'.

INPUT X'F' (CA LEVEL 3 INTERRUPT REQUESTS)

The X'F' register indicates which channel adapter is currently selected, and the status of its level 3 interrupts.

An input X'F' also initiates an autoselection in the channel adapters. When an input X'F' is executed, the channel adapter with the highest level 3 priority is selected.

Note: If an input X'F' is performed, the level 3 interrupt for the resultant selected channel adapter must be reset before the autoselect mechanism will allow a new channel adapter to select via another input X'F'.

The latch is reset by:

- An output X'0'
- An output X'2' with byte 0, either bit 5 or 6 set to 1
- An output X'7' with byte 1, bit 3 set to 1 (for system reset only)
- An output X'B'

(For full description of the autoselection mechanism, see "Channel Adapter Selection" page 12-065.)

The bits of the register have the following meaning:

Byte	Bit	Meaning
0	0	(Not used)
	1	TPS installed
	2	Selected CA initial selection level 3 interrupt request
	3	Selected CA data/status level 3 interrupt request
	4	Channel adapter address bit 0
	5	Channel adapter address bit 1
	6	Channel adapter address bit 2
	7	(Not used)
1	0-7	(Not used)

Byte 0, Bit 1: TPS Installed

This bit, when on, indicates that the currently-selected channel adapter is equipped with a TPS.

Byte 0, Bit 2: Selected CA Initial Selection L3 Interrupt Request

This bit, when on, indicates that the currently-selected channel adapter has an initial selection level 3 interrupt request pending.

Byte 0, Bit 3: Selected CA Data/Status L3 Interrupt Request

This bit, when on, indicates that the currently selected channel adapter has a data/status level 3 interrupt request pending.

Byte 0, Bits 4-6: Channel Adapter Address Bits

These bits identify the current by selected channel adapter, as for output X'7'.

CA Functional Partitioning and Selection

Note: For the jumper locations on the logic cards listed below, refer to Chapter 4.

CCIN Card

IOC bus buffer
IOH/IOHI input output decoder
IOC bus tag and command verification
CA data/control storage and controls
CA internal bus gating
Autoselection logic
CA internal timing
AIO (cycle steal) controls
Channel burst length control and jumpering
Residual byte count

CHIN Card

ESC address jumpering
Data in/data out jumpering
Channel tag controls
Channel clock timing
Channel data/status control
Channel data buffer
Channel command decodes

CADR Card

NSC address jumpering
Select out/select in jumpering
Channel drivers and receivers

CVTL Card

Miscellaneous timing and control logic

CHANNEL ADAPTER SELECTION

Although there may be up to six channel adapters installed, the control program can perform channel adapter input and output on only one channel adapter at a time. In order to do so, the control program must first select that channel adapter. Two methods are available:

- Output X'7'
- Autoselection

OUTPUT X'7'

As explained in "Accessing Channel Adapter Registers", page 12-016, the IOH and IOHI instructions are used to access the channel adapter registers. These two instructions do not contain an explicitly defined channel adapter address, but are performed on the selected channel adapter. In order to perform channel adapter input and output, the first IOH or IOHI issued must be a CA output X'7' with the applicable select bits set. The following is a description of the CA output X'7' using an IOH or IOHI.

Input/Output (IOH)

IOH Halfword Instruction:

0	1	3	4	5	7	8								15
0	R2			0	R1		0	1	0	1	0	0	0	0

Register R2:

0	1		3	4	5		7	8			11	12		14	15
0	0	0	0	1	0	0	0	x	x	x	x	0	0	0	x

Bit 4=1, indicates a channel adapter IOH

Bits 8-11=0111, indicate a CA output X'7'

Bit 15=0, indicates an output

Register R1:

0	1	2	3	4	5	6	7	8							15
x	x	1	0				x	x	x	x	x	x	x	x	

Bit 2=1, indicates select CA addressed by bits 4-6

Bits 4-6= 000 select CA#1
001 select CA#2
010 select CA#3
011 select CA#4
100 select CA#5
101 select CA#6

Refer to "Channel Adapter Input/Output Registers" for a definition of the bits that may be 0 or 1, indicated by x.

Input/Output Immediate (IOHI)

First halfword of IOHI:

0				4	5		7	8			11	12		14	15
0	0	0	0	0		R		0	1	1	1	0	0	0	0

Second halfword of IOHI:

0				4	5		7	8			11	12		14	15
0	0	0	0	1	0	0	0					0	0	0	x

Bit 4=1, indicates a channel adapter IOHI

Bits 8-11=0111, indicate a CA output X'7'

Bit 15=0, indicates an output

Register R:

0	1	2	3	4	5	6	7	8							15
x	x	1	0				x	x	x	x	x	x	x	x	

The bit definition of R of an IOHI is the same as the bit definition of R1 of an IOH.

AUTOSELECTION

Note: Refer to the channel adapter troubleshooting section for detailed autoselection timing, signal propagation, and signal description.

The autoselection hardware is a mechanism by which the channel adapter with the highest priority level 3 interrupt will become selected when a CA input X'F' is executed via an IOH or IOHI instruction.

When a CA level 3 interrupt occurs, the control program needs to select the channel adapter with the interrupt in order to service it. By using a CA input X'F' via an IOH or IOHI instruction, the channel adapter with the level 3 interrupt will become selected and the currently selected channel adapter will deselect. In addition, if more than one channel adapter has a level 3 interrupt pending, the channel adapter with the highest priority will be selected. The level 3 interrupts are listed below by priority order from highest to lowest.

Priority outbound data transfer
Outbound data transfer
Initial select
Inbound data transfer
Any other data/status interrupt
(for example, PI, or stacked status)

CA Enabling/Disabling, Interrupt Requests and Control Characters

CA ENABLING/DISABLING

Note: Refer to the "Channel Adapter Troubleshooting", starting on page 12-800, for detailed routing of the enable and disable signals. Refer to page 6-010 for an illustration of the panel showing the enable/disable switches.

The channel interface for each channel adapter must be enabled before communication can take place with a host channel that the channel adapter is attached to. Switches located on the control panel control the enabling and disabling of the channel interfaces.

Notes:

1. Setting the switches on the control panel in the enable position will not enable a channel adapter unless a CA output X'7' with byte 1, bit 4 on has been issued, using an IOH or IOHI instruction to the channel adapter being enabled. In addition, an output X'7' with byte 1, bit 5 on must be executed to enable the ESC addresses.
2. If a channel adapter has a TPS and is attached to a loosely-coupled host, or hosts, only one interface may be enabled at a time on the channel adapter affected.

LEVEL 1 INTERRUPT REQUESTS

When an error condition is detected in the CA, a bit is set in the CA level 1 interrupt check bit register X'D' to indicate the type of error:

Byte	Bit	Meaning
0	0	PIO bus parity error*
	1	Internal bus parity error
	2	CCIN card check
	3	(Not used)
	4	CHIN card check
	5	Address compare error
	6	Initiate service latch ungated
	7	(Not used)
1	0	Output exception check *
	1	PIO halt remember latch *
	2	Cycle steal halt remember latch *
	3	Bus in check interface A
	4	Ground fault error
	5	Bus in check interface B
	6	CADR card check interface A
	7	CADR card check interface B

The CA raises a level 1 interrupt request for all CA errors (except for those marked by an *). These errors combine to cause an IOC bus check, which raises a level 1 interrupt request to the CCU.

LEVEL 3 INTERRUPT REQUESTS

A CA raises a level 3 interrupt request to the CCU by activating IOC bus byte 1, bit 0 when the I/O is not active. There are two types of level 3 interrupt request:

1. CA initial selection request
2. Data/status transfer request

An initial selection interrupt request may be determined by executing an input X'0'. A data/status interrupt request may be determined by executing an input X'2'.

CONTROL CHARACTER RECOGNITION

Certain control characters from the host channel are recognized by the CA hardware, which sets corresponding bits in register X'C'. The control program modifies its monitoring action according to these bits.

Name	EBCDIC	ASCII
Circle B	3D	
2848 ETX	03	
DLE	10	10 or 90
STX	02	02 or 82
ETB	26	17 or 97
ETX	03	03 or 83
SYN	32	16 or 96

Circle B monitoring is started by an output X'2' with byte 1, bit 0 on and 2848 ETX with byte 1, bit 2 on. BSC EBCDIC and ASCII DLE, STX, ETB and ETX monitoring are initiated by a output X'C' with byte 0, bit 2 or 3 on. Both bits cannot be set at the same time. SYN monitoring requires output X'C' byte 0, bit 0 to be on along with either byte 0, bits 2 or 3.

For a Circle B, 2848 ETX, EBCDIC or ASCII ETB or ETX, the CA hardware does not accept any further data from the host after receiving the control character, and sets a CA level 3 interrupt request with channel stop indicated in register X'2'.

When a DLE is immediately followed by an STX, the CA resets all monitoring for EBCDIC or ASCII characters. The DLE STX sequence indicates that BSC transparency mode has been activated. In transparency mode, data may be sent that has the same bit pattern as a control character, which is why monitoring must be stopped.

If a DLE is the last character received before a level 3 interrupt occurs due to the CA byte count reaching zero, the CA hardware sets the DLE remember bit in register X'C' byte 0, bit 1. When the control program initiates the continuation of the command, this bit must be set with an output X'C'. If the first character received is an STX, the CA resets control character monitoring.

Four consecutive SYN characters received at the start of an inbound transfer cause a level 3 interrupt request and set the channel stop condition X'2' byte 0, bit 5. Any non-SYN character resets monitoring for SYN characters.

Two-Processor Switches (Part 1 of 2)

TWO-PROCESSOR SWITCH (LOOSELY-COUPLED HOST ATTACHMENT)

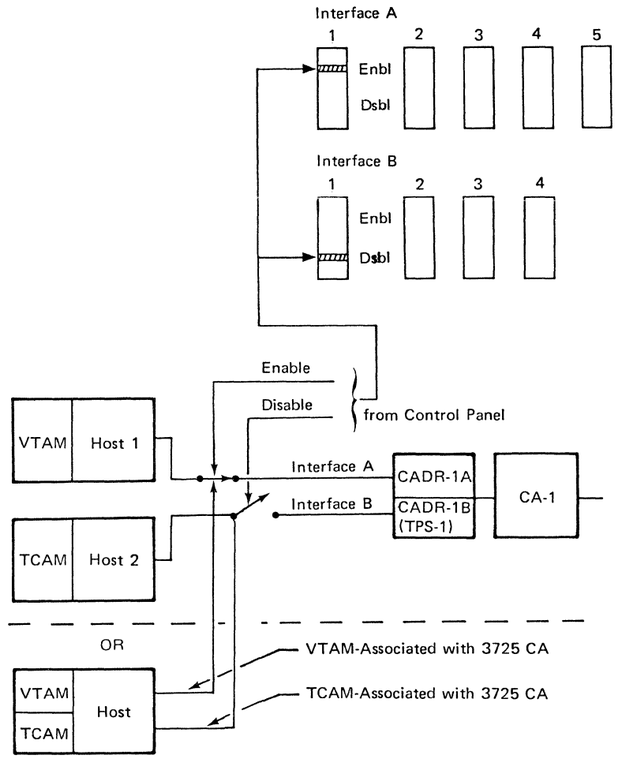
Note: This information does not apply to the 3725 Model 2.

The term "loosely-coupled" means either:

a. Two separate hosts each running its own access method (TCAM/VTAM) with each host attached to an interface of a CA with a two-processor switch, or

b. A single host running different access methods, one access method associated with one interface of a CA via a two-processor switch, and the other access method associated with the other interface of the same CA.

When running in a loosely-coupled environment, only one of the interfaces may be enabled. The following figure shows this environment.



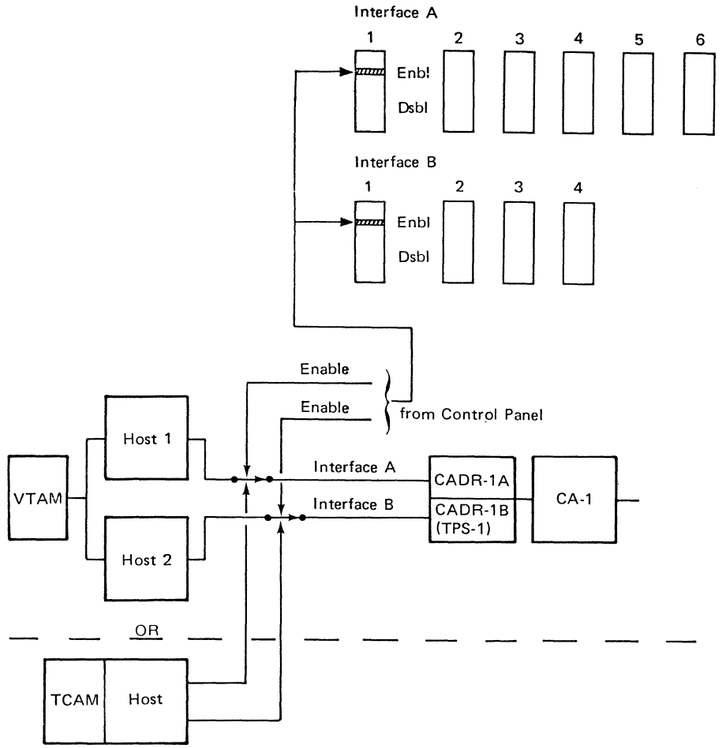
TWO-PROCESSOR SWITCH (TIGHTLY-COUPLED/ALTERNATE PATH HOST ATTACHMENT)

The term "tightly-coupled" is defined as two hosts in a multihost system both using the same access method (TCAM or VTAM) to access both interfaces of a CA having a two-processor switch. Either host can access the CA.

Alternate path attachment is a single access method in a single host having access to both interfaces of a CA that is equipped with two-processor switch. With alternate path, if the access method issues a start I/O over one channel, and that channel is busy with another device, the start I/O can then be issued over the alternate channel.

Note: More than one access method can operate in tightly-coupled or alternate path hosts, but only one can access both interfaces of a CA with a two-processor switch.

When running in a tightly-coupled or alternate path environment, both CA interfaces can be enabled. The following figure shows this environment.



TIGHTLY-COUPLED/ALTERNATE PATH HOST OPERATIONS

In order to understand these operations, the following definitions are required:

1. Neutral: The CA is not in communication with either interface, and no active commands exist in the CA.

2. Switched: The CA is actively communicating with an interface. If the CA is active with a command, all additional communication is to be done over this interface. The switched state comprises three substates:

   a. Short-term or instantaneous allegiance: The CA enters this state when it traps 'select out' during a channel-initiated sequence, or a poll during a control-unit-initiated sequence. If the CA presents DE without Unit Check (UC) during initial selection, and chaining, is not indicated, the CA returns to the neutral state.

   b. Long-term or implicit allegiance: The CA enters this state when accepting a command without presenting DE, or when a No-Op is chained to the next command. This state lasts until the CA presents DE without UC to the channel. The CA remains switched to the same interface throughout. If chaining is not indicated, and DE is not stacked, the CA returns to the neutral state.

   c. Contingent allegiance: This state is an extension of long-term allegiance. When a command ends with a UC status (because of some error), the CA enters this state, even though DE is presented to the command. The CA remains switched to the same interface, and does not return to neutral. The host, on receiving the UC status, issues a sense command to determine the cause of the error. The CA returns to neutral when receiving a command other than No-Op or Test I/O.

The following table summarizes the various states of allegiance.

State	Substate	Meaning
Neutral		No allegiance
Switched	short-term	Initial select seq present async status present tagged DE
	long-term	Service out during initial select seq except No-Op
	contingent	UC status presented to host during init sel or final status
Return Neutral		At command end with DE acceptance and no chaining. DE accepted or stacked after busy. Async status accepted or stacked. Command other than No-Op or Test I/O in contingent state.

Two-Processor Switches (Part 2 of 2)

New ending statuses to the host are as follows:

1. Normal tagged status: This status is presented to the host in response to an initial selection command. The CA remains switched in longterm allegiance to the interface until the control program presents the normal ending status tagged for that interface.

2. Tagged DE status: When the CA has presented a busy status to an interface because the other was in long-term or contingent state, that interface presents DE status to the host when the CA returns to the neutral state. This DE status is tagged for that interface by the hardware to clear the busy status previously presented to the host. The tagged DE status frees the host so that it can issue the next command.

3. Untagged asynchronous status. When operating with tightly-coupled processors, the control program can present asynchronous status to the host to initiate a command. This occurs when the CA is in the neutral state with no commands active.

   'Request in' is raised to both interfaces, and both interfaces compete for service. The first channel to poll the CA wins. Acceptance of the status causes a level 3 interrupt request, and the CA returns to neutral. If the channel stacks the status, the CA returns to neutral and interrupts the control program. The untagged asynchronous status is then re-offered to both interfaces by the CA when the interrupt is serviced by the control program.

PRESENTATION OF STATUS

Although the CA has two channel interfaces, simultaneous operation of both interfaces is impossible. Therefore, when the CA is switched to one interface, they cannot both operate simultaneously. The other interface refuses all channel-initiated sequences. When the CA is in long-term or contingent allegiance on one interface, The other interfaces present the status X'00' to the channel. When 'select out' is trapped by the non-switched interface, it raises 'status in' and presents X'10'. The host channel responds by dropping 'hold out' or 'select out' and 'address out', thereby disconnecting the interface. The host does not issue further commands until a tagged DE status is received. This status is presented when the active CA interface returns to the neutral state. The CA hardware raises 'request in' and presents the DE status. If the DE status is stacked instead of accepted, the CA hardware automatically tries to represent the status, independently of the control program.

EFFECT OF SYSTEM RESET

System Reset over Interface Having Allegiance

When the CA recognizes the system reset, the CA is completely reset, ending any allegiance condition, and sets an initial selection interrupt request on level 3. However, if a DE status resulting from a previous busy status on the opposite interface is still pending, it is not reset.

During system reset, if the opposite channel polls the CA in response to a 'request in' from some other control unit, the resulting 'select out' tag is bypassed. Similarly, any channel-initiated initial selection sequence to either interface causes the CA to switch to that interface, to present the busy status, and to return to the neutral state.

System Reset over Interface Not Having Allegiance

When a system reset occurs on the interface not having allegiance, any pending tagged DE status is reset. The rest of the CA is not reset, and no level 3 initial selection interrupt request is made.

System Reset when the CA is in Neutral

When a system reset occurs on a neutral CA, only the pending tagged DE status is reset on the interface, if any. The rest of the CA hardware is not reset, and no level 3 initial selection interrupt request is made.

Note: If a system reset is presented to both interfaces simultaneously, the CA is completely reset, and a level 3 interrupt request is made.

EFFECT OF SELECTIVE RESET

Selective Reset over Interface Not Having Allegiance

A selective reset cannot occur on the interface not having allegiance. Therefore, the CA hardware is not reset, and no level 3 initial selection|interrupt request is made.

	Command from Host Channel			
CA State	System Reset A	System Reset B	Select, Reset A	Select, Reset B
Switched to A	Reset all CADR A and CHIN hardware Return to neutral state, and IS interrupt	Reset tagged DE on CADR B	Reset Bus In & Tag In & tagged DE status. Return to neutral state, and IS interrupt	N.A.
Switched to B	Reset tagged DE on CADR A.	Reset all CADR B & CHIN hardware Return to neutral state, and IS interrupt	N.A.	Reset Bus In & Tag In & tagged DE status. Return to neutral state, and IS interrupt
Neutral	Reset tagged DE on CADR A	Reset tagged DE on CADR B	N.A.	N.A.
	Simultaneous System Reset			
	Resets all CHIN and both CADR hardware, and issues an IS interrupt request.			

IS = Initial selection

Selective Reset over Interface Having Allegiance

When the CA recognizes the selective reset, it returns to the neutral state, and issues a level 3 initial selection interrupt request.

No hardware reset occurs, except for a tagged DE status caused by a previous busy status on the opposite interface.

During the initial selection, if the opposite channel polls the CA in response to 'request in' from some other control unit, the resulting 'select out' tag is bypassed.

Similarly, if a channel-initiated initial selection sequence occurs on either interface, the CA switches to that interface, enters the short-term allegiance state, presents control unit busy X'70' as the initial status, and returns to the neutral state.

CA Error Condition

Any CA-detected error sets the appropriate bit on in register X'D'.

IOC bus parity error, output exception check, PIO halt, and cycle steal halt do not cause CA level interrupts. Instead, the CA forces an IOC line error, the IOC bus hardware in the CCU requests a level 1 interrupt, indicated by CCU Input X'7E' byte 0 bit 7 with a further definition of the IOC bus error in CCU register X'76'.

CA Input X'D' can be accessed to determine the cause of the interrupt. All other CA error conditions described cause level 1 interrupts.

In X'D' Bit	Error Description	Level 1 Interrupt
0.0	IOC bus parity error	No
1	Internal bus parity error	Yes
2	CCIN card check	Yes
4	CHIN card check	Yes
5	Address compare error	Yes
6	Initiate service latch ungated	No
1.0	Output exception check	No
1	PIO halt remember latch	No
2	Cycle steal halt remember latch	No
3-5	Bus-in check (A and B)	Yes
4	Ground fault error	Yes
6-7	CADR card check (A and B)	Yes

There is no checking built into the CVIL card.

The interrupt handler in the CCU:

- Determines the CA in error and the type of error by issuing:
  - Input X'E' to find out which CA raised the level 1 interrupt request
  - Output X'7' to select the CA, if not selected
  - Input X'D' to get type of CA error
- Completes the operation in progress by issuing:
  - Output X'7' with byte 1, bit 2 (set pgm request interrupt)
  - Output X'7' to reset the level 1 interrupt request
  - Output X'3' to set the ESC status, or
  - Output X'6' to set the NSC status
  - Output X'2' to set the transfer status (CE, DE, UC)

IOC Bus Parity Error

This error is caused by bad parity on the IOC bus during byte 0 and byte 1 transfers.

Internal Bus Parity Error

This error is caused by bad parity on the internal bus (on the CCIN card, the CHIN card, or between the cards).

CCIN Card Check

This indicator is set for four different reasons:

1. Inbound PIO parity error
2. Internal bus parity error outbound
3. I/O command decoder failure
4. Counter failure

CHIN Card Check

This indicator is caused by bad parity on the interface bus within the CHIN card.

Address Compare Error

This error is caused by the CCU issuing an ESC address out of range for this CA. The CCU resets the error indication and the interrupt, and tries to restart the operation.

Initiate Service Latch Ungated

This condition indicates that an initial selection has been made, or in the process of being made, on the channel adapter.

Output Exception Check

The CA checks for CCU output commands (except output X'7') that may interfere with a host data or status transfer. Similarly, output X'B' is allowed only during initial selection or program-requested interruption.

PIO Halt Remember Latch

If the CCU detects an error during a PIO operation, it raises the halt tag line to stop the operation. The CA sets the PIO halt remember latch.

Cycle Steal Halt Remember Latch

If the CCU detects an error during an AIO operation, it raises the halt tag line to stop the operation. The CA sets the CS halt remember latch.

Bus-In Check (A and B)

'Bus-in' parity is generated and checked during data transfers. Bad parity sets 'channel bus-in check' and stops the data transfer.

Bad parity during initial selection has no effect and initial selection terminates normally. Status transfers are not parity checked.

Bad parity during the address part of a channel command has no effect, and the address sequence ends normally.

Ground Fault Error

If the CA detects that a 'bus in' line or 'tag in' line (except for the 'select in' line) is at ground potential when the line is attempting a logical 1 potential, byte 1, bit 4 (ground fault) is set in input X'D'. The CCU attempts to reset the error by output X'7' with byte 1, bit 2 (reset CA interrupt level 1), and tests if the reset took place with input X'D' commands.

CADR Card Check (A and B)

The 3725 interface drivers to the host are checked for the correct ON and OFF conditions. These checks are performed continuously after 3725 power on. Except for 'select in', all the 'tag in' and 'bus in' drivers are checked at the input (base) of the final transistor driver to the interface. This check point is compared with a point earlier in the logic of the CADR card, and is not scopable.

Initial Selection (Part 1 of 2)

During 'address out' time at the channel interface, an enabled CA compares the address on bus out against its own jumpered NSC address (or jumpered ESC address if the ESC range is enabled). The CA traps 'select out' if the addresses match. The next command will be accepted unless a short busy sequence exits (X'70' status), and the CA normally causes a level 3 initial selection interrupt request to inform the control program. If the addresses do not match, 'select out' is propagated to the next CA.

INITIAL STATUS TRANSFER

Note: For final status, see "Final Status" later in this chapter.

The following tables show the status that is presented to the host channel during initial selection sequence. The status presented depends on the command. Unless otherwise stated, level 3 interrupt requests are not issued.

The native subchannel (NSC) is active when initial status to an NSC command (except TIO and No-Op) has been accepted by the host but final status has not yet been presented, or when the NSC asynchronous status set up by the control program is pending. A TIO or No-Op does not cause the NSC to become active. The NSC does not become active if a channel bus out check occurs during the NSC command transfer, and the subsequent UC status (initial or final) is accepted by the host.

Note: The CA returns busy X'10' to the channel if it is already in long-term allegiance on the other interface.

STACKED INITIAL STATUS

Any initial status (except for X'00') can be stacked in response to a command except TIO. When an initial status is stacked, the CA causes a level 3 interrupt request with input X'0' byte 0, bit 5 on to inform the control program. The control program then reads the stacked status (hardware-generated) and presents the status as though it were a final status transfer (as described under "NSC Final Status Transfer" and "ESC Final Status Transfer". If the ESC addresses are enabled, the control program sets up a 'suppress out monitor' sequence as described under "ESC Stacked Final Status".

ESC Initial Status Chart - Command to ESC Address

	CA Status (See Notes 1 and 2)				
CA STATE / Command	ESC status not available and IS or PI or sel rst	ESC status available and command to same ESC address and not IS, not PI and not sel rst	ESC status available and command to different ESC address and not IS, not PI and not sel rst	ESC status available and IS or PI or sel rst	ESC status not available and not IS, not PI and not sel rst
TIO	70	Available status	70 (Note 3)	70	70 (Note 3)
No-Op	70	0C (Note 4)	0C (Note 4)	70	0C (Note 4)
All other commands not listed above	70	00 (Note 3)	00 (Note 3)	70	00 (Note 3)

For explanation of notes, see NSC initial status table.

Initial Selection (Part 2 of 2)

NSC Initial Status - Command to NSC Address

	CA Status (See Notes 1 and 2)						
CA STATE / Command	(TPS Op)	Not active & IS or PI or sel rst & not NSC status available	Active & status pending	Active & NCS status available & not PI & not sel rst	Active & IS, PI & sel rst & not NSC status available	Not active, not NSC status available and not IS, not PI, & not sel rst	Active & not NSC status available and not IS, not PI, & not sel rst
TIO	10	70	70	Available status (Note 3)	70	00	10
No-Op	10	70	70	Available status + busy (Note 4)	70	0C (Note 4)	10
Sense Sense ID Read Write Write Br	10	70	70	Available status + busy (Note 3)	70	00 (Note 3)	10
All other commands Not listed above	10	70	70	Available status + busy (Note 3)	70	08 (Note 3)	10

All codes are in hexadecimal
IS=initial selection

Notes:

1. Status codes:

   10=busy
   0C=CE, DE
   08=CE
   70=CUE, M and Busy

2. CA states:

   Active=NSC active.
   IS=initial selection level 3 interrupt request.
   PI=program interrupt on level 3.
   Sel Rst=data transfer selective reset (input X'2' byte 1, bit 1).
   (TPS Op)=Two-processor switch operation.
   Busy is present when the other interface is in long term allegiance.
   Pending status=Any other NSC pending status (CA is active).

3. A level 3 interrupt occurs.

4. A level 3 interrupt occurs only if the X'0C' status is started.

Data Transfer Sequences (Part 1 of 4)

There are four possible data transfer sequences, depending on whether the transfer is to or from the host, and whether the transfer is an adapter-initiated operation (AIO) or a program-initiated operation (PIO). AIO mode uses cycle stealing, but PIO mode does not. In addition, there are some minor differences between NSC and ESC modes of operation.

DATA TRANSFER IN AIO MODE (CYCLE STEALING)

In AIO mode, two 8-bit data storage positions are used for data transfer. The CA starts data transfer in AIO mode by raising the line 'cycle steal request'. After all CAs have removed 'valid halfword' (VH) from the bus, the IOC sends 'cycle steal grant' (CSG). The CA then puts the CSCW on the IOC bus with VH and PV (parity valid) on.

The CSCW indicates where in main storage the data for this transfer starts. Once started, the operation continues until the CA tells the IOC to stop.

The procedure for exchanging data between the IOC and the adapter is the same as for PIO mode, except for the last transfer. If the last transfer is for two bytes only, the CA sets 'end-of-chain' (EOC) instead of VH. If the last transfer is for one byte only, the CA sets the 'modifier' (M) bit as well as 'valid byte' (VB). When the IOC recognizes either EOC or M+VB, it terminates the data transfer.

The two data storage positions are used in flip-flop mode: while one is being loaded, the other may be read, and conversely.

Outbound Data Transfer Sequence in AIO Mode

When responding to a read type I/O command from the host (data transfer from 3725 to host), the control program executes the following sequence of instructions:

1. Output X'3n' Fixed Pointer Register (n=0 to 5)

   This instruction loads the CCU pointer address register X'3n' with the cycle steal data address for the CA. The 'n' corresponds to the CA address, as follows:

'n'	CA#
0	1
1	2
2	3
3	4
4	5
5	6

2. Output X'3' ESC Address and Status Byte Register

   This instruction is for ESC operations only. It loads byte 0 of the X'3' with the emulation subchannel address for the data transfer. Byte 1 is not used.

   Note: This instruction is not required for NSC operations as the assigned hardware NSC address is always used.

3. Output X'C' Cycle Steal Mode Control Register

   This instruction loads the X'C' register with the transfer byte count. The byte count must be in the range 1 through 255.

4. Output X'2' Data/Status Control Register

   This instruction loads the X'2' register with controls for the AIO transfer, as follows:

   - Byte 0, bit 0=1 means outbound transfer.
   - Byte 0, bits 1 and 2 must both be zero.
   - Byte 0, bit 3=1 if ESC operation, otherwise NSC operation.
   - Byte 0, bit 4=1 if PIO mode, otherwise AIO mode.
   - Byte 0, bit 5=1 means the control program has already recognized an initial selection sequence, and needs to reset the resulting CA initial selection level 3 interrupt request.
   - Byte 0, bit 6=1 resets all control latches used in the preceding transfer.
   - Byte 0, bit 7 and byte 1, bits 1-3 and bits 5-7 must be set to zero.

When the output X'2' instruction is executed, the CA hardware cycle steals bytes from the CCU storage, and sends them to the channel for transfer to the host. At completion of the transfer, the CA hardware raises a level 3 data/status interrupt request to inform the control program that the transfer is complete.

At the end of an outbound data transfer on the NSC, the CA hardware may present a CE status before the level 3 data/status interrupt is generated. This CE status lets the CA disconnect from a block multiplex channel at the end of a transfer sequence, whether data or status. The output X'2' instruction should have byte 0, bit 0 outbound and bit 2 status transfer both on, to present the hardware-generated CE status. The control program should include the CE status in the final status. If a hardware-generated CE has already been presented, the CA blocks the CE status set up by the control program.

Note: After completion of the outbound cycle steal operation, the CA fixed pointer register in the CCU (reg X'3x') cannot be used to determine from what CCU storage location the last data byte transferred over the channel was taken. Instead, the last data byte storage location can be determined by using an Input X'C' and the initial setting of the fixed pointer register (reg X'3x').

5. Input X'2'-Data/Status Control Register

   For a data transfer sequence that has ended normally, the register is as follows:

   - Byte 0, bit 0=1 indicates outbound
   - Byte 0, bit 2=1 if the CE status was presented
   - Byte 0, bit 3=1 if ESC, 0 if NSC
   - Byte 0, bit 4=1 if CE status was sent to the channel.
     This bit may be stacked for use by the control program later.

6. Input X'C' Cycle Steal Mode Control Register

   This instruction is used to read the residual byte count at data transfer end. The count is zero if the transfer ended normally but may range up to 255 otherwise.

   If any unusual condition occurred during the transfer, the corresponding bit(s) are set in register X'2', and the residual byte count in X'C' represents the number of bytes NOT transferred.

Inbound Data Transfer Sequence in AIO Mode

When responding to a write type I/O command from the host (data transfer from host to 3725) the control program must execute the following sequence of instructions:

1. Output X'3n' Fixed Pointer Register (n=0 to 5)

   This instruction loads the CCU pointer address register X'3n' with the cycle steal data address for the channel adapter. The 'n' corresponds to the address of the channel adapter:

'n'	Channel Adapter
0	1
1	2
2	3
3	4
4	5
5	6

2. Output X'3' ESC Address and Status Byte Register

   This instruction is used for ESC operations only; it is not required for the NSC. It loads byte 0 of the ESC Address and Status Byte register with the emulation subchannel address to be used for the transfer. Byte 1 (status) is not used.

   Note: This instruction is not required for NSC operations as the assigned hardware NSC address is always used.

3. Output X'C' Cycle Steal Mode Control Register

   This instruction loads the register with the number of bytes to be transferred across the interface. The byte count must be in the range 1 through 255 (X'01' through X'FF'). The instruction also initializes the BSC character recognition hardware as follows:

   a. Byte 0, bit 0: Monitor for SYN characters
   b. Byte 0, bit 1: Set the DLE remember latch
   c. Byte 0, bit 2: Monitor for USASCII control characters
   d. Byte 0, bit 3: Monitor for EBCDIC control characters

Data Transfer Sequences (Part 2 of 4)

4. Output X'2' Data/Status Control Register

   This instruction loads the data/status control register with the information needed to control the operation, and starts the transfer. The bits of this register should be set as follows:

   - Byte 0, bit 1 must be set to 1 to indicate an inbound data transfer sequence.
   - Byte 0, bits 0 and 2 must both be set to 0.
   - Byte 0, bit 3 must be set to 0 to indicate NSC operation, or to 1 to indicate ESC operation.
   - Byte 0, bit 4 must be set to 0 to indicate AIO operation.
   - Byte 0, bit 5 must be set to 1 if the control program has recognized an initial selection sequence and wants to reset the resulting channel adapter Initial Selection Level 3 Interrupt Request.
   - Byte 0, bit 6 must be set to 1 in order to reset any hardware latches that were set during the last data or status transfer sequence.
   - Byte 0, bit 7, and byte 1, bits 0 through 7 must all be set to 0.

When the output X'2' instruction is executed, the channel adapter hardware obtains bytes from the channel adapter and cycle steals them to CCU storage. At the completion of the transfer, the channel adapter hardware raises a channel adapter data/status level 3 interrupt request to inform the control program that the transfer is complete.

Note: After completion of the inbound cycle steal operation, the CA fixed pointer register in the CCU (reg X'3x') cannot be used to determine where the last data byte received from the host was stored in CCU storage. Instead, the final storage location can be determined by using an Input X'C' and the initial setting of the fixed pointer register (reg X'3x').

5. Input X'2' - Data/Status Control Register

   For a data transfer sequence that ended normally, the bits of this register have the following status:

   - Byte 0, bit 1 is 1 indicating an inbound data transfer sequence.
   - Byte 0, bit 3 is 0 indicating an NSC operation, or 1 indicating an ESC operation.

6. Input X'C' - Cycle Steal Mode Control Register

   This instruction is used to obtain the residual byte count at the end of the transfer. The residual byte count will be in the range 1 through 255 (X'01' through X'FF'), but will normally be zero if the transfer was successful.

   If any unusual conditions occurred during the transfer, such as channel stop, selective reset, or interface disconnect, the corresponding bit will have been found on when register X'2' was examined. Byte 1, bits 0 through 7 of register X'C' then indicate the number of bytes that were NOT transferred.

The instruction also indicates the current status of the BSC recognition hardware as follows:

- Byte 0, bit 0: SYN monitor control latch
- Byte 0, bit 1: DLE remember control latch
- Byte 0, bit 2: USASCII monitor control latch
- Byte 0, bit 3: EBCDIC monitor control latch

DATA TRANSFER IN PIO MODE

In PIO mode, four storage positions are used for data transfer. Bytes 0 and 1 are transferred between storage and register X'4' and bytes 2 and 3 are transferred between storage and X'5'.

During an inbound operation (host to controller) the 4 bytes are loaded by the channel, then a level 3 data/status interrupt request is made by the CA. The CCU then reads the 4 bytes using Input X'4' and Input X'5'. Before executing the X'4' and X'5', the control program examines the the residual byte count in input register X'2', and compares it to the requested byte count in output register X'2'. The data is valid if the following conditions are met:

Data Storage Position	Comparison between Residual and Requested Byte Count
0	Residual < requested
1	Residual at least 2 less than requested
2	Residual at least 3 less than requested
3	Residual=0

During an outbound transfer (controller to host) operation, the control program loads the four storage positions with 4 bytes using output X'4' and X'5'. The data transfer is started by the CA hardware when the control program sets 'outbound' (X'2' byte 0 bit 0). When the 4 bytes have been transferred to the channel, the CA hardware raises a level 3 data/status interrupt request to the control program to continue the data transfer. The control program verifies the requested byte count against the actual byte count, and when the two are equal, stops the data transfer.

Outbound Data Transfer Sequence in PIO Mode

When responding to a read type I/O command from the host (data transfer from 3725 to host) the control program executes the following sequence of instructions:

1. Output X'3' - ESC Address and Status Byte Register

   This instruction is used for ESC operations only; it is not required for the NSC. It loads byte 0 of the ESC address and status byte register with the emulation subchannel address to be used for the transfer. Byte 1 (status) is not used.

   Note: This instruction is not required for NSC operations as the assigned hardware NSC address is always used.

2. Output X'4' - Data Buffer Bytes 1 and 2

   This instruction loads the first two bytes of the four-byte PIO buffer register in the channel adapter hardware.

3. Output X'5' - Data Buffer Bytes 3 and 4

   This instruction loads the last two bytes of the four-byte PIO buffer register in the channel adapter hardware.

4. Output X'2' - Data/Status Control Register

   This instruction loads the data/status control register with the information needed to control the operation, and starts the transfer. The bits of this register should be set as follows:

   - Byte 0, bit 0 must be set to 1 to indicate an outbound data transfer sequence.
   - Byte 0, bits 1 and 2 must both be set to 0.
   - Byte 0, bit 3 must be set to 0 to indicate NSC operation, or to 1 to indicate ESC operation.
   - Byte 0, bit 4 must be set to 1 to indicate PIO operation.
   - Byte 0, bit 5 must be set to 1 if the control program has recognized an initial selection sequence and wants to reset the resulting channel adapter initial selection level 3 interrupt request.
   - Byte 0, bit 6 must be set to 1 in order to reset any hardware latches that were set during the last data or status transfer sequence.
   - Byte 0, bit 7, and byte 1, bits 1, 3, and 4 must all be set to 0.
   - Byte 1, bits 5 through 7 specify the number of bytes to be transferred. A maximum of four bytes may be transferred during any one sequence:

Byte 1 Bits 5 6 7	Meaning
0 0 1	1-byte transfer
0 1 0	2-byte transfer
0 1 1	3-byte transfer
1 0 0	4-byte transfer

Data Transfer Sequences (Part 3 of 4)

Byte Multiplex Channel

When the output X'2' instruction is executed, the channel adapter hardware notifies the channel that it needs service. It then reselects to the channel interface, identifies itself by gating its I/O device address to the channel bus in, sends the specified number of data bytes to the channel, and deselects from the channel. The channel adapter hardware then raises a channel adapter data/status Level 3 interrupt request to inform the control program that the transfer is complete.

The control program should react by executing an Input X'2' instruction to determine the cause of the interrupt request. For a data transfer sequence that has ended normally, the bits of this register will have the following status:

- Byte 0, bit 0 is 1 indicating an outbound data transfer sequence.
- Byte 0, bit 3 is 0 indicating an NSC operation, or 1 indicating an ESC operation.
- Byte 1, bits 5 through 7 indicate the residual byte count. It should be zero, indicating that all the requested bytes have been transferred.

If any unusual conditions occurred during the transfer, such as channel stop, selective reset, or interface disconnect, the corresponding bit will be found on in register X'2'. Byte 1, bits 5 through 7 then indicate the number of bytes that were <u>not</u> transferred:

Byte 1 Bits 5 6 7	Meaning
0 0 1	1 byte not transferred
0 1 0	2 bytes not transferred
0 1 1	3 bytes not transferred
1 0 0	4 bytes not transferred

The byte multiplexor channel may be forced into burst mode by the channel adapter. The number of bytes per transfer is then determined by two jumpers set at installation time. Burst length is up to 4 bytes in PIO mode, and 8, 16, 32, or 64 bytes in AIO mode.

Block Multiplex or Selector Channel

Only the NSC address can be used; the output X'3' instruction is therefore not required. When the output X'2' instruction is executed, the channel adapter remains selected to the channel, the 'Select Out' and 'Operational In' tag lines remaining on for the whole selection time. The 'Service Out' tag is also on. The channel adapter hardware places the first byte of data on bus out and raises the 'Service In' tag.

When the transfer is complete, the channel adapter hardware raises a data/status level 3 interrupt request to inform the control program of the result of the transfer. The control program should react by executing an input X'2' instruction, followed by the sequence Output X'4', output X'5', and output X'2' to transfer each byte to the channel. Each sequence of these three instructions is followed by a data/status level 3 interrupt.

Transfer Termination

At the end of an outbound data transfer on the NSC, and irrespective of the channel type, the hardware may present a Channel End status before the data/status level 3 interrupt is generated. This Channel End status is particularly useful to a block multiplex type channel, as it allows the channel adapter to disconnect from the channel at the end of the last transfer sequence, which should be for both a data and status transfer. The last output X'2' instruction should have byte 0, bits 0 (outbound data transfer sequence) and 2 (status transfer sequence) set to 1 in order to present the hardware-generated Channel End. The control program should include Channel End in the final status. If a hardware Channel End has already been presented, the channel adapter blocks the Channel End set up by the control program.

Inbound Data Transfer Sequence in PIO Mode

When responding to a write type I/O command from the host (data transfer from host to 3725) the control program executes the following sequence of instructions:

1. Output X'3' - ESC Address and Status Byte Register

   This instruction is used for ESC operations only; it is not required for the NSC. It loads byte 0 of the ESC address and status byte register with the emulation subchannel address to be used for the transfer. Byte 1 (status) is not used.

   <u>Note:</u> This instruction is not required for NSC operations because the assigned hardware NSC address is always used.

2. Output X'2' - Data/Status Control Register

   This instruction loads the data/status control register with the information needed to control the operation, and starts the transfer. The bits of this register should be set as follows:

   - Byte 0, bit 0 must be set to 0.
   - Byte 0, bit 1 must be set to 1 to indicate an inbound data transfer sequence.
   - Byte 0, bit 2 must be set to 0.
   - Byte 0, bit 3 must be set to 0 to indicate NSC operation, or to 1 to indicate ESC operation.
   - Byte 0, bit 4 must be set to 1 to indicate PIO operation.
   - Byte 0, bit 5 must be set to 1 if the control program has recognized an initial selection sequence and wants to reset the resulting channel adapter Initial Selection Level 3 Interrupt Request.
   - Byte 0, bit 6 must be set to 1 in order to reset any hardware latches that were set during the last data or status transfer sequence.
   - Byte 0, bit 7, and byte 1, bits 0 through 4 must all be set to 0.
   - Byte 1, bits 5 through 7 specify the number of bytes to be transferred across the channel by this output X'2' instruction. A maximum of four bytes may be transferred during any one sequence:

Byte 1 Bits 5 6 7	Meaning
0 0 1	1-byte transfer
0 1 0	2-byte transfer
0 1 1	3-byte transfer
1 0 0	4-byte transfer

Data Transfer Sequences (Part 4 of 4)

1. Input X'4' - Data Buffer Bytes 1 and 2

   This instruction obtains the first two bytes of the 4-byte PIO from the buffer register in the channel adapter hardware.

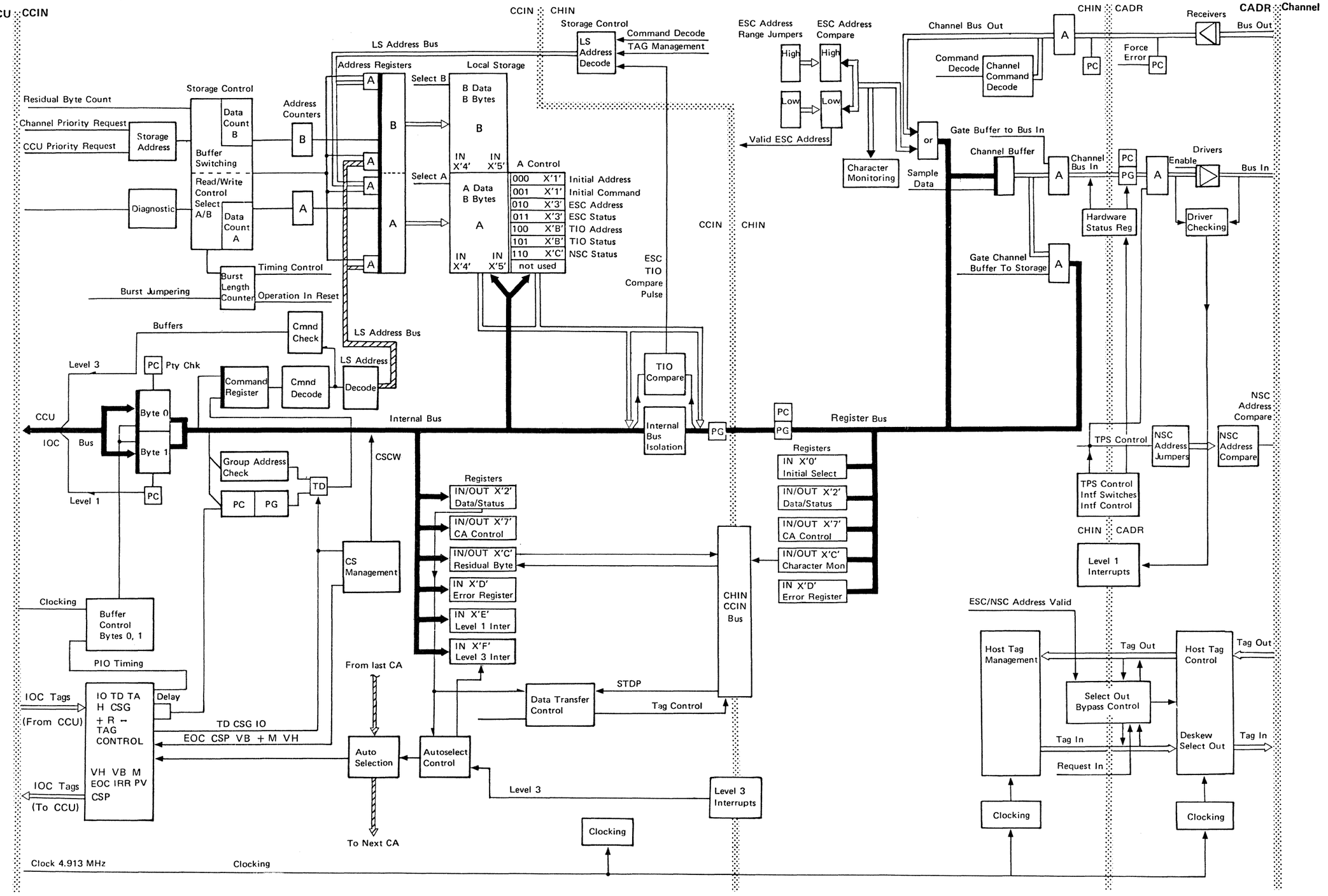
2. Input X'5' - Data Buffer Bytes 3 and 4

   This instruction obtains the last two bytes of the 4-byte PIO from the buffer register in the channel adapter hardware.

Block Multiplex or Selector Channel

Only the NSC address can be used; the Output X'3' instruction is therefore not required. When the Output X'2' instruction is executed, the channel adapter remains selected to the channel, the 'Select Out' and 'Operational In' tag lines remaining on for the whole selection time. The channel places the first byte of data on bus out and raises the 'Service Out' tag. The channel adapter hardware accepts the byte of data from bus out and raises the 'Service In' tag. When the transfer is complete, the channel adapter hardware raises a data/status level 3 interrupt request to inform the control program of the result of the transfer. The control program should react to this interrupt in the same way as for the byte multiplex channel.

CA Data Flow


CCU
CCIN
CHIN
CADR
Channel
Storage Control
LS Address Bus
Address Registers
Local Storage
Select B
Select A
B Data
B Bytes
A Data
B Bytes
A Control
000 X'1' Initial Address
001 X'1' Initial Command
010 X'3' ESC Address
011 X'3' ESC Status
100 X'B' TIO Address
101 X'B' TIO Status
110 X'C' NSC Status
not used
IN X'4'
IN X'5'
Residual Byte Count
Channel Priority Request
CCU Priority Request
Storage Address
Buffer Switching
Data Count B
Data Count A
Read/Write Control Select A/B
Diagnostic
Address Counters
Timing Control
Burst Length Counter
Burst Jumpering
Operation In Reset
Buffers
Cmnd Check
LS Address
Command Register
Cmnd Decode
Decode
Level 3
PC
Pty Chk
Level 1
Byte 0
Byte 1
Internal Bus
CCU IOC Bus
Group Address Check
TD
PG
CSCW
CS Management
Registers
IN/OUT X'2' Data/Status
IN/OUT X'7' CA Control
IN/OUT X'C' Residual Byte
IN X'D' Error Register
IN X'E' Level 1 Inter
IN X'F' Level 3 Inter
Clocking
Buffer Control Bytes 0, 1
PIO Timing
IOC Tags (From CCU)
IO TD TA H CSG + R - TAG CONTROL
Delay
TD CSG IO
EOC CSP VB + M VH
VH VB M EOC IRR PV CSP
IOC Tags (To CCU)
From last CA
Auto Selection
Autoselect Control
To Next CA
Level 3
Level 3 Interrupts
Clocking
Clock 4.913 MHz
Clocking
Storage Control
LS Address Decode
Command Decode
TAG Management
ESC Address Range Jumpers
ESC Address Compare
High
Low
Valid ESC Address
Character Monitoring
Sample Data
or
Gate Buffer to Bus In
Channel Buffer
Gate Channel Buffer To Storage
ESC TIO Compare Pulse
TIO Compare
Internal Bus Isolation
Register Bus
Registers
IN X'0' Initial Select
IN/OUT X'2' Data/Status
IN/OUT X'7' CA Control
IN/OUT X'C' Character Mon
IN X'D' Error Register
CHIN CCIN Bus
STDP
Data Transfer Control
Tag Control
Channel Bus Out
Command Decode
Channel Command Decode
Force Error
Receivers
Bus Out
Channel Bus In
Drivers
Enable
Bus In
Hardware Status Reg
Driver Checking
TPS Control
NSC Address Jumpers
NSC Address Compare
TPS Control Intf Switches Intf Control
Level 1 Interrupts
ESC/NSC Address Valid
Host Tag Management
Tag Out
Host Tag Control
Select Out Bypass Control
Tag In
Request In
Deskew Select Out
Clocking

Final Transfer Status/Sense Definition (Part 1 of 2)

NSC FINAL STATUS TRANSFER

When the control program has finished processing an NSC command, it presents the final status for that command by executing the following sequence of instructions:

1. Output X'6': NSC Status/Control Register

   This instruction loads byte 1 of register X'6' with the status to be presented.

2. Output X'2': Data/Status Control Register

   This instruction loads register X'2' with the information needed to control the operation, and starts the transfer. The bits of this register should be set as follows:

Byte	Bit	Setting
0	0	Off
0	1	Off
0	2	On
0	3	Off
0	4	Off
0	5	On if initialized
0	6	On to reset hardware latches
0	7	Off
1	0-7	Off

When the Output X'2' instruction is executed, the CA hardware transfers the NSC status from register X'6' to the channel. At the completion of the transfer, the CA hardware raises a data/status interrupt request on level 3 to inform the control program that the transfer is complete.

The control program reacts by executing an Input X'2' instruction to determine the cause of the interrupt request. For a status transfer sequence that has ended normally, the bits of the register have the following status:

- Byte 0, bit 2 is on (status transfer)
- Byte 0, bit 3 is off (NSC operation)

NSC STACKED FINAL STATUS

When an NSC final status is stacked, the hardware causes a CA data/status interrupt request level 3. The control program reacts to this request by executing an Input X'2' instruction to determine the cause of the interrupt request, by examining the following bits:

- Byte 0, bit 2 (on = status transfer)
- Byte 0, bit 3 (off= NSC operation)
- Byte 1, bit 3 (on = stacked ending status)

The control program may now take one of the following options:

1. Reinitiate the final status transfer (if an ESC address is not enabled), or
2. Set up suppress out monitor sequence (if an NSC operation)

ESC FINAL STATUS TRANSFER

When the control program has finished processing an ESC command, it presents the final status for the command by executing the following sequence:

1. Output X'3' ESC Address and Status Byte Register

   This instruction loads X'3' with the I/O device address and its associated status.

2. Output X'2' Data/Status Control Register

   This instruction loads X'2' with the information needed to control the operation, and starts the transfer. The bits of the register should be set as follows:

Byte	Bit	Setting
0	0	Off
0	1	Off
0	2	On if status transfer
0	3	On if ESC operation
0	4	Off
0	5	On if selected
0	6	On to reset hardware latches
0	7	Off
1	0-7	Off

When the Output X'2' instruction is executed, if the CA is deselected from the channel, the hardware notifies the channel that it needs service.

When the CA is reselected ('select in' and 'operational in' both on) the CA identifies itself by gating its I/O address to channel bus in, sending its status, and deselecting. The CA hardware then raises a data/status interrupt request on level 3 to inform the control program that the transfer is complete.

The control program reacts by executing an Input X'2' as already described. For a status transfer that has ended normally, the register bits are as follows:

- Byte 0, bit 2 is on (status transfer)
- Byte 0, bit 3 is on (ESC operation)

If any unusual condition occurred during the transfer, the corresponding bits will be found in X'2'.

ESC STACKED FINAL STATUS

When an ESC final status is stacked, the CA hardware loads the ESC address and status into X'B', sets the ESC TIO status available latch, and raises a data/status interrupt request level 3. The control program reacts by executing an Input X'2' instruction. The status should be as follows:

- Byte 0, bit 2 on (status transfer)
- Byte 0, bit 3 on (ESC operation)
- Byte 1, bit 3 on (stacked ending status)

The control program sets up a suppress out monitor sequence to complete the transfer, as follows:

1. Output X'2' to reset the level interrupt request
2. Output X'7' to set up suppress out monitor
3. Input X'2' to monitor the 'suppress out' tag line

   and when 'suppress out' drops

4. Output X'2' to set 'suppressible status' and to transfer the ESC stacked status to the channel.

Final Transfer Status/Sense Definition (Part 2 of 2)

CA MONITORING TASK FINAL STATUS/SENSE INFORMATION

Note: For all initial status presentations including TIO and No, refer to "Initial Status Transfer", page 12-085.

The CA monitoring task is a 3725 program that runs during IPL from phase 1 through the end of phase 2. The presentation of status and sense information depends on how the IPL is started.

1. If IPL results from power on, or power-on reset from the control panel, the CAs are disabled and return condition code 3, and bypass 'select out'
2. If IPL results from CCU hardcheck, the CAs are not monitored during the IPL sequence. The status returned depends on the condition of the CA when the hardcheck occurred, and operations since the hardcheck.
3. If IPL results from an IPL command from the host, or remote IPL (link-attached), or control program abend, or the 'I' function on the 3727 operator console while the control program is running, or pushing the function start button on the control panel while the function select switch is in the 'normal' position, any CAs that were enabled are left enabled, and are monitored during IPL until the channel loader/dump program is loaded from diskette. The CA monitoring task resets any ESC interrupts, and responds with the final status as shown in the following table:

Status	Command	Explanation
CE+DE (sense X'02' not initialized	Sense	IPL not yet received over a remote link, or via another CA
CE+DE	Sense ID	Normal ending status
CE+DE+UE (with cmd reject sense X'82' or X'02' not init	Sense	IPL in progress on another channel, or remote link, and is not required on this CA
CE+DE+UC	Read Write Write break	Indicates that the command cannot be accepted because the 3725 is not IPLed
No final status after CE initial status	Write IPL	The level 3 interrupt due to Write IPL is not serviced until the dump program takes control
DE+UC	All commands other than TIO Sense No-Op Sense ID Write Write Break Write IPL Read	Indicates that no control is loaded, and the command is not accepted
DE	Asynchronous	Occurs after synchronous busy (an interface returned busy to the host because of init selection on the other interface in long-term allegiance) generated by hardware

CHANNEL LOADER DUMP PROGRAM FINAL STATUS/SENSE INFORMATION

Note: For all initial status presentations, including TIO and No-Op, refer to "Initial Status Transfer", page 12-085.

This program loads the control program from the host, or dumps the contents of main storage at the host.

Status	Command	Explanation
CE then DE (Sense X'02' not initialized	Sense	IPL not yet received over a remote link, or via another CA
CE then DE	Sense ID	Normal ending status
CE then DE+UE (with cmd reject sense X'82' or X'02' not init	Sense	IPL is in progress over another CA, or remote link, and is not required on this CA
CE then DE	Write Write Break Read	Normal ending status for a Write or Write Break after a Write IPL is received, and for a Read after a Write is received (dump function)
CE+DE+UC	Read Write Write Break	Returned to a Write or Write Break command if Write IPL is not received, and to Read if Write not received (dump)
DE	Asynchronous	Occurs after synchronous busy (an interface returned busy to the host because of init selection on the other interface in longterm allegiance) generated by hardware
DE	Write IPL	Normal ending status
DE+UC	Asynchronous	Indicates IPL is required
DE+UC (with cmd reject Sense X'82' not initialized)	Commands other than: TIO Sense No-Op Sense ID Write Write Break Write IPL Read	Command reject condition

NCP Channel Commands Information

Command Code	Command	Description
X'00'	TIO	Requests the 3725 to present pending status.
X'01'	Write	The Write command is initiated to the NCP. Data in the host processor main storage is transferred to the NCP.
X'02'	Read	The Read command is initiated at the NCP. Data at controller storage is transferred to host processor main storage.
X'03'	No-Op	This command is required as the last CCW in a Read or Write CCW chain.
X'04'	Sense	The host initiates this command. One byte of sense data is transferred to the host.
X'05'	Write IPL	Host command to the 3725 indicating that an NCP/PEP/EP is to be loaded.
X'09'	Write Break	The Write Break command is identical to the Write command except that it indicates that it is the last or only Write command in a chain of Write CCWs.
X'31'	Write Start 0	This is the first command expected in the Write Channel program after IPL of the NCP. It is also expected after each successful Write Start 1 command.
X'32'	Read Start 0	This is the first command expected in the Write channel program after IPL of the NCP. It is also expected after each successful Read Start 1 command.
X'51'	Write Start 1	This is the second command expected in the Write Channel program after IPL of the NCP. It is also expected after each successful Read Start 0 command.
X'52'	Read Start 1	This is the second command expected in the Read channel program after IPL of the NCP. It is also expected after each successful Read Start 0 command.
X'61'	WXID	The host sends the Write XID command to signal NCP that a channel contact sequence is beginning and prepare to receive host's XID.
X'62'	RXID	The host sends the Write XID command to signal NCP that a channel contact sequence is beginning and prepare to receive host's XID.
X'93'	Reset Restart	This command causes the NCP to reset its switches to indicate that the last Write Start and Read Start commands were Write Start 1 and Read Start 1.
X'A3'	Discontact	This command tells the NCP to exit the contacted state with the host. The host: - Indicates that the channel is no longer contacted. - Indicates that attachment to the transmission group should be broken. - Release PIUs on the channel hold and the intermediate queues.
X'C3'	Contact	This command tells the NCP to set up for operation with the host identification data
X'E4'	Sense ID	This command requests the machine type and model number the NCP returns four bytes (X'FF372500') to the host.

Data transfer does not occur on Read Start and Write Start commands.

NCP Status Information

The following four tables show the final and asynchronous status generated by the NCP.

Note: For all initial status presentations including TIO and NOOP, refer to "Initial Status Transfer", page 12-085.

Status	Command	Explanation
CE + DE	Read Write Write break Sense Sense ID	Normal Final Status for these commands.
DE	Write IPL Write start 0 Write start 1 Read start 0 Read start 1 WXID RXID Reset Restart Discontact Contact	Normal Final Status for these commands.
DE	Asynchronous	1. Given after synchronous busy (i.e. one channel interface had given busy to the host during an initial select because the other interface on the same CA was in long term allegiance (hardware generated DE). 2. Given after an IPL over the non IPL'ing CA's.
CE+DE+UE	Read	Normal end of channel transfer unit (CTU).
	Write Write break	Buffer depletion occurred on this PIU of the CTU. The PIU associated with this CCW has been rejected.
CE+DE+UE (with command reject and not initialized sense X'82')	Read Write Write break	The host attempted some operation over a channel while IPL was in progress over another channel adapter or a remote link.
CE+DE+UE (with not initialized sense X'02')	Sense	The 3725 is currently being loaded over another CA of a remote link. This status is given by the channel loader dump program (CLDP).
CE+DE+UE (with sense = X'00')	Sense	The 3725 has been loaded and no additional load attempt is required.

Status	Command	Explanation
DE + UE	Read Start 0 Read Start 1	A channel transfer unit (CTU) is not available (attention status has not been presented). Attention may accompany this status.
	Write Start 0 Write Start 1	No NCP buffers are available for host Writes (buffer depletion). Attention may accompany this status.
	WXID	No NCP buffers are available to receive the host identification data.
CE+DE+UC (with data check sense X'0F')	Read	Host buffer unit is too small to contain the data to be transferred or the read has been halted by a host I/O.
	Write/Write Break	The count field included with the data transfer did not agree with the number of bytes sent by the host.
CE+DE+UC (with abort sense X'01')	Read Write Write break	The host attempted some operation when contact was lost.
CE+DE+UC (with not initialized X'02' or intervention required sense (X'40')	Read Write Write break	The NCP aborted. This status is given by the channel loader dump program (CLDP) that was loaded into storage from the 3725 diskette during the IPL caused by the abend.
CE+DE+UC (with command reject sense X'80')	Read Write Write break	The channel adapter hardware is declared inoperative by the NCP.
DE+UC (with about sense X'01')	Read start 0 Read start 1 Write start 0 Write start 1 WXID RXID Reset restart Discontact	The host attempted some operation when contact was lost
DE+UC (with not initialized X'02' or intervention required sense (X'40')	Read start 0 Read start 1 Write start 0 Write start 1 WXID RXID Reset restart discontact contact or asynchronous	The NCP aborted. This status is given by the channel loader dump program (CLDP) that was loaded into storage from the 3725 diskette during the IPL caused by the abort.

Status	Command	Explanation
DE+UC (with command reject sense X'80')	Read start 0 Read start 1 Write start 0 Write start 1 WXID RXID Reset restart Discontact Contact	The channel adapter hardware is declared inoperative by the NCP.
ATTN	Asynchronous	The NCP has data ready to transfer to the host.
CE+DE+ ATTN	Read	Normal end but NCP has more data to send to the host indicator. Another read is required.
	Write break	Indicates NCP has data to send to the host and requires a read.
CE+DE+ +SM	Write break	Normal end and allows chaining to a read channel program.
(CE)+DE+ UE+SM	Read	Normal end of channel transfer unit (CTU) and buffers are now available for host Writes.
DE+UE+SM	Read start 0 Read start 1	A read was attempted but no data was available.
CE+DE+ ATTN+SM	Read	Normal end but indicates NCP has more data to send to the host and a read is required. Also indicates buffers are available for host writes.

NCP Sense Information

0 – Command Reject
1 – Intervention Required
2 – Bus Out Check
3 – Equipment Check
4 – Data Check
5 – Not Used
6 – Not Initialized
7 – Abort

Sense bits	Error	Probable cause	Error Description
0	Command reject	Host program failure	Command reject. The control unit has detected an invalid command from the channel.
1	Intervention required	Program failure	Intervention required. The control unit has detected an internal programming error.
2	Bus out check	Channel failure	Bus out check. The control unit has detected data from the channel that is not in proper parity.
3	Equipment check	I/O control unit failure	Equipment check. The control unit has detected an internal hardware error.
4	Data check	Host program failure	Data check. Host buffer too small, PIU incorrect, possibly caused by Halt I/Os or channel errors.
5	Not used		
6	Not initialized	Host program failure	Not initialized. The control unit does not have a control program loaded.
7	Abort	I/O control unit failure	Abort. The control unit has halted its channel operation abnormally.

EP Channel Commands

Command Code	Command Name	Description
X'00'	TIO	Requests pending status for addressed line.
X'01'	Write	Data from the host is transferred to the addressed EP line
X'02'	Read	Data from the addressed EP line is transferred to the host
X'03'	No-Op	This command is sent from the host to the 3725. The 3725 responds with an immediate CE+DE; or, if an initial selection is present or, if a programmed interrupt is pending, short busy is presented.
X'04'	Sense	A byte of sense data pertaining to the addressed EP line is transferred to the host.
X'05'	Diag Write	This command is treated by EP as an No-Op.
X'05'	Write IPL	Host command to the 3725 indicating that an NCP/PEP/EP is to be loaded.
X'06'	Prepare	When a wake-up or line break signal is received on the line addressed by the prepare command, the command is terminated by a CE+DE.
X'09'	Poll	This command causes EP to request polling characters from the host.
X'0A'	Inhibit	An X'02' operation is performed except that line timeout is inhibited.
X'0D'	Break	A break signal is sent over the addressed EP line.
X'0E'	Search	Data from the addressed communication line is not transferred to the host except for characters of an ending sequence. This command is valid only if the CSP controlling the line is in emulation mode.
X'12'	Diag Read	This command is treated by EP as a No-Op.
X'13'	Sad Zero Set addr0	This command is treated by EP as a No-Op.
X'17'	Sad One Set addr1	This command is treated by EP as a No-Op.

Command Code	Command Name	Description
X'19'	Poll SOH	An X'09' operation is performed except that an SOH character is sent to the addressed line before the poll characters.
X'1B'	Sad Two Set addr2	This command is treated by EP as a No-Op.
X'1D'	Diag Poll	This command is treated by EP as a No-Op.
X'1E'	Adprep Address Prepare	The host prepares the 3725 for polling by a BSC terminal on the addressed line. This command is valid only if the CSP controlling the line is in emulation mode.
X'1F'	Sad Three Set addr3	This command is treated by EP as a No-Op.
X'23'	SETMODE	The host specifies the operating mode of the 2701 or 2703 being emulated by EP.
X'27'	Enable	The host requests that the addressed EP line be enabled. No data transfer occurs.
X'29'	Dial	Dial digits are transferred from the host to the dial equipment.
X'2F'	Disable	The host requests that the addressed EP line be disabled. No data transfer occurs.
X'41'	Write Break	Execution is similar to X'01' except that no data is sent to the addressed EP line. The command ends when carrier from the line drops, or when a timeout occurs.
X'58'	Read Clear	Execution is identical to X'02' except that no decoding of control characters is performed.

EP Ending Statuses (Part 1 of 2)

The following tables summarize status information that the Emulation Program can present to the host processor. The status byte bit designations are:

Bit 0 Attention (not used)

1 Status Modifier (SM)
2 Control Unit End (CUE)
3 Busy
4 Channel End (CE)
5 Device End (DE)
6 Unit Check (UC)
7 Unit Exception (UE)

Ending status combination

Status	Meaning
CE+DE+SM	Poll Command Termination: this three-bit combination indicates that either (1) a start-stop or BSC Poll command was terminated due to a positive response, or (2) a BCS Poll command was terminated due to no response (that is, a timeout occurred).
CE+DE	Command Termination: This 2-bit combination indicates that the command was executed and terminated normally.
CE+DE+UE	Unusual Condition: This 3-bit combination indicates that an exceptional condition has occurred. This condition is unique for each command.
CE+DE+UC	Command Termination: This 3-bit combination indicates that the command was terminated abnormally. The sense byte further defines defines the error condition.

Ending Status to Halt I/O (Start-Stop)

Command	Channel End, Device End Status	Channel End, Device End and Unit Exception Status	Channel End, Device End, and Unit Check Status
READ INHIBIT SEARCH POLL (receive)	Halt I/O was received before first character was received.	N.A.	Halt I/O was received after the receiving bit was set. Lost data is set in the sense byte.
PREPARE	Halt I/O was received after the line went to space. (command ends normally).	Halt I/O was received before a true start bit was received.	N.A.
WRITE POLL (xmit) BREAK	Halt I/O acts in the same manner as a stop sequence for these commands. It does not cause Unit Exception or Unit Check. Note: UE and UC may be set as a result of these commands.	N.A.	N.A.
DIAL	Halt I/O was received after the channel issued stop. (Entire number was dialed).	Halt I/O was received before the channel issued Stop. A connection was not established	N.A.
ENABLE	Switched network; Halt I/O was received after the attached data set established a connection (generated Data Set Ready). The Line remains enabled.	Non-Switched network; Halt I/O was received before the attached data set could establish a connection. The line remains disabled.	N.A.
	Non-Switched network; Halt I/O was received after Data Set Ready was received. The line remains enabled.	Non-Switched network; Halt I/O was received before Data Set Ready was received. The line remains disabled.	N.A.
DISABLE	Halt I/O was received after Data Set Ready dropped (command was not aborted).	N/A	N/A

Ending Status to Halt I/O (BSC)

Command	Channel End, Device End Status	Channel End, Device End and Unit Exception Status	Channel End, Device End, and Unit Check Status
READ SEARCH	Halt I/O was received before character phase was detected.	N.A.	Halt I/O was received after character phase was detected. Lost data is set in the sense byte.
PREPARE	The command ended after character phase (normal end) was detected.	Halt I/O was received before character phase was detected.	N.A.
SET MODE	The command ended before halt I/O was received	Halt I/O was received before the command ended normally	N.A.
WRITE	Halt I/O acts in the same manner as a Stop sequence for this command	N.A.	Halt I/O acts in the same manner as Read for receive position of poll.
AD PREP	Halt I/O was received after a group, selection, or polling address was recognized	Halt I/O was received before a group, selection, or polling address was recognized	N.A.
POLL	Halt I/O acts in the same manner as Stop for transmit portion of poll	N.A.	Halt I/O acts in the same manner as Read for receive portion of poll
DIAL	Halt I/O was received after the channel issued Stop (entire number was dialed).	Halt I/O was received before the channel issued Stop. A connection was not established.	N.A.

Note: if a command signals Channel End and Device End only, the operation continues as though Halt I/O were never issued.

EP Ending Statuses (Part 2 of 2)

Command	Channel End, Device End Status	Channel End, Device End and Unit Exception Status	Channel End, Device End, and Unit Check Status
ENABLE	Switched network. Halt I/O was received after the attached data set established a connection (generated Data Set Ready). The line remains enabled.	Switched network. Halt I/O was received after the attached data set established a connection (generated Data Set Ready). The line remains disabled.	N.A.
	Non-switched network; Halt I/O was received after Data Set Ready was received. The line remains enabled.	Non-switched network: Halt I/O was received before Data Set Ready was received. The line remains disabled.	N.A.
DISABLE	N.A.	N.A.	N.A.

Ending Status CE, DE, and UE (Start-Stop)

Command	Channel End, Device End, and Unit Exception Status			
	1050,1060 2740 and 2741	1030	TTY 33 and 35	83B2/83B3 and WU 115A
READ INHIBIT	C was received N was received while in control mode.		EOT character was received	FIGS-H-LTRS was received. (For World Trade operation, the H may be any customer selected character A-Z)
SEARCH	Invalid command			FIGS-H-LTRS was received.
WRITE DIAG WRITE DIAL POLL (xmit)	The addressed communication line was receiving when the command was accepted at initial selection.			
PREPARE	Halt I/O was issued to the addressed communication line before it received a valid start bit			
ENABLE	Switched network: Halt I/O was issued before the attached data set established a connection. Non-switched network: Halt I/O was issued before the Enable function was completed.		Switched network: Halt I/O was issued before the attached data set established a connection	Non-switched network: Halt I/O was issued before the Enable function was completed.
DIAL	Halt I/O was issued before all the digits necessary to complete the call were presented to the ACU.	N/A	Halt I/O was issued before all the digits necessary to complete the call were presented to the ACU.	N/A

Ending Status CE, DE, and UE (BSC)

Command	Channel End, Device End, and Unit Exception Status
READ SEARCH	EOT character was received while the communication line was in control mode.
SET MODE	Halt I/O was received before the command ended.
WRITE DIAL POLL (xmit)	The addressed communication line was receiving when the command was accepted at initial selection.
PREPARE	Halt I/O was received before character phase was detected.
ADDRESS PREPARE	Halt I/O was received before group, selection, or polling address was recognized.
ENABLE	Switched Network: Halt I/O was issued before the attached data set established a connection. Non-Switched Network: Halt I/O was issued before the Enable function was completed.
DIAL	Halt I/O was issued before all the digits necessary to complete the call were presented to the ACU.

EP Sense Information (Part 1 of 2)

SENSE BIT = 0: COMMAND REJECT

Command	Line Type	Probable Cause	Error Description
01	BSC	Host Program Failure	Line disabled
02	BSC	Host Program Failure	Line in trans parent wait mode or disabled
06	BSC	Host Program Failure	Line in trans parent wait mode or disabled
09	BSC	Host Program Failure	Line in trans parent wait mode or disabled
09	SS	Host Program Failure	Command invalid for some SS terminals
0A	BSC	Host Program Failure	Command invalid for some BSC terminals
0D	BSC	Host Program Failure	Command invalid for some BSC terminals
0D	SS	Host Program Failure	Command invalid for some SS terminals
0E	BSC	Host Program Failure	Line in trans parent wait mode, no station selection capability or line disabled.
0E	SS	Host Program Failure	Command invalid for some SS terminals
1E	SS	Host Program Failure	Command invalid for SS terminals.
1E	BSC	Host Program Failure	Line in transparent wait mode, no station selection capability or line disabled.
23	SS	Host Program Failure	Command invalid for S/S terminals.

SENSE BIT = 0 (CONTINUED)

Command	Line Type	Probable Cause	Error Description
23	BSC	Host Program Failure	Line in trans parent wait mode.
27	BSC	Host Program Failure	Line in trans parent wait mode.
29	SS	Host Program Failure	No auto call feature installed.
29	BSC	Host Program Failure	Line in trans parent wait mode, or no auto call feature installed.
2F	BSC	Host Program Failure	Line in trans parent wait mode.
42	BSC	Host Program Failure	Command invalid for BSC terminals.
All other command	SS and BSC	Host Program Failure	Command invalid.

SENSE BIT = 1: INTERVENTION REQUIRED

Command	Line Type	Probable Cause	Error Description
01,09	BSC	Modem Interface Failure	Line (not enabled) Modem-(power off, on hook, not in data mode, not attached, not operational, CTS dropped).
0D	SS	Modem Interface Failure	Line (not enabled) Modem-(Power off, on hook, not in data mode, not attached, not operational, CTS dropped).

SENSE BIT = 1 (CONTINUED)

Command	Line Type	Probable Cause	Error Description
01,09	SS	Modem Failure	Line (not enabled) Modem-(power off, on hook, not in data mode, not attached, not operational, CTS dropped). Break signal received.
02,06 0E,1E	BSC	Modem Failure	Line (not enabled) Modem-(power off, on hook, not in data mode, not attached, not operational).
02,0A 0E	SS	Modem Failure	Line (not enabled) Modem-(power off, on hook, not in data mode, not attached, not operational). Space for over 1 character time.
06	SS	Modem Failure	Line (not enabled) Modem-(power off, on hook, not in data mode, not attached, not operational).
27	SS and BSC	Modem Failure	DSR does not rise after one second (non-switched line).
29	SS and BSC	Modem Failure	Auto Call Unit-(Power off, not attached, DLO on).

EP Sense Information (Part 2 of 2)

SENSE BIT = 2: BUS-OUT PARITY CHECK

Command	Line Type	Probable Cause	Error Description
All	NA	Channel Failure	Wrong bus-out parity when output data (from the channel) is being presented.

SENSE BIT = 3: EQUIPMENT CHECK

Command	Line Type	Probable Cause	Error Description
All	SS and BSC	Hardware Failure	Hardware detected error, logged in EP error log. Also when EP loaded to unhang subchannel and when command already active

SENSE BIT = 4: DATA CHECK

Command	Line Type	Probable Cause	Error Description
02,0E 1E,09	BSC	Communications Failure	Check Character did not compare. Incorrect control sequence transparent mode, or VRC error on ASCII.
01	SS	Communications Failure	VRC or echo check.
02,0A	SS	Communications Failure	VRC, LRC, Circle N response to text or line at space at stop bit time.
09	SS	Communications Failure	VRC, echo check, line at space at stop bit time or response received other than Circle N or D
0E	SS	Communications Failure	Line at space at bit time.

SENSE BIT = 5: OVERRUN

Command	Line Type	Probable Cause	Error Description
02,0A	SS	Host Program/ Program Failure	Data service is not honored before next character or buffer is received.
02,0E	BSC	Host Program/ Program Failure	Data service is not honored before next character or buffer is received.
01	BSC	Host Program/ Program Failure	Data service is not honored before next character or buffer is received.
09	BSC	Host Program/ Program Failure	Data service is not honored before next character or buffer is received/ transmitted.

SENSE BIT = 6: LOST DATA

Command	Line Type	Probable Cause	Error Description
02,06 0A,0E	SS and BSC	Host Program/ Failure	Data service request or receiving bit is on when Read or Halt I/O issued. Channel issues Stop during read service operation.

SENSE BIT = 7: TIMEOUT

Command	Line Type	Probable Cause	Error Description
01,09 0D	BSC	Modem Interface Failure	Line does not become transmit operational within 25.5 seconds.
01,09 0D	SS	Modem Interface Failure	Line does not become transmit operational within 25.5 seconds. or no reply received-(control timeout 09).
02,0E	SS	Communications Secondary Failure	No character received within specified time interval.
02,0E	BSC	Communications Secondary Failure	No character received within 3 seconds.
29	SS and BSC	Modem Interface Failure	Auto Call unit returns Abandon Call and Retry.
2F	SS	Modem Interface Failure	DSR does not go off within 25.6 seconds.
2F	SS	Modem Interface Failure	DSR does not go off within 3 seconds.
06	SS	Communications Failure	Line open (continuous space) for 25.6 seconds.

Channel Tag Line Definitions (Part 1 of 2)

Address In

'Address in' is a tag line from the attached CAs to the channel. It signals the channel when the address of the currently selected CA has been placed on 'bus in'. During an initial selection sequence, or a CA-initiated sequence, the channel responds to 'address in' by raising 'command out'. 'Address in' stays up until the rise of 'command out'. 'Address in' must fall so that 'command out' may fall. 'Address in' is not up concurrently with any other inbound tag line.

Address Out

'Address out' is a tag line from the channel to attached CAs. It signals all the control units to decode the address on 'bus out'. If any CA recognizes the address, it responds by raising 'operational in' when 'select out' and 'hold out' rises with 'address out' still up (except during the short-busy sequence). 'Address out' rises 250 nanoseconds after the address is placed on 'bus out' or after the rise of 'operational out', whichever occurs later. 'Address out' is down for at least 250 nanoseconds before its rise for CA selection. If 'address out' falls before 'select out' rises, the CA selection is cancelled.

'Address out' can rise only when 'select out' and 'hold out', 'select in', 'status in', and 'operational in' are down at the channel, except for interface disconnect. Ultimate use of the address on 'bus out' at the CA is timed by the next rise of 'select out' at the addressed CA. The rise of 'address out' is delayed at least 250 nanoseconds after the address is placed on 'bus out'. Once 'address out' and 'select out' and 'hold out' are up, 'address out' stays up until either 'select in' or 'operational in' rises, or 'status in' falls during a short-busy sequence. Except when interface disconnect is being signaled during CA selection, 'address out' cannot be up concurrently with any other out tag line.

Command Out

The 'command out' tag line from the channel to the CAs signals the selected CA in response to 'address in', 'status in', 'data in',or 'service in'. The rise of 'command out' indicates that any information on 'bus in' is no longer required to be valid. 'Command out' stays up until the fall of the signal that raised it.

During an initial selection sequence, 'command out' rising in response to the rise of 'address in' indicates to the selected CA that the channel has placed a command byte on 'bus out'.

During 'interface disconnect', 'address out' may be up concurrently with 'command out', but normally no other outbound tag line is up.

'Command out' up in response to 'data in' or 'service in' always means STOP. 'Command out' in response to 'status in' means STACK.

When 'command out' is raised to indicate PROCEED, STACK or STOP, 'bus out' has all zeros but not necessarily correct parity. 'Bus out' should not be parity checked under these conditions.

Data In

'Data in' is a tag line to the channel from attached CAs. It is used to signal the channel that the selected CA requires the transmission of a byte of information. The information depends on the I/O operation in progress. The channel responds to the rise of 'data in' by raising either 'data out' or 'command out'.

During execution of an I/O operation specified by READ or SENSE, the CA places a data byte on 'bus in' and raises 'data in'. During execution of a WRITE command, the CA raises 'data in' and the channel places a byte on 'bus out'. When 'data in' is alternated with 'service in', 'data in' may rise when 'service out' is raised in response to 'service in'. However, 'data in' is not considered valid until 'service in' is dropped. Similarly, 'service in' may rise when 'data out' is dropped in response to 'data in'; however, 'service in' is not considered valid until 'data in' has dropped.

The conditions that apply to 'service in' concerning overrun also apply to 'data in'.

Data Out

'Data out' is a tag line from the channel to attached CAs, and is used in response to the rise of 'data in'. During READ or SENSE operations, the rise of 'data out' indicates that the channel has accepted the information on 'bus in'. During execution of a WRITE operation, the rise of 'data out' indicates that the channel has placed on 'bus out' the data requested by 'data in'.

When 'data out' is sent in response to 'data in' during execution of an I/O operation specified by a READ or SENSE command, 'data out' rises after the channel accepts the information on 'bus in'. In these cases, the rise of 'data out' indicates that the information is no longer required to be valid on 'bus in', and is not associated with any information on 'bus out'. When 'data out' is sent in response to 'data in' during execution of a WRITE command, the rise of 'data out' indicates that the channel has provided the information requested on 'bus out'.

'Data out' remains up until the fall of 'data in'.

Operational In

Attached CAs have a line to the channel to signal that the CA has been selected. 'Operational in' stays up for the duration of the selection. The selected CA is identified by the address byte on 'bus in' when 'address in' was raised.

The rise of 'operational in' indicates that a CA is selected and communicating actively with the channel. 'Operational in' rises only when incoming 'select out' to the CA is up and the outgoing 'select out' is down. That is, the CA traps 'select out', and does not propagate it. The line 'operational in' can drop only after 'select out' drops.

When 'operational in' is raised for a particular signal sequence, it stays up until all the required information, data or status, is transmitted between the CA and the channel. If 'select out' drops, 'operational in' drops after the rise of the outbound tag associated with the transfer of the last byte of information.

Operational Out

'Operational Out' is a tag line from the channel to attached CAs, and is used for interlocking purposes. Except for 'suppress out', all lines from the channel are significant only when 'operational out' is up. Whenever 'operational out' is down, all inbound lines from the CA drop, and any operation currently in progress is reset. Under these conditions, all CA-generated interface signals are dropped within 1.5 microseconds after the fall of 'operational out' at the CA.

Request In

'Request in' is a line from all CAs to the channel. When up, this line indicates that the CA requests service and a selection sequence.

The line is dropped when:

- 'Operational in' rises, unless additional CA-initiated sequences are required
- The CA is no longer ready to present data or status information
- The selection requirement is satisfied by another path

'Request in' never falls later than 250 nanoseconds after the fall of 'operational in' if the sequence satisfies the service requirements of the CA.

'Request in' does not remain up when 'suppress out' is up if the request is for suppressible status. When the CA is requesting a selection sequence to present suppressible status, 'request in' falls at the CA within 1.5 microseconds after the rise of 'suppress out' at the CA.

'Request in' can be signaled by more than one CA at a time.

Channel Tag Line Definitions (Part 2 of 2)

Select Out, Select In, and Hold Out

CA selection is controlled by 'select out', 'select in' and 'hold out'. 'Select out' and 'select in' form a loop from the channel through each CA to the cable terminator block ('select out') and again through each CA back to the channel ('select in'). CA selection circuitry may be attached to either 'select out' or 'select in', though it is assumed to be always connected to 'select out'.

If selection is not required, the selection signal is propagated in turn by each CA to the next CA on the line.

When an operation is being initiated by the channel, 'select out' is raised not less than 400 nanoseconds after the rise of 'address out', which indicates which CA is being selected.

The channel keeps 'select out' up until 'select in' rises, or until 'address in' and 'operational in' rise, or until 'status in' rises. When 'select in' rises, 'select out' drops and does not rise again until after 'select in' falls.

A CA becomes selected only when it raises its 'operational in' tag line. After the drop of 'select out', the CA keeps 'operational in' up until the current signal sequence is completed. If a CA raises 'operational in', it suppresses the propagation of 'select out' to the next CA. If the CA does not require selection, it propagates 'select out' to the next CA on the line within 1.8 microseconds.

When 'status in' rises in response to 'select out' in the short-busy sequence, 'select out' drops and does not rise again until after 'address out' has dropped.

To prevent overlapping of interface sequences, either:

1. 'Select out' is not raised until all inbound signals for the preceding sequence are in the down state, or
2. The In tags are considered valid until 1.5 microseconds after the fall of 'operational in' for the preceding sequence.

'Hold out' is a line from the channel to attached control units and is used with 'select out' to synchronize CA selection. 'Hold out' is also used to minimize the propagation of the fall of 'select out' by purging the signal from its path. Once 'hold out' drops, it cannot rise again until 4 microseconds later.

Service In

'Service in' is a tag line from attached CAs to the channel and is used to signal to the channel when the selected CA is ready to send or receive a byte of information. The information associated with 'service in' depends on the operation. The channel responds to the rise of 'service in' by raising either 'service out' or 'command out'.

Service Out

'Service out' is a tag line from the channel to attached CAs and is raised to signal the selected CA when 'service in' or 'status in' has been recognized. A signal on 'service out' indicates to the selected CA that the channel has accepted the information on 'bus in' or has provided on 'bus out' the data requested by 'service in'.

Status In

'Status in' is a tag line from the attached CA to the channel to signal that the selected CA has placed a byte of status information on 'bus in'. The status byte has a fixed format and contains bits describing the current status of the control unit. The channel responds by raising either 'service out' or 'command out', or by dropping 'select out' in the case of a short-busy sequence.

'Status in' does not rise concurrently with any other inbound tag line. It stays up until the rise of another outbound tag line, or until 'select out' drops in the case of a short-busy sequence.

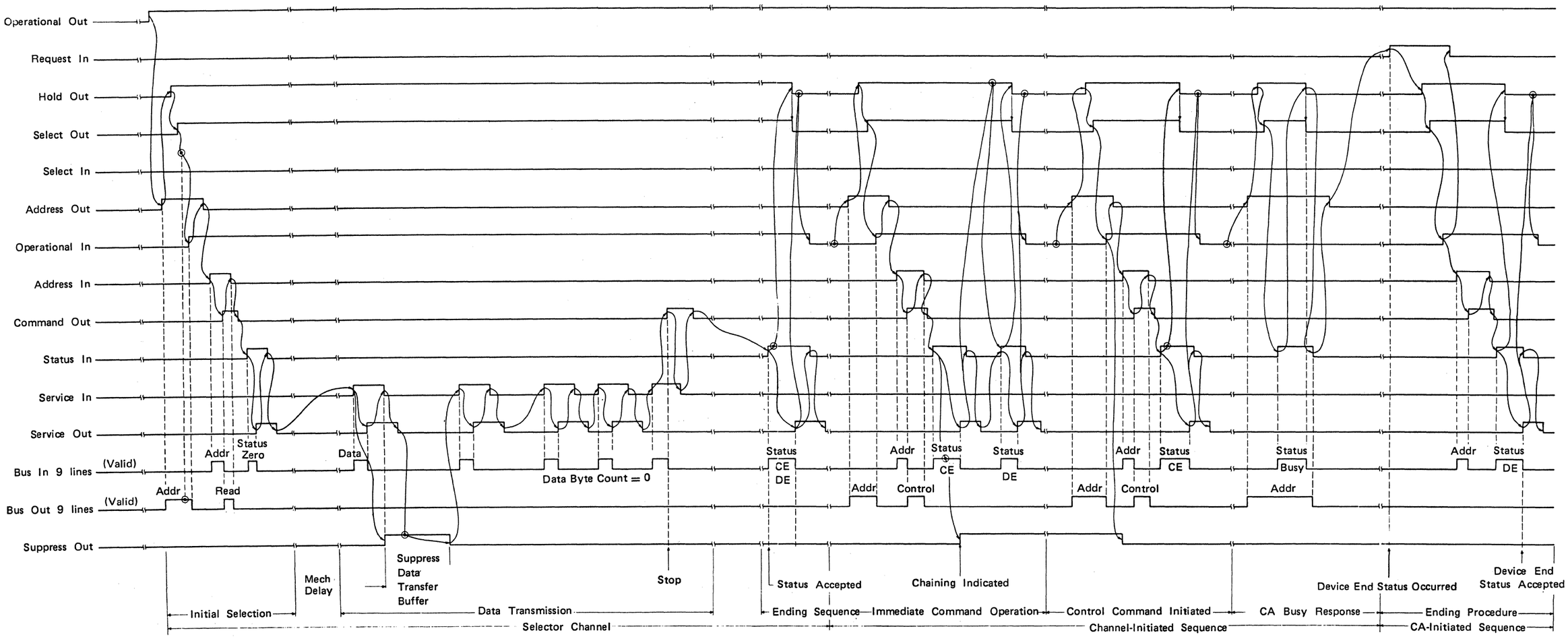
'Status in' must fall so that the corresponding out tag may fall. In the case of a short-busy sequence, that status information stays valid until 'select out' drops.

Suppress Out

'Suppress out' is a line from the channel to attached CAs: it may rise or fall at any time. This line is used both alone and with the out tag lines to provide the following special functions:

- Suppress data
- Suppress status
- Command chaining
- Selective reset

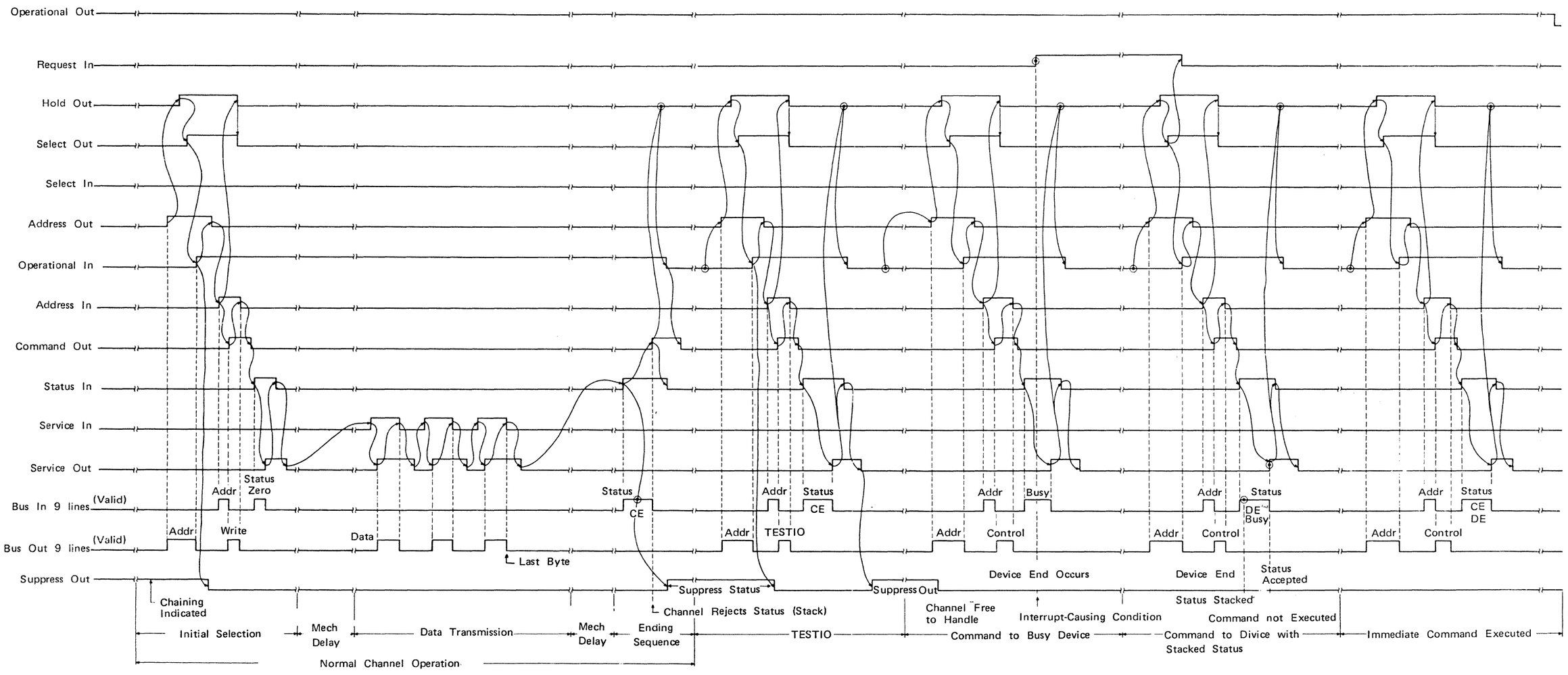
Selector Channel



Note : The Circled dot ⊙ is used in the sequence chart to indicate the checking of that signal's level before proceeding.

Legend
CE = Channel end
DE = Device end

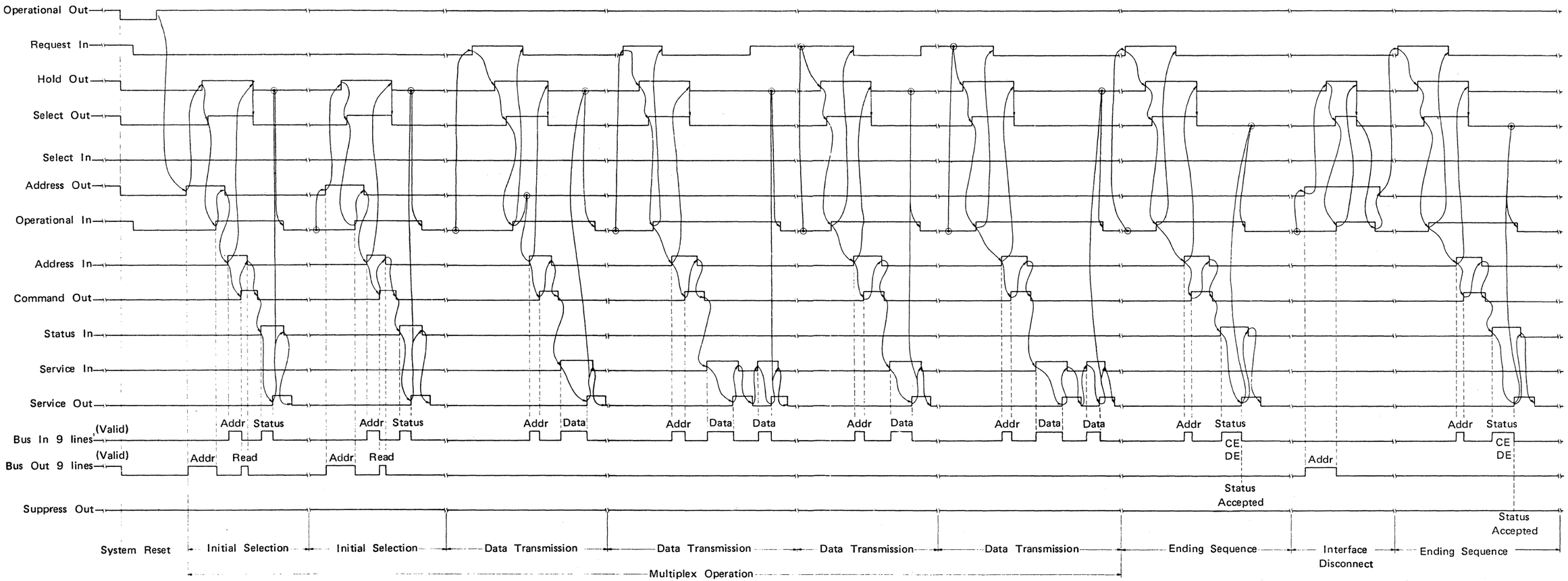
CU Forced Burst Mode



Note : The circled dot ⊙ is used in the sequence chart to indicate the checking of that signal's level before proceeding.

Legend
CE = Channel end
DE = Device end

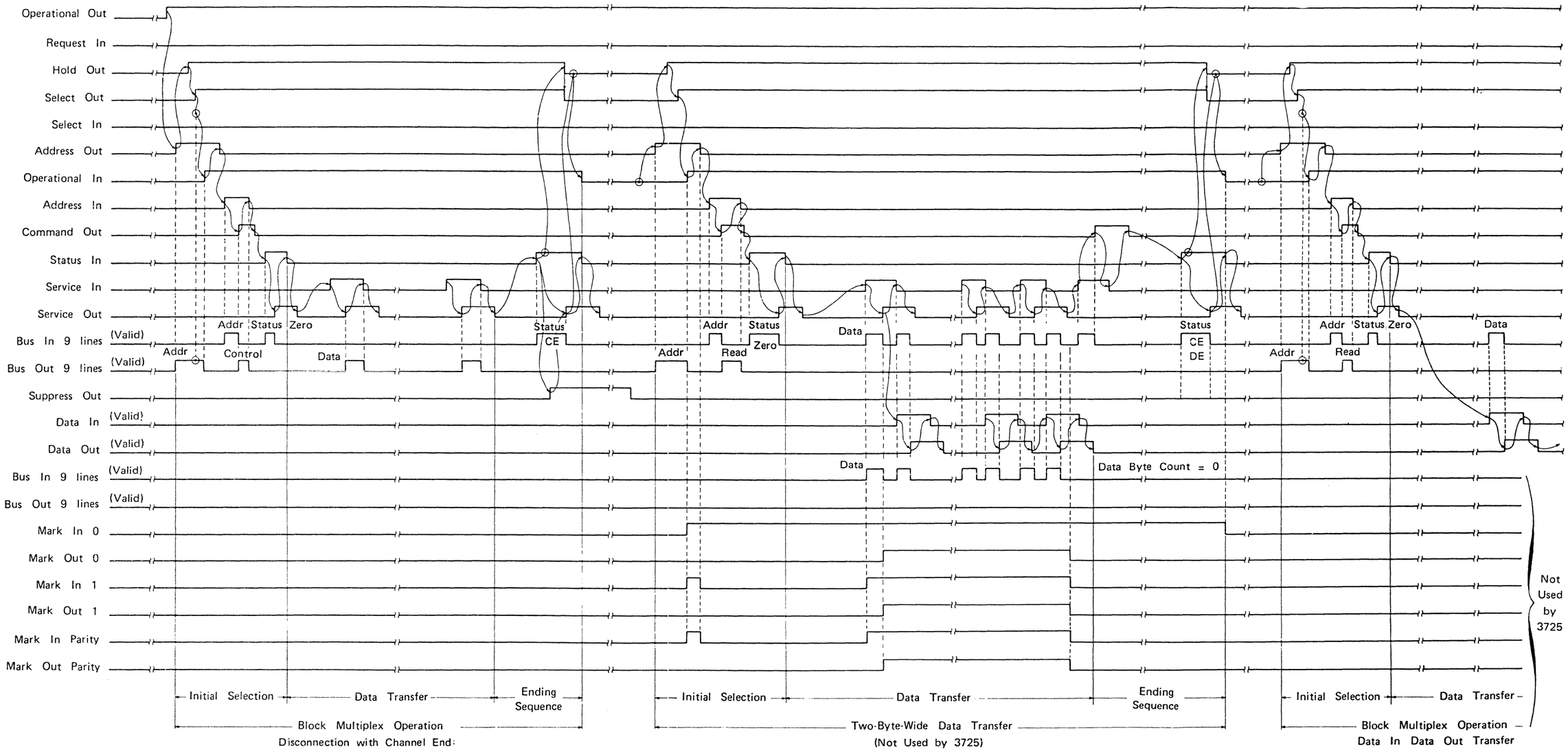
Multiplexer Channel



Note : The circled dot ⊙ is used in the sequence chart to indicate the checking of that signal's level before proceeding.

Legend
CE = Channel end
DE = Device end

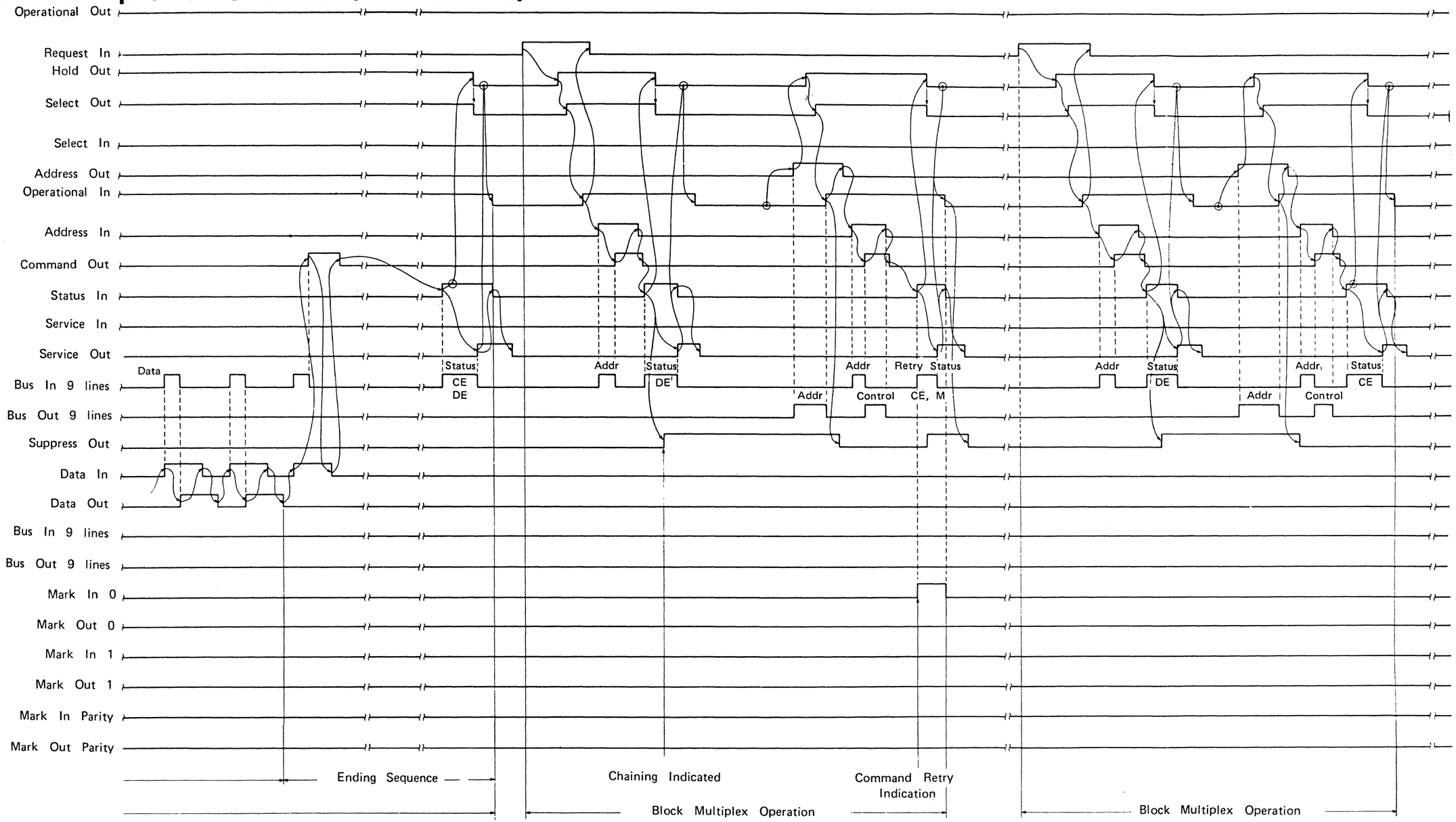
Block-Multiplexer Channel (Part 1 of 2)



Note : The circled dot ⊙ is used in tha sequence chart to indicate the checking of that signal's level before proceeding.

Legend
CE = Channel end
DE = Device end

Block-Multiplexer Channel (Part 2 of 2)



Note : The circled dot ⊙ is used in the sequence chart to indicate the checking of that signal's level before proceeding.

Legend
CE = Channel end
DE = Device end
M = Modifier

Chapter 12. Channel Adapter
Section 2. Troubleshooting Guidelines

DC Voltages and Tolerances at Board Pin Level

Vdc	Vmin	Vmax	Ripple (max)
−12.0	−10.92	−13.20	0.45V p−p
−8.5	−7.73	−9.35	0.25V p−p
−5.0	−4.55	−5.50	0.15V p−p
−4.3	−4.19	−4.48	0.07V p−p
−1.5	−1.48	−1.56	0.03V p−p
+5.0 (Note 1)	+4.55	+5.50	0.20V p−p
+5.0	+4.75	+5.25	0.13V p−p
+8.5	+7.73	+9.35	0.35V p−p
+12.0	+10.92	+13.20	0.40V p−p
+12.0 (Note 2)	+11.40	+13.20	0.40V p−p
+24.0	+21.00	+27.60	0.30V p−p

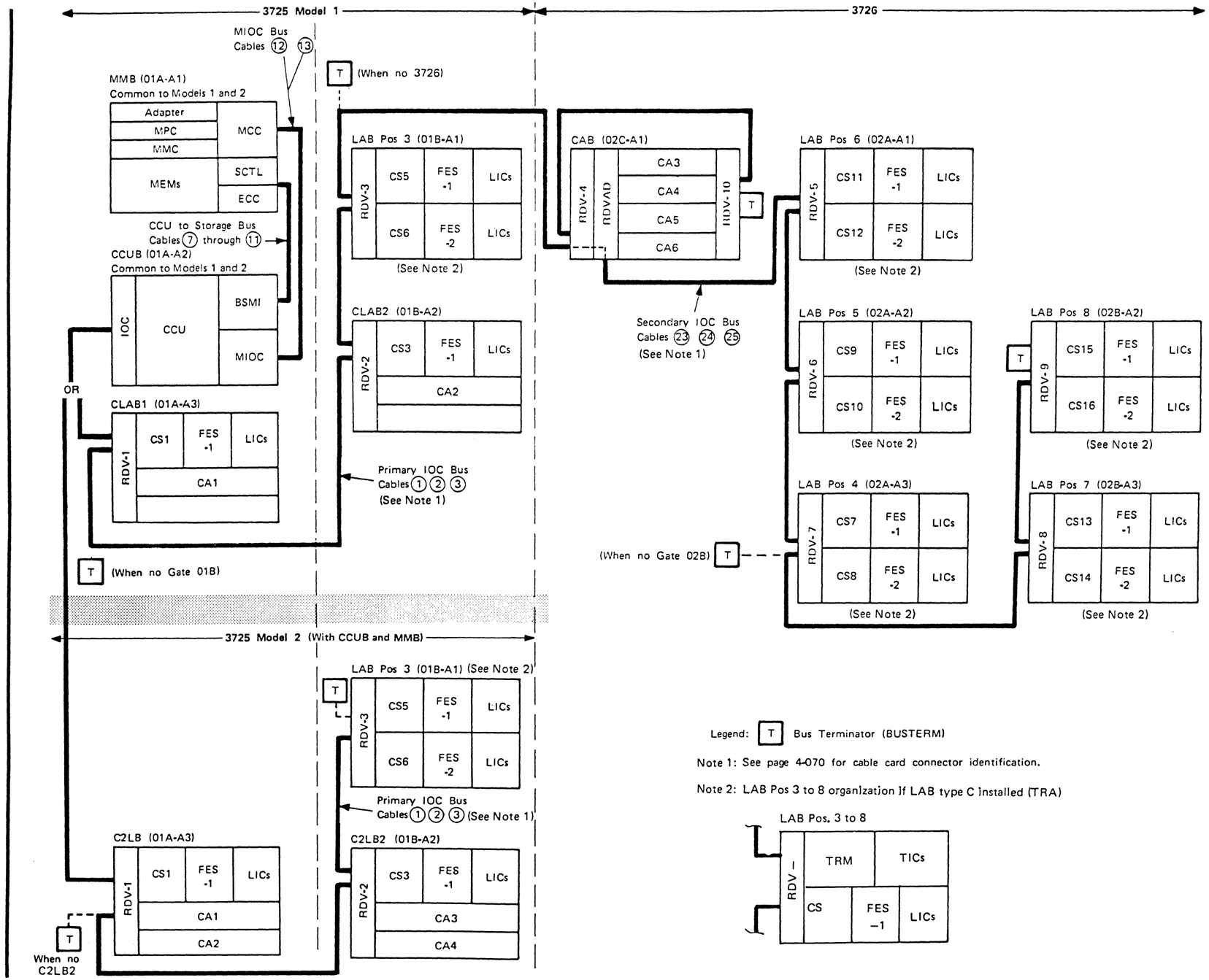
Notes:

1. 02−PS7 only
2. 01−PS4 only

Channel Adapter Troubleshooting Techniques

The channel adapter troubleshooting is performed using the following techniques:

- Checking the channel adapter enabling/disabling operation (see page 12-800)
- Checking the channel adapter autoselection mechanism (see page 12-803)
- Running a scoping routine to check the autoselection mechanism (see page 12-807)
- Checking the cycle steal mechanism (see page 12-809)
- Running a scoping routine to check the cycle steal mechanism (see page 12-830)



Legend: T Bus Terminator (BUSTERM)

Note 1: See page 4-070 for cable card connector identification.

Note 2: LAB Pos 3 to 8 organization if LAB type C installed (TRA)

Enabling/Disabling Checks (Part 1 of 3)

CLOCKING

Clock signals driving the channel adapters are 5 MHz and 100 ms. The CCIN card receives 5 MHz on pin M07 and the CHIN card receives 100 ms on pin P07. (See page 5-050 for clock scoping details.)

CHANNEL ENABLING/DISABLING

Note: The switch(es) on the control panel do not enable a channel adapter when in the 'Enable' position unless an IOH/IOHI Output X'7' instruction with byte 1, bit 4 set on has been issued to the channel adapter that is to be enabled.

The channel adapter enable switch circuit for each host interface covers a path from the control panel circuit board to the CHIN card in that channel adapter. Using the cabling between boards chart on page 4-070 as a general reference, and the detailed routing drawing on the next page, the channel adapter enable circuit trace and measure from the control panel circuit board through to the CHIN card input pin.

The channel adapter switches are part of the control panel circuit board. The common points of the switches connect to the dc ground plane of the board at 01A-A1 via two parallel paths as shown on the detailed routing figure. Each switch feeds a separate latch which is also located on the control panel circuit board. The purpose of this latch is to eliminate switch bounce. When the switch is down (to disable the interface), the output of the latch is at +3.8V; when the switch is up (to enable the interface), the output of the latch should be between 0 and 0.5V. Each latch is routed to the CHIN card of the corresponding channel adapter as shown on the detailed routing figure.

Note: If the line is open circuit at any point, the level at the CHIN card input will float to about +1V to +2.2V as shown.

All Channel Adapters Disabled Lamp

This lamp is a LED with a driver which is part of the control panel circuit card. The input to the LED driver is a large 'DOT OR' circuit. A line from each channel adapter feeds the 'DOT OR' as shown on the detailed routing drawing. When an interface is enabled for any channel adapter, the CHIN card for that channel adapter sends a ground level to the 'DOT OR', turning off the LED.

If no channel adapter is enabled, the 'DOT OR' line should rise to about +1.7V, turning on the LED.

Hexadecimal Display Codes X'EE1' through XEEA'

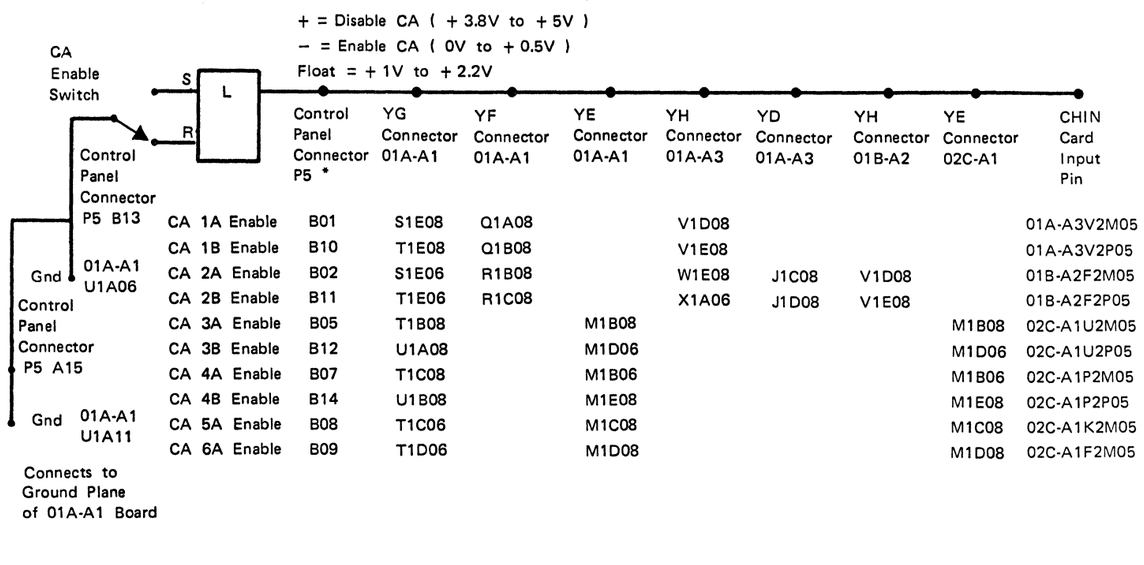
Immediately after a power on reset, the MOSS code (in ROS) makes a check of all the installed channel adapters to ensure that none of them has its 'Enabled' line active. If an 'Enabled' line is found to be active, an error code is displayed on the hexadecimal indicators. The correspondence between the error code and the channel interface is as follows:

Interface A		Interface B (TPS)	
Error Code	CA	Error Code	CA
X'EE1'	1A	X'EE7'	1B
X'EE2'	2A	X'EE8'	2B
X'EE3'	3A	X'EE9'	3B
X'EE4'	4A	X'EEA'	4B
X'EE5'	5A	–	–
X'EE6'	6A	–	–

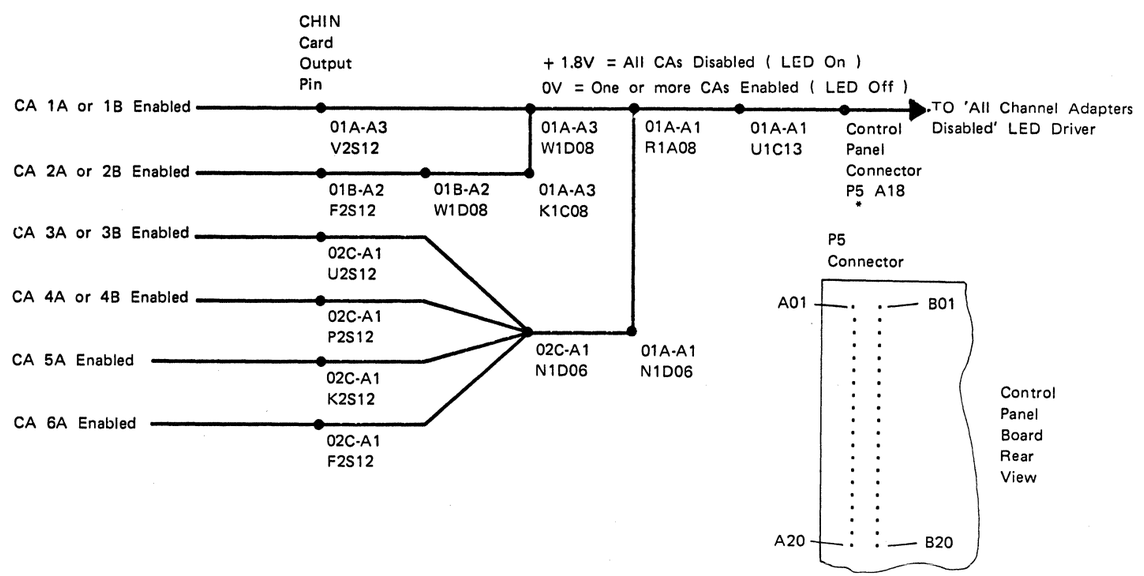
These displays could be caused by the CHIN or CADR cards. A grounded or open circuit 'Channel Adapter Enabled' signal(s) from each channel adapter to the MMB board may also be suspected; refer to the detailed routing diagram on the next page.

3725/3726

Channel Adapter Enable Switch Circuit Routing



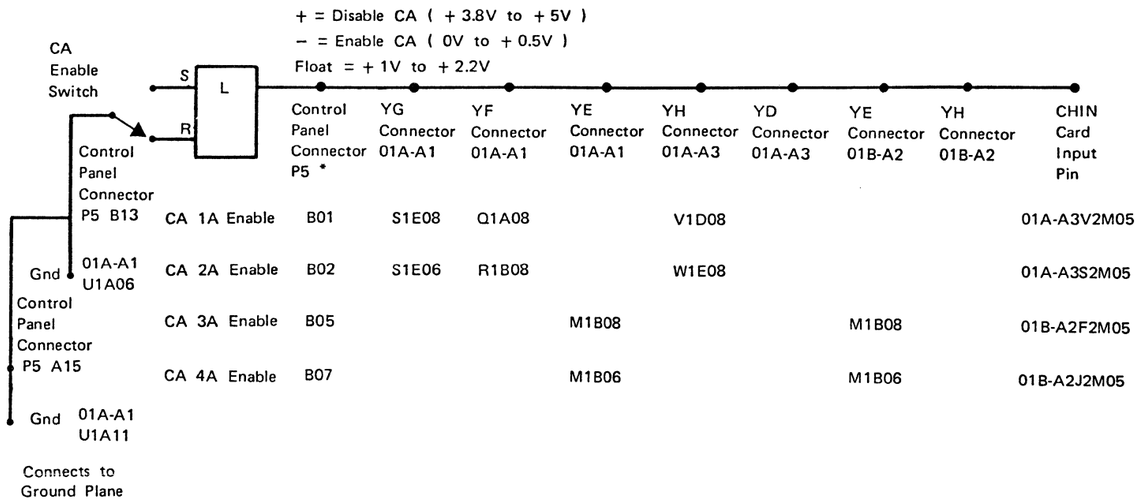
All Channel Adapters Disabled 'Dot Or' Circuit Routing



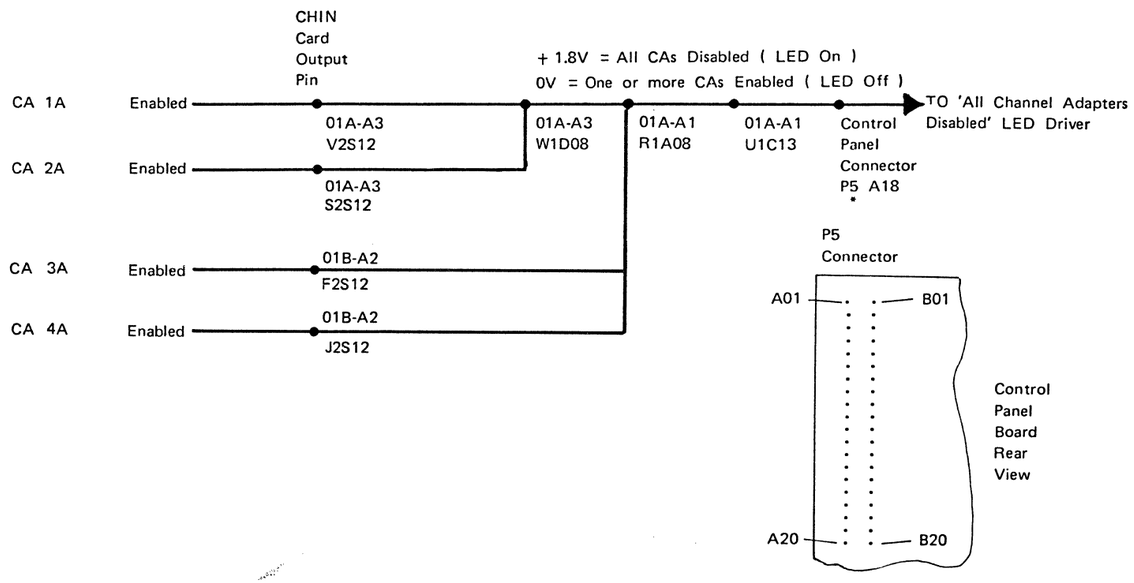
Enabling/Disabling Checks (Part 2 of 3)

3725 MODEL 2

Channel Adapter Enable Switch Circuit Routing



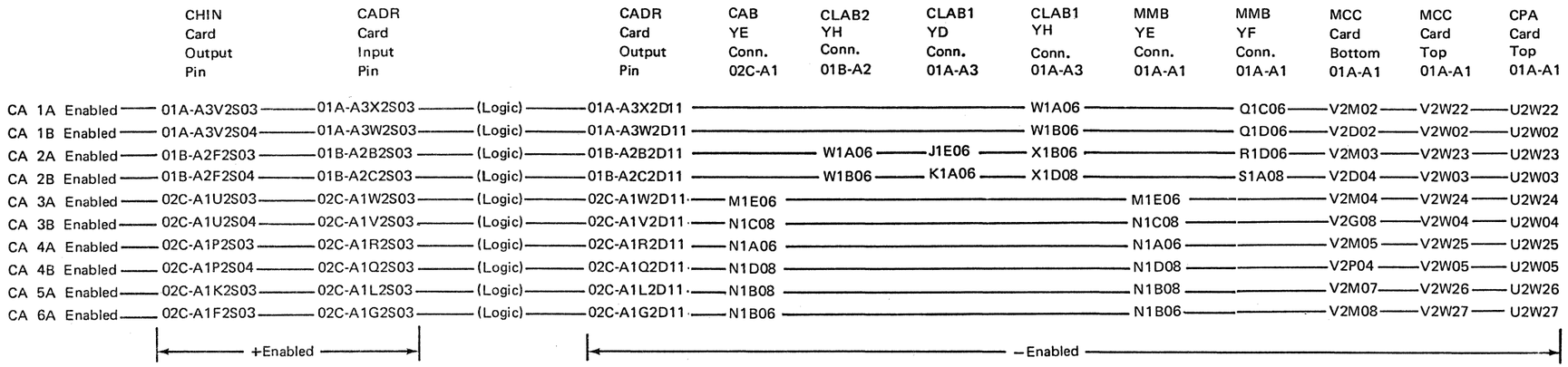
All Channel Adapters Disabled 'DOT OR' Circuit Routing



Enabling/Disabling Checks (Part 3 of 3)

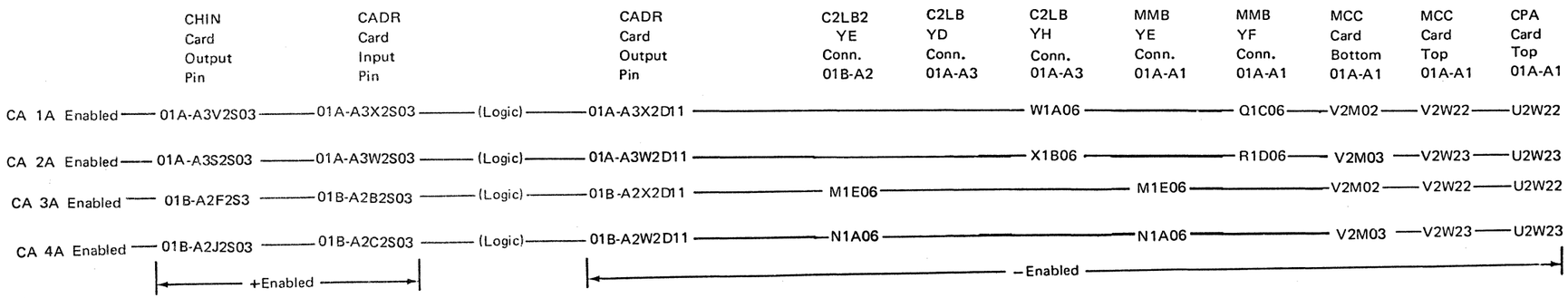
CHANNEL ADAPTER ENABLED SIGNAL ROUTING (CA BOARDS TO MMB BOARD)

3725/3726



Legend :
+ = + 4.5 V to + 5 V
– = 0 V to + 0.5 V
Float = + 1 V to + 2.2 V

3725 Model 2



Legend :
+ = + 4.5 V to + 5 V
– = 0 V to + 0.5 V
Float = + 1 V to + 2.2 V

Autoselection Troubleshooting

CA AUTOSELECTION TROUBLESHOOTING

See page 12-065 for a general description of the channel adapter autoselection mechanism.

The following symptoms are an indication that the autoselection mechanism has failed:

- The channel adapter diagnostic routine LA20 fails.
- An IOC bus error occurs and the lagging address register (Input X'74') indicates that an IOH/IOHI instruction failed while performing a channel adapter Input X'F'.
- An IOH/IOHI Input X'F' instruction was done because of a level 3 interrupt, but the channel adapter selected does not have a level 3 interrupt.

When the FRUs indicated by the maintenance procedure have been replaced, the autoselect interconnect cable(s) may be suspected as causing the problem. The following information is available in this section for isolating autoselection problems:

- Autoselection signal description
- Channel adapter autoselect signal routing diagram
- Autoselect timing chart
- Manual intervention scoping routine description

AUTOSELECTION SIGNAL DESCRIPTION

A detailed description of each of the autoselect signals follows:

Priority Bus Bits 4 through 7 to Interface

These four signals indicate the level 3 interrupt priority of all channel adapters. They are individually dotted by all channel adapters so that each CA can determine whether its own level 3 priority is greater than, equal to, or less than that of all the other channel adapters. These signals are driven and received by each channel adapter. When a level 3 interrupt occurs in a channel adapter, that CA activates the applicable priority bus bit (if any) as long as the autoselect signal 'Hold' is inactive. The priority bus bits are defined in the table below:

Note: If a channel adapter level 3 is caused by any interrupt other than those defined by the priority bus bits as above, that channel adapter does not activate any of the priority bus bits. As shown in the table, this type of level 3 interrupt has the lowest priority of all.

Priority bus bit	Priority	Type of operation
4	1 (high)	Priority outbound data transfer
5	2	Outbound data transfer
6	3	Initial selection
7	4	Inbound data transfer
none	5 (low)	All others

Sample Out to Interface
Sample In
Sample Out Wrap Dot
CA Installed Sent
CA Installed Received

These signals are described as a group because of their logical interaction.

The channel adapter that was selected before the execution of the channel adapter Input X'F' IOH/IOHI instruction was started generates the 'Sample Out to Interface' signal which propagates to the next CA as 'Sample In'. The 'Sample Out to Interface' signal is generated even if the channel adapter originating this signal has the highest priority. If the CA receiving the 'Sample In' signal has a priority equal to or greater than that of all the other CAs (as determined by comparing its own priority with the priority bus bits) it does not propagate 'Sample Out to Interface' to the next channel adapter. Instead, it becomes selected itself, and resets the selected latches in the other channel adapters by generating the 'CA Sample Trap to Interface' signal.

The 'CA Sample Trap to Interface' signal remains active until the level 3 interrupt in the CA selected by the autoselect mechanism has been reset, and inhibits a new Input X'F' from initiating another autoselect sequence as long as it is active.

Note: If the last physically installed channel adapter receives the 'Sample In' signal, but does not have greater or equal priority, the last CA also generates the 'Sample Out Wrap Dot' in addition to the 'Sample Out to Interface' signal, since this signal has no load when generated by the last CA. The last CA is the highest numbered CA.

The 'Sample Out Wrap Dot' signal is dotted at all channel adapters, and arrives at CA#1 as 'Sample In'. A channel adapter determines that it is the last physically installed CA by examining the 'CA Installed Received' signal which is generated by a higher numbered channel adapter as the 'CA Installed Sent' signal. When the 'CA Installed Received' signal is inactive, it indicates that the CA is the last one in the chain; for example, if CA#5 is not installed, CA#4 does not receive the 'CA Installed Received' signal, and thus knows that it is the last in the chain.

Hold to Interface

This signal inhibits all channel adapters from changing the priority bus bits until the CA with the highest priority has become selected. The 'Hold to Interface' signal is generated by the channel adapter that was selected before the IOH/IOHI Input X'F' instruction execution was started.

The 'Hold to Interface' signal is reset as follows:

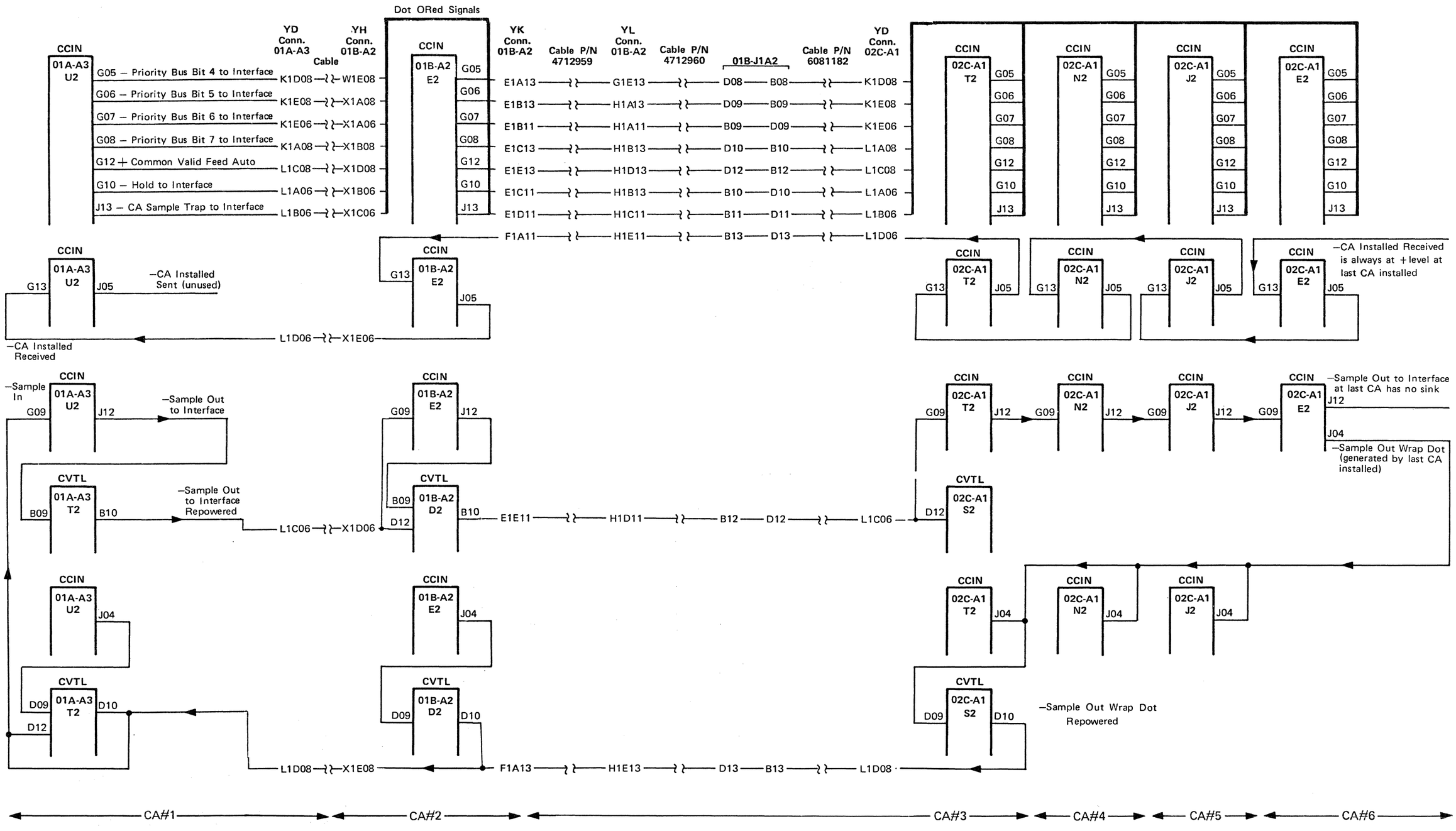
- When the CA that generated the 'Hold to Interface' signal receives 'Sample In', the 'CA Sample Trap' signal was not generated by any of the other channel adapters, and the CA remained selected throughout the autoselection sequence. This means that the already selected CA has the highest level 3 priority, or that no CA had a level 3 interrupt pending when the Input X'F' was executed.
- When the channel adapter that generated the 'Hold to Interface' signal receives the 'CA Sample Trap to Interface' signal, and is no longer selected. This means that the channel adapter with the highest level 3 priority has selected.

Common Valid Feed Auto

The 'Common Valid Feed Auto' line is a dotted signal that indicates that all channel adapters have decoded the Input X'F' instruction. The CA that was selected before the Input X'F' instruction was started uses the line to generate the valid tag on the IOC bus in response to the TA tag during the Input X'F' instruction. If any of the channel adapters have not recognized the input, 'Valid Tag' is not raised, and an IOC bus error occurs. The CA that becomes selected during the Input X'F' instruction uses the 'Common Valid Feed Auto' signal to generate the 'Valid Tag' response to the TD tag.

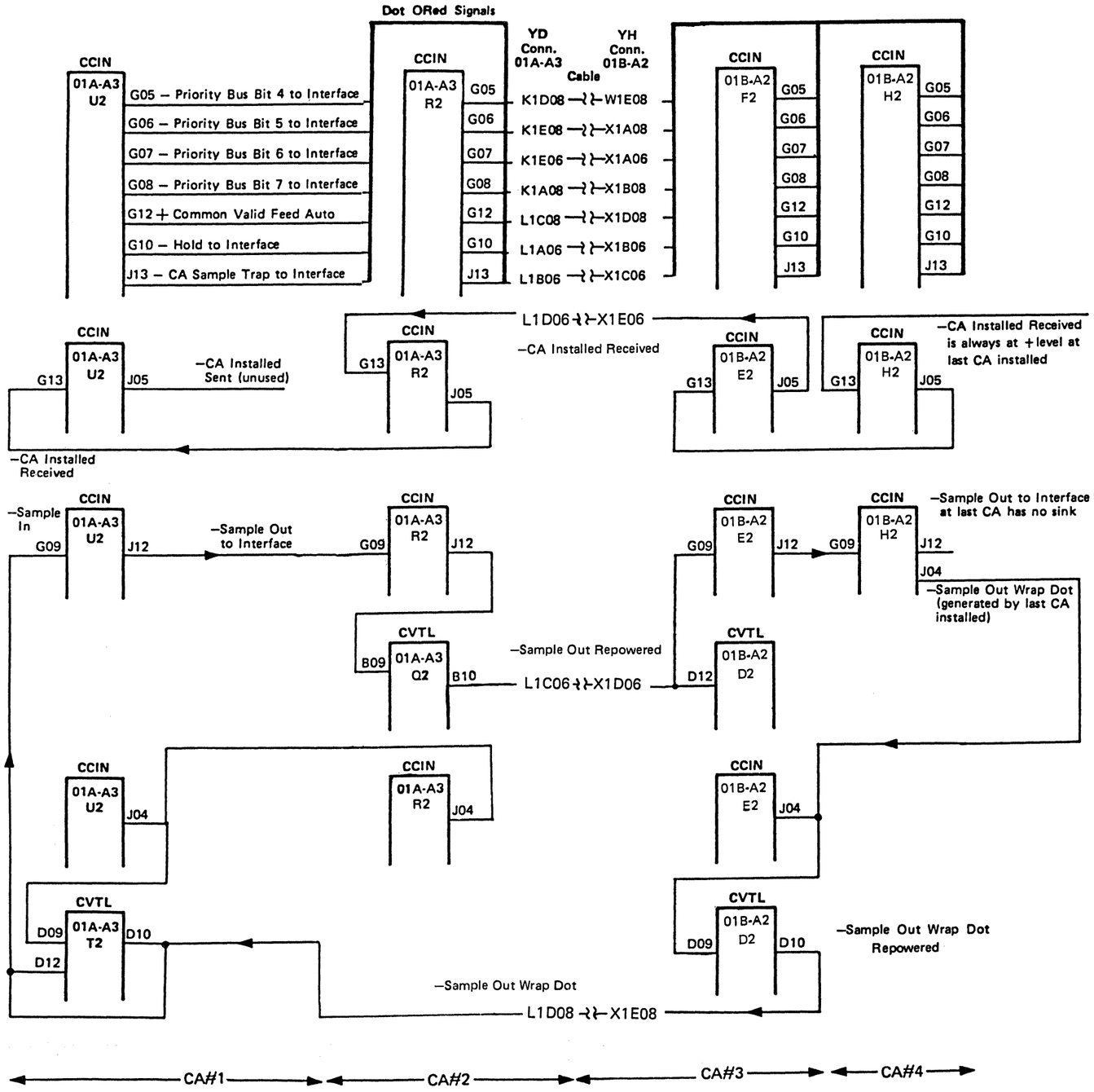
Autoselection Signal Routing (Part 1 of 2)

CHANNEL ADAPTER AUTOSELECT SIGNAL ROUTING (3725/3726)



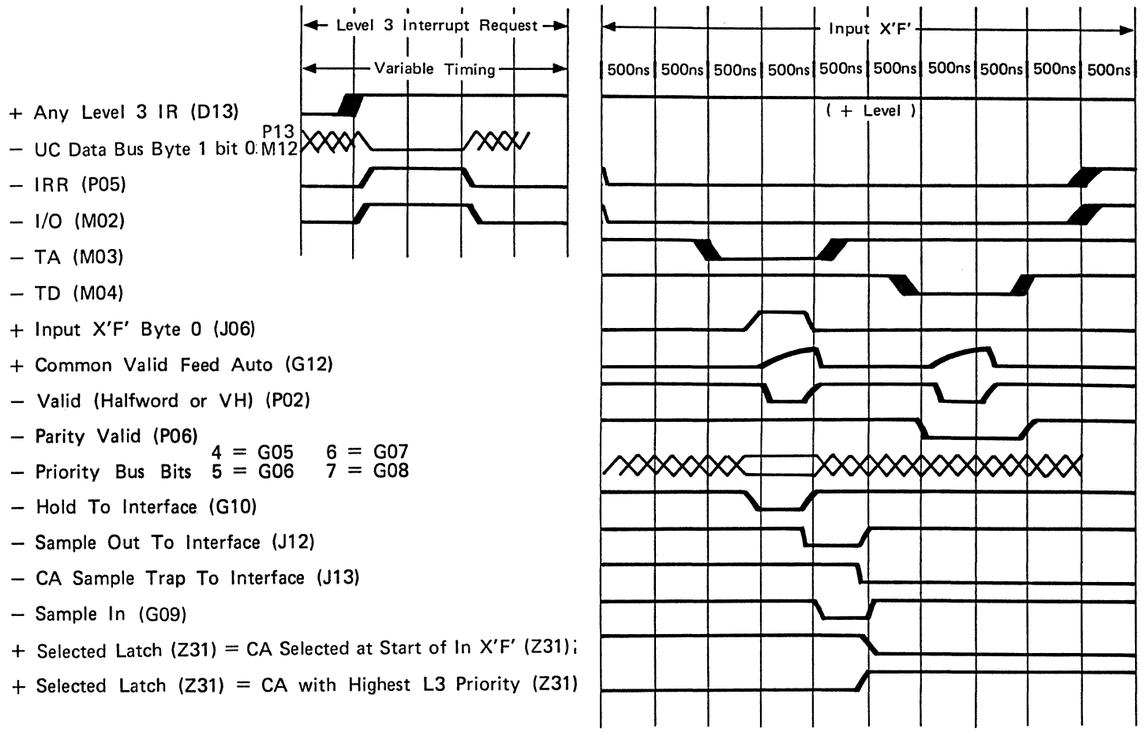
Autoselection Signal Routing (Part 2 of 2)

CHANNEL ADAPTER AUTOSELECT SIGNAL ROUTING (3725 MODEL 2)



Autoselection Timing

Pins Listed are on the CCIN card



Notes:

1. If an IOC bus check occurs, the IOC bus tags stay at the signal level present when the error occurred, and remain there until the IOC bus check is reset.

2. The 'Sample Out to Interface' and 'Sample In' signals are not present at each channel adapter (see under "Autoselection Signal Description" on page 12-803).

3. IOC bus byte 1, bit 0, indicates that a channel adapter level 3 is present on the bus when I/O is not active. Scoping pins P13 and M12 are on the channel adapter side of the RDV card. Byte 1, bit 0 is not active unless a level 3 interrupt is being raised by a channel adapter on the board being scoped.

4. The 'Sample Out Wrap Dot' signal is not shown on the timing diagram; it is the same as 'Sample Out to Interface'.

5. The channel adapter with the level 3 interrupt will continue to present its interrupt on byte 1, bit 0 of the IOC bus (when I/O is inactive) until the channel adapter's level 3 interrupt is reset.

Scoping Levels

+ = +4.5 to +5V

– = 0 to +0.5V

A line that is floating will be between +1 and +2.2V.

Autoselection, Scoping Routine (Part 1 of 2)

CHANNEL ADAPTER SCOPING ROUTINE FOR AUTO-SELECT MECHANISM

The purpose of this scoping routine is to allow you to troubleshoot problems related to the channel adapter autoselection mechanism. The routine sets up a channel adapter autoselection test as follows:

1. The first channel adapter to be tested is specifically selected with an IOH Output X'7' instruction and autoselection is enabled.
2. A level 3 interrupt request is generated in the other channel adapter.
3. An IOH Input X'F' instruction is performed. This triggers the autoselection mechanism to select the channel adapter with the level 3 request and deselect the channel adapter previously selected.
4. If the looping option is selected, steps 2 and 3 are repeated, which results in the two channel adapters being alternately selected.

Running the Routine

The routine may be run by calling manual intervention routine MB01 with the 'CA' option (this is the same manual intervention routine that is used for the IOC bus).

When the message 'TO TEST CA AUTOSELECT TYPE: CA THEN PRESS SEND' is displayed, reply by typing 'RCA' (the other scoping routines are selected by simply pressing SEND). When the message 'ENTER CA NBRS FOR AUTOSELECT THEN PRESS SEND' is displayed, the parameters should be entered in the following format:

'Rooxxyy

where:

oo = the selected option

xx = the first channel adapter to be tested

yy = the second channel adapter to be tested

Option

The option 'oo' may be one of the following values:

01 The requested function is executed once only. The '01' option may be repeated as often as required.

02 The requested function is executed and loops until an error is found. The error is reported once only via an RAC 675 or 676. If you wish to continue, type 'G' (for Go); the routine then loops indefinitely until the 'ATTN' key is pressed.

03 This option is identical to option 02 above, except that if an error is found, it is not reported.

XX and YY

The remaining parameters are 'XX' and 'YY', where XX is the first channel adapter to be tested, and YY is the second. The parameters XX and YY should be entered in the form X'0n' where n is the channel adapter number (CA1 = X'01').

Notes:

1. To test only one channel adapter, YY should be X'00'.
2. If XX is not in the range X'01' through X'06', or if YY is not in the range X'00' through X'06', the following message is displayed: 'INVALID CA NBR PRESS SEND TO RETRY'.
3. If XX is the same as YY the following message is displayed: 'INVALID REQUEST PRESS SEND TO RETRY'.

Autoselection, Scoping Routine (Part 2 of 2)

Error Reporting

The following RAC-ERC codes are displayed when an error is found by the scoping routine:

RAC	ERC	Meaning
675	7FFA	The request was for one channel adapter, but the adapter found by the autoselect mechanism was not the one expected. Action: type G to proceed.
675	7FFB	The request was for two channel adapters, but the first adapter found by the autoselect mechanism was not the one expected. Action: type G to proceed.
675	7FFC	The request was for two channel adapters. The first one found by the autoselect mechanism is correct, but the second one is not. Action: type G to proceed.
675	7FFD	A level 1 interrupt occurred while trying to issue a PIO. The error bit gives the reason for the level 1 interrupt. Action: type G to proceed.
676	7FE1	A permanent level 3 interrupt is present. It is impossible to exit this level even when the channel adapters have been reset by the program via an Output X'7' instruction with byte 0, bit 7 set to 1. Action: restart the routine.
676	7FE2	A level 1 interrupt occurred while executing a PIO in level 3. Action: type G to proceed.

The error bit field displayed is the contents of In X'76'. This register latches the error conditions found during PIO execution. The following combinations may occur:

Error field	Suspected IOC Line*	Meaning
X'0400'	Data bus bits	IOC Bus parity check
X'0800'	VH	I/O tag is off
X'1800'	IRR	I/O tag raised on a new IOH
X'2800'	IRR, VH, EOC	No response to TA
X'4800'	IRR	VH did not fall after TD dropped
X'5800'	IRR, VH, EOC, VB, M	No response to TD for PIO read
X'6800'	IRR, VH, EOC, VB, M	No response to TD for PIO read
X'F800'	VH	I/O tag is off, VH must rise (PIO end after a data exchange)
X'xxFF'		Autoselect failure. X'xx' is the channel adapter address (00-05)

Where:

Data bus bits = bytes 0 and 1, bits 0 through 7 and parity
EOC = end of chain
IRR = interrupt request removed
M = modifier
VB = valid byte
VH = valid halfword

CA Cycle Steal Troubleshooting

The symptoms of channel adapter cycle steal failures are:

- Channel adapter diagnostic routine LA20 fails.
- IOC Bus errors (register X'7E' byte 0, bit 7 on and register X'76') and register X'75' byte 0, bit 0 off indicating that the channel adapter was cycle stealing.
- BER type 10 with ID 14, 16, 91 or 9A, or BER type 14 with ID 92.

When the FRUs indcated by the maintenance procedures have been replaced, the signal paths between the channel adapter and the IOC bus may be suspected of causing the problem. Some specific suspect items are the top card connectors, the bottom card connectors, and the on board nets. The following information is available for isolating channel adapter cycle steal problems.

- CA cycle steal operations description
- Diagrams of the IOC bus lines used during cycle steal operations with the channel adapter
- Channel adapter cycle steal timing chart
- Manual intervention scoping routines description
- Pin/Net listings in the YZ pages

CA CYCLE STEAL OPERATIONS (AIO)

Note: The RDV card must be enabled by an IOH/IOHI instruction before the channel adapter can operate with the IOC bus.

Before a cycle steal operation can be started for a given channel adapter, that CA's pointer register in the CCU must be loaded with the cycle steal data address using an Output X'3x' instruction. Also, registers X'C' and X'2' within this CA must be loaded with the byte count and the controls using IOH/IOHI instructions from the CCU. The last instruction executed should be an IOH/IOHI Output X'2' instruction to set byte 0, bit 4 off, indicating an AIO data/status operation.

The CA may now raise a cycle steal request to transfer data between the CA and the CCU storage. The CA raises the 'Cycle Steal Request' line at pin P10 of its CCIN card, which is connected to pin J11 of its corresponding RDV card. If the RDV card is not disabled, it forwards the request to the CCU as the 'Cycle Steal Low' signal on the IOC bus pin X32. This IOC bus line is a 'dor-OR' of the signals coming from all the channel adapters. When the CCU sees the 'Cycle Steal Request Low' line, it raises the 'IO Tag' line which inhibits any additional requests from trapping the cycle steal grant when it is sent out. The signal 'IO Tag' coming up also causes all adapters on the IOC bus to clear the data bus bytes 0 and 1 and respond with 'Interrupt Request Removed' (IRR) and drop 'Valid'. The CCU then honors the request by sending out 'Cycle Steal Grant Low' on the IOC bus to pin Y03 of the RDV card for channel adapter 1. The grant signal proceeds in priority order from CA#1 (highest) to CA#6 (lowest). As each RDV card (for CAs 1 and 2), or CCIN (for CAs 3 through 6) receives 'Cycle Steal Grant Low' on pin Y03 (M08 for CCIN) it checks its cycle steal request latch. If the request latch is on, the grant signal is trapped and used by that adapter. If the request latch is off, the grant signal is passed on to the next lower priority channel adapter.

When a channel adapter sees the cycle steal grant, it places its Cycle Steal Control Word (CSCW) on the IOC data bus and raises the 'Valid Tag' signal. If the parity on the data bus is good, the CCU acknowledges by dropping cycle steal grant. The CA drops the request when the grant was received, and drops valid when the grant drops.

Note: If the CCU detects bad parity on the data bus, it raises the Halt tag.

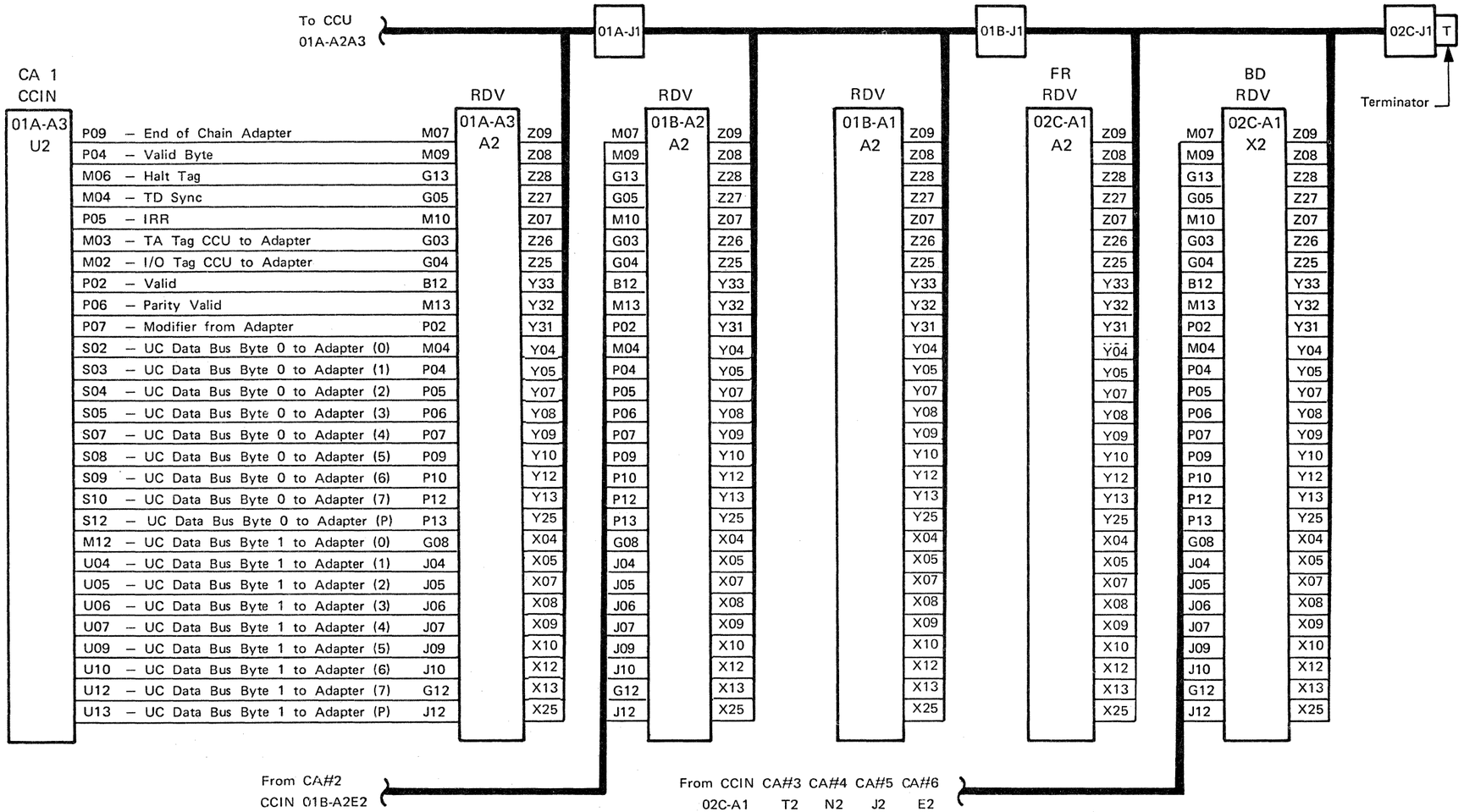
The data transfer now takes place in 8-byte bursts, except for the last one, which may be less than 8 bytes. The CCU starts the data transfer by raising the TD Tag. Depending on the direction of the transfer, into the CCU (inbound) or out of the CCU (outbound), the transfer sequence may vary. The channel adapter responds with 'Valid Tag' during the data transfer, and the CCU replies with the TD tag. The data bus is loaded for each response from the sending end, either CCU or channel adapter; each response transfers 2 bytes. On the last transfer, the CA responds with 'End of Chain Tag' (EOC) instead of 'Valid Tag'. If the last transfer contains only 1 byte instead of 2, the CA responds with 'Modifier' and 'Valid Byte' instead of 'Valid Tag'.

After this last transfer response, the CCU drops I/O. The CA raises 'Valid' and drops IRR. The CCU then raises TD, which allows another cycle steal request to start.

Note: If the CCU detects bad parity on any transfer, it responds with 'Halt' instead of the TD, tag.

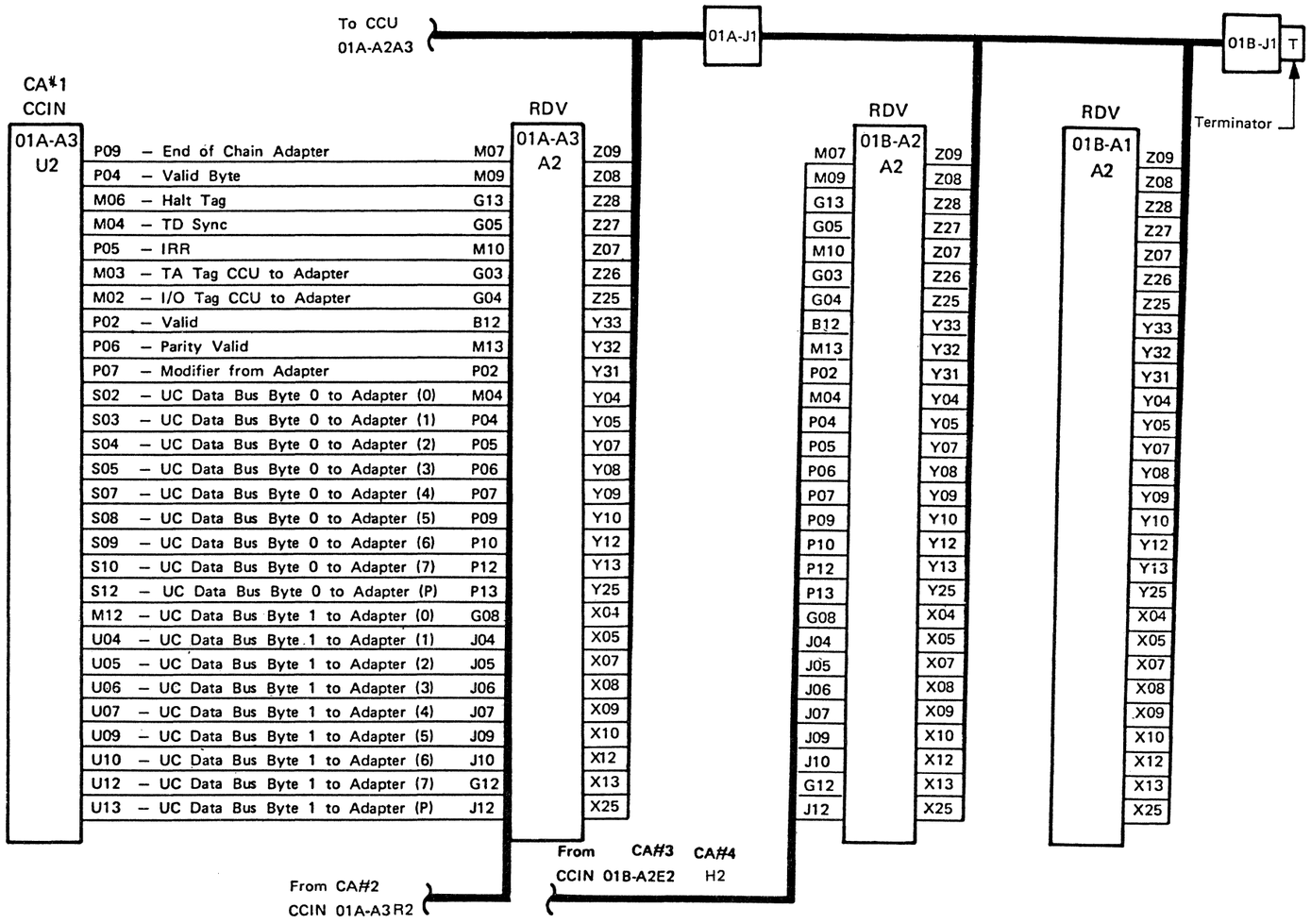
Cycle Steal, Signal Routing (Part 1 of 4)

CYCLE STEAL ROUTING (3725/3726)



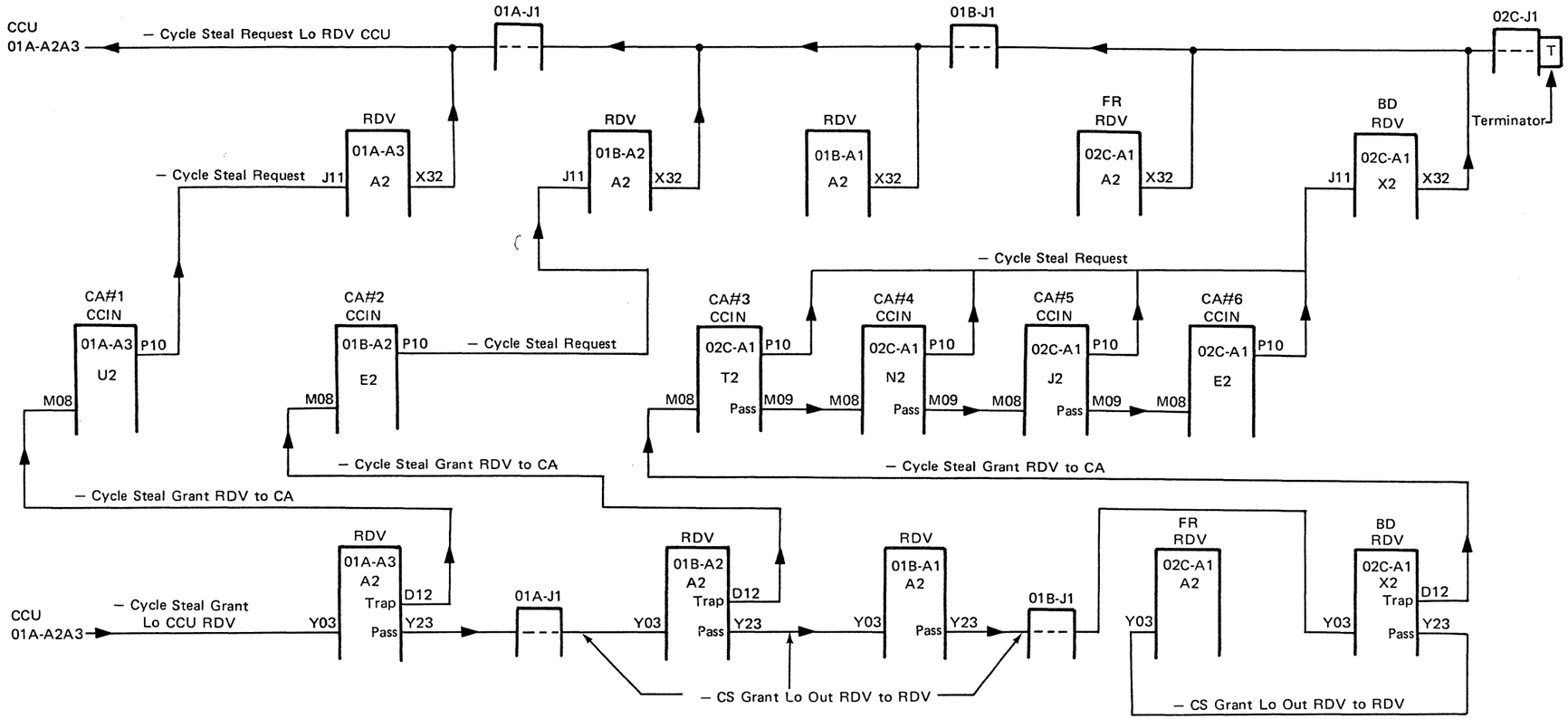
Cycle Steal, Signal Routing (Part 2 of 4)

CYCLE STEAL ROUTING (3725 MODEL 2)



Cycle Steal, Signal Routing (Part 3 of 4)

CYCLE STEAL SIGNAL ROUTING DURING AIO
(3725/3726)



Notes:

1. When a RDV card receives a cycle steal request on pin J11, it passes on the request to the CCU on the dot-OR at pin X32.

2. When a RDV card receives 'Cycle Steal Grant Lo' at pin Y03, it may either trap it, or pass it on:

   a. If the CA corresponding to the RDV card has a cycle steal request active, the RDV traps the grant signal and sends it out on pin D12 to its channel adapter.

   b. If the CA corresponding to the RDV card does not have a cycle steal request active, the RDV passes the grant signal via its pin Y23 to the next RDV.

3. When the CCIN card of one of CAs 3 through 6 receives 'Cycle Steal Grant Lo' at pin M08, it may either trap it, or pass it on:

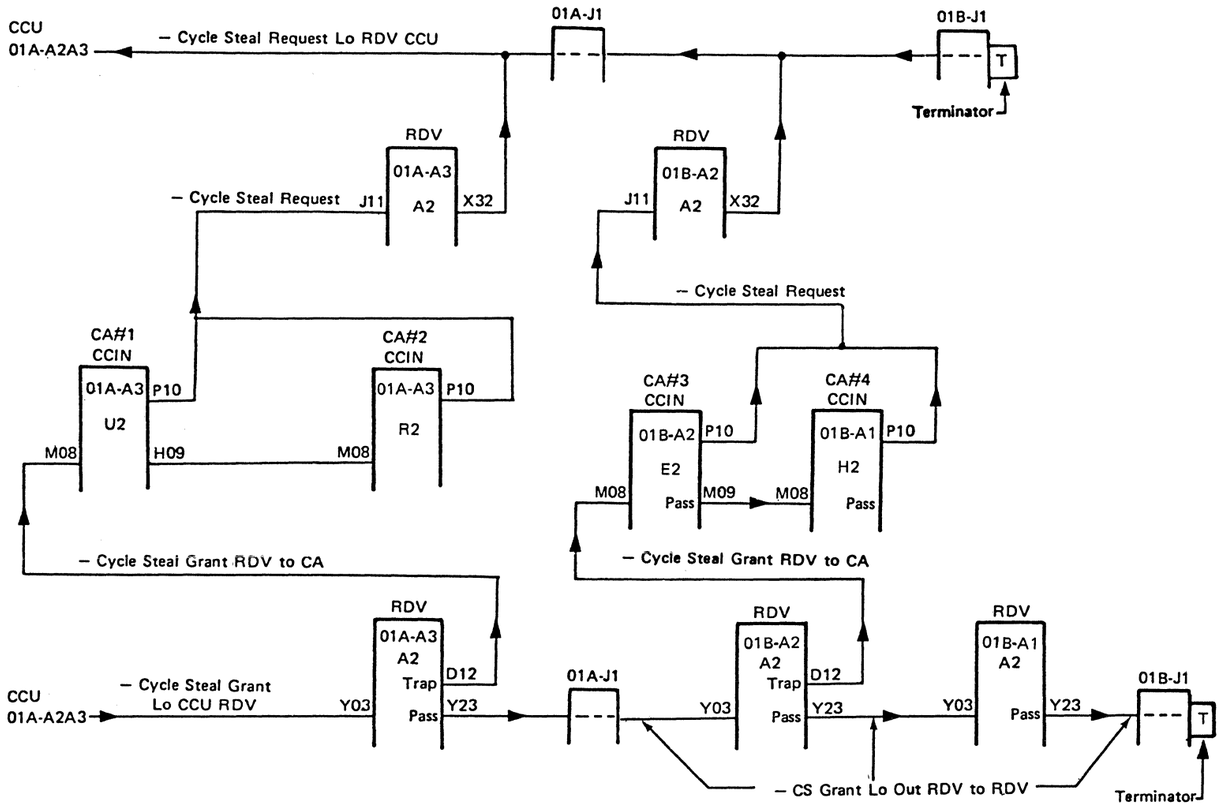
   a. If the CA has a cycle steal request active, it traps the grant signal internally.

   b. If the CA does not have a cycle steal request active, it passes the grant signal to the next channel adapter.

4. 'Cycle Steal Request' and 'Cycle Steal Grant Lo' are both tied of at the FRRDV and RDV-AD cards so that the secondary IOC bus has no access to these lines.

Cycle Steal, Signal Routing (Part 4 of 4)

CYCLE STEAL SIGNAL ROUTING DURING AIO
(3725 MODEL 2)



Notes:

1. When a RDV card receives a cycle steal request on pin J11, it passes on the request to the CCU on the dot-OR at pin X32.

2. When a RDV card receives 'Cycle Steal Grant Lo' at pin Y03, it may either trap it, or pass it on:

   a. If the CA corresponding to the RDV card has a cycle steal request active, the RDV traps the grant signal and sends it out on pin D12 to its channel adapter.

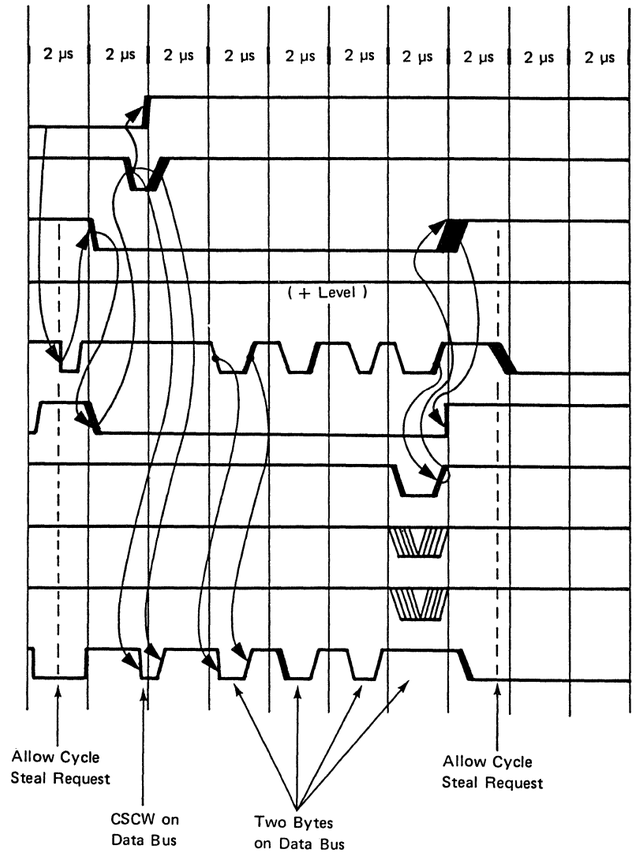
   b. If the CA corresponding to the RDV card does not have a cycle steal request active, the RDV passes the grant signal via its pin Y23 to the next RDV.

Cycle Steal, Timing and Scoping

CA CYCLE STEAL TIMING

Pins listed are on the CCIN card

- Cycle Steal Request (P10)
- Cycle Steal Grant (M08)
- I/O (M02)
- TA (M03)
- TD (M04)
- IRR (P05)
- EOC (P09)
- Modifier (P07)
- Valid Byte (P04)
- Valid (Halfword or VH) (P02)



Notes:

1. The timing diagram shows an 8-byte AIO write operation (CCU to CA).
2. The 'Modifier' and 'Valid Byte' signals are shown on the diagram, but do not occur for an 8-byte transfer. These two signals are only used during an odd byte transfer when the last data transfer has only one byte valid.

Scoping Levels

+ = +4.5 to +5V

− = 0 to +0.5V

A line that is floating will be between +1 and +2.2V.

CHANNEL INTERFACE SCOPING

Refer to Chapter 4 "CA Cable Routing" for the host interface signal internal routing and scoping points.

CA I/O INSTRUCTION FAILURE SCOPING

The channel adapter input/output instruction may be cycled and scoped using the PIO scoping routine for the IOC bus (refer to Chapter 11).

Cycle Steal, Scoping Routine (Part 1 of 2)

CHANNEL ADAPTER SCOPING ROUTINE FOR CYCLE STEAL MECHANISM

The purpose of this scoping routine is to allow the CE to troubleshoot problems related to the channel adapter cycle steal mechanism. The routine sets up an eight data byte cycle steal from CCU storage into the data buffers of the selected channel adapter. The cycle steal is performed according to the option selected. Option 06 does a data compare in the CCU by performing a PIO read of the channel adapter data buffers and comparing it with the data sent.

Running the Routine

The routine may be run by calling manual intervention routine MB01 with the 'CA' option (this is the same manual intervention routine that is used for the IOC bus).

When the message 'TO TEST CA AUTOSELECT TYPE: CA THEN PRESS SEND' is displayed, reply by pressing SEND.

When the message 'TO TEST CA CYCLE STL TYPE: CE THEN PRESS SEND' is displayed, set the cursor to the reply area, type RCE, then press SEND.

When the message 'ENTER OPT-C.A. NBR-ADDR-DATA THEN PRESS SEND' is displayed, the parameters should be entered in the following format:

'Rooxxaaaaaaddddddddddddddd

where:

oo = the selected option

xx = the channel adapter to be tested

aa = the storage address for cycle stealing

dd = 8 bytes of data

Option

The option 'oo' may be one of the following values:

01 Cycle steal with error reporting is executed once only. The '01' option may be repeated as often as required.

02 The cycle steal function is executed and loops until an error is found. The error is reported once only via an RAC 675. If you wish to continue, type 'G' (for Go); the routine then loops indefinitely until the 'BREAK' key is pressed.

03 This option is identical to option 02 above, except that if an error is found, it is not reported.

06 This option is identical to option 02 above, except that data comparison occurs between the contents of the channel adapter data buffers and the data sent.

Channel Adapter Number

The channel adapter number may take any value in the range 01 through 06.

Storage Address for Cycle Steal

The parameter 'aaaaaa' is the storage address (3 bytes : 6 hex digits) from which the data will be cycle stolen; this address must be greater than X'008000'.

Data for Cycle Steal

The parameter 'dddddddddddddddd' is the data (8 bytes : 16 hex digits).

Notes:

1. A total of 13 bytes (26 hex digits) must be input.
2. If an invalid option is selected, the following message is displayed: 'INVALID FORMAT PRESS SEND TO RETRY'. Re-enter the request.
3. If xx is not in the range X'01' through X'06', the following message is displayed: 'INVALID CA NBR PRESS SEND TO RETRY'.
4. If an invalid storage address is entered, the following message is displayed: 'INVALID MEMORY ADDRESS PRESS SEND TO RETRY'.

Example:

To select option 01 for channel adapter no. 03, using storage address X'00A000' and data string X'11112222AAAACCCC', enter:

R010300A00011112222AAAACCCC

Cycle Steal, Scoping Routine (Part 2 of 2)

Error Reporting

The following RAC-ERC codes are displayed when an error is found by the scoping routine:

RAC	ERC	Meaning
675	7F00	Tried to enable a RDV, but a level 1 interrupt occurred. The error bits indicate the cause. Action: request the PIO scoping routine for the IOC bus.
675	7F11	A level 1 interrupt occurred while a PIO was being issued to the channel. The error bits indicate the cause. Action: type G to proceed.
675	7F22	The data entered from the keyboard does not match the data that was cycle stolen. Action: type G to proceed.

The error bit field displayed is the contents of In X'76'. This register latches the error conditions found during PIO execution. The following combinations may occur:

Error Field	Suspected IOC Line*	Meaning
X'0400'	Data bus bits	IOC Bus parity check
X'0800'	VH	I/O tag is off
X'1800'	IRR	I/O tag raised on a new IOH
X'2800'	IRR, VH, EOC	No response to TA
X'4800'	IRR	VH did not fall after TD dropped
X'5800'	IRR, VH, EOC, VB, M	No response to TD for PIO read
X'6800'	IRR, VH, EOC, VB, M	No response to TD for PIO read
X'F800'	VH	I/O tag is off, VH must rise (PIO end after a data exchange)

Where:

Data bus bits = bytes 0 and 1, bits 0 through 7 and parity
EOC = end of chain
IRR = interrupt request removed
M = modifier
VB = valid byte
VH = valid halfword

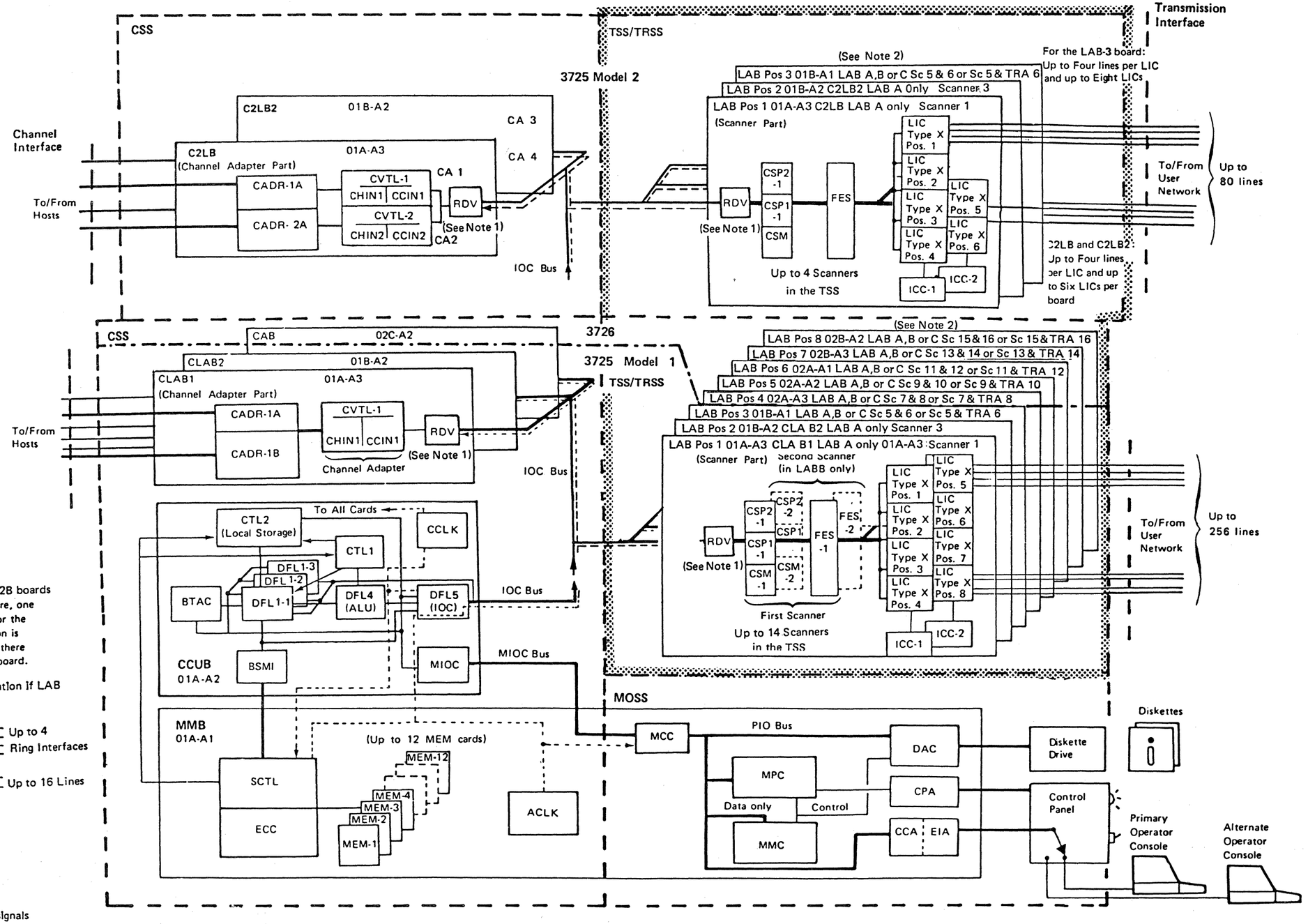
Contents

Chapter 13. Transmission Subsystem

Section 1. Unit Description

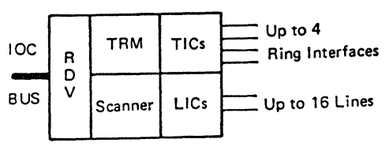
Contents

TSS in 3725 Data Flow



Note 1: As the CLAB and the CL2B boards are split into two parts in this figure, one for the channel adapter and one for the scanner, the redrive (RDV) function is shown twice for clarity. However, there is one RDV per CLAB or C2LB board.

Note 2: LAB Pos. 3 to 8 organization if LAB type C installed



Legend:

----- Clock signals

——— Data/control signals

Communication Scanner Processor (Part 1 of 4)

PURPOSE

The communication scanner processor (CSP) is a processor that operates under the control of the scanner microcode loaded from the diskette drive into CSP storage during IPL. The front-end scanner (FES) is an adapter of the CSP. One CSP and one FES is called a "scanner".

PACKAGING

One CSP is packaged on three cards:

- Communication scanner processor card 1 (CSP1)
- Communication scanner processor card 2 (CSP2)
- Communication scanner storage (CSM)

COMPONENTS

The CSP components are shown on the data flow (right). They:

(A) Connect to the IOC bus to receive or transmit data and controls

(B) Execute the instructions of the microcode to transmit and receive data, support the link protocols, translate characters, and control the communication interface

(C) Store the microcode, the transmitted/received data, the line interface parameters, and the diagnostics

(D) Provide registers in local storage for current CSP operation and permanent code in the ROS for starting the microcode IPL

(E) Control the storage and manage (by hardware only) the cycle steal with the FES

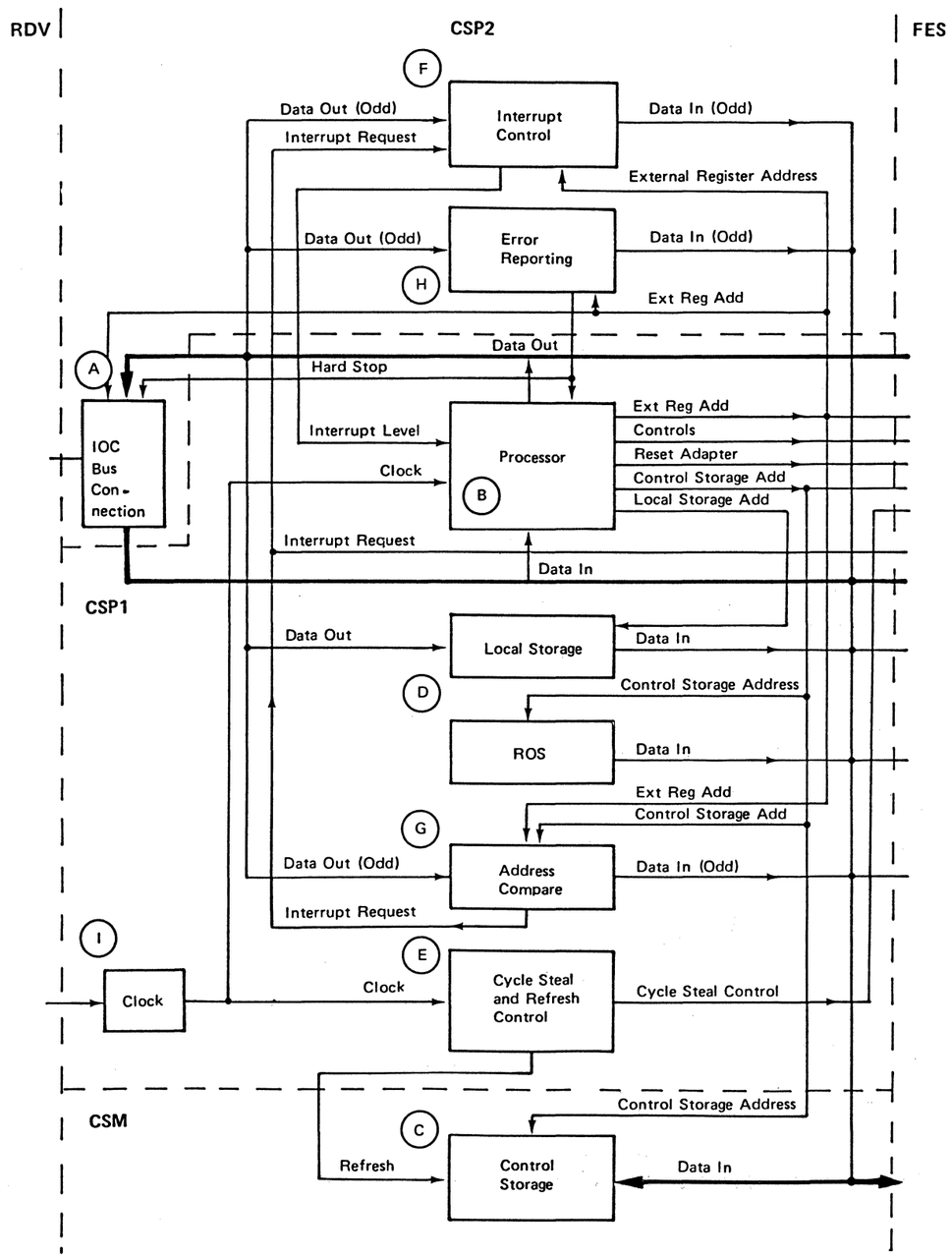
(F) Control the CSP interrupts

(G) Provide control storage address compare, and generate a CSP interrupt when the selected address is encountered

(H) Detect and report the CSP errors via the CSP error register

(I) Generate clocking signals to the processor and its associated logic from the basic clock (29.49 MHz) supplied by the adapter clock (ACLK)

DATA FLOW



LOCAL STORAGE

The local storage consists of 128 bytes (64 halfwords) of storage addressable by byte or by halfword. It is organized in 16 blocks (or pages) of eight local storage registers (LSRs) each one byte long, used to store the microcode pointers, the control blocks, and the program status words.

The local storage registers can be displayed or altered from the operator console using the TSS functions (see page 2-370).

READ-ONLY STORAGE

The read-only storage (ROS) consists of 8K bytes (4K halfwords) of storage addressable by halfwords. It includes all the permanent code needed for microcode IPL and dump, and to perform diagnostics.

The ROS can be displayed from the operator console using the TSS functions (see page 2-370).

CONTROL STORAGE

The control storage consists of up to 128K bytes (64K halfwords). It is addressable by halfword with an access time of 200 ns. An error correction code (ECC) mechanism detects and corrects all single-bit errors. The control storage is used to store:

- The scanner microcode (about 34K bytes)
- The buffers for the transmitted and received data, and the line control blocks and parameters (about 13K bytes)
- Tables and service buffers (about 17K bytes)

The control storage can be displayed or altered from the operator console using the TSS functions (see page 2-370).

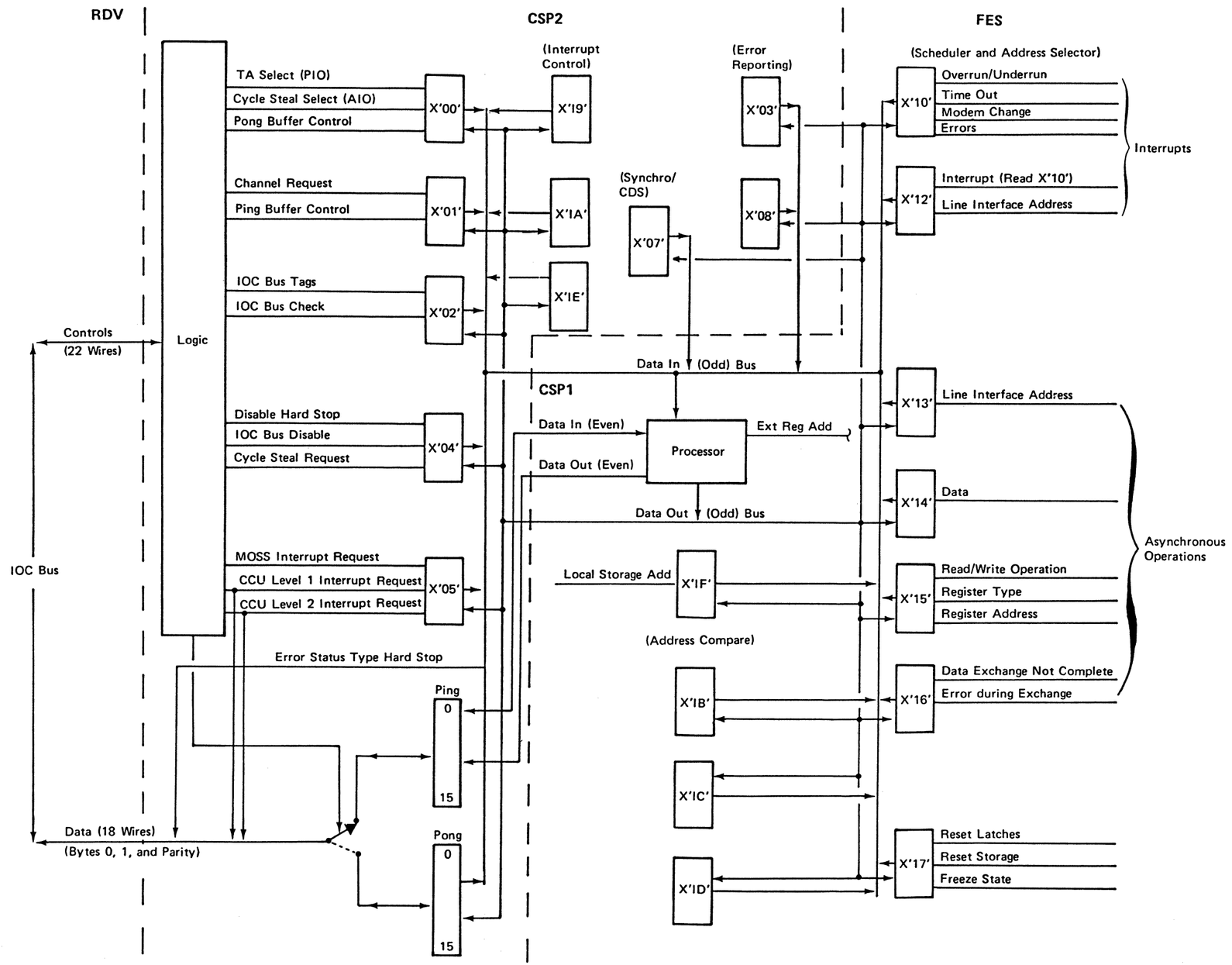
Communication Scanner Processor (Part 2 of 4)

CSP EXTERNAL REGISTERS

The CSP uses 32 eight-bit external registers located in the main communication scanner components: IOC bus connection, address compare, error reporting, and front-end scanner. The external registers are connected to the CSP via the data in (odd) bus during read operations, and data out (odd) bus during write operations. Their bits are set on or off by the hardware or the microcode.

The external registers can be displayed or altered from the operator console using the TSS functions (see page 2-370).

Address (Hex)	Function	Card
00	IOC bus control 1	CSP2
01	IOC bus control 2	CSP2
02	IOC bus service	CSP2
03	CSP error	CSP2
04	Miscellaneous/ adapter address	CSP2
05	External interrupt request/priority	CSP2
06	(Reserved)	
07	Synchro/configuration data set	CSP2
08	Error indication to the CE indicator card, and bad parity generator	CSP2
09-0F	(Reserved)	
10	Extended interrupt request	FES
11	(Reserved)	
12	Interrupt request	FES
13	Line interface address	FES
14	Data in/out	FES
15	Sync operation command	FES
16	Async operation status	FES
17	FES general command	FES
18	(Reserved)	
19	CSP interrupt request	CSP2
1A	Current CSP interrupt	CSP2
1B	Address compare control	CSP1
1C	Address compare byte 0	CSP1
1D	Address compare byte 1	CSP1
1E	CSP interrupt masks	CSP2
1F	Local storage page reg	CSP1



Communication Scanner Processor (Part 3 of 4)

CCU/CSP Register Use

The CSP external registers are used as follows:

- Program-initiated operation (PIO):

  Register X'00' indicates the PIO operation to the microcode when 'TA select' is on, and controls the pong buffer operation.

  Register X'02' reflects the status of the IOC bus tags and indicates IOC bus check when parity errors are detected on the data.

  Register X'04' provides miscellaneous IOC controls.

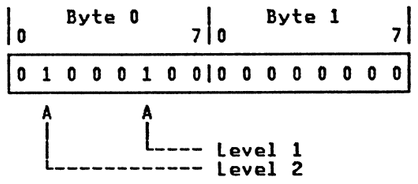
- Adapter-initiated operation (AIO):

  Register X'00' indicates the AIO operation to the microcode when 'cycle steal request' is on.

  Register X'01' controls the cycle steal sequence on the IOC bus and the ping buffer operation.

- Interrupts:

  Register X'05' controls the interrupts from the CSP to the CCU or MOSS. CCU level 1 or 2 interrupt requests are reported via bytes 0 and 1 of the IOC bus.



CSP/FES Register Use

The CSP external registers are used as follows:

- Asynchronous actions:

  Register X'13' specifies the line interface address.

  Register X'14' transfers the parameters and status in both directions.

  Register X'15' specifies the type of storage or register to be accessed in the FES (RAM, LIC, or ICC) and starts the read or write operation.

  Register X'16' gives the status of the FES at the end of the read or write operation, and indicates the errors.

- Cycle Stealing:

  Cycle stealing between the FES and CSP is performed via the CSP data bus. It is controlled by hardware only, no external registers being used in the operation.

- Interrupts:

  Register X'10' and X'12' control the interrupts from the FES on microcode level 2. The interrupt condition is indicated by X'12' bits 0 and 1, the address of the line interface that has initiated the interrupt is in X'12' bits 2 through 7, and the type of interrupt is given by X'10' bits 1 through 3.

- Reset:

  Register X'17' is used by the microcode to reset the components of the FES.

PING/PONG BUFFERS

Two half-word registers, called the ping and pong buffers, are alternately connected to the IOC bus in flip/flop mode. They are used for transferring commands, data, and control words between the IOC bus and the CSP. The switching from one buffer to the other is controlled from the IOC logic.

Although the ping/pong buffers are physically located in card CSP2, for the microcode they are logically located in the local storage (card CSP1), page 3, registers 0 through 3.

PROCESSOR CHARACTERISTICS

- Instruction Time:

  476 ns through 1258 ns depending on the type of instruction

- Program Levels:

  The interrupts are raised by the microcode (all levels) or by hardware (levels 0, 1, and 2). Levels 0 through 3 can be masked by the microcode.

  The levels are used as follows:

Level	Function
0	Error handling and address compare
1	CCU instruction processing
2	FES interrupt processing
3	Queue and command processing
4	(Not used)
5	(Not used)
6	(Not implemented)
7	Timer control and disconnect stop initialization

- Clock:

  The clock consists of a 29.4912 MHz square wave provided from the redrive card (see Chapter 5).

HARDSTOP

CSP internal errors detected by the CSP hardware cause a CSP hardstop. Two bytes of status information are presented by the CSP either to the CCU control program (NCP) or to the MOSS after the hardstop condition is detected. If the CSP is in 'Connect' mode, the error status is presented to the CCU control program (NCP). If the CSP is in 'Disconnect' mode, the error status is presented to the MOSS. The errors that cause a hardstop are listed under "CSP Internal Errors" on page 13-321. The bit definitions of the two status bytes are given under "Error Status Type Hardstop" on page 13-351. Flow diagrams are on pages 13-330 and 13-340 for reporting errors to CCU and MOSS.

A hardstop condition is handled by the CSP hardware as follows:

1. A permanent cycle steal request is forced internally to the CSP processor. No cycle steal operation takes place. However, since cycle steal has the highest priority with CSP storage, this causes the CSP microcode processing to stop, but keeps the CSP clocks running.
2.
   a. A level 1 interrupt request is presented to the CCU control program (NCP) if the CSP is in 'Connect' mode.

   b. A level 4 interrupt request is presented to the MOSS processor control program if the CSP is in 'Disconnect' mode.

Communication Scanner Processor (Part 4 of 4)

3.

a. The CCU control program (NCP) responds to its level 1 interrupt request by sending a 'Get Error Status' command. The CSP then sends the two bytes of 'error status type hardstop' if the CSP clocks are not stopped by the hardstop condition.

b. The MOSS control program responds to its level 4 interrupt request by sending a 'Get Command Completion' command. The CSP then sends the two bytes of 'MOSS command completion' (defined on page 13-352) if the CSP clocks are not stopped by the hardstop condition. Then the MOSS control program sends a 'Get Error Status' command to which the CSP responds with the two bytes of 'error status type hardstop'.

The error status type hardstop does not transit through the ping/pong buffers, but is directly presented on the IOC bus data bytes. A BER type 11, ID95, is generated, and can be displayed.

The hardstop is reset from the CCU or the MOSS via a programmed reset command or a general reset.

SCANNER STATE

The state of a scanner is shown on the console display. Once the scanner has been selected by the operator, its state appears in the 'm' field of the machine status area on the screen. (Refer to the 3725 Operating Guide for MSA description.) The state may be:

- Inoperative
- Initialized
- Connected
- Disconnected (stop or go)
- Reset

Inoperative

The scanner is inoperative when it is not in any other state.

Initialized

The scanner is initialized when the CSP is loaded with the microcode and the FES storage is initialized to all zeros. There is no operation with the control program.

Connected

The scanner is connected when it runs under the control of the control program. Errors on CCU I/O instructions are reported to the control program, and errors on MOSS I/O instructions to the MOSS.

Disconnected

The scanner is disconnected when it does not run under the control of the control program but under the control of the MOSS microcode. Only MOSS I/O instructions are executed. Any instructions from the CCU are not answered.

When disconnected from the control program, the scanner may be:

- Running (disconnected-go): The scanner can be in this state while being IPLed, dumped, or debugged using the TSS services.
- Stopped (disconnected-stop): The microcode continues to react to the MOSS instructions. The scanner can be in this state while being debugged using the TSS services.

Any errors are reported to the MOSS.

The scanner may be disconnected by:

- The operator when entering a service command (stop or address compare with the stop option selected, for example)
- The microcode when certain errors are detected (a CCU interrupt level 1 is requested with the disconnect indication)

Reset

The scanner may be reset by a general reset which resets all the scanners, or by a programmed reset which resets one specific scanner.

The operator can reset a specific scanner from the console using the TSS services.

1. General Reset:

   When the CCU activates the reset tag of the IOC bus, during a general IPL or at power-on, every scanner:

   a. Forces a CSP interrupt level 0.

   b. Resets the external registers X'03' (CSP error) and X'08' (error indicators/bad parity generator).

   c. Disables the IOC bus and resets the FES by setting on 'reset adapter. The LICs and ICCs are reset at the same time.

   d. Starts microcode execution at level 0 address 0. The first task of the microcode is to initialize the CSP external registers.

2. Programmed Reset:

   The control program can reset one scanner by sending a write command in the PIO mode with operation code equal 'Programmed Reset' and the scanner address. When this command is received, the selected scanner:

   a. Disables the hardstop (X'04' bit 3 on).

   b. Restarts the microcode operation at level 0 address 0.

   c. Resets the interrupt requests to the MOSS and CCU (X'05' bits 0 and 2 on).

The connected FES is stopped (FES status: freeze) but the storages of the CSP and FES are not reset: the CSP storage can be dumped.

SCANNER COMMANDS

The following commands may be used from the operator console to modify the scanner state.

Current State	Possible Scanner Commands	Resulting State
Connected	Stop Reset Dump IML	Disconnected/stop Reset Reset Initialized
Disconnected/go	Stop Reset Dump IML	Disconnected/stop Reset Reset Initialized
Disconnected/stop	Start Reset Dump IML	Disconnected/go Reset Reset Initialized
Reset (or Unknown mode)	Reset Dump IML	Reset Reset Initialized
Initialized	Stop Connect Reset Dump IML	Disconnected/stop Connected Reset Reset Initialized
Inoperative	Reset Dump IML	Reset Reset Initialized

Front-End Scanner (Part 1 of 2)

PURPOSE

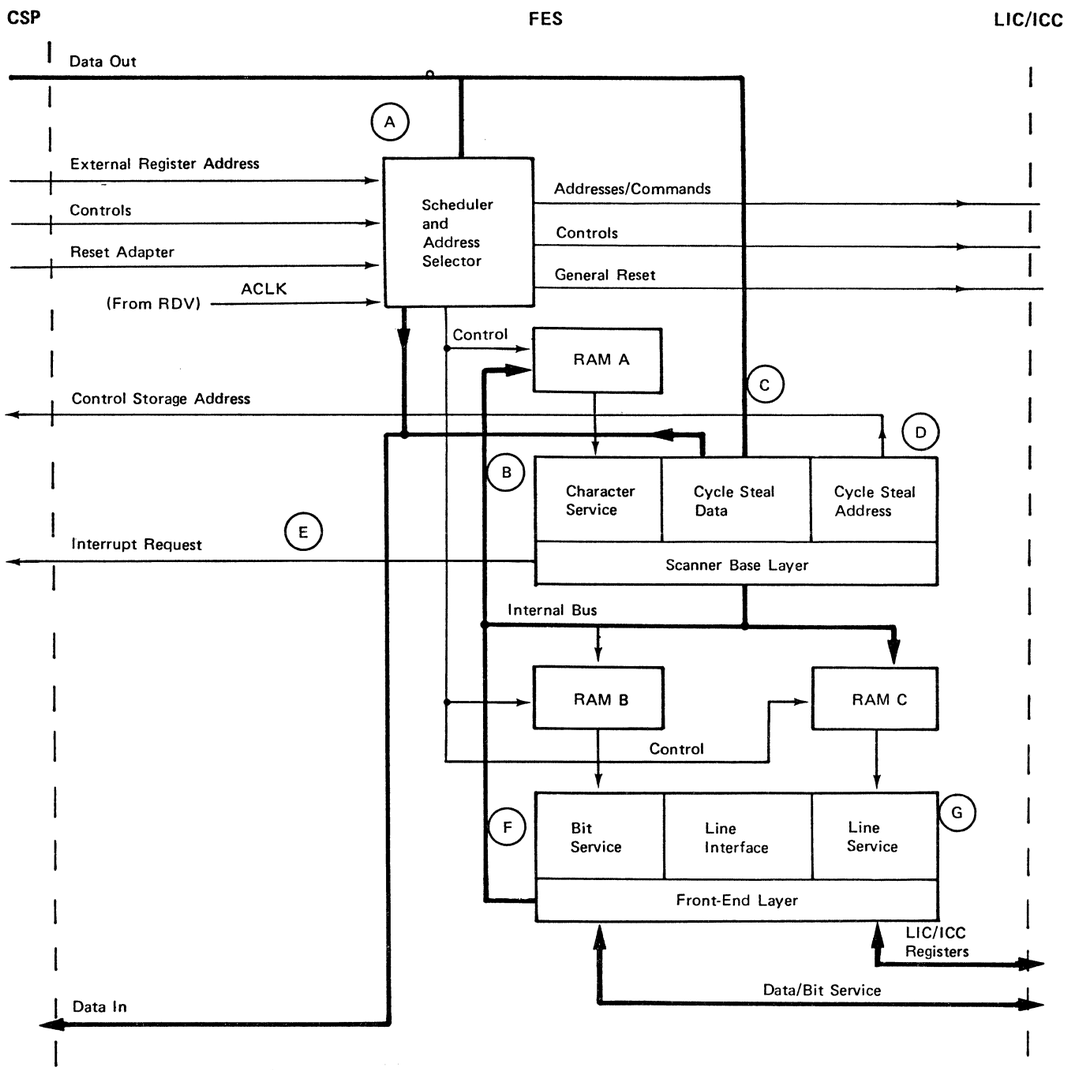
The front-end scanner (FES) is a scanning circuit that supports, under the control of the CSP, a wide range of protocols and line interfaces. The FES is an adapter of the CSP.

THROUGHPUT

The FES maximum throughput varies from 256 kbps (for one line attached), to 307.2 kbps (for 32 lines at 9600 bps distributed over eight LICs). The throughput is spread over the LICs up to the last one installed on the LAB; if only one LIC is installed, all the throughput is devoted to this LIC (see also "LIC Weight" on page 1-051).

The FES uses the 'LIC present' wire to scan up to the last LIC even if the intermediate LICs are not plugged. In addition, the 'wide-band' wire used with high-speed lines attributes all the scanning throughput to one line on the LIC.

DATA FLOW



PACKAGING

One FES is packaged on one card.

COMPONENTS

The FES components are shown on the data flow and have the following functions:

(A) Control the FES timing, the exchange with the CSP, and the RAM operation

(B) Control the transfer of characters between the front end and the CSP

(C) Handle the data halfwords coming from or going to the control storage

(D) Provide cycle steal control for the CSP

(E) Interrupt the CSP on level 2 for buffer and data management, and for error reporting

(F) Serialize/deserialize the bits transmitted to or received from the LICs

(G) Provide line services related to link protocol and modem control

Front-End Scanner (Part 2 of 2)

FUNCTIONS

The FES functions are organized in relation to the three main components as follows:

Scheduler

The scheduler:

- Communicates with the CSP through the external registers (asynchronous path) to:
  - Send general commands to the FES using X'17'
  - Read interrupt information using X'10' and X'12'
  - Access the RAMs and LIC/ICC registers using X'13', X'14', X'15', and X'16'
- Generates the FES timing
- Scans the lines on two levels, character service and bit service, depending on the processing speed (see also "Scanning Operation" on page 13-261).
- Controls the FES RAMs

Scanner Base Layer

The scanner base layer:

- Transfers characters to and from the control storage in cycle steal
- Reports line status and fetches transmission parameters using cycle steals
- Reports line status and errors by interrupting the microcode on level 2
- Manages timers

Front-End Layer

The front-end layer:

- Serializes received bits into characters and deserializes characters to be transmitted into bits (bit service)
- Supports link protocols and line interfaces (line service)

STORAGES

The FES includes three random access storages (RAMs):

- RAM A is used for character service
- RAM B is used for bit service
- RAM C is used for line service

In each RAM, four halfwords are assigned to each line interface (receive and transmit), this gives a total of 4x64 = 256 halfwords per RAM.

The FES storages can be displayed or altered from the operator console using the TSS Functions (see page 2-370).

RESET

The FES can be reset from the CSP by:

- General reset
- Programmed reset

General Reset

The 'reset adapter' signal is activated for the CSP/FES at CSP power-on reset or, in normal operation, by the microcode. This signal:

1. Resets the FES latches.
2. Disables the lines between the CSP and the FES.
3. Resets the FES storages: all bits are set to zero and the correct parities are written.
4. Stops all scanning (the FES is in the freeze state).

The LIC and ICC registers are reset at the same time. At the end of the general reset, X'17' shows:

- Bit 1 off (storage reset complete), unless an error has been detected
- Bit 2 on (freeze state)
- Bit 4 on if an error has been detected during storage reset

Programmed Reset

The scanner microcode can reset the FES using the FES general command register (X'17').

- X'17' Bit 0:

  This bit, when on, resets the FES latches and disables the CSP/FES lines. The FES storages are not reset and the FES is not in the freeze state at the end of the process.

- X'17' Bit 1:

  This bit, when on, resets the FES storages when bit 0 is off. At the end of the process, bit 1 is reset by hardware, except if an error has been detected (bit 4 is on), and the FES is not automatically in the freeze state.

- X'17' Bit 2:

  This bit, when on, sets the FES in freeze state when bits 0 and 1 are off. All scanning is stopped at the end of the current line operation. The microcode must reset bit 2 to restart the scanning.

Line Interface Card (Part 1 of 2)

PURPOSE

The line interface card (LIC) attaches up to four communication interface cables to the controller. The number of lines connected to a LIC depends on the line throughput.

For information about:

- LIC types and characteristics, see Chapter 1
- LIC interface physical description, see Chapter 4

PACKAGING

One LIC is packaged on one card.

COMPONENTS

The LIC components are shown on the data flow. They:

(A) Interface the DCEs and direct attached terminals.

(B) Transmit to and receive from the FES the data, bit-by-bit, over the data/bit service bus.

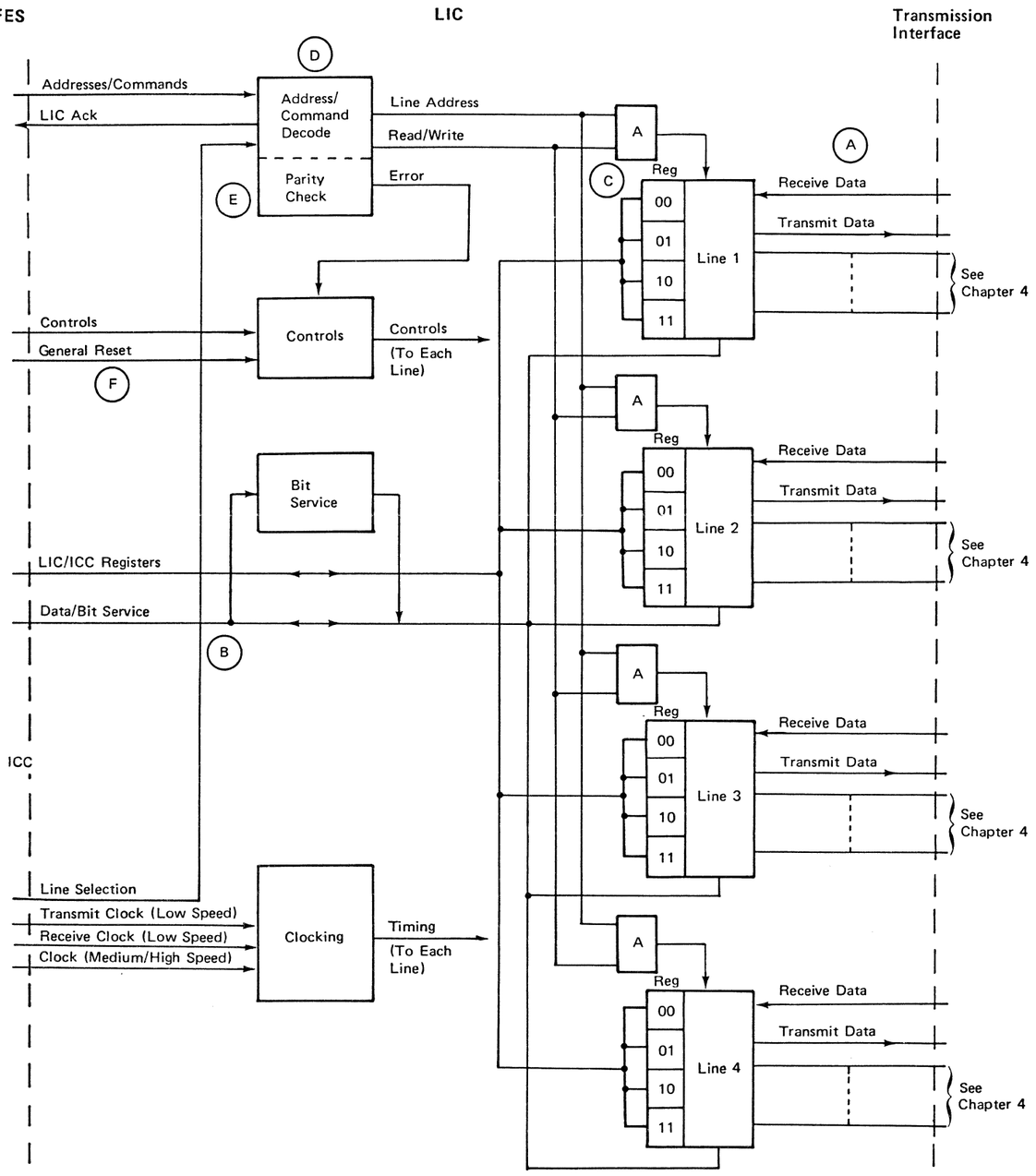
(C) Provide the CSP with the line status and characteristics via registers (four 6-bit registers per connected line).

(D) Receive the addresses and commands from the FES and decode them.

(E) Check the bit parities and activate 'LIC Ack' if the parity is correct. The absence of 'LIC Ack' is reported by the FES to the CSP which then disables the failing line.

(F) Reset the line interfaces when 'general reset' is on.

DATA FLOW



LIC REGISTERS

Each line of the LIC is provided with four 6-bit registers numbered '00' through '11'. They are used to give information about the hardware configuration to the microcode and the FES logic (see page 13-540 for a bit-by-bit description).

The LIC registers can be displayed or altered from the operator console using the TSS Functions (see page 2-370).

The microcode can read and/or write the LIC registers via the FES front end and the LIC/ICC register bus.

Register '00'

This register contains the modem-in status. It is read-only.

Register '01'

This register contains the modem-out status. It can be read or written and is also used for error detection.

Bit 0, when off, sets the line in wrap mode: transmit is connected to receive. This function is used for tests and diagnostics, and is accessible from the operator console.

Register '10'

This register provides information about the LIC position within the LAB and the cable attachment types. It is read-only.

Register '11'

This register provides information about the card identifiers and the clock selection. It can be read and written, and is used to reset one specific line on the LIC.

Line Interface Card (Part 2 of 2)

RESET

The LIC or one connected line can be reset from the CSP via the FES by:

- Activating the 'general reset' line
- Using LIC register '11' (individual line reset)

General Reset

'General reset' is activated automatically, from the FES at CSP power-on (power-on reset). In normal operation, 'general reset' may be activated by the CSP microcode. This signal resets all the LICs attached to the FES: all the lines are set to 0, but 'transmitted data' is set on except for LIC4, where this bit is also set off.

Line Reset

The setting on by the microcode of register '11' bit 0 for one specific line resets the line. Register 01 is forced to zero and 'transmitted data' is set on except for LIC4, where this bit is also set off. The three other lines are not reset.

Line Disabled/No-Operation

With NCP, the 'disable' command resets one specific line. The line is marked as disabled in the NCP tables.

With EP, the 'disable' command stops the current line operation, but the next data transfer (transmit or receive) is accepted, provided that the line is not faulty.

COMMUNICATION INTERFACE

For communication interface details, refer to "Communication Interfaces" starting on page 4-130.

Internal Clock Card

PURPOSE

The internal clock card (ICC) feature supplies clocking to the lines interfacing:

- DTE clocking (transmission speeds up to 1200 bps). For these transmission speeds, jumpering on the board is not used.
- Direct-attached terminals (transmission speeds from 2400 bps through 56 kbps). For these transmission speeds, jumpering, on the board is mandatory (see jumpering, that follows).

An ICC can supply up to four LICs (16 lines). Two ICCs are provided per feature.

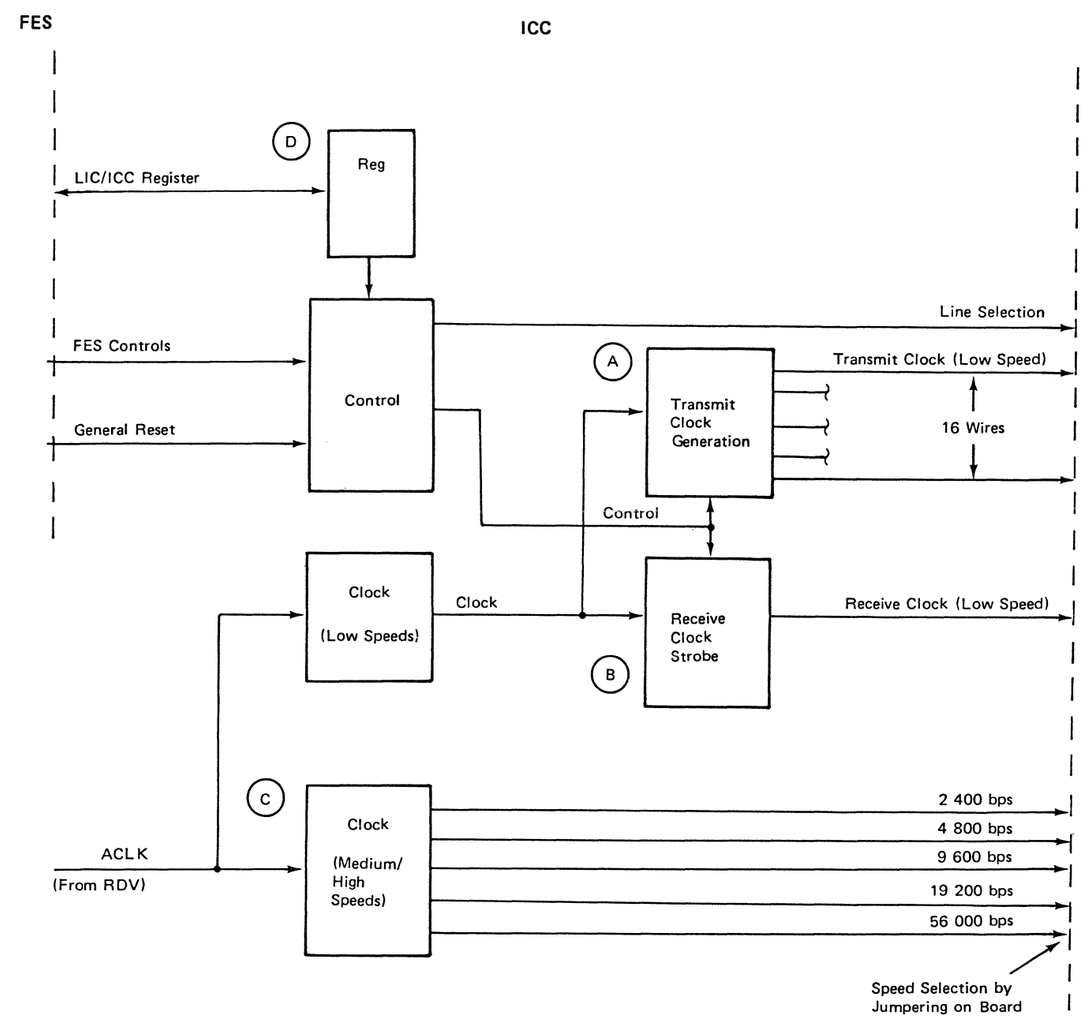
PACKAGING

One ICC is packaged on one card. It can be installed on the customer's site. 2400 bps through 56 kbps clock signals are distributed to LICs through jumpers.

JUMPERING (FOR DIRECT-ATTACHED TERMINALS)

For distributing the clock at speeds from 2400 bps through 56 kbps, jumpers are needed on the board. They are set at 9600 bps at manufacture and can be adjusted at installation time (see page 4-270 for details). Because only one connection is set between the ICC and one LIC, all the terminals directly attached to a LIC must have the same transmission speed.

DATA FLOW



COMPONENTS

The ICC components are shown on the data flow. They:

(A) Provide the clock to the transmit interfaces for DTE clocking or low speed direct-attached terminals (50 bps through 1200 bps)

(B) Retrieve the receive clock for DTE clocking (up to 16 interfaces) by indicating to the LIC when to sample the received bit (50 bps through 1200 bps)

(C) Generate the clock for the medium and high speed direct-attached terminals (2400 bps through 56 kbps)

(D) Store the line addresses and speed information for DTE clocking. The ICC register can be written and read from the CSP.

REGISTER (FOR MODEMS WITHOUT CLOCKS)

The ICC has a 5-bit register that allows the CSP microcode to select clock speeds from 50 bps to 1200 bps for unclocked modems. The register is loaded at microcode IPL from the CSP via the FES and the LIC/ICC register bus by a write operation.

For tests and diagnostics, the register contents can be read from the CSP (see page 13-541 for transmission speed coding).

The ICC register can be displayed or altered from the operator console using the TSS Functions (see page 2-370).

RESET

The ICC latches and register are reset from the CSP via the FES when 'general reset' is activated.

Scanner Microcode Organization (Part 1 of 2)

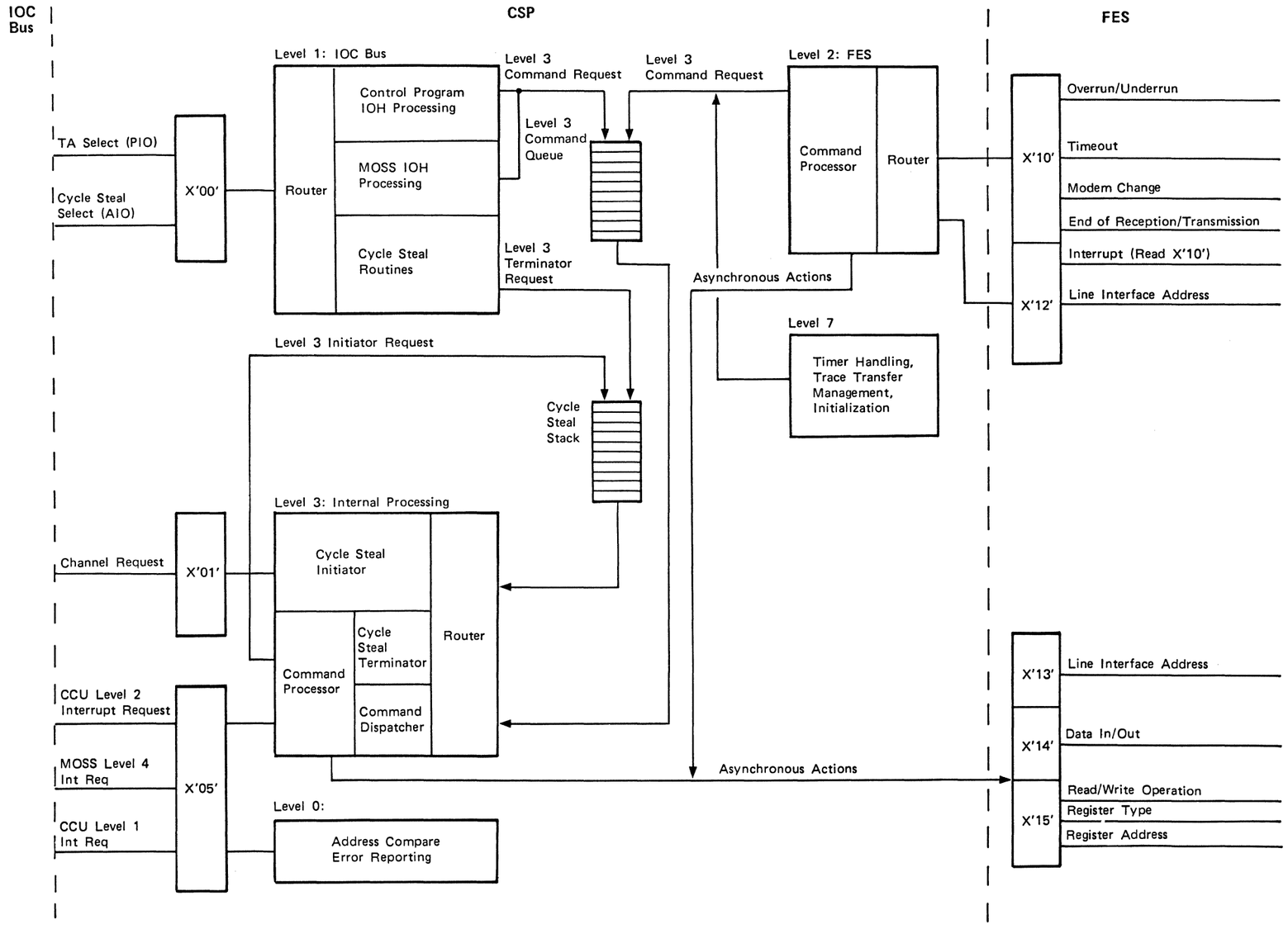
FUNCTIONS

The main functions of the scanner microcode are to:

- Manage the data buffers in the control storage for the transmitted and received data
- Support the link protocols
- Control the line interfaces
- Provide service facilities under the control of the NCP/EP or the MOSS

GENERAL DATA FLOW

(See also page 13-011)



Scanner Microcode Organization (Part 2 of 2)

LINE OPERATING MODES

The microcode operates the lines in one of the following modes:

- Normal mode
- Character mode
- Burst mode
- Service mode

The operating mode is selected on a line-by-line basis and the scanner may run all three modes at the same time.

Normal Mode

The microcode normally uses this mode to transfer data using messages (several characters) in a burst. The supported link protocols are:

SDLC:

Half-duplex or duplex
CRC management
Zero-bit insertion and deletion
Flag abort and idle detection
NRZI encoding and decoding

BSC:

Half-duplex
EBCDIC (NCP/EP)
ASCII (NCP/EP)
EBCDIC transparency (NCP/EP)
ASCII-7 transparency (EP)
Control character recognition
CRC management
VRC/LRC management (ASCII)
Synchronization character insertion and deletion
Timeout on continuous synchronization

X.21:

Leased lines
Switched lines

Autocall:

NCP/EP
Automatic call origination

Character Mode

This mode is similar to the operating mode of the communication scanner type 2 of the 3705 which transfers data character by character. The supported link protocols are:

BSC:

No control character recognition
No CRC management
No synchronization character insertion
No timeout on continuous synchronization

Start-stop:

Start bit insertion and deletion
Five through eight information bits
Insertion of one or two stop bits

Burst Mode

This mode supports start-stop protocol only. It is similar to character mode, except that the characters are exchanged by bursts of up to four characters between the scanner and the CCU. The detection of the ending character is performed by the scanner microcode.

Service Mode

In service mode, the scanner executes the commands sent from the MOSS. The scanner may be connected or disconnected.

MICROCODE LEVELS

The microcode operates on five levels numbered 0, 1, 2, 3, and 7 (levels 4, 5, and 6 are not used).

Level 0

This level is used for error reporting and for address compare. When errors are detected, the 'CCU level 1 interrupt request' bit is activated in external register X'05' and a CCU level 1 interrupt is requested via the IOC bus (see also "Reporting Errors to the CCU" on page 13-330).

Level 1

This level handles the CCU and MOSS I/O instructions received from the IOC bus. The router transmits the following information to the processing functions:

- CCU and MOSS I/O instructions received via the 'TA select' tag
- AIO operations initiated via the cycle steal grant tag

The output PIO and MIO commands are transmitted after checking to level 3 via the command queue. The input PIO and MIO are handled on level 1. CCU cycle steal operations are handled on level 1, and then resume level 3 via the cycle steal stack.

Level 2

This level handles FES interrupts received via register X'12'. The address of the line interface that requested the interrupt is given in register X'12'.

Interrupts are requested from the FES for:

- End of receive or transmit operations
- Microcode request
- Time out
- Modem-in change
- Overrun or underrun condition
- Error

After interrupt level 2 processing, a command request is loaded into the command queue to resume level 3 processing.

Level 3

This level:

- Manages the command queues and the cycle steal stack.
- Executes the control program commands. Commands for the line interfaces are transmitted to the FES asynchronously via registers X'13', X'14', and X'15'.
- Executes the MOSS commands.

Level 3 interacts with the IOC bus via registers X'01' for channel requests and X'05' for interrupt requests.

The asynchronous operations are mainly:

- Set mode operations for initialization of the line interfaces and FES functions
- Commands for receive, transmit, and reset operations
- Modem management operations
- Timer management operations

Level 7

This level handles the timer routines, initializes the disconnected stop status, and manages trace transfer (SIT).

Microcode Interaction with Control Program (Part 1 of 5)

The scanner microcode operates with the CCU control program (in normal mode or character mode) using:

- Reserved areas in the CCU main storage
- Reserved areas in the scanner storage
- Instructions (IOH or IOHI) to move control information between the CCU and the scanner
- Commands (specified in register R1 of the associated IOH/IOHI instruction) to start a line, or start a line with initialization

PIO is used to transfer commands to the line interfaces (4 bytes) and to get the line identification and error status (2 bytes).

AIO is used to transfer data, parameters, and status between the scanner and CCU storages in cycle steal mode.

The scanner interrupts the control program on level 2 for normal operation, and on level 1 for error reporting.

RESERVED CCU STORAGE AREAS

In the CCU main storage, three types of area are reserved for communication with the scanners:

- Parameter/status areas
- Line vector table
- Data buffers

Parameter/Status Area (PSA)

For each line interface (transmit or receive), the control program reserves one parameter/status area. A duplex lines has two PSAs, a half-duplex line one PSA. The scanner accesses the PSA in cycle steal.

The PSA is divided into two areas:

1. The parameter area (16 bytes, 4 words), used to transfer control parameters from the CCU to the scanner
2. The status area (12 bytes, 3 words), used to transfer the command status from the scanner to the CCU

The 28 bytes (7 words) making up the PSA must be contiguous, but may be anywhere in main storage. More than one PSA may be prepared for each line interface to allow quick network reconfiguration.

Line Vector Table (LNVT)

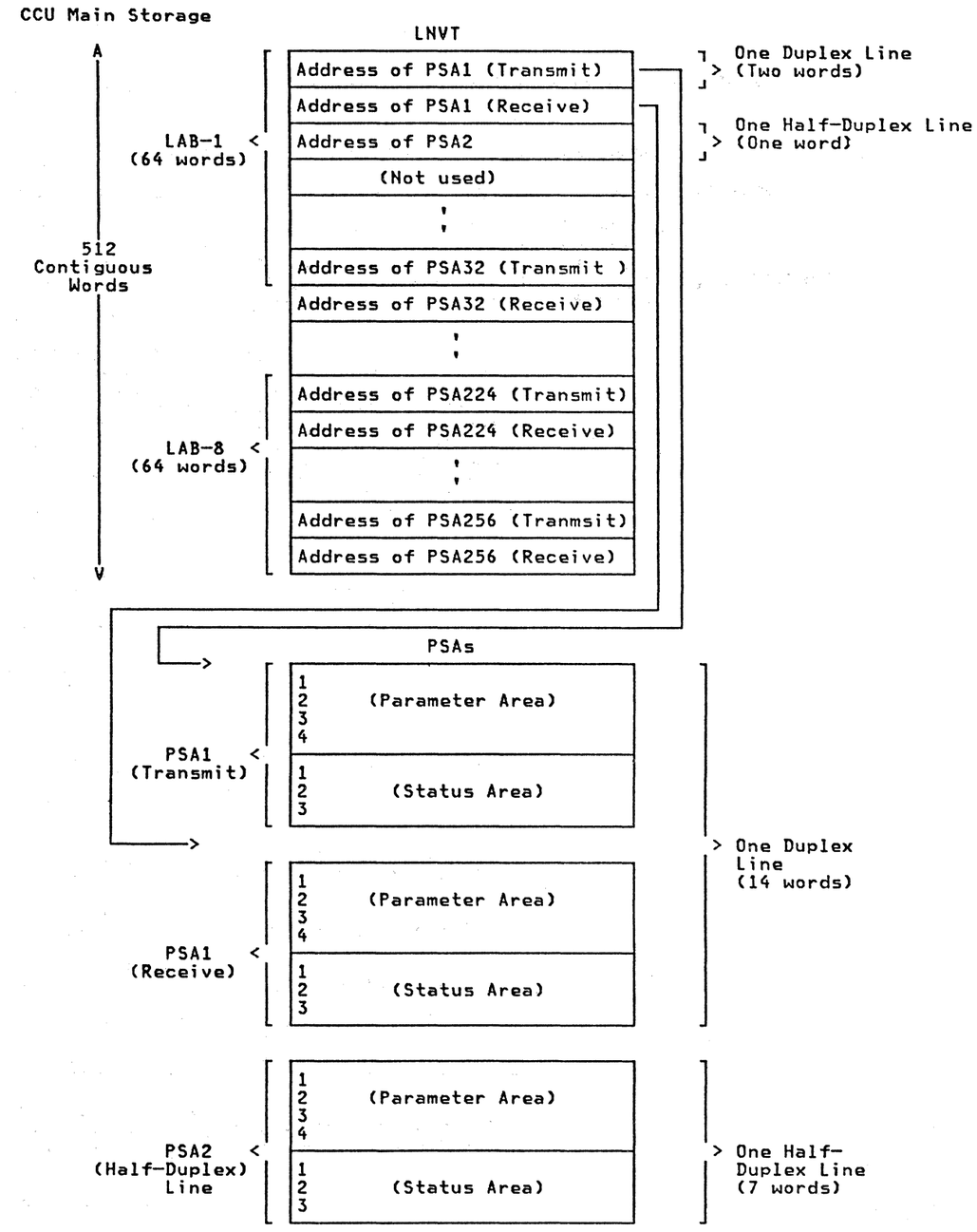
The line vector table consists of 512 locations, two for each of the 256 lines in the controller. There are two entries per line: one for the transmit interface (even address), another one for the receive interface (odd address). For a half-duplex line, the first LNVT entry points to the unique PSA that is used for both the transmit and receive interfaces. The second entry is not used (except during wrap processing).

The starting address of the LNVT is set by default to X'880'. The scanner calculates the LNVT address of its lines. If necessary, the LNVT starting address may be changed using a set line vector table high/low instruction.

LNVT Default Addresses

LAB Position	Line Address	Address (Hex)
1	000-015 016-031	880-8FC 900-97C
2	032-047 048-063	980-9FC A00-A7C
3	064-079 080-095	A80-AFC B00-B7C
4	096-111 112-127	B80-BFC C00-C7C
5	128-143 144-159	C80-CFC D00-D7C
6	160-175 176-191	D80-DFC E00-E7C
7	192-207 208-223	E80-EFC F00-F7C
8	224-239 240-255	F80-FFC 1000-107C

The 32 words preceding the LNVT (addresses X'800' through X'880') are used for the trace LNVT during scanner interface trace.



Microcode with Control Program (Part 2 of 5)

Data Buffers

These are areas reserved for the temporary storage of data and other information in transit through the controller. They are accessed by the scanner in cycle steal mode. The address of the buffers to be used by the scanner is part of the parameter area of the PSA. The format of the buffers depends on the control program (NCP or EP).

Buffer Format (No prefix-offset for EP)

Byte	Contents
0-3 4-5 6 7	Prefix: Address of the next buffer in the chain (not used) Offset value in bytes Buffer byte count
	Offset
Up to 256	Data (varies with control program)

For the first buffer in a chain, the offset and byte count are provided by the PSA. For the other buffers:

- In transmit operation: the link pointer, offset and byte count are taken from the buffer prefix.
- In receive operation: the link pointer is taken for the buffer prefix, the offset is zero, and the byte count is provided to the scanner by the Set Mode information.

RESERVED SCANNER STORAGE AREAS

Storage MAP

See page 13-550.

Scanner Control Block

See pages 13-551 and 13-552.

Line Control Block

See page 13-553.

Interface Control/Parameter Status Area

See page 13-554.

Line Interface Buffer

See page 13-555.

Parameter/Status Area

The PSAs of the CCU storage are duplicated in the scanner control storage. The parameters related to the line interface (parameter area) are transferred from the CCU to the scanner with the command; the status of the line after execution of the command (status area) is transferred from the scanner to the CCU. All transfers are made in cycle steal.

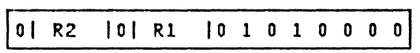
CCU INSTRUCTIONS

The following CCU instructions are used to move control information between the CCU and the scanner:

- Start line initial (output initialization of interface)
- Start line (CCU output instruction)
- Get error status (CCU level 1 input instruction
- Get line identification (CCU level 2 input instruction)
- Set line vector table high/low (relocation of the line vector table)

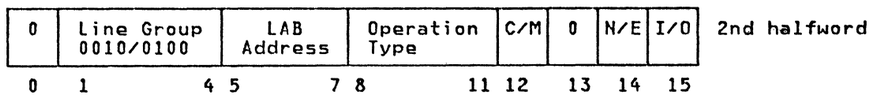
The CCU I/O instructions may be IOH or IOHI. In the following descriptions, the output instructions are IOH and the input instructions are IOHI.

IOH Format



This instruction transfers the contents of the register specified by RI to the communication scanner, or places information coming from the communication scanner into the register specified by RI. The scanner, the scanner command or register, and the direction of data movement are all specified by the contents of R2. For the instruction to execute correctly, R2 must be loaded as follows:

R2 Contents



C/M = CCU/MOSS bit: 0 = initiated by CCU, 1 = initiated by MOSS

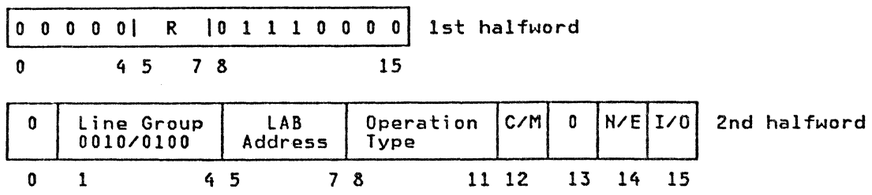
N/E: 0 = normal mode, 1 = character mode or burst mode

I/O = input/output bit: 0 = output, 1 = input

Refer to Chapter 11 for IOH operation and address/command format.

Microcode Interaction with Control Program (Part 3 of 5)

IOHI Format



C/M = CCU/MOSS Bit: 0 = initiated by CCU, 1 = initiated by MOSS

N/E: 0 = normal mode, 1 = character mode or burst mode

I/O = input/output bit: 0 = output, 1 = input

Refer to Chapter 11 for IOHI operation and address/command format.

At TA time, this instruction transfers the contents of the register. specified by R to the scanner, or places information coming from the scanner into the register specified by R. The scanner, the scanner command or register, and the direction of data movement are all specified by the contents of the second halfword.

R2 and R Contents

The IOH (R2 contents), IOHI (R contents) on the IOC bus are as follows:

	DB0		DB1		
	123	4567	123	567	Comments
CSP	0..	0...	CCCC	....	Format
	....	....	0000	0..0	Start line
	....	....	0001	0..0	Start line initial
	....	....	0010	0..0	Set LNVT high
	....	....	0011	0..0	Set LNVT low
	....	....	....	0...	NCP cmd
	....	....	....	1...	MOSS cmd
	....	....	....	.0..	n/a
	....	....	....	..0.	Normal mode
	....	....	....	..1.	Character mode
	....	....	....	...0	Write
	....	....	....	...1	Read
CSP/1st	..01	....	....	....	Any LAB type
CSP/2nd	..10	....	....	....	LAB type B
CSP-1	..01	.000	....	....	C2LB 3725-2
CSP-1	..01	.000	....	....	CLAB1 3725
CSP-3	..01	.001	....	....	C2LB2 3725-2
CSP-3	..01	.001	....	....	CLAB2 3725
CSP-5	..01	.010	....	....	LAB-3 3725
CSP-6	..10	.010	....	....	LAB-3 3725 Model 1 or 2
CSP-7	..01	.011	....	....	LAB-4 3726
CSP-8	..10	.011	....	....	LAB-4 3726
CSP-9	..01	.100	....	....	LAB-5 3726
CSP-10	..10	.100	....	....	LAB-5 3726
CSP-11	..01	.101	....	....	LAB-6 3726
CSP-12	..10	.101	....	....	LAB-6 3726
CSP-13	..01	.110	....	....	LAB-7 3726
CSP-14	..10	.110	....	....	LAB-7 3726
CSP-15	..01	.111	....	....	LAB-8 3726
CSP-16	..10	.111	....	....	LAB-8 3726

Note: CSP-2 and CSP-4 do not exist.

Start Line Initial

This instruction is used the first time a line is addressed from the control program. The contents of R1 provides the scanner with the line interface address. The scanner calculates the related LNVT entry, then cycle steals the PSA address of the line interface (see also "Instruction Operation" on page 13-230).

The PSA address is saved in the ICB for subsequent start line instructions. The start line initial instruction is issued after IPL and each time a PSA is changed for a line.

R1 or DB0/DB1 at TD Time

Bit	Function
0-7	High level command code
8-10	0 0 0
11-15	Line interface address

Bits 0 through 7: Form the command code (see "Commands" on page 13-124).

Bits 11 through 15: Form the line interface address in the LAB: 00 (X'00') through 64 (X'3F'). Bit 15 is off for a transmit interface, on for a receive interface. For a half-duplex line, bit 15 is always off.

R2 or DB0/DB1 at TA Time

Bit	Function
0	0
1-4	Line group
5-7	LAB address
8-11	0 0 0 1 (operation type)
12-13	0 0
14	Character mode
15	0 (output)

Bits 1 through 4: Select the first or second group of 16 lines in a LAB.

0010 : First group
0100 : Second group

In a LAB type B, this division corresponds to the two scanners.

Bits 5 through 7: Select the LAB:

000 : CLAB1
001 : CLAB2
010 : LAB pos 3
011 : LAB pos 4
100 : LAB pos 5
101 : LAB pos 6
110 : LAB pos 7
111 : LAB pos 8

Bits 8 through 11: Define the type of operation performed with the line.

Bit 14: Specifies the line operating mode:

0 : Normal mode
1 : Character mode or burst mode

Microcode Interaction with Control Program (Part 4 of 5)

Get Error Status

This instruction is issued to the scanner that requested a CCU interrupt level 1. The scanner responds with an error status which is loaded into register R.

Register R contains the error status (see "Error Status Description" on page 13-350).

Immediate Field

Bit	Function
0	0
1-4	Line group
5-7	LAB address (operation type)
8-11	0 0 0 1
12-14	0 0 0
15	1 (input)

Bits 1 through 11: (Same as R2 for start line initial).

Start Line

This instruction is used to start a line operation when the PSA address is already known by the scanner. The scanner uses the line interface address to locate the PSA address following the ICB, cycle steals the PSA contents from the CCU storage, then executes the command (see also "Instruction Operation" on page 13-230).

R1 or DB0/DB1 at TD Time

Bit	Function
0-7	Command code
8-10	0 0 0
11-15	Line interface address

Bits 0 through 15: (same as in R1 for start line initial).

R2 or DB0/DB1 at TD Time

Bit	Function
0	0
1-4	Line group
5-7	LAB address
8-11	0 0 0 0 (operation type)
12-13	0 0
14	Character mode or burst mode
15	0 (output)

Bits 1 through 11 and 14: (Same as in R2 for start line initial).

Get Line Identification

This instruction is issued to all the scanners when a CCU level 2 interrupt has been requested for servicing one line interface.

At the end of a command execution on a line interface, the scanner transfers the status zone related to this line interface into the CCU storage in cycle steal mode, and requests a CCU interrupt level 2. The CCU sends a 'get line identification' to all the scanners (see also "Instruction Operation" on page 13-230).

The interrupt priority mechanism of the scanner ensures that the instruction is accepted by the scanner with the most urgent request. The scanner transfers the address of the PSA related to the line interface that raised the interrupt, then the CCU reads the status for the line in the PSA.

Register R contains the LNVT entry address of the line interface that caused the interrupt.

Immediate Field

Bit	Function
0	0
1-4	0 1 1 0
5-7	0 0 0 (All LABs)
8-11	0 0 0 0 (operation type)
12-14	0 0 0
15	1 (input)

Set LNVT High/Low

This instruction is used to provide a scanner with the LNVT address for its lines. By default, the LNVT starting address is X'880'.

'Set line vector table high' modifies byte X of the LNVT address.

R1 (Set LNVT High)

Bit	Function
0-7	All zeros
8-15	Byte X

R2 (Set LNVT High)

Bit	Function
0	0
1-4	Line group
5-7	LAB address
8-11	0 0 1 0 (operation type)
12-14	0 0 0
15	0 (output)

Bits 1 through 7: (Same as R2 for start line initial).

'Set line vector table low' modifies bytes 0 and 1 of the LNVT address.

R1 (Set LNVT Low)

Bit	Function
0-7	Byte 0
8-15	Byte 1

R2 (Set LNVT Low)

Bit	Function
0	0
1-4	Line group
5-7	LAB address
8-11	0 0 1 1 (operation type)
12-14	0 0 0
15	0 (output)

Bits 1 through 11: (Same as R2 for start line initial).

Microcode Interaction with Control Program (Part 5 of 5)

COMMANDS

Commands are used by the start line and start line initial instructions. The command code is given by bits 0 through 7 of register R1 of the IOH instruction. All commands use the PSA. Some commands need a buffer in addition. Once a scanner has received a command from the control program, the command remains outstanding until the scanner ends the command execution and requests a CCU interrupt level 2. The 'Halt' and 'Halt Immediate' commands may be issued at any time, even if another command is outstanding.

The commands are organized in the following subsets:

Common Commands

Common commands can be issued by the NCP or the EP in normal or character mode. Instructions issued in character mode have the character mode bit (bit 14 of the second halfword) set to 1 in the instruction.

Command	Code (Hex)
Set Mode	01
Enable	02
Disable	03
Monitor Incoming Call	04
Dial (normal mode only)	05
Change	06
Wrap	07
Raise DTR	08
Flush Data	09
Reset-D	0B
Reset-N	0C
Halt	F0
Halt Immediate	F1

NCP Commands

NCP commands can be issued by the NCP on SDLC, BSC, or X.21 lines working in normal mode.

Command	Code (Hex)
SDLC Transmit Control	10
SDLC Transmit Data	11
SDLC Transmit Continue	1D
SDLC Receive Monitor	12
SDLC Receive	13
SDLC Receive Continue	14
X.21 Call Request	15
X.21 Monitor Incoming Call	16
X.21 Clear Request	17
NCP BSC Control	18
NCP BSC Transmit	19
NCP BSC Transmit Continue	1A
NCP BSC Receive	1B
NCP BSC Receive Continue	1C

EP Commands

EP commands can be issued by the EP on BSC lines operating in normal mode.

Command	Code (Hex)
EP BSC Transmit Initial	20
EP BSC Transmit SYN	21
EP BSC Transmit Data	22
EP BSC Poll	23
EP BSC Receive	24
EP BSC Receive Continue	25
EP BSC Prepare	26
EP BSC Monitor for Phase	27

Character Mode Command

Character mode commands can be issued by the NCP or EP on BSC or start-stop lines (both working in character mode).

Command	Code (Hex)
Write ICW	40

Burst Mode Command

Burst mode commands can be issued by the NCP for EP on start-stop lines.

Command	Code (Hex)
Start-Stop Transfer	41

Miscellaneous Commands

These commands are used for all protocols line tracing or modem testing.

Commands F4 and F5 are issued by NCP when a scanner time-out occurs.
F4 issues a BER 11 ID B1, and F5 attempts a line recovery and issues a BER 11 ID A4.

With a modem testing command (2B), the microcode can request the status of local or remote 386x modems, or their attached lines.

For more description, refer to the 386x documentation.

Command	Code (Hex)
386X Modem Test	2B
Trace	2C
Stop Trace	2D
Wrap	2E
Line Status	30
F4	F4
F5	F5

Microcode Interaction with FES (Part 1 of 2)

The scanner microcode operates with the FES hardware using:

- Reserved areas in the control storage
- FES storages (RAM A, RAM B, and RAM C)
- Commands

The asynchronous path through the CSP external registers is used for transferring commands, initializing the lines, and managing the modems and timers.

Cycle stealing is used for the transmission of data, parameters, and status information between the control storage and the FES storages.

The FES interrupts the microcode on level 2 in normal operation (for example, end of message), and for error reporting.

RESERVED SCANNER STORAGE AREAS

FES Parameter/Status

See page 13-554.

Line Interface Buffers

See page 13-555.

FES STORAGES

See page 13-510 for a bit-by-bit description of the fields.

RAM A

RAM A is devoted to the character service function of the scanner base. Four halfwords are assigned to each line interface (receive and transmit).

- Receive:

Hw	Byte 0 (Even address)	Byte 1 (Odd address)
0	New PDF	SCF
1	Old PDF	Interrupt request
2	Control storage addressing	
3	Scanner base cntl. Parameters	Timer

- Transmit:

Hw	Byte 0 (Even address)	Byte 1 (Odd address)
0	Timer	SCF
1	Next PDF	Interrupt request
2	Control storage addressing	
3	Scanner base control (parameters)	

- Parallel Data Field (PDF):

  On the receive line interface, the old PDF and the new PDF contain the next two characters to be loaded into the control storage. On the transmit line interface, the next PDF contains the last character received from the control storage.

- Secondary Control Field (SCF):

  The SCF in RAM A is a duplication of the SCF in RAM B. It is used by the character service function to know the status of the bit service function after character deserialization and serialization.

  The SCF gives the second byte of the status transferred to the CSP at burst end (the first byte is given by the scanner base control).

- Interrupt Request:

  This field reports the interrupt or error conditions that occurred during data reception or transmission. The field contents are transferred to register X'10' to indicate the cause of the interrupt.

- Control Storage Addressing:

  This field contains the address of the line interface buffers in the control storage. It is used during cycle steal transfers between the FES and the CSP.

- Scanner Base Control:

  This field contains the parameters provided by the microcode to the scanner base layer for processing the burst according to the link protocol. Some parameters of the scanner base control are duplicated in the PCF (RAM B).

  At burst end, the scanner base control gives the first byte of the status to be transferred to the CSP (the second byte is given by the SCF).

- Timer:

  The timer fields are used by the BSC protocol:

  - On the receive interfaces, to check that two SYN characters are separated by less than three seconds
  - On the transmit interfaces, to insert a SYN character every second in the transmitted data
  - The timer fields can also be used by the microcode to generate timeouts.

RAM B

RAM B is devoted to the bit service function of the front-end layer. Four halfwords (Hw) are assigned to each line interface (receive and transmit).

- Receive:

Hw	Byte 0 (Even address)	Byte 1 (Odd address)
0	SDF	SCF
1	PDF	PCF
2	Spare	Spare
3	SYN 2	SYN 1

- Transmit:

Hw	Byte 0 (Even address)	Byte 1 (Odd address)
0	SDF	SCF
1	PDF	PCF
2	DLE	Spare
3	SYN	Spare

- Serial Data Field (SDF):

  On the receive line interface, the SDF is used to deserialize the bits received from the line to form a character. On the transmit line interface, the SDF is used to serialize the character into bits for transmission over the line. The work of the SDF is controlled by the SCF.

- Parallel Data Field (PDF):

  On the receive line interface, the PDF contains the last character deserialized from the SDF. On the transmit interface, the next character to be serialized is in the SDF.

- Secondary Control Field (SCF):

  This field is used as a working zone to control the character serialization and deserialization in the SDF. At burst end, it gives the status of the bit service function.

Microcode Interaction with FES (Part 2 of 2)

- Primary Control Field (PCF):

  The PCF is used as a communication zone between RAM A and RAM B, and duplicates some parameters of the scanner base control field. On the receive line interface, the PCF contains the parameters to be used for deserialization. On the transmit line interface, the PCF contains the parameters to be used for serialization.

- Synchronization:

  These fields contain the SYN character(s) to be used during data transmission in BSC.

  On the receive line interface, the SYN character is used as a reference for detection of the SYN character received from the line. Two SYN characters are needed in BSC protocol, one in BSC-like protocols (usually non-standard, and non-IBM BSC protocols).

  On the transmit line interface, the SYN character is stacked for transmission over the line once per second.

- Data Link Escape (DLE):

  This field contains the DLE character used in BSC transmission. Sequences containing the DLE character initiate and terminate the transparent text, and provide an active control character within the transparent text.

RAM C

RAM C is devoted to the line service function of the front-end layer. Four half-words (Hw) are assigned to each line interface (receive and transmit).

- Receive:

	Byte 0 (Even address)	Byte 1 Hw (Odd address)
0	Set mode	Control
1	BCC2 (see note)	BCC1 (see note)
2	Modem-in pattern	Mask
3	Spare	Spare

Note: Not used in start-stop protocol.

- Transmit:

Hw	Byte 0 (Even address)	Byte 1 (Odd address)
0	Set mode	Control
1	BCC2 (see note)	BCC1 (see note)
2	Modem-out immediate	Mask
3	Modem-out stacked	Mask

Note: Not used in start-stop protocol.

- Set Mode:

  This field contains the link protocol parameters specific to the line interface.

- Control:

  This field contains the parameters provided by the microcode to the line service function for processing the burst according to the link protocol.

  In BSC ASCII or EBCDIC, bit 7 of the set mode field is used as an additional bit of the control field.

- Modem-In:

  This field contains the status of the line interface wires activated from the modem. It is compared with the modem-in information coming from the LIC to detect any modem change. The modem-in status depends on the interface type: V.24, V.25, or X.21. The modem change indication (bits 6 and 7) causes an interrupt request to the CSP via register X'10'.

- Modem-Out:

  The modem-out immediate field contains the interface configuration that is supplied to the modem via the LIC registers. The modem-out information depends on the interface type: V.24, V.25, or X.21. Modem-out stacked contains the next modem-out pattern that will be transmitted when bit 15 of the PCF is set on.

  The modem-out field of the RAM is compared with the modem-out echo coming from the LIC to detect any failures in the modem drivers of the LIC (LIC driver check).

- Mask:

  This field is used by the microcode to select the bits of the modem-in or modem-out field that must be checked for modem change or LIC driver check.

- Block Check Character (BCC):

  These fields (BCC2 and BCC1) are used for cyclic redundancy checking in SDLC and BSC protocols. They accumulate the block check characters during transmission or reception. The block check character is reset at the beginning of the block; block check accumulation is then performed until the end of the block or message.

  On a receive line interface:

  - In BSC protocol, the accumulated BCCs are compared with the received BCCs for error detection.
  - In SDLC, the result of the accumulation performed during message reception (including the BCCs) must be X'F0B8'.

On a transmit line interface, the accumulated BCCs are transmitted at the end of the block or message.

COMMANDS

The scanner microcode uses commands to operate the FES. All commands are transferred via the external registers X'13', X'14', X'15', and X'17'.

Receive and Transmit Commands

The microcode sets the FES RAMs for receive and transmit operation through the FES asynchronous path, using external registers X'13', X'14', and X'15', then starts the FES by loading bit 1 of the scanner base control field.

Reset/Freeze Command

This command is transmitted to the FES via external register X'17'. Depending on the setting of the bits in the register, the components of the FES may be reset or frozen (see "Reset" on page 13-021 for details).

Microcode Interaction with MOSS

When the scanner is in the service mode, the scanner microcode operates with the MOSS microcode using:

- Reserved areas in the CCU main storage (mailboxes) as described in Chapter 14
- Reserved areas in the scanner control storage (72 halfwords starting at storage address X'9600')

The scanner interrupts the MOSS on level 4 for normal operation and error reporting. Cycle stealing is used for transferring (via the mailboxes in CCU storage) the data, parameters, and status between the scanner and the MOSS storages.

Control information is exchanged via MOSS I/O instructions (see also Chapter 14).

Control Block Relationship

The figure (right) shows the relationship between the control blocks in the CCU main storage, the CSP control storage, the FES storage, and the LIC registers.

Legend

```
----> Controls
====> Data
FPS   FES parameter/status
ICB   interface control block
LCB   line control block
LIB   line interface buffer
LNVT  line vector table
PCF   primary control field
PDF   parallel data field
PSA   parameter/status area
RAM   random access storage
SCB   scanner control block
SCF   secondary control field
SDF   serial data field
```

DATA TRANSFERS

For a duplex line the control blocks are related as follows:

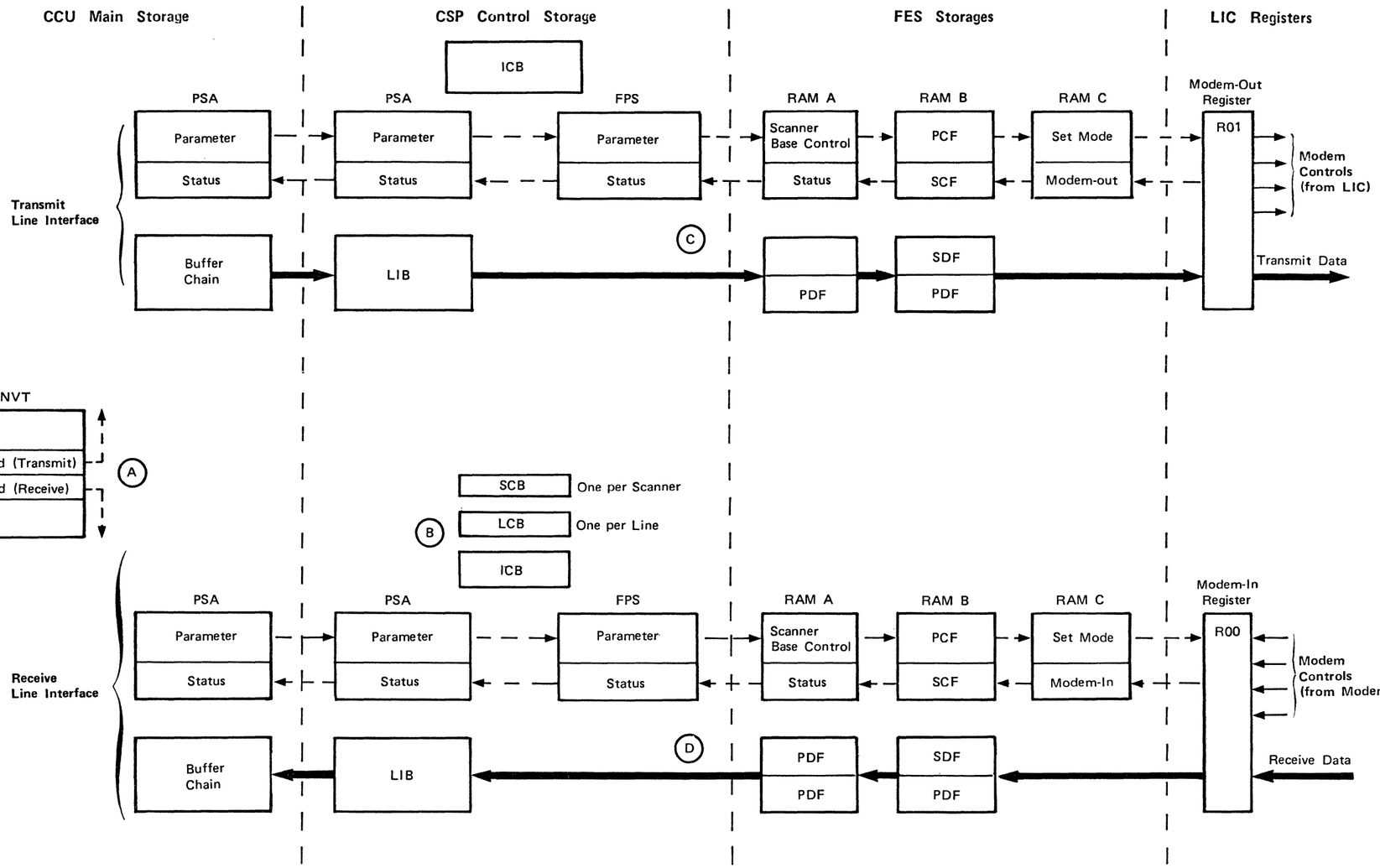
(A) The LNVT addresses two PSAs, one for transmit, the other for receive.

(B) The line parameters given in the PSA are transferred to the line control block (LCB) at scanner initialization.

(C) The transmitted data transits through the parallel data field (PDF) from the CCU transmit buffer to the transmit line interface.

(D) The received data transits through the PDF from the receive line interface to the CCU receive buffer.

For a half-duplex line, one PSA only is addressed and the transmit and receive line interfaces are related to the same PSA (the transmit one).

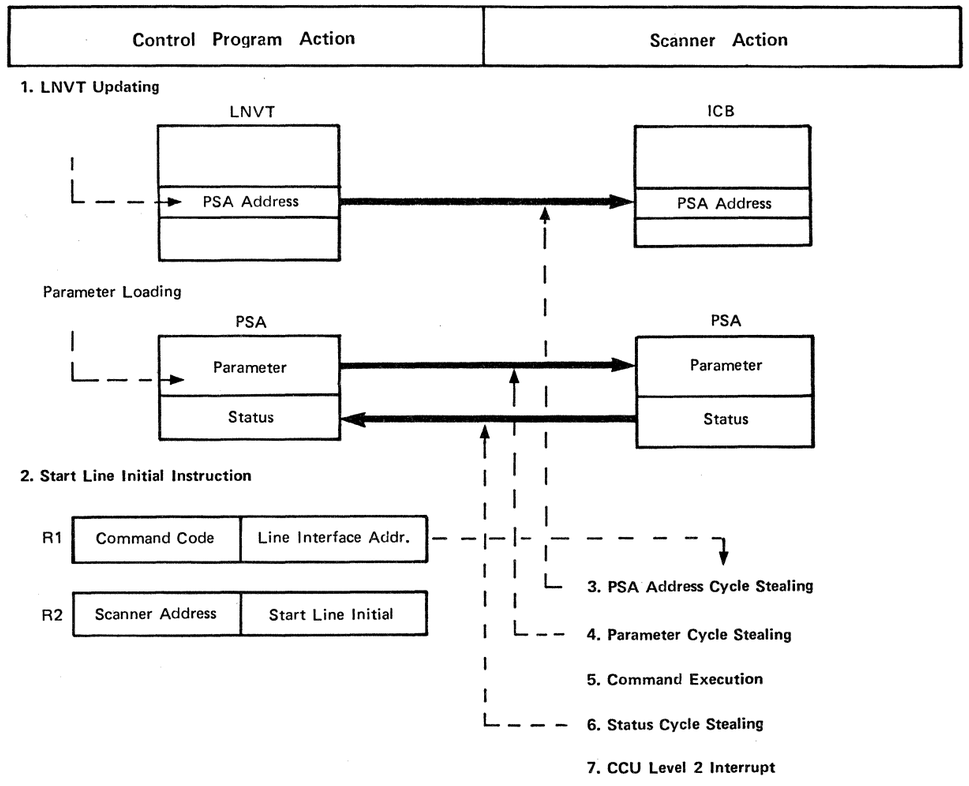


Instruction Operation (Part 1 of 2)

START LINE INITIAL

The operating sequence for the Start Line Initial instruction is as follows:

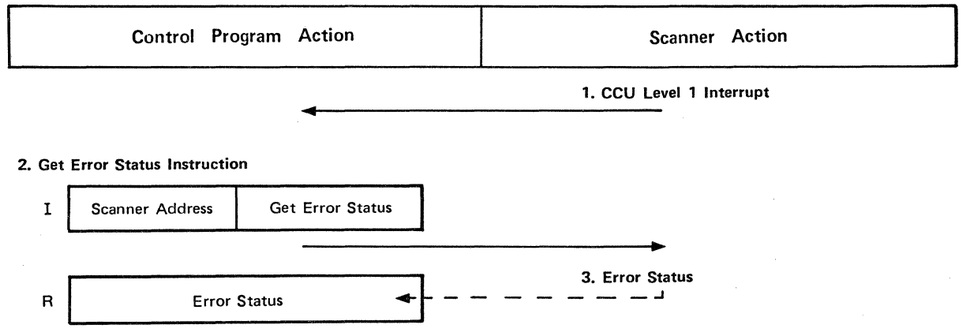
1. The control program updates the PSA address in the LNVT (after IPL or when a new PSA is used), and loads the command parameters in the PSA of the line interface to be addressed.
2. The control program issues a Start Line Initial instruction; the contents of R1 are transmitted via the IOC bus to the scanner, according to the contents of R2.
3. From the line interface address, the scanner identifies the LNVT entry that contains the PSA address, and cycle steals the PSA address into the ICB for the subsequent start line instruction(s).
4. The scanner cycle steals the parameters from the PSA.
5. The scanner executes the command according to the command code and the parameters.
6. When the command has been executed, the scanner prepares the status and cycle steals it into the status area of the PSA.
7. The scanner interrupts the microcode at level 2.



GET ERROR STATUS

The operating sequence for the Get Error Status instruction is as follows:

1. The scanner interrupts the control program at level 1.
2. The control program issues a Get Error Status instruction to the scanner that has requested the interrupt level 1.
3. The scanner loads register R with the error status.



Instruction Operation (Part 2 of 2)

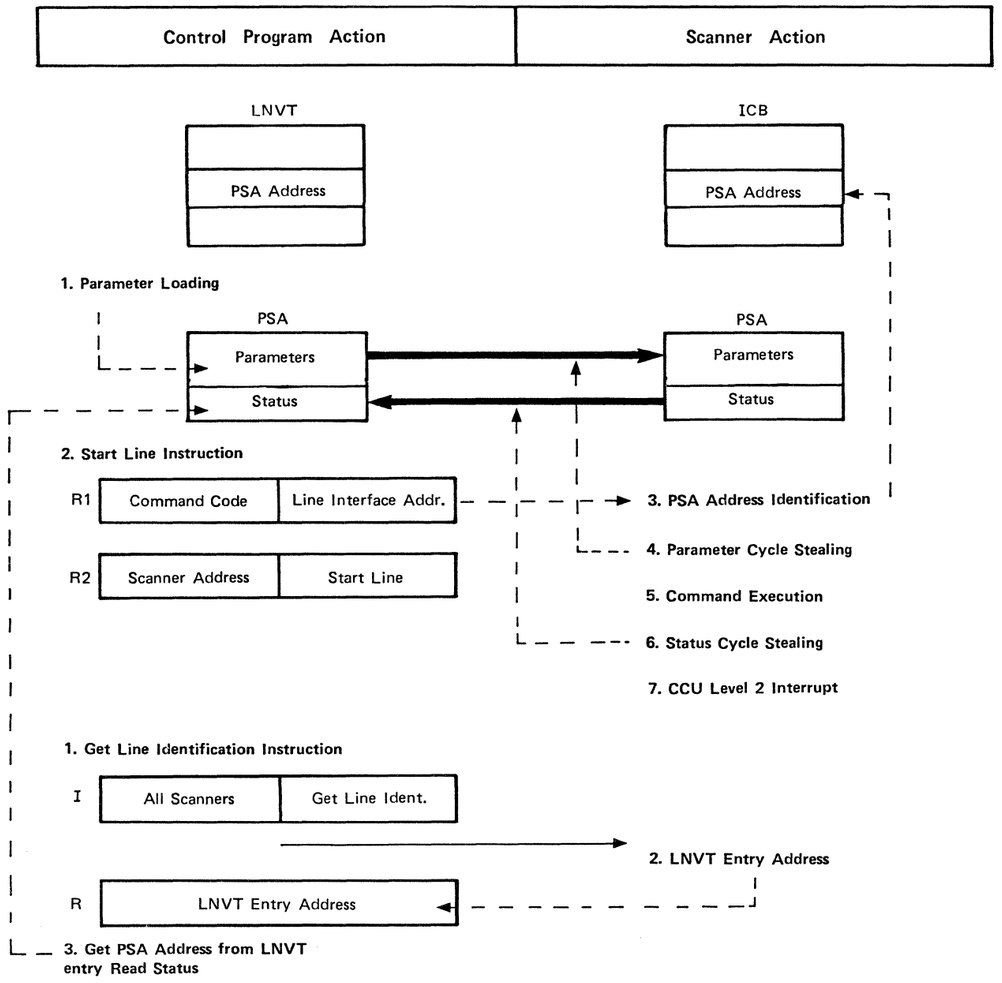
START LINE

The operating sequence for the Start Line and Get Line Identification instructions is as follows:

1. The control program loads the parameters of the command into the PSA of the line interface to be addressed.
2. The control program issues a Start Line instruction; the contents of R1 are transmitted via the IOC bus to the scanner, according to the contents of R2.
3. From the line interface address, the scanner identifies the PSA address in the ICB.
4. The scanner cycle steals the parameters from the PSA.
5. The scanner executes the command according to the command code and the parameters.
6. When the command has been executed, the scanner prepares the status and cycle steals it into the status area of the PSA.
7. The scanner interrupts the microcode at level 2.

GET LINE IDENTIFICATION

1. The control program issues a Get Line Identification to all the scanners.
2. The scanner loads register R with the address of the LNVT entry for the addressed line interface.
3. The control program reads the status area from the PSA.



FES Operation (Part 1 of 2)

PARAMETER TRANSFER

At the beginning of a receive or transmit operation, the microcode sets the parameters to be used by the line interface during the burst. The control storage address to be used for cycle stealing may be transferred at the same time. If not, the address following that last used will be the new address.

The parameters and the control storage address are loaded into RAM A in the scanner base control and control storage address fields respectively. Some parameters are duplicated in the primary control field (PCF) (RAM B).

The remaining parameters are transferred from the FES parameter status area in cycle steal when the receive or transmit operation starts. Each time a cycle steal operation is performed (every two characters), and until the end of the burst, the control storage address is updated by hardware.

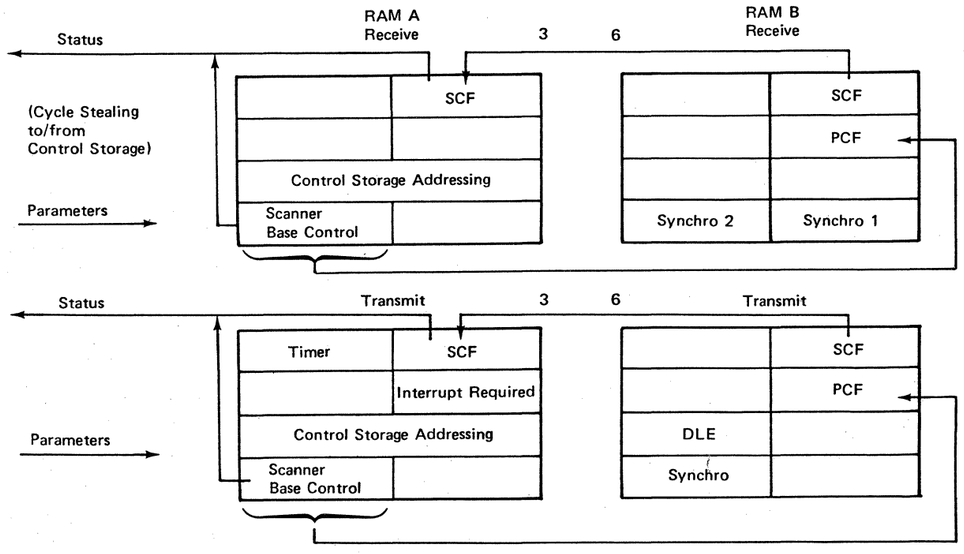
STATUS TRANSFER

At the end of the burst:

- The first byte of the status is given by the scanner base control (SBC) field.
- The second byte of the status is given by the secondary control field (SCF).

The status is loaded by the FES in cycle steal mode into the parameter/status area of the scanner control storage. An interrupt may be requested if required by the microcode through the interrupt request bit in RAM A.

PARAMETER/STATUS TRANSFER MECHANISM



DATA TRANSFERS

Receive Operation

During a receive operation, the data transfer sequence between RAM A and RAM B is as follows:

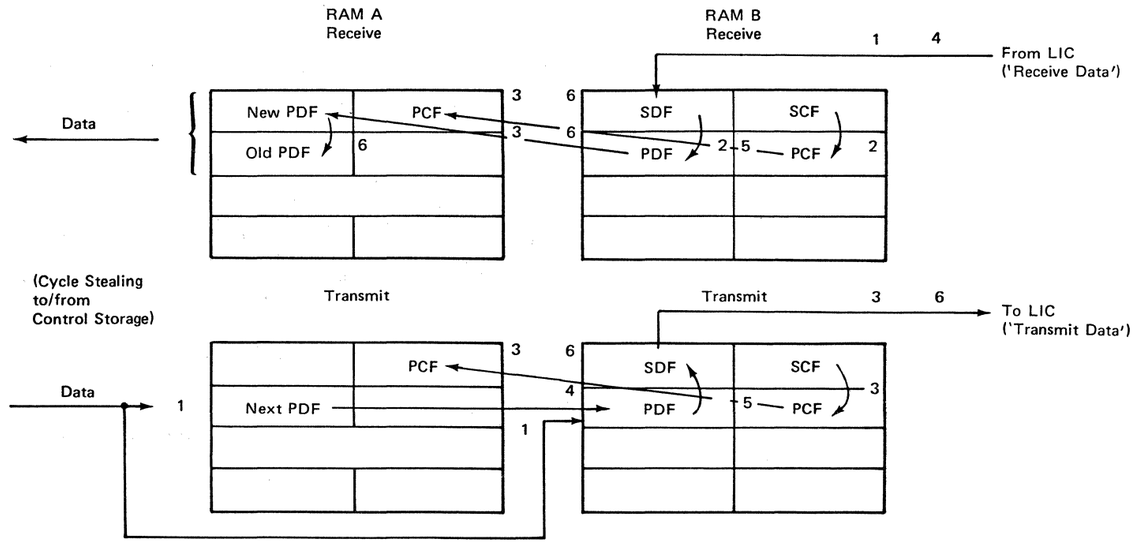
1. The data is received bit-by-bit from 'receive data' via the LIC. The bits are deserialized into an 8-bit character in the SDF under the control of the SCF (RAM B).
2. When the SDF is full, the character is transferred to the PDF. At the same time, the SCF is transferred to the PCF.
3. The character is transferred to the new PDF in RAM A. The PCF (RAM A) is updated with the PCF (RAM B) contents.
4. A second character is received in the SDF and assembled.
5. The second character is transferred to the PDF.
6. The first character is shifted in RAM A from the new PDF to the old PDF. The second character is transferred to the new PDF. The PCF (RAM A) is updated with the PCF (RAM B) contents.
7. Two characters are transferred to the CSP control storage in cycle steal mode according to the control storage addressing field.

Transmit Operation

The transfer sequence is as follows:

1. Two characters are transferred in cycle steal from the control storage. The first one is transferred to the PDF (RAM B), the second one to the next PDF (RAM A).
2. The first character is transferred to the SDF where it is serialized under the control of the SCF (RAM B).
3. The bits are transmitted to 'transmit data' via the LIC. When the SDF has been serialized, the SCF is transferred to the PCF. The PCF (RAM A) is updated with the PCF (RAM B) contents.
4. The second character is transferred from RAM A to RAM B.
5. The second character is transferred to the SDF.
6. The second character is serialized and transmitted to the modem via the LIC. The PCF (RAM A) is updated with the PCF (RAM B) contents.

DATA TRANSFER MECHANISM



FES Operation (Part 2 of 2)

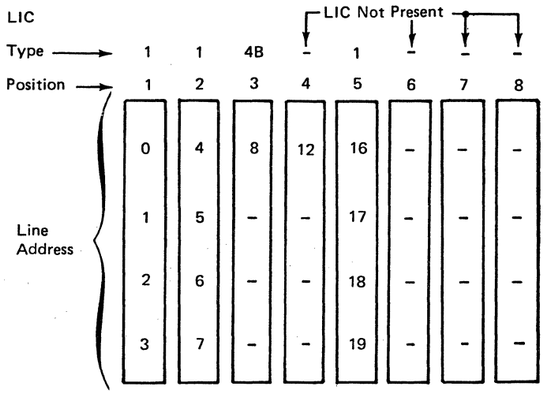
SCANNING OPERATION

The scanning mechanism generates:

- Signals used by the front-end layer to scan bits on the lines
- Signals used by the scanner base layer to scan characters assembled in the front-end layer

Bit scanning and character scanning operate independently. They loop from LIC position 1 (line address 0) up to the last LIC (and the last line) present in the scanner.

Scanner Configuration Example



LIC positions 1, 2, 3, and 5 are present.

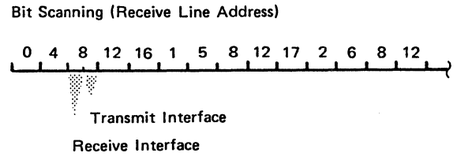
LIC positions 4, 6, 7, and 8 are missing.

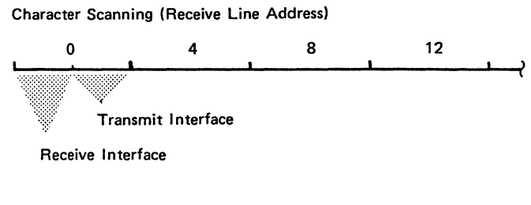
LIC position 4 is scanned as if having one line only as in LIC type 4B, or LIC type 3 with direct-attached terminal. The physical position of the line on the LIC does not affect the scanning.

LIC positions 6, 7, and 8 are not scanned.

Scanning Sequence

In the scanner configuration example, bit and character scanning proceeds as follows: for each line address, the receive interface is scanned first, followed by the transmit interface.





Character scanning is four to eight times slower than bit scanning, depending on the cycle steal activity of the scanner base layer on the scanned interfaces.

MODEM LINE MANAGEMENT

Modem Change Detection

Each time a line (receive interface) is scanned, the information on the leads coming from the modem of that line is compared to the corresponding information stored in the modem-in pattern in RAM C. The information is masked by the mask configuration previously loaded by the microcode into RAM C.

Any change detected raises the line 'modem change', which generates a level 2 interrupt to the CSP. The new modem-in pattern is loaded by hardware into the modem-in field of RAM C. Bits 6 and 7 are set to inhibit further modem-in comparisons until the microcode reads the stored modem-in pattern and resets bits 6 and 7.

LIC Driver Check

Each time a transmit line interface is scanned, the modem-out information previously loaded by the microcode into RAM C is sent to the LIC.

The echo of the modem-out pattern sent by previous scan is compared with the modem-out information in RAM C, taking into account the mask pattern. Any difference raises the line 'driver check', which causes a level 2 interrupt to the CSP.

Bits 6 and 7 of the modem-out field form a 2-bit position counter, which allows a comparison to take place only once every four scans.

Usually, the modem-out pattern sent is the 'modem-out immediate' pattern. At the request of the microcode 'modem-out stacked' is sent instead of 'modem-out immediate', and is loaded into the modem-out immediate field of RAM C.

Error Detection (Part 1 of 2)

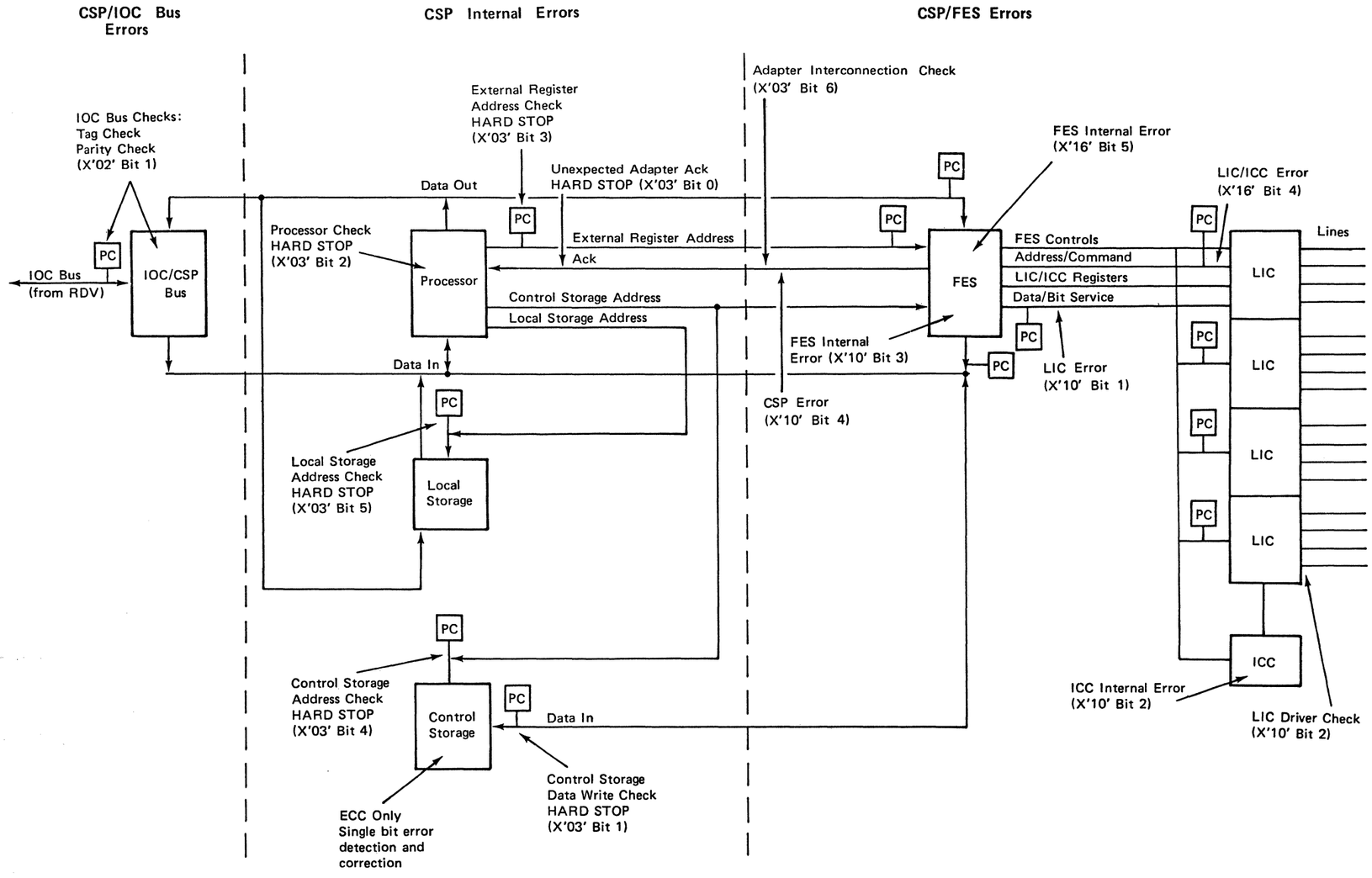
TSS HARDWARE ERRORS: DATA FLOW

This data flow summarizes the errors detected by the TSS hardware and their effect on the external registers. The errors marked 'hardstop' cause a scanner hardstop when they occur. The error information is gathered in two bytes, which are described on page 13-350.

TSS errors are classified as follows:

- CSP/IOC bus errors
- CSP internal errors
- CSP/FES errors

The information on an error occurring in the TSS is gathered in a BER type 11. For BER display and contents analysis, see Chapter 2.



Error Detection (Part 2 of 2)

CSP/IOC BUS ERRORS

CSP/IOC bus errors are detected as follows:

By the CCU or IOC hardware:

- IOC bus check (invalid data) during PIO or AIO
- Timeout during PIO or AIO
- Invalid CSCW during AIO
- Storage protection during AIO
- Address exception during AIO

By the CSP hardware (see the data flow):

- IOC bus check: invalid data
- IOC bus check: invalid address

By the scanner microcode:

- Invalid input PIO from NCP/EP
- Invalid input PIO from MOSS

CSP INTERNAL ERRORS

CSP internal errors are detected as follows:

By the CSP hardware (see the data flow on page 13-320):

- Unexpected adapter acknowledge
- Control storage data write check
- Processor check
- External register address check
- Control storage address check
- Local storage address check

All these errors cause a scanner hardstop.

By the scanner microcode:

- Invalid output PIO from the NCP/EP
- Invalid output PIO from the MOSS
- Invalid CSP interrupt level 0, 1, or 2 request
- Cycle steal stack overflow
- Interrupt level 2 stack overflow
- Invalid cycle steal length
- Command rejected
- Invalid IOH sequence
- Adapter interconnection check with unidentified line

CSP/FES ERRORS

CSP/FES errors are detected as follows:

By the CSP hardware (see the data flow):

- Adapter interconnection check with line identified

By the scanner microcode:

- FES failed to answer
- FES error reporting path check

By the FES hardware (see the data flow on page 13-320):

- FES internal error
- LIC/ICC error

These two errors are reported to the CSP via register X'16' bit 1 without a CSP interrupt. The microcode tests register X'16' bit 1 at the end of each asynchronous operation.

The following five errors are reported to the CSP on a line interface basis using a CSP interrupt level 2 via register X'10' and X'12'.

- FES internal error
- FES/LIC error
- LIC driver check
- ICC internal error
- CSP/FES Error

Reporting Errors to the CCU (Part 1 of 2)

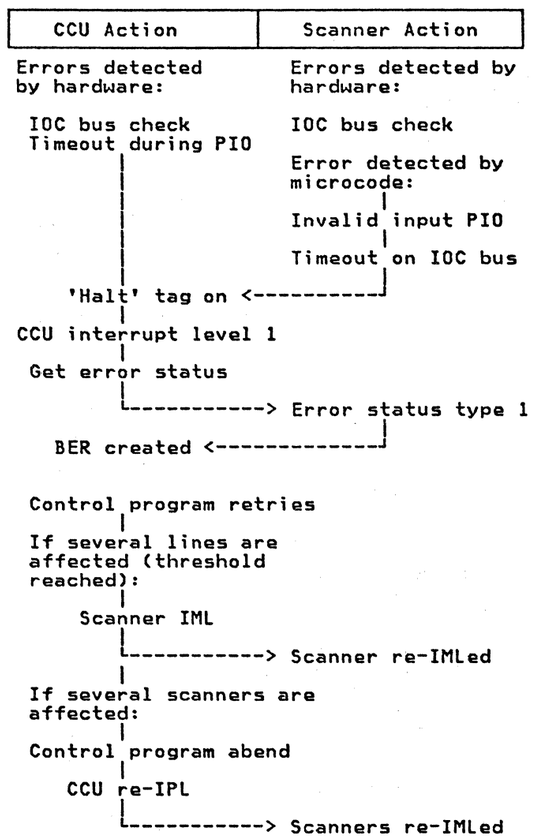
Errors on CCU I/O instructions are reported as follows:

- 3725/3726 errors affecting only one specific line are identified by an error status in the response to the command and are reported to the control program via a CCU interrupt level 2. The line affected is disabled by the NCP or set to no-operating by the EP.
- Errors affecting one scanner are identified in the error registers and are reported to the control program via a CCU interrupt level 1. When these errors affect all the lines connected to a scanner, the scanner is re-IMLed. When several scanners are affected, the control program may abend, and the controller must be re-IPLed.

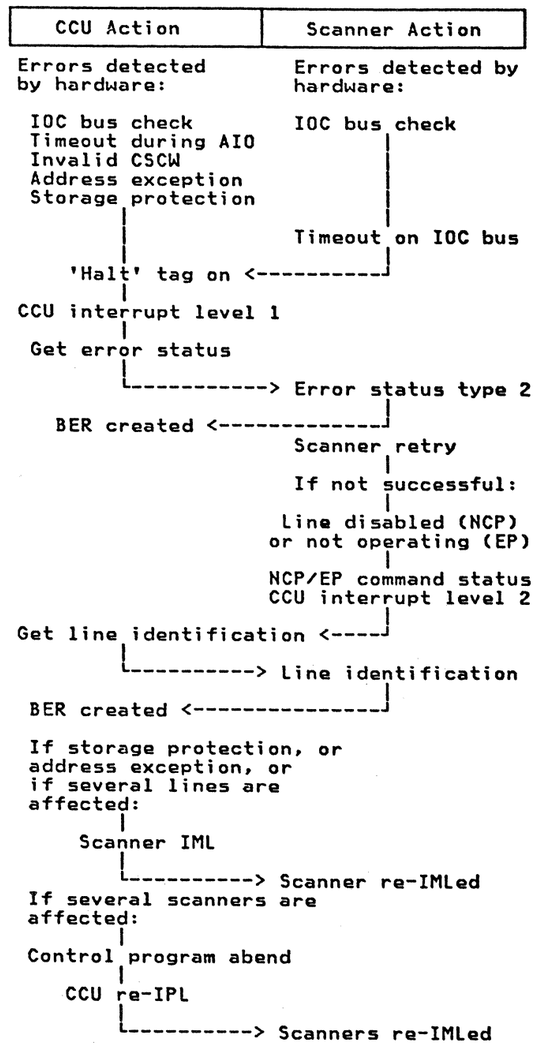
One BER is generated for each TSS error. The scanner must be in the connect mode.

CSP/IOC BUS ERRORS

Errors during PIO



Errors during AIO



Reporting Errors to the CCU (Part 2 of 2)

CSP INTERNAL ERRORS

Detected by Hardware

CCU Action	Scanner Action

Errors detected by hardware:

Unexpected adapter ack
CSP storage check
Processor check
External register
Address check
Local storage check

Hardstop

CCU interrupt level 1

Get error status <----------

----> Error status type hardstop

BER created <----------

Scanner IML

----------> Scanner re-IMLed

Detected by Microcode

CCU Action	Scanner Action

Errors detected by scanner microcode:

Invalid output PIO
Invalid CSP interrupt (lvl 0,2)
Disconnect mode
NCP buffer validity check
Cycle steal stack overflow
Interrupt level 2 stack overflow
Invalid cycle steal length
Invalid IOH sequence
force scanner dump condition
Command rejected or adapter interconnection check with line not identified

CCU interrupt level 1

Get error status <----------

----------> Error status type 3

BER created <----------

If stack overflow or invalid interrupt request:

Scanner IML

----------> Scanner re-IMLed

If invalid output PIO for all scanners:

Control program abend

CCU re-IPL

----------> Scanners re-IMLed

If command rejected with line identified:

Line disabled (NCP) or not operating (EP)

NCP/EP command status
CCU interrupt level 2

Get line identification <-----

----------> Line identification

BER created <----------

CSP/FES ERRORS

CCU Action	Scanner Action

Errors detected by hardware:

Adapter interconnection check with line identified with line identified

Errors detected by microcode:

FES failing to answer
FES error reporting path check

Errors detected by FES:

FES internal error
LIC/ICC error
LIC error
LIC driver check
ICC internal error

Line disabled (NCP) or not operational (EP)

NCP/EP command status
CCU interrupt level 2

Get line identification <-----

----------> Line identification

BER created <----------

Reporting Errors to the MOSS

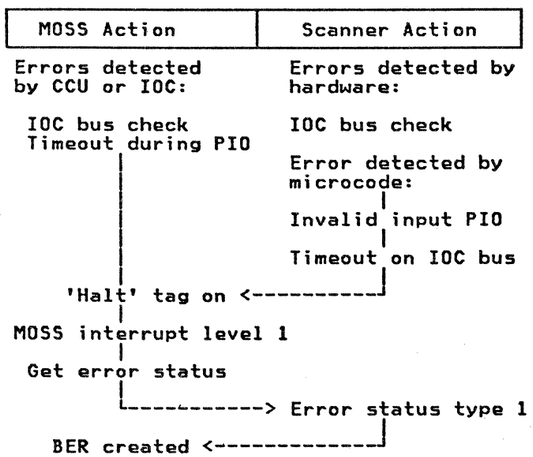
Errors on MOSS I/O instructions are reported as follows:

- Errors affecting only one specific line are identified by an error status in the response to the command, and are reported to the MOSS via a MOSS interrupt level 4.
- Errors affecting only one scanner are identified in error registers, and are reported to the MOSS via a MOSS interrupt level 1.

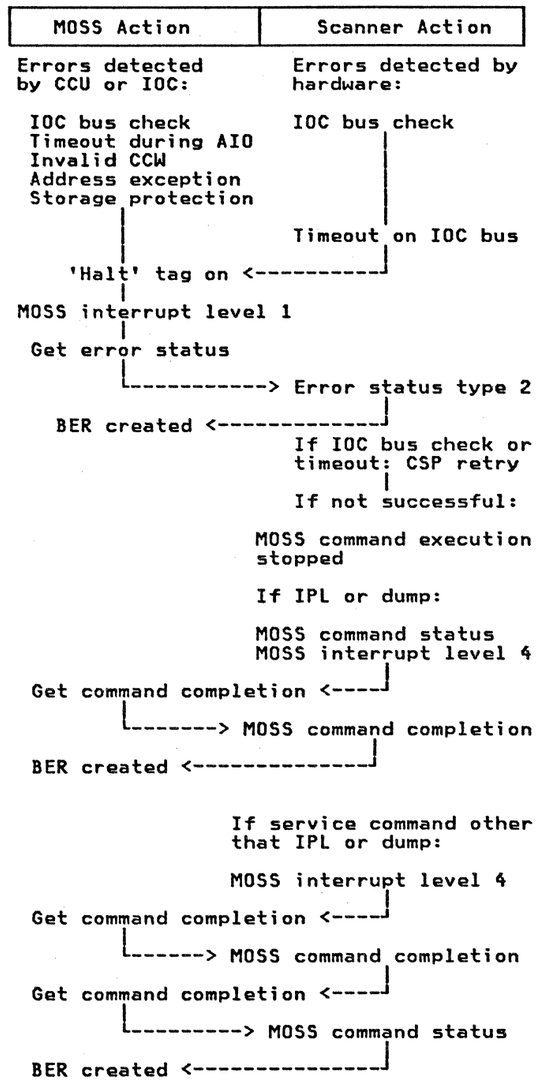
One BER is generated for each TSS error. The scanner may be in connect or in disconnect mode.

CSP/IOC BUS ERRORS

Errors during PIO

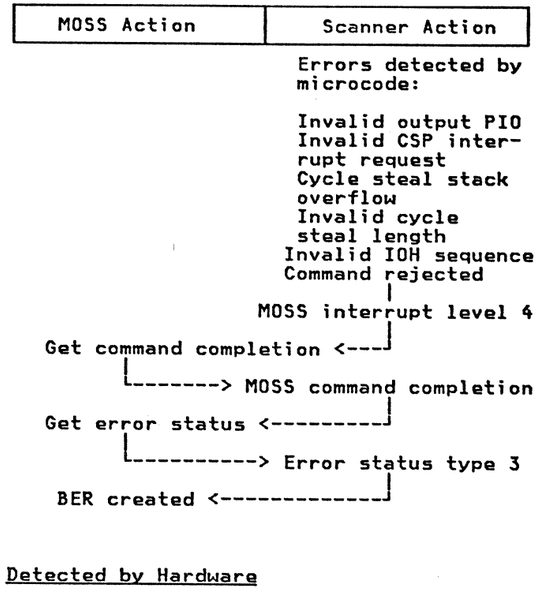


Errors during AIO

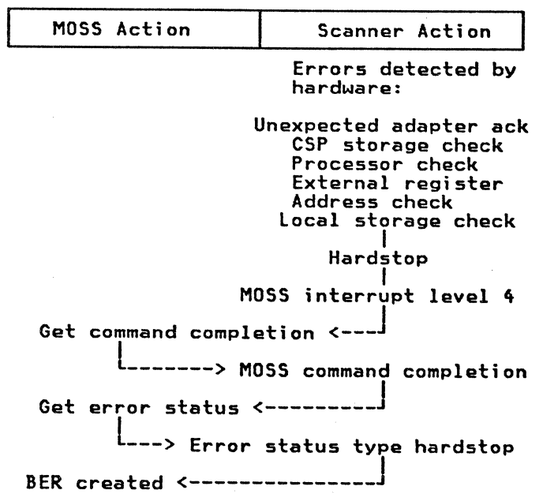


CSP INTERNAL ERRORS

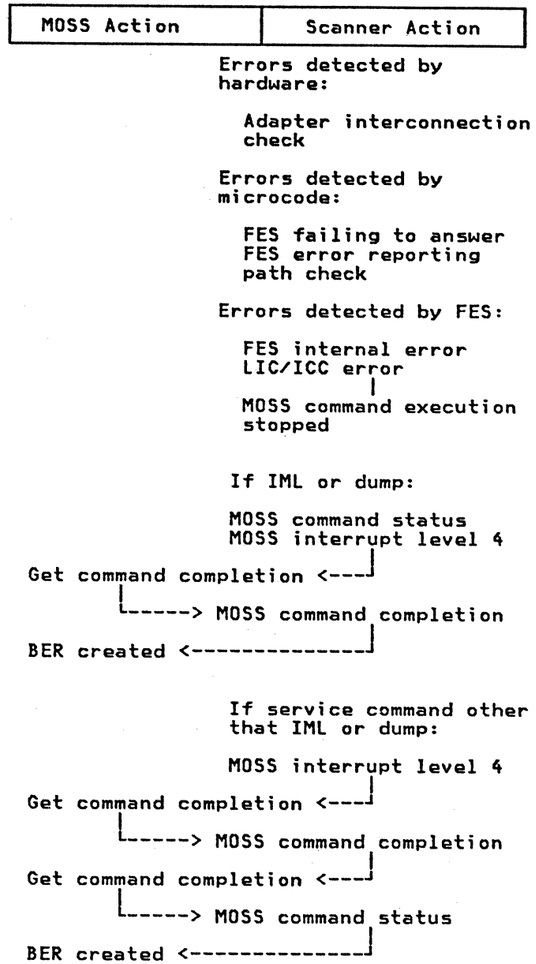
Detected by Microcode



Detected by Hardware



CSP/FES ERRORS



Error Status Description (Part 1 of 3)

To display the scanner external registers covered by error status type 1, refer to TSS functions on page 2-370.

ERROR STATUS TYPE 1 (TWO BYTES)

Error status type 1 is built by the scanner microcode when a halt signal is received from the CCU via the IOC bus during a PIO operation. The PIO may be issued from the CCU or from the MOSS.

The BER created is type 11, with ID 18, 1B, 97, 98, or 9C.

Error status type 1 is reported on CCU or MOSS interrupt level 1 by the CCU or IOC hardware.

Byte 0

Bit	Function
0	Read/write (X'02' bit 0)
1	IOC bus check (X'02' bit 1)
2	(Not used)
3	I/O tag (X'02' bit 3)
4	Halt (X'02' bit 4)
5	TA tag (X'02' bit 5)
6	TD tag (X'02' bit 6)
7	(Not used)

Byte 1

Bit	Function
0	TA time select (X'00' bit 0)
1	(Not used)
2	LAB type (X'07' bit 5)
3	Disconnect mode (X'01' bit 5)
4	(Not used)
5	(Not used)
6	(not used)
7	Invalid input IOH

Bits 6 and 7: Set by the scanner microcode.

ERROR STATUS TYPE 2 (TWO BYTES)

Error status type 2 is built by the scanner microcode when a halt signal is received from the CCU via the IOC bus during an AIO operation. The AIO may be issued from the control program or the MOSS.

The BER created is type 11, with ID 14, 16, 91, 92, or 93.

Error status type 2 is reported on CCU or MOSS interrupt level 1 requested by the CCU or IOC hardware.

Byte 0

Bit	Function
0	Read/write (X'02' bit 0)
1	IOC bus check (X'02' bit 1)
2	Cycle steal grant (X'02' bit 2)
3	I/O tag (X'02' bit 3)
4	Halt (X'02' bit 4)
5	(Not used)
6	TD tag (X'02' bit 6)
7	(Not used)

Byte 1: CCU

Bit	Function
0	(Not used)
1	Cycle steal select (X'00' bit 1)
2	LAB type (X'07' bit 5)
3-7	Line interface address bits 0-4

Byte 1: MOSS

Bit	Function
0	(Not used)
1	Cycle steal select (X'00' bit 1)
2	LAB type (X'07' bit 5)
3	(Not used)
4-7	Last IOH TD byte 0 bits 0-3

ERROR STATUS TYPE 3 (TWO BYTES)

Error status type 3 is built by the scanner microcode when an invalid output PIO, an invalid interrupt request, or an invalid IOH sequence is detected. The output PIO may be issued from the control program or the MOSS.

The BER created is type 11, with ID 1E, 1F, 99, 9A, or 9B.

Error status type 3 is reported on CCU interrupt level 1 or MOSS interrupt level 4 requested by the scanner microcode.

Byte 0

Bit	Function
0	Error status type 3 (always on)
1-3	Invalid interrupt levels 0-2
4	Invalid cycle steal length
5	Cycle steal/CCU level 2 stack overflow
6	Cmd 2 while Cmd 1 in process
7	Disconnect mode

Bit 1: Set to indicate an invalid CSP interrupt level 0 or an unexpected branch to address '0000' at any level.

Bit 7: Set when the disconnect mode is entered via a service command.

Byte 0 bits 1 and 7 both on: CSP has detected an NCP buffer validity check.

Byte 0 bit 7, and Byte 1 bit 1 both on: CSP received an "F2" output IOH from NCP and has entered a disconnect mode (forces a CSP dump).

Byte 1:

Bit 0 = 0: command not rejected

Bit	Function
1	Invalid output IOH
2	LAB type (X'07' bit 5)
3-6	Not used
7	Adapter interface check

Bit 7: Set when errors are detected during an asynchronous operation to the FES.
Bit 0 = 1: command rejected

Bit	Function
1	Trace or line command
2-7	Line interface address (line command)
4-7	Trace slot number (trace command)

Bit 0: Set when:

- A command is in process and another command is received from the control program for the addressed line.
- A Start Line Initial was not the first command received for the addressed line (PIO command at TA time).
- A Set Mode command (PIO command at TD time) was not the first command received during the Start Line Initial for the addressed line.
- A MOSS mailbox exchange was initiated, but no mailbox was available.

Bit 1: Set when a trace command is rejected, reset when a line command is rejected.

Bits 2-7: Give the line interface address (00-3F), when a line command is rejected.

Bits 4-7: Give the slot number, when a trace command is rejected. Bits 2-3 are 00.

Error Status Description (Part 2 of 3)

ERROR STATUS TYPE HARDSTOP (TWO BYTES)

An error status type hardstop is returned by the CSP hardware when an error causing a hardstop has been detected.

The BER created is type 11, with ID 95.

It is reported on CCU interrupt level 1 or MOSS interrupt level 4 requested by the CSP hardware.

Byte 0

Bit	Function
0	Error status type hardstop (always off)
1-4	Set to 0
5	Control store data check
6	Local storage/external register parity check
7	Internal check

Bits 5 through 7: Present only if byte 1 bit 2 is set on (CSP processor check). All zeros in bits 5 through 7 identify a hardstop forced by the scanner microcode.

Byte 1

Bit	Function
0	Unexpected adapter check (X'03' bit 0)
1	CSP storage write data check (X'03' bit 1)
2	Processor check (X'03' bit 2)
3	External register address check (X'03' bit 3)
4	CSP storage address check (X'03' bit 4)
5	Local storage address check (X'03' bit 5)
6-7	(Not used)

NCP/EP COMMAND STATUS

The NCP/EP command status is transferred to the control program when a CSP/FES error or unsuccessful scanner microcode retries have caused the microcode to terminate a pending command.

The BER created is type 11, with ID A1, A2, or A4.

The NCP/EP command status is reported via CCU interrupt level 2 or MOSS interrupt level 4.

Command Status (except Character and Burst Modes)

SCF (X'0x')	CCMD
SES (X'00')	LCS
R count	
Last buffer used	

Character and Burst Modes Command Status

SCF (X'0x')	PDF
	LCS
LCD/PCF	SDF
Modem-in	Modem-out

Line Communication Status (LCS)

Bits 0-2 = 100: Special Status

Bits 0-7 (Hex)	Function
80	Timeout (nothing received), or X'21' timeout on ready for data
81	X'21' timeout on ready for data and timeout during clear
82	End of reception
86	386X test control active, or X'21' timeout on proceed to select
87	X'21' timeout on proceed to select and timeout during clear
88	DLE-EOT disconnect sequence
8A	Lost data
9A	X'21' call proceed signal error
9B	X'21' call proceed signal error and timeout during clear
9C	Disconnected
9E	Connected
A0	TI on during LPDA2 command

Bits 0-2 = 110: Internal Box Errors

Bits 0-7 (Hex)	Function
C0	Error already recorded in BER (AIO error)
C2	Adapter Interface check
C4	CSP/FES error
C6	FES failing to answer
C8	FES internal error
CA	LIC driver check/ICC internal error
CC	LIC interface error
CE	LIC/ICC error
D0	No interrupt from FES
D2	Command rejected
D4	Trace already active on this interface
D6	FES error reporting path check
D8	Invalid level 2 interrupt
DA	Modem already in test mode
DC	Internal clock failure

Bits 0-2 = 111: Hardware Error Status

Bits 0-7 (Hex)	Function
E2	CTS dropped, modem retrain
E6	RLSD failed to drop
EE	DSR dropped
EE	External clock failure for BER 11 ID A4
F2	TI/CTS failed to come up
F4	DSR failed to come up
F6	Cable not installed
F8	DSR/CTS/TI failed to drop
FA	X'21' disconnected after clear signal received
FB	X'21' disconnected after clear signal received, and timeout during clear
FC	Autocall check

See NCP/EP Handbook for more details about LCS.

Error Status Description (Part 3 of 3)

MOSS COMMAND COMPLETION (TWO BYTES)

The MOSS command completion is transferred to the MOSS to identify the scanner that requested the MOSS interrupt level 4.

The BER created is type 01, with ID 05.

Byte 0

Bit	Function
0-7	Current MOSS command

Byte 1

Bit	Function
0	Scanner request
1	Command failed (MOSS command status follows)
2	Error during status transfer (AIO error)
3	Error status type 3 available (get error status must be used)
4	Address compare hit on cycle steal operation
5-7	Snapshot entry number

MOSS COMMAND STATUS (TWO BYTES)

If an error occurs during the processing of a MOSS service, the MOSS command status is transferred to the MOSS, in addition to the usual MOSS completion. Byte 1 bit 1 of the MOSS command completion is set on to indicate that the command failed, and that a MOSS command status is present.

The BER created is type 01, with ID 05.

Byte 0

Bit	Function
0-7	Current MOSS command

Byte 1

Bit	Function
0	AIO error
1	FES failing to answer
2	LIC/ICC error (X'16' bit 4)
3	FES internal error (X'16' bit 5)
4	FES error reporting path check (X'17' bit 4)
5	Adapter interface check (X'03' bit 6)
6-7	(Not used)

Bits 0 and 1: Set by the microcode.

Local Storage Register Description

	Page→V	0	1	2	3	4	5	6	7
Level 0	0	0	1	2	3	4	5	6	7
	1	8	9	A	B	C	D	E	F
Level 1	2	CHCW		CCU address 1/2		CCU address 2/2		CSP address	
	3	Ping Addressing		Pong		L1 cycle steal routine address		BALR to cycle steal routines	
Level 1	4								CSP'chk result
	5					CCU level 2 control block First on stack		Next available	
Level 2	6			ICB Pointer					
	7								
Level 3	8			ICB Pointer					Request Reg.
	9								
Level 7	A								
	B								
PSW Area	C	PSW 0 X'9B20'				PSW 1 X'A000'			
	D	PSW 2 X'A300'				PSW 3 X'A700'			
	E	Cycle steal control block First on stack		Next available		Command queue control block First on queue		Last on queue	
	F	ROS work registers Disconnect work register				PSW 7 X'Cxxx'			

To locate and display local store registers from the storage, refer to "TSS Functions, DPLY/ALT LSR" in page 2-376. The local store registers are at the head of a storage dump.

External Register Description (Part 1 of 5)

Refer to page 13-011 for the organization of the external registers, and to Chapter 11 for a description of the IOC bus lines.

X'00': IOC BUS CONTROL 1

This register indicates to the scanner microcode the type of exchange that is performed with the CCU: PIO ('TA select' on) or AIO ('cycle steal select' on). It also controls the operation of the pong buffer. The bits are not checked for parity. They have the following meaning:

Bit	Function
0	TA time select
1	Cycle steal select
2	(Not used)
3	Channel address valid halfword
4	ETX, ETB, ENQ (ASCII)
5	ETX, ETB, ENQ (EBCDIC)
6	Pong buffer busy
7	Pong buffer end-of-chain

Bit 0: Set at TA time of a PIO read/write operation when the scanner has decoded its address and no parity error has been detected. It is reset by the microcode.

Bit 1: Set when the scanner has detected the 'cycle steal grant' tag on. It is reset by the microcode.

Bit 3: Automatically set on when the microcode sets on the extended channel address valid halfword (external register X'01' bit 4). It is reset at the same time as the pong buffer busy bit.

Bit 4: Set when an ETX, ETB, and ENQ ASCII character is decoded in the ping or pong buffer during a data transfer from the IOC bus to the scanner. It is reset at the beginning of the channel grant transfer.

Bit 5: Same as for bit 4 for EBCDIC characters.

Bit 6: Set by hardware to indicate to the microcode that the pong buffer is busy. It is automatically reset when the valid halfword or end-of-chain tag drops.

Bit 7: Set when the last data halfword is to be received into or sent from the pong buffer. It is automatically reset when the end-of-chain has been sent to the CCU, or by a PIO selection (TA time).

X'01': IOC BUS CONTROL 2

This register controls the cycle steal operation on the IOC bus and the ping buffer operation. The bits are not checked for parity. They have the following meaning:

Bit	Function
0	(Not used)
1	Channel request ready
2	Channel request
3	Ping buffer selected
4	Extended channel address VH
5	Disconnect mode (PIO disabled)
6	Ping buffer busy
7	Ping buffer end-of-chain

Bit 1: Set by the microcode after the cycle steal control word and cycle steal address have been loaded into the ping and pong buffers. It is reset by the cycle steal grant tag when cycle stealing starts.

Bit 2: Set and reset at the same time as bit 1. It is used to define the priority level associated with the request (see also external register X'05' bit 5).

Bit 3: Set when the ping buffer is connected to the IOC bus. It is reset when the pong buffer is connected.

Bit 4: Set by the microcode when more than 16 bits of address are required. It is reset at the same time as the ping buffer busy bit.

Bit 5: Set and reset by the microcode. When on, this bit prevents any PIO operation from the CCU and level 1 or 2 interrupt requests from the scanner (disconnect mode). PIO operations initiated from the MOSS are, however, still accepted. When set off, this bit indicates connect mode.

Bit 6: Set by hardware to indicate to the microcode that the ping buffer is busy. It is automatically reset when the valid halfword or end-of-chain tag drops.

Bit 7: Set when the last data halfword is to be received into or sent from the ping buffer. It is automatically reset when the end-of-chain has been sent to the CCU, or by a PIO selection (TA time).

X'02': IOC BUS SERVICE

This register reflects the status of the IOC bus service tags to the microcode, and indicates the IOC bus check when parity errors are detected on the data. The bits are not checked for parity. They have the following meaning:

Bit	Function
0	Read/write (input/output IOH)
1	IOC bus check
2	Cycle steal grant
3	I/O tag
4	Halt
5	TA tag
6	TD tag
7	Interrupt priority check

Bit 0: Reflects the status of the read/write bit (IOC bus byte 1 bit 7) of a PIO or cycle steal operation: bit 0 = 0 indicates a write operation; bit 0 = 1 indicates a read operation.

Bit 1: Set when the TA and TD tags are active at the same time, or a data parity check occurs. It is reset by the microcode at the end of the level 0 routine.

Bit 2: Used by the microcode to read the status of the cycle steal grant on the IOC bus.

Bit 3: Used by the microcode to read the status of the I/O tag on the IOC bus.

Bit 4: Set when the halt tag is active. When the bit is on, the IOC bus status is frozen, a level 0 interrupt to the CSP occurs, and the IOC tag drivers are disabled. It is reset by the microcode.

Bit 5: Used by the microcode to read the status of the TA tag on the IOC bus.

Bit 6: Used by the microcode to read the status of the TD tag on the IOC bus.

Bit 7: Set when the priority latch (on the redrive card), the disable IOC bus latch, the halt latch, or the reset latch is set. No level 0 interrupt is requested.

During diagnostics, this register is isolated from the IOC bus. All bits except bit 7 can be set and reset by the microcode. Bit 4 means TD tag.

X'03': CSP ERROR

This register reports the CSP errors. The bits are not checked for parity. They have the following meaning:

Bit	Function
0	Unexpected adapter acknowledge
1	Control storage data write check
2	Processor check
3	External register address check
4	Control storage address check
5	Local storage address check
6	Adapter interconnection check
7	External adapter check

Bit 0: Set when an acknowledge signal is erroneously received from the FES while one external register in the CSP address range is accessed. It causes a hardstop in the CSP. It is reset by the microcode.

Bit 1: Set when a parity check is detected during a control storage write operation. It causes a hardstop in the CSP. It is reset by the microcode.

Bit 2: Set when the error line is active in the CSP. It causes a CSP hardstop and must be reset by the microcode. This bit sets LED 4 on the CELIA card.

Bit 3: Set when a parity check is detected on the external register address bus. It causes a hardstop and is reset by the microcode.

Bit 4: Set when a parity check is detected in the control storage address bus. It causes a hardstop and is reset by the microcode.

Bit 5: Set when a parity check is detected in the local storage address bus. It causes a hardstop and is reset by the microcode.

Bit 6: Set when the acknowledge signal from the FES is not received. It causes a CSP level 0 interrupt and is reset by the microcode.

Bit 7: Set when a CSP interrupt level 0 is requested by the FES. It causes a level 0 interrupt and is reset by the microcode.

External Register Description (Part 2 of 5)

X'04': MISCELLANEOUS/ADAPTER ADDRESS

This register provides the microcode with miscellaneous IOC bus controls. The bits are not checked for parity. They have the following meaning:

Bit	Function
0	Timer pulse read
1	Programmed reset
2	Latch reset and FES reset
3	Disable hardstop
4	Valid byte
5	IOC bus disable
6	Adapter address
7	Cycle steal request

Bit 0: Reflects the status of the 100 ms timer pulse.

Bit 1: Set when a programmed reset command from the CCU is decoded. It is reset by the microcode.

Bit 2: Set when the reset tag from the redrive card is active. When on, it disables the IOC bus and the FES. It is reset by the microcode.

Bit 3: Set by a programmed reset command or a general reset from the CCU. It disables the CSP hardstop, and is reset by the microcode.

Bit 4: Indicates that the last data transferred is a byte. It is reset at the same time as the ping/pong end-of-chain or at the beginning of a PIO operation.

Bit 5: Set at power on reset or by the microcode. When it is on, the IOC bus is disabled and external register X'02' is isolated from the IOC bus. It is reset by the microcode.

Bit 6: Used by the microcode to address the FES.

Bit 7: Reflects the status of the cycle steal request at the IOC bus.

X'05': EXTERNAL INTERRUPT REQUEST/PRIORITY

This register controls the MOSS and CCU interrupts requested by the CSP. The CCU interrupts are reported via the data bytes of the IOC bus. The bits are not checked for parity. They have the following meaning:

Bit	Function
0	MOSS interrupt request (level 4)
1	CCU level 1 interrupt request
2	CCU level 2 interrupt request
3	Level 1 interrupt wrap read
4	Level 2 interrupt wrap read
5	Level 2 and cycle steal priority
6	Diagnostic priority compare
7	(Not used)

Bit 0: Used by the CSP microcode to request an interrupt to the MOSS. It is reset by the CSP microcode.

Bit 1: Causes a level 1 interrupt to the CCU. It is controlled by the CSP microcode.

Bit 2: Causes a level 2 interrupt to the CCU. It is controlled by the CSP microcode.

Bit 3: Gives the state of the level 1 interrupt to the CCU. It can only be read by the CSP microcode.

Bit 4: Gives the state of the level 2 interrupt to the CCU. It can only be read by the CSP microcode.

Bit 5: Set by the microcode to indicate a high priority level associated with the level 2 or cycle steal request.

Bit 6: Reflects the output of the level 2/cycle steal priority compare circuit.

X'07': SYNCHRO/CONFIGURATION DATA SET

This register can be used for clock check. It also provides the scanner configuration to the microcode. The bits are not checked for parity. They have the following meaning:

Bit	Function
0	Channel request bid
1	100-ms interval timer
2	Sync 1
3	Sync 2
4	Not used
5	LAB type
6	Configuration data set 1
7	Configuration data set 2

Bit 0: Set by the microcode to prevent any PIO operation at the beginning of a cycle stealing operation. It is reset by the microcode when setting on bit 1.

Bit 1: Set by the 100-ms timer pulse from the CCU. It is reset by the microcode.

Bit 2: Set by the microcode and causes a 200-ns pulse to be sent on the CSP2 card (board connector side). This pulse is used for maintenance purposes by the diagnostics (see Chapter 2).

Bit 3: Same function as bit 2.

Bit 5: When on, indicates a LAB type A; when off, indicates a LAB type B.

Bits 6 and 7: Provide the encoded scanner configuration to the microcode.

X'08': ERROR INDICATORS/BAD PARITY GENERATOR

This register is used by the microcode to:

1. In write, set on three error indicators of the CELIA card (LED 1, 2, and 3 related to bits 5, 6, and 7 respectively). Bits 5 to 7 carry error codes for scanner troubleshooting using the MIM part 2.

   For description and use of the CELIA card, see page 3-011.

   The bits are checked for parity. Bits 5 through 7 are disabled when the disable hardstop bit (X'04' bit 3) is on.

Bit	Function
0	(Not used)
1	(Not used)
2	(Not used)
3	(Not used)
4	(Not used)
5	1, set on CELIA LED 1
6	1, set on CELIA LED 2
7	1, set on CELIA LED 3

2. In read, force a bad parity by inverting the parity bit that corresponds to the bit pattern written in the register. The hardstop bit is disabled previously.

Bit	Function
0	Bit Pattern 0
1	Bit Pattern 1
2	Bit Pattern 2
3	Bit Pattern 3
4	Bit Pattern 4
5	Bit Pattern 5
6	Bit Pattern 6
7	Bit Pattern 7

External Register Description (Part 3 of 5)

X'10': EXTENDED INTERRUPT REQUEST

This register reports FES interrupts when X'12' bits 0 and 1 are on, and FES errors when X'10' bit 0 is on. The CSP can only read this register. The bits are not checked for parity if bit 0 is on. They have the following meaning:

Bit 0 = 0: Various conditions

Bit	Function
1	Overrun or underrun
2	Time out (transmit)
3	Modem change
4	Ending configuration/transparent ending/end of transmission
5-7	(Not used)

Bit 0 = 1: Errors

Bit	Function
1	FES/LIC error
2	LIC driver check/ICC internal check
3	FES internal error
4	CSP/FES error
5-7	(Not used)

Bit 0: Set by hardware to indicate FES/LIC or internal errors, or LIC/ICC errors on bits 1 through 4. When off, it indicates overrun/underrun, timeout, or modem change on bits 1 through 3.

Bit 2: When bit 0 is set, bit 2 indicates a LIC driver check for a transmit line interface, or an ICC internal error for a receive line interface. It is reset by the microcode.

Bit 1: on, to report underrun.

Bit 1: off, to signal transmission end.

Bit 4: Set by hardware to indicate an ending configuration (end bits 1 through 3) in the burst status on receive, or to report a transparency ending condition on transmit.

X'12': INTERRUPT REQUEST

This register indicates that X'10' contains supplementary interrupt information; bits 2 through 7 contain the address of the line interface that requested a CSP interrupt. The bits are checked for parity. They have the following meaning:

Bit	Function
0-1	Data/read X'10'
2	Line interface address bit 0
3	Line interface address bit 1
4	Line interface address bit 2
5	Line interface address bit 3
6	Line interface address bit 4
7	Line interface address bit 5

Bits 0 and 1: Set by hardware to indicate that X'10' contains interrupt information and must be read immediately afterwards.

Bits 2 through 7: Contain the line interface address. They are reset by the microcode. Bit 7 is set off for a transmit line interface, and is set on for a receive line interface.

X'13': LINE INTERFACE ADDRESS (READ/WRITE)

This register contains the line interface address during CSP asynchronous actions (read or write). The bits are checked for parity. They have the following meaning:

Bit	Function
0-1	(Not used: set to 00)
2	Line interface address bit 0
3	Line interface address bit 1
4	Line interface address bit 2
5	Line interface address bit 3
6	Line interface address bit 4
7	Line interface address bit 5

Bits 2 through 7: Contain the line interface address. They are reset by the microcode. Bit 7 is set off for a transmit line interface, and is set on for a receive line interface.

X'14': DATA IN/OUT

This register is used by the microcode for transferring data to be loaded into, or coming from, the RAMs or the LIC/ICC registers, during asynchronous operations. The bits are checked for parity. They are not described because the X'14' register is used as a data buffer.

X'15': ASYNCHRONOUS OPERATION COMMAND

This register is used by the microcode to specify to the FES the asynchronous operation to be executed. The bits are checked for parity. They have the following meaning:

Bit	Function
0	Read/write operation
1-3	Storage element to be accessed
4-5	Register address
6-7	(Not used)

Bit 0: Set by the microcode to indicate a write operation. It is reset to indicate a read operation.

Bits 1 and 2: Set by the microcode to indicate:

01 : RAM A access
00 : RAM B access
10 : RAM C access
11 : LIC/ICC or FES diagnostic register

Bit 3: Set by the microcode to indicate:

RAM access : 0 even byte
1 odd byte
LIC/ICC access : 0 LIC access
1 ICC or diagnostic register

Bits 4 and 5: Set by the microcode to indicate:

Storage : Halfword displacement address
LIC : Register address
ICC : Register address

If bits 4 and 5 are set to 11, the diagnostic register of the FES is accessed.

X'16': ASYNCHRONOUS OPERATION STATUS

This register gives the status of the last asynchronous operation initiated by the microcode. It can be reset by a microcode write operation. In a read operation, if bit 1 is equal to 1, the bits are not checked for parity. The bits have the following meaning:

Bit	Function
0	Data exchange not complete
1	Error during data exchange
2-3	(Not used)
4	LIC/ICC error
5	FES internal error
6-7	(Not used)

Bit 0: Set by hardware during asynchronous operation when the microcode specifies a command via register X'15'. It is reset when the operation is complete.

Bit 1: Set by hardware when an error occurs during data byte processing.

Bits 4 and 5: When bit 1 is set, bits 4 and 5 indicate the type of error that occurred.

Bits 1, 4, and 5 can be reset either by the microcode or when the asynchronous operation that follows is executed without error.

External Register Description (Part 4 of 5)

X'17': FES GENERAL COMMANDS

This register receives the FES general commands sent from the CSP. The bits are not checked for parity. They have the following meaning:

Bit	Function
0	Reset FES latches and disable CSP/FES
1	Reset FES storage
2	Freeze state
3	(Not used)
4	FES storage reset error
5-7	(Not used, set to 0)

Bit 0: Set by the microcode to reset the FES latches and disable the CSP/FES wires. The RAM information is not reset. The LIC and ICC registers are reset.

Bit 1: Set by the microcode to reset the FES storage. It is set off by hardware when the reset is completed.

Bit 2: Set by the microcode to stop the FES scanning at the end of the current line process by the scanner base. All storage and register information is kept. Resetting bit 2 restarts the scanning.

Bit 4: Set by hardware to report any error detected during FES storage reset.

Bits 5 through 7: Set off and reserved for CDS update.

X'19': CSP INTERRUPT REQUEST

This register contains pending CSP interrupt requests. The bits are not checked for parity. They have the following meaning:

Bit	Function
0	Set level 0
1	Set level 1
2	Set level 2
3	Set level 3 (prog. control only)
4-7	(Not used)

Bits 0 through 3: Set by the hardware or the microcode to request an interrupt from the CSP. When several interrupt requests are presented at the same time, the highest unmasked level is taken into account. When all bits 0 through 3 are off, the microcode runs on level 7.

X'1A': CURRENT CSP INTERRUPT LEVEL

This register contains the current CSP interrupt level and stacks this level when a level 0 occurs. The bits are checked for parity and have the following meaning:

Bit	Function
0	Current level stack parity bit
1	Current level stack bit 0
2	Current level stack bit 1
3	Current level stack bit 2
4	Current level parity bit
5	Current level bit 0
6	Current level bit 1
7	Current level bit 2

Bits 0 through 3: Set by hardware to stack the current interrupt level when a CSP level 0 interrupt occurs.

Bits 4 through 7: Represent the current interrupt level. They are reset after a power on reset or a general reset.

X'1B': ADDRESS COMPARE CONTROL

This register controls the address compare function of the CSP. The bits are not checked for parity. They have the following meaning:

Bit	Function
0	Address compare
1	Address compare
2	(Not used)
3	(Not used)
4	Instruction/data fetch
5	Data storage
6	Cycle steal fetch
7	Cycle steal storage

Bit 0: Set by hardware when the control storage address compares with the address contained in external registers '1C' and '1D', and if one of bits 4 through 7 of the register is set. When set on, this bit requests a level 0 interrupt from the CSP. It is reset by the microcode.

Bit 1: Set by hardware when the control storage address compares with the address contained in external registers '1C' and '1D', and if one of bits 6 or 7 of the register is set. Bit 0 is set at the same time. Bit 1 is reset by the microcode.

Bit 4: Set by the microcode to ask for an address compare during instruction or data fetch. Bit 0 is set on if the address compare is detected. Bit 4 is reset by the microcode.

Bit 5: Set by the microcode to ask for an address compare during a data storage instruction. Bit 0 is set on if the address compare is detected. Bit 5 is reset by the microcode.

Bit 6: Set by the microcode to ask for an address compare during a cycle steal fetch. Bits 0 and 1 are set on if the address compare is detected. Bit 6 is reset by the microcode.

Bit 7: Set by the microcode to ask for an address compare during cycle steal store. Bits 0 and 1 are set on if the address compare is detected. Bit 7 is reset by the microcode.

External Register Description (Part 5 of 5)

X'1C': ADDRESS COMPARE BYTE 0

This register contains byte 0 of the address to be compared with the actual control storage address. The bits are checked for parity and have the following meaning:

Bit	Function
0	Main storage address high 0
1	Main storage address high 1
2	Main storage address high 2
3	Main storage address high 3
4	Main storage address high 4
5	Main storage address high 5
6	Main storage address high 6
7	Main storage address high 7

X'1D': ADDRESS COMPARE BYTE 1

This register contains byte 1 of the address to be compared with the actual control storage address. The bits are checked for parity and have the following meaning:

Bit	Function
0	Main storage address low 0
1	Main storage address low 1
2	Main storage address low 2
3	Main storage address low 3
4	Main storage address low 4
5	Main storage address low 5
6	Main storage address low 6
7	Main storage address low 7

X'1E': CSP INTERRUPT MASKS

This register uses masks to control CSP interrupts. The bits are checked for parity and have the following meaning:

Bit	Function
0	Mask interrupt level 0
1	Mask interrupt level 1
2	Mask interrupt level 2
3	Mask interrupt level 3
4	Mask interrupt level 4 (not used)
5	Mask interrupt level 5 (not used)
6	(Not used)
7	Master mask

Bits 0 through 5: Control the masking of levels 0 through 5.

Bit 7: Masks all levels except level 0 and the current level.

X'1F': LOCAL STORAGE ADDRESS

This register contains the primary (bits 0 through 3) and secondary (bits 4 through 7) local storage address. The bits are checked for parity and have the following meaning:

Bit	Function
	Primary local storage address:
0	Bit 0
1	Bit 1
2	Bit 2
3	Bit 3
	Secondary local storage address:
4	Bit 0
5	Bit 1
6	Bit 2
7	Bit 3

FES RAM A Description (Part 1 of 3)

The fields of the FES RAM A are described bit-by-bit in this section (see page 13-130 for microcode/FES interaction).

RAM A RECEIVE

Hw	Byte 0 (Even address)	Byte 1 (Odd address)
0	New PDF	SCF
1	Old PDF	Interrupt request
2	Control storage addressing	
3	Scanner base ctl.	Timer

Parallel Data Field (PDF)

The old PDF and the new PDF contain the next two characters to be loaded into the control storage of the CSP.

Secondary Control Field (SCF)

Bit	Function
8	End 1
9	Character service request
10	Overrun detected
11	Modem-in change detected
12	End 2
13	End 3
14	(Not used)
15	Timer forced to 3 seconds (BSC)

Bits 8, 12, and 13: Set by hardware to indicate to the microcode the end of data transfer. They are coded as follows, depending on the protocol:

Protocol	End Bit 1 2 3	Meaning
All	0 0 0	Data
SS	1 0 0	Stop check
SDLC	0 0 1	Flag OK
	0 1 1	Flag off boundary
	1 0 1	Data check
	1 0 0	Abort
	1 1 0	Idle
BSC	1 1 1	Force timer to 0
	0 1 1	Normal text
	1 1 0	Transparent text
	X 0 1	Quit control
	1 0 0	Data check

Bits 9 and 10: Bit 9 is set by the front-end layer to request character service from the scanner base layer. Bit 10 is set by hardware to indicate an overrun condition (too many characters received) to the microcode. Bits 9 and 10 are associated as follows to indicate:

00 : No scanner base operation
01 : Characters received during overrun are deleted (BSC only)
10 : Character service request
11 : Overrun with characters lost

Bit 11: Set by hardware to indicate a modem-in change.

Bit 15: In BSC, is set by hardware to force the timer to three seconds to check that the SYN characters are separated by less than three seconds. It is reset by hardware.

Interrupt Request

Bit 8 = 0: Errors

Bit	Function
9	Overrun
10	(Not used)
11	Modem-in change detected
12	End bit present (or ending conf.)
13	Interrupt pending
14	(Not used)
15	Forced to 1

Bit 8 = 1: Other CSP L2 interrupts

Bit	Function
9	FES/LIC error
10	ICC internal error
11	FES internal error
12	CSP/FES error
13	Interrupt pending
14-15	Forced to 00

Bit 8: Set by hardware to indicate an error condition. It is off for other CSP level 2 interrupt conditions.

Bit 9: Set by hardware to indicate an overrun when bit 8 is 0, or a FES/LIC error when bit 8 is 1.

Bit 10: Set by hardware to indicate an ICC internal error when bit 8 is 1.

Bit 11: This bit is set on by hardware to indicate a modem-in change when bit 8 is 0, or an FES internal error when bit 8 is 1.

Bit 12: Set by hardware to indicate that an end bit is present in the SCF when bit 8 is 0, or to indicate a CSP/FES error when bit 8 is 1. It is reset by the scanner base when the interrupt has been processed.

Bit 13: Set by hardware to indicate to the scanner base that an interrupt is to be processed.

Control Storage Addressing

Bit	Function
0-1	Buffer length
2-4	Forced to 100
5-10	Buffer starting address
11-13	Burst number
14-15	Halfword number

Bits 0 and 1: Set by the microcode to indicate the buffer length:

00 : 32 bytes (4 bursts)
01 : 64 bytes (8 bursts)
10 : 128 bytes (16 bursts)
11 : 256 bytes (32 bursts)

Bits 5 through 10: Set by the microcode to indicate the buffer starting address to the FES for cycle stealing.

Bits 11 through 13: Set by the microcode to indicate the burst number (0 through 7) to the FES.

Bits 14 and 15: Set by the microcode to indicate the halfword number per burst (0 through 3) to the FES. There are four halfwords (8 bytes) in one burst.

Scanner Base Control

This field is loaded by the microcode to start the receive operation and to provide parameters for the current burst. Parameters for the following bursts are taken from the parameter/status area of the CSP control store, and loaded into this field by hardware.

Parameters (Bit 0 = 0: Burst valid)

Bit	Function
1	Start receive operation
2	Interrupt requested
3	EP mode
4	Stacked PDF full
5-7	Burst length (in bytes)

Status (Bit 0 = 1: Burst not valid)

Bit	Function
1-7	Invalid information

Bit 0 When off, indicates that the current burst is free and valid for data cycle steal. The scanner base control field contains valid parameters in bits 1 through 7.

When on, it indicates that the current burst has not been processed by the microcode, and is not valid. The scanner base control field contains old status information in bits 1 through 7; this information is invalid.

Bit 1: Set by the microcode to start the receive operation in the scanner base layer. It is reset by hardware.

Bit 2: Set to indicate that a CSP level 2 interrupt must be requested at the end of the burst.

Bit 3: Set to indicate that the line is working in emulation program mode. It decides whether or not a CSP interrupt level 2 is to be requested at character end.

Bit 4: Set by hardware to indicate that the old PDF is full. It is reset by hardware.

Bits 5 through 7: Indicate the burst length in bytes.

FES RAM A Description (Part 2 of 3)

Timer

Bit	Function
8	Timer active
9	Timer mode (100 ms/2 ms)
10	Clock service phase
11-15	Timer value

Bit 8: In BSC, set by hardware to start the timer, and reset to stop the timer. In other protocols, it can be set by the microcode to start the timer, and reset by the hardware at timeout.

Bit 9: Set by the microcode to indicate the unit of time to be used for the timer:

0 : 100 ms
1 : 2 ms

In BSC, bit 9 is always 0.

Bit 10: Set by hardware and is used for internal processing of the timer. It is reset by hardware.

Bits 11 through 15: In BSC, set by hardware to:

- Manage the insertion of a SYN character every second in transmit
- Check the occurrence of the SYN character less than once every 3 seconds in receive. They are reset by hardware. In other protocols, bits 11 through 15 can be set by the microcode to provide the timer value according to the unit specified via bit 9.

RAM A TRANSMIT

Hw	Byte 0 (Even address)	Byte 1 (Odd address)
0	Timer Control	SCF
1	Next PDF	Interrupt request
2	Control storage addressing	
3	Scanner base control	

Timer Control

Bit	Function
0	SYN insertion
1-7	(Not used)

Bit 0: Set by the microcode in BSC to allow the hardware to insert a SYN character every second in the transmitted data.

Secondary Control Field (SCF)

Bit	Function
8	(Not used)
9	Character service request
10	Underrun detected
11	Modem-in change detected
12	EOT sent
13	Transparency ending
14-15	(Not used)

Bits 9 and 10: Bit 9 is set by the front-end layer to request character service from the scanner base layer.

Bit 10 is set by hardware to indicate an underrun condition (lack of characters) to the microcode. When the FES is in the underrun condition:

- SDLC: The line is stopped.
- BSC: SYN characters are transmitted.
- Start-stop: Mark bits are transmitted.

Bits 9 and 10 are associated to indicate:

00 : No scanner base operation
01 : Underrun without operation
10 : Character service request
11 : Underrun with cycle steal

Bit 11: Set by hardware to indicate a modem-in change.

Bit 12: Set by hardware to indicate that the EOT character has been transmitted.

Bit 13: Set by hardware to indicate end of transparent text.

Parallel Data Field (PDF)

The next PDF contains the last character received from the control storage (and the next to be transmitted).

Interrupt Request

Bit 8 = 0: Various conditions

Bit	Function
9	Underrun
10	Time out
11	Modem-in change detected
12	Transparency ending
13	Interrupt pending
14	(Not used)
15	Forced to 1

Bit 8 = 1: Other CSP L2 interrupts

Bit	Function
9	FES/LIC error
10	LIC driver check
11	FES internal error
12	CSP/FES error
13	Interrupt pending
14	Forced to 0
15	Forced to 0

Bit 8: Set by hardware to indicate an error condition. It is off for the other CSP level 2 interrupt conditions.

Bit 9: Set by hardware to indicate an overrun when bit 8 is 0, or a FES/LIC error when bit 8 is 1.

Bit 10: Set by hardware to indicate an ICC internal error when bit 8 is 1.

Bit 11: Set by hardware to indicate a modem-in change when bit 8 is 0, or an FES internal error when bit 8 is 1.

Bit 12: Set by hardware to indicate that an end bit is present in the SCF when bit 8 is 0, or to indicate a CSP/FES error when bit 8 is 1. It is reset by the scanner base when the interrupt has been processed.

Bit 13: Set by hardware to indicate to the scanner base that an interrupt is to be processed.

Control Storage Addressing

Bit	Function
0-1	Buffer length
2-4	Forced to 100
5-10	Buffer starting address
11-13	Burst number
14-15	Halfword number

Bits 0 and 1: Set by the microcode to indicate the buffer length:

00 : 32 bytes (4 bursts)
01 : 64 bytes (8 bursts)
10 : 128 bytes (16 bursts)
11 : 256 bytes (32 bursts)

Bits 5 through 10: Set by the microcode to indicate the buffer starting address to the FES for cycle stealing.

Bits 11 through 13: Set by the microcode to indicate the burst number (0 through 7) to the FES.

Bits 14 and 15: Set by the microcode to indicate the halfword number per burst (0 through 3) to the FES. There are four halfwords (eight bytes) in one burst.

FES RAM A Description (Part 3 of 3)

Scanner Base Control

Parameters (Bit 0 = 0: Burst valid)

Bit	Function
1	Start transmit operation
2	Interrupt requested
3	Modem-in change detected
4	Stacked PDF full
5-7	Burst length (in bytes)
8	Send CRC
9	Start on odd byte
10	(Not used)
11-12	Start transmit operation
13	Option protocol
14	Start timer
15	Send modem-out stacked

Status (Bit 0 = 1: Burst not valid)

Bit	Function
1-15	Invalid information

Bit 1: Set by the microcode to start the transmit operation. It is reset by hardware.

Bit 2: Set to indicate that a CSP level 2 interrupt must be requested at the end of the burst.

Bit 3: Set by hardware to indicate a modem-in change during data transmission. The modem-in status will be examined at the next CSP interrupt.

Bit 4: Set by hardware to indicate that the old PDF field is full. It is reset by hardware.

Bits 5 through 7: Indicate the burst length in bytes.

Bit 8: When on, causes the CRC to be sent at the end of the burst.

Bit 9: When on, causes transmission to start with the odd byte on the first half-word of a burst. It is reset by the microcode.

Bits 11 and 12: Specify the transmit operation as follows:

```
00  : Normal
01  : End of message
10  : Transmit continuous
11  : End of message with turnaround
```

Bit 13: Specifies the following protocol options:

```
SDLC: No zero insert
SS  : Transmit break
BSC : Transparency end
```

Bit 14: When on, starts the timer at burst end or end of transmission.

Bit 15: When on, requests the modem-out stacked pattern to be sent on the line interface.

FES RAM B Description (Part 1 of 2)

The fields of the FES storage RAM B are described bit-by-bit in this section (see page 13-130 for microcode/FES interaction).

RAM B RECEIVE

Hw	Byte 0 (Even address)	Byte 1 (Odd address)
0	SDF	SCF
1	PDF	PCF
2	Spare	Spare
3	SYN 2	SYN 1

Serial Data Field (SDF)

This field is used for the deserialization of the bits received from the 'receive data' line via the LIC.

Secondary Control Field (SCF)

Bit	Function
8	End 1
9	Bit counter 2
10	Bit counter 1
11	Modem-in change detected
12	End 2
13	End 3
14	Bit counter 0
15	(Not used)

Bits 8, 12 and 13: Set by hardware to indicate to the microcode the end of data transfer. They are coded as follows, depending on the protocol:

Protocol	End Bit 1 2 3	Meaning
All	0 0 0	Data
SS	1 0 0	Stop check
SDLC	0 0 1	Flag OK
	0 1 1	Flag off boundary
	1 0 1	Data check
	1 0 0	Abort
	1 1 0	Idle
BSC	1 1 1	Force timer to 0
	0 1 1	Normal text
	1 1 0	Transparent text
	X 0 1	Quit control
	1 0 0	Data check

Bit 9, 10, and 14: Set by hardware to count the bits in the SDF. They are reset by hardware at the end of the character.

Bit 11: Set by hardware to indicate a modem-in change and prevent the receive until the modem-in status is processed by the microcode.

Parallel Data Field (PDF)

This field contains one character after deserialization.

Primary Control Field (PCF)

Bit	Function
8	End 1
9	Character service request
10	Overrun detection
11	Modem-in change detected
12	End 2
13	End 3
14	Start receive operation
15	Force timer to 3 seconds (BSC)

Bits 8, 12, and 13: These bits indicate the end of data transfer and are coded as follows, depending on the protocol:

Protocol	End Bit 1 2 3	Meaning
All	0 0 0	Data
SS	1 0 0	Stop check
SDLC	0 0 1	Flag OK
	0 1 1	Flag off boundary
	1 0 1	Data check
	1 0 0	Abort
	1 1 0	Idle
BSC	1 1 1	Force timer to 0
	0 1 1	Normal text
	1 1 0	Transparent text
	X 0 1	Quit control
	1 0 0	Data check

Bits 9 and 10: Set by hardware to indicate:

00 : Not used
01 : Characters received during overrun are deleted (BSC only)
10 : Character service
11 : Overrun with characters lost

Bit 11: Set by hardware to indicate a modem-in change and to prevent the receive until the modem-in status is processed by the microcode.

Bit 14: Set by hardware to start the receive operation.

Bit 15: Set by hardware to force the timer to 3 seconds, for BSC SYN character in receive, for example. It is reset by hardware.

Synchronization

This field contains the SYN characters to be used during receipt of BSC characters. These characters vary depending on the protocol: in BSC ASCII, they are X'1616', in BSC EBCDIC X'3232'.

FES RAM B Description (Part 2 of 2)

RAM B TRANSMIT

Hw	Byte 0 (Even address)	Byte 1 (Odd address)
0	SDF	SCF
1	PDF	PCF
2	DLE	Spare
3	SYN	Spare

Serial Data Field (SDF)

This field is used to serialize the character to be transmitted on the 'transmit data' line via the LIC.

Secondary Control Field (SCF)

Bit	Function
8	Send CRC
9	Bit counter 2
10	Bit counter 1
11	Modem-in change detected
12	Send EOT
13	SDLC : no zero insert BSC : transparency end SS : transmit break
14	Bit counter 0
15	Send modem-out stacked

Bit 8: Set by hardware to request the CRC to be sent at the end of the burst. It is reset by hardware.

Bit 9, 10, and 14: Set by hardware to count the bits in the SDF. They are reset by hardware at the end of the character.

Bit 11: Set by hardware to indicate a modem-in change and prevent starting the transmission until the modem-in status is processed by the microcode.

Bit 12: Set by hardware to request the EOT character to be sent. It is reset by hardware.

Bit 13: Set by hardware to indicate the selected protocol options.

Bit 15: Set by hardware to request the modem-out stacked pattern to be sent on the line interface. It is reset by hardware.

Parallel Data Field (PDF)

This field contains one character before serialization.

Primary Control Field (PCF)

Bit	Function
8	Send CRC
9	Character service request
10	Underrun detection
11	Modem-in change detected
12	Send EOT
13	SDLC : no zero insert BSC : transparency end SS : transmit break
14	Start transmit operation
15	Send modem-out stacked

Bit 8: Set by hardware to request the CRC to be sent at the end of the burst. It is reset by hardware.

Bits 9 and 10: Set by hardware to indicate:

00 : Not used
01 : Characters received during underrun are deleted (BSC only)
10 : Character service
11 : Underrun with characters lost

Bit 11: Set by hardware to indicate a modem-in change and prevent starting the transmission until the modem-in status is processed by the microcode.

Bit 12: Set by hardware to request the EOT character to be sent. It is reset by hardware.

Bit 13: Set by hardware to select the protocol options. It is reset by hardware.

Bit 14: Set by hardware to start the line operation. It is reset by hardware.

Bit 15: Set by hardware to request the modem-out stacked pattern to be sent on the line interface. It is reset by hardware.

Data Link Escape (DLE)

This field contains the data link escape (DLE) character used in BSC transmission. The sequences with a DLE character initiate and terminate the transparent text, and provide an active control character within the transparent text.

Synchronization

This field contains the SYN characters to be used during the transmission of BSC characters. These characters vary depending on the protocol; in BSC ASCII, they are X'1616', in BSC EBCDIC X'3232'.

FES RAM C Description (Part 1 of 5)

The fields of the FES storage RAM C are described bit-by-bit in this section (see page 13-130 for microcode/FES interaction).

LINE TYPE IDENTIFICATION

In the following descriptions, use bits 0, 1, and 2 of the set mode byte to select the right RAM C description.

Set Mode	RAM C Description
00x	BSC ASCII
01x	BSC EBDCIC
100	BSC normal
101	Start-stop
11x	SDLC

SDLC RECEIVE

Hw	Byte 0 (Even address)	Byte 1 (Odd address)
0	Set mode	Control
1	BCC2	BCC1
2	Modem-in pattern	Mask
3	Spare	Spare

Set Mode

Bit	Function
0-1	Line type: SDLC
2	Interrupt on first flag
3	Receive : no zero delete option Transmit : not used
4	NRZI
5-7	Ones counter

Bits 0 and 1: Set to 11 by the microcode to indicate that the line uses the SDLC protocol.

Bit 2: Set by the microcode to request a CSP level 2 interrupt when the first SDLC flag is detected.

Bits 3: Set by the microcode to indicate the zero delete option:

0 : Zero delete option
1 : No zero delete option

Bit 4: Set by the microcode to indicate the non-return to zero inverted (NRZI) option:

0 : No NRZI option
1 : NRZI option

Bits 5 through 7: Set by hardware to count the received 1-bits in order to detect SDLC flags, abort characters, and idle characters.

Control

Bit	Function
8	(Not used)
9	Character phase
10	First bit flag
11-12	SDLC state
13	Last line state
14-15	Line state

Bit 9: Set by hardware when the first character is detected. It is reset by hardware when no more characters are received. It may be set on by the microcode to request the bit service function to return all incoming bits in bytes without taking true character boundaries into account.

Bit 10: Set by hardware when the first bit of a character is detected. It is reset by hardware when the last bit is detected.

Bits 11 and 12: Initialized by the microcode and updated by hardware to indicate the SDLC state:

00 : Data
01 : First flag
10 : Abort
11 : Idle

Bit 13: Updated by hardware to indicate the last state of the line: 0 or 1. It is used with the NRZI option only.

Bits 14 and 15: Initialized by the microcode and updated by hardware to indicate the line state:

00 : Inhibited
01 : Diagnostic mode
10 : Modem without data
11 : Modem and data

Block Check Character (BCC)

These fields (BCC1 and BCC2) are used for CRC checking.

Modem-In

Bit	Function
0	V.24: Data set ready V.25: Power indication X.21: (Not used)
1	V.24: Ready for sending V.25: Data line occupied X.21: Indication
2	V.24: Ring indicator V.25: Present next digit X.21: (Not used)
3	V.24: Receive line signal detector V.25: Abandon call and retry X.21: (Not used)
4	V.24: Test indicator V.25: Call origination status X.21: (Not used)
5	V.24: Received data V.25: (Not used) X.21: Receive
6-7	Modem-in change detected

Bits 0 through 5: Modem-in status.

Bits 6 and 7: Set when a change is detected with respect to the mask pattern. Other changes are no longer detected until the microcode has processed the new modem-in pattern and reset bits 6 and 7.

Mask

Bit	Function
8-13	Mask bits
14	Start conditional
15	No modem change report

Bits 8 through 13: Set by the microcode to monitor the modem-in pattern for changes. A mask bit set to 1 means "look for a change".

Bit 14: Set by the microcode to start the receive operation only if the modem is ready.

Bit 15: Set by the microcode to select the transmit line interface for modem-in change reporting. No modem-in change is reported on the receive line interface.

FES RAM C Description (Part 2 of 5)

SDLC TRANSMIT

Hw	Byte 0 (Even address)	Byte 1 (Odd address)
0	Set mode	Control
1	BCC2	BCC1
2	Modem-out immediate	Mask
3	Modem-out stacked	

Set Mode

As for receive.

Control

Bit	Function
8	(Not used)
9	Modem-in change detected
10	First bit flag
11	BCC1 sent
12	(Not used)
13	Last line state
14-15	Line state

Bit 9: Set by hardware to indicate a modem-in change during data transmission. The modem-in status is examined at the next CSP interrupt. Bit 9 is reset by hardware after processing.

Bit 10: Set by hardware when the first bit of a character is transmitted. It is reset by hardware at the end of the bit time.

Bit 11: Set by hardware to indicate that BCC1 has been transmitted. It is reset by hardware when BCC2 has been transmitted.

Bit 13: Updated by hardware to indicate the last state of the line: 0 or 1. It is used with the NRZI option only.

Bits 14 and 15: Initialized by the microcode and updated by hardware to indicate the line state:

00 : Inhibited
01 : Diagnostic mode
10 : Modem without data
11 : Modem and data

Block Check Character (BCC)

These fields (BCC1 and BCC2) are used for CRC checking.

Modem-Out

Bit	Function
0	V.24: Data terminal ready V.25: Digit signal 8 X.21: (Not used)
1	V.24: Request to send V.25: Digit signal 4 X.21: Control
2	V.24: New synchronization V.25: Digit signal 2 X.21: (Not used)
3	V.24: Data rate selector V.25: Digit signal 1 X.21: (Not used)
4	V.24: Modem test V.25: Call request X.21: (Not used)
5	V.24: Diag bit strobe V.25: Digit present X.21: Transmit
6-7	Service request (LIC dvr. check)

Bits 0 through 5: Modem-out status.

Bits 6 and 7: Incremented each time a modem-out pattern is sent to the LIC. When bits 6 and 7 are equal to 11, a comparison is made by hardware between the transmitted pattern and the LIC driver outputs. If the patterns are different, the LIC driver check bit (X'10' bit 2) is set on.

Mask

Bit	Function
8-13	Mask bits
14	Start conditional
15	Mask of mark bit

Bits 8 through 13: Set by the microcode when the same modem-in lines are to be taken into account.

Bit 14: Set by the microcode to start the receive operation when the modem is ready.

Bit 15: Set by the microcode to mask the mark bit at the LIC driver output, for example, for checking.

FES RAM C Description (Part 3 of 5)

BSC RECEIVE

Hw	Byte 0 (Even address)	Byte 1 (Odd address)
0	Set mode	Control
1	BCC2	BCC1
2	Modem-in pattern	Mask
3	Spare	Spare

Set Mode: BSC, Normal

In BSC normal mode, the data is exchanged by messages.

Bit	Function
0-2	Line type: BSC normal
3-4	(Not used)
5	One SYN character
6-7	Character length

Bits 0 through 2: Set to 100 by the microcode to indicate that the line type is BSC.

Bit 5: Set by the microcode to indicate that only one SYN character must be transmitted. When off, it indicates that two SYN characters must be sent.

Bits 6 and 7: Set by the microcode to indicate the character length:

00 : Five bits
01 : Six bits
10 : Seven bits
11 : Eight bits

Set Mode: BSC, ASCII or EBCDIC

In BSC ASCII or EBDCIC mode, the data is exchanged by characters.

Bit	Function
0-1	Line type: BSC, ASCII or EBCDIC
2-3	ITB/EIB encoding
4-5	CRC type
6	ASCII 8-bits

Bits 0 and 1: Set by the microcode to indicate the line type:

00 : BSC ASCII
01 : BSC EBCDIC

Bits 2 and 3: Set by the microcode to indicate the ITB/EIB status:

00 : ITB is data
01 : ITB mode
10 : EIB mode
11 : EIB mode with burst change

Bits 4 and 5: Indicate the CRC type as follows:

00 : STX included
CRC/S if ASCII
CRC/B if EBCDIC
01 : CRC/B
10 : CRC/S
11 : LRC with VRC (ASCII 7-bits)

Bit 6: Set by the microcode to indicate ASCII 8-bits.

Control

Bit	Function
7	D (ASCII or EBCDIC)
8	S (ASCII or EBCDIC)
9	Character phase
10	First bit flag
11	C (ASCII or EBCDIC)
12-13	NX (ASCII or EBCDIC)
14-15	Line state

Bit 7: In BSC ASCII or EBCDIC, it is set by hardware when a DLE character is detected. It is reset by hardware.

Bit 8: In BSC ASCII or EBCDIC, it is set by hardware when a SYN character is detected. It is reset by hardware.

Bit 9: Set by hardware when the first character is detected. It is reset by hardware when no more characters are received. It may be set on by the microcode to request the bit service function to return all incoming bits in bytes without true character boundaries.

Bit 10: Set by hardware when the first bit of a character is detected. It is reset by hardware at the end of the bit time.

Bit 11: In BSC ASCII or EBCDIC, it is set by hardware when in control mode. It is reset by hardware.

Bits 12 and 13: In BSC ASCII or EBCDIC, they are set by hardware to indicate the status of NX as follows:

00 : Normal mode
01 : Transparent text mode
10 : Normal text mode
11 : CRC phase

Bits 14 and 15: Initialized by the microcode and updated by hardware to indicate the line state:

00 : Inhibited
01 : Diagnostic mode
10 : Modem without data
11 : Modem and data

Block Check Character (BCC)

These fields (BCC1 and BCC2) are used for CRC checking.

Modem-In

Same as SDLC receive.

Mask

Same as SDLC receive.

FES RAM C Description (Part 4 of 5)

BSC TRANSMIT

Hw	Byte 0 (Even address)	Byte 1 (Odd address)
0	Set mode	Control
1	BCC2	BCC1
2	Modem-out immediate	Mask
3	Modem-out stacked	

Set Mode

Same as receive.

Control

Bit	Function
7	D (ASCII or EBCDIC)
8	S (ASCII or EBCDIC)
9	Modem-in change detected
10	First bit flag
11	C (ASCII or EBCDIC)
12-13	NX (ASCII or EBCDIC)
14-15	Line state

Bit 7: In BSC ASCII or EBCDIC, it is set by hardware when a DLE character is detected. It is reset by hardware.

Bit 8: In BSC ASCII or EBCDIC, it is set by hardware when a SYN character is detected. It is reset by hardware.

Bit 9: Set by hardware to indicate a modem-in change during data transmission. The modem-in status is examined at the next CSP interrupt. Bit 9 is reset by hardware after processing.

Bit 10: Set by hardware when the first bit of a character is transmitted. It is reset by hardware when the last bit is detected.

Bit 11: In BSC ASCII or EBCDIC, it is set by hardware when in control mode. It is reset by hardware.

Bits 12 and 13: In BSC ASCII or EBCDIC, they are set by hardware to indicate the status of NX as follows:

00 : Normal mode
01 : Transparent text mode
10 : Normal text mode
11 : CRC phase

Bits 14 and 15: Initialized by the microcode and updated by hardware to indicate the line state:

00 : Inhibited
01 : Diagnostic mode
10 : Modem without data
11 : Modem and data

Block Check Character (BCC)

These fields (BCC1 and BCC2) are used for CRC checking.

Modem-Out

Same as SDLC transmit.

Mask

Same as SDLC transmit.

FES RAM C Description (Part 5 of 5)

START-STOP RECEIVE

Hw	Byte 0 (Even address)	Byte 1 (Odd address)
0	Set mode	Control
1	Spare	Spare
2	Modem-in pattern	Mask
3	Spare	Spare

Set Mode

Bit	Function
0-2	Line type: Start-stop
3	(Not used)
4	Stop length
5	(Not used)
6-7	Character length

Bits 0 through 2: Set to 101 by the microcode to indicate the line type start-stop.

Bit 4: Set by the microcode to indicate the stop length:

0 : One bit
1 : Two bits

Bits 6 and 7: Set by the microcode to indicate the character length:

00 : Five bits
01 : Six bits
10 : Seven bits
11 : Eight bits

The start and stop bits are not counted in the character length.

Control

Bit	Function
8	(Not used)
9	Character phase
10	First bit flag
11-12	(Not used)
13	Stop phase
14-15	Line state

Bit 9: Set by hardware when the start bit is detected. It is reset by hardware at the character end. It may be set on by the microcode to request the bit service function to return all incoming bits in bytes without true character boundaries.

Bit 10: Set by hardware when the first bit of a character is detected. It is reset by hardware at the end of the bit time.

Bit 13: Set by hardware to indicate that stop bits are being sent. It is reset by hardware.

Bits 14 and 15: Initialized by the microcode and updated by hardware to indicate the line state:

00 : Inhibited
01 : Diagnostic mode
10 : Modem without data
11 : Modem and data

Modem-In

Same as SDLC receive.

Mask

Same as SDLC receive.

START-STOP TRANSMIT

Hw	Byte 0 (Even address)	Byte 1 (Odd address)
0	Set mode	Control
1	Spare	Spare
2	Modem-out immediate	Mask
3	Modem-out stacked	

Set Mode

Same as receive.

Control

Bit	Function
8	(Not used)
9	Modem-in change detected
10	First bit flag
11	Underrun
12	(Not used)
13	Stop phase
14-15	Line state

Bit 9: Set by hardware to indicate a modem-in change during data transmission. The modem-in status is examined at the next CSP interrupt. Bit 9 is reset by hardware after processing.

Bit 10: Set by hardware when the first bit of a character is transmitted. It is reset by hardware at the end of the bit time.

Bit 11: Set by hardware to indicate underrun condition (lack of characters from the CSP). Mark bits are transmitted over the line. Bit 11 is reset by hardware when characters are received from the CSP.

Bit 13: Set by hardware to indicate that stop bits are being sent. It is reset by hardware.

Bits 14 and 15: Initialized by the microcode and updated by hardware to indicate the line state:

00 : Inhibited
01 : Diagnostic mode
10 : Modem without data
11 : Modem and data

Modem-Out

Same as SDLC transmit.

Mask

Same as SDLC transmit.

LIC/ICC Register Description (Part 1 of 2)

The six bits (bit 0 through bit 5) of the LIC registers are used as follows (see also pages 13-030 and 13-040 for the LIC/ICC register organization):

LIC Type 1

Register 00 (V.24)

Bit	Function
0	Data set ready
1	Ready for sending
2	Ring indicator
3	Receive line signal detector
4	Test indicator
5	Received data

Register 00 (V.25)

Bit	Function
0	Auto call box is powered on
1	Data line occupied
2	Present next digit
3	Abandon call and retry
4	Call origination status
5	(Not used)

Register 01 (V.24)

Bit	Function
0	Line wrap/data terminal ready
1	Request to send
2	New sync
3	Data rate select
4	Modem test
5	Write: diag bit strobe Read: transmit bit echo

Bit 0: Set on by the microcode to indicate data terminal ready to the DCE. It is set off to put the line in wrap mode: transmit is connected to receive.

Register 01 (V.25)

Bit	Function
0	Call number 8
1	Digit signal 4
2	Digit signal 2
3	Digit signal 1
4	Call request
5	Digit present

Register 10

Bit	Function
0-2	LIC position within the scanner
3-5	Cable attachment type

Bits 0 through 2: Give the LIC position within the scanner:

000 : LIC position 8} {Not for C2LB and
001 : LIC position 7} {C2LB2 boards
010 : LIC position 6
011 : LIC position 5
100 : LIC position 4
101 : LIC position 3
110 : LIC position 2
111 : LIC position 1

Bits 3 through 5: Give the cable attachment type:

100 : DCE attachment (V.24)
110 : autocall DCE (V.25)
101 : direct attachment (V.24)

Register 11

Bit	Function
0	Line reset
1-3	LIC type
4-5	Clock selection

Bit 0: Set on by the microcode to reset all the LIC/ICC registers associated with that line. Register 11 is reset at the same time.

Bits 1 through 3: Give the LIC type: 001 (LIC type 1).

Bits 4 and 5: Give the clock selection:

00 : Diag bit strobe
01 : ICC clock (up to 1200 bps) (Note 1)
10 : DCE clock (Note 2)
11 : ICC clock (2400 bps through 19 200 bps) (Note 3)

Notes:

1. Called "business machine clock" on CDF information screen (page 2-405).
2. Called "external clock" on CDF information screen (page 2-405).
3. Called "Direct attachment" on CDF information screen (page 2-405).

LIC Type 2

Register 00

Bit	Function
0	Data set ready
1	Ready for sending
2	Ring indicator
3	Automatic gain control
4	(Not used)
5	Received data

Register 01

Bit	Function
0	Line wrap/data terminal ready
1	Request to send
2	(Not used)
3	(Not used)
4	Modem test
5	Write: diag bit strobe Read: transmit bit echo

Bit 0: Set on by the microcode to indicate data terminal ready to the DCE. It is set off to put the line in wrap mode: transmit is connected to receive.

Register 10

Bit	Function
0-2	LIC position within the scanner
3-5	Cable attachment type

Bits 0 through 2: Give the LIC position within the scanner:

000 : LIC position 8} {Not for C2LB and
001 : LIC position 7} {C2LB2 boards
010 : LIC position 6
011 : LIC position 5
100 : LIC position 4
101 : LIC position 3
110 : LIC position 2
111 : LIC position 1

Bits 3 through 5: Give the cable attachment type:

100 : DCE attachment

Register 11

Bit	Function
0	Line reset
1-3	LIC type
4-5	Clock selection

Bit 0: Set on by the microcode to reset an individual line and all the LIC/ICC registers associated with that line. Register 11 is reset at the same time.

Bits 1 through 3: Give the LIC type: 010 (LIC type 2).

Bits 4 and 5: Give the clock selection:

00 : Diag bit strobe
01 : (Not used)
10 : DCE clock
11 : (Not used)

LIC/ICC Register Description (Part 2 of 2)

LIC Type 3

Register 00

Bit	Function
0	Data set ready
1	Ready for sending (clear to send)
2	(Not used)
3	Receive line signal detector
4	Test indicator
5	Received data

Register 01

Bit	Function
0	Line wrap/data terminal ready
1	Request to send
2	Write : local attachment clock Read : transmit clock echo
3	Write : not used Read : receive clock echo
4	Modem test
5	Write: diag bit strobe Read: transmit bit echo

Bit 0: Set on by the microcode to indicate data terminal ready to the DCE. It is set off to put the line in wrap mode: transmit is connected to receive.

Register 10

Bit	Function
0-2	LIC position within the scanner
3-5	Cable attachment type

Bits 0 through 2: Give the LIC position within the scanner:

000 : LIC position 8} {Not for C2LB and
001 : LIC position 7} {C2LB2 boards
010 : LIC position 6
011 : LIC position 5
100 : LIC position 4
101 : LIC position 3
110 : LIC position 2
111 : LIC position 1

Bits 3 through 5: Give the cable attachment type:

100 : DCE attachment
101 : Direct attachment

Register 11

Bit	Function
0	Line reset
1-3	Card identifier
4-5	Clock selection

Bit 0: Set on by the microcode to reset an individual line and all the LIC/ICC registers associated with that line. Register 11 is reset at the same time.

Bits 1 through 3: Give the card identifier: 011 (LIC3).

Bits 4 and 5: Give the clock selection:

00 : Diag bit strobe
01 : (Not used)
10 : DCE clock
11 : ICC clock (2400 bps through 245 760 bps)

LIC Type 4 (A or B)

Register 00

Bit	Function
0	(Not used)
1	Indication
2-4	(Not used)
5	Received data

Register 01

Bit	Function
0	Line wrap/data terminal ready
1	Control
2-4	(Not used)
5	Write: diag bit strobe Read: transmit bit echo

Bit 0: Set on by the microcode to indicate data terminal ready to the DCE. It is set off to put the line in wrap mode: transmit is connected to receive.

Register 10

Bit	Function
0-2	LIC position within the scanner
3-5	Cable attachment type

Bits 0 through 2: Give the LIC position within the scanner:

000 : LIC position 8} {Not for boards C2LB
001 : LIC position 7} {and C2LB2
010 : LIC position 6
011 : LIC position 5
100 : LIC position 4
101 : LIC position 3
110 : LIC position 2
111 : LIC position 1

Bits 3 through 5: Give the cable attachment type:

100 : DCE attachment
101 : Direct attachment

Register 11

Bit	Function
0	Write : line reset Read : transmission speed
1-3	Card identifier
4-5	Clock selection

Bit 0: In write mode, this bit is set on by the microcode to reset an individual line and all the LIC/ICC registers associated with that line. Register 11 is reset at the same time.

In read mode, bit 0 indicates the transmission speed:

0 : Up to 9600 bps (LIC4A)
1 : Above 9600 bps (LIC4B)

Bits 1 through 3: Give the card identifier: 100 (LIC4).

Bits 4 and 5: Give the clock selection:

00 : Diag bit strobe
01 : (Not used)
10 : DCE clock
11 : LIC4A, ICC clock (2400 bps through 9600 bps)
LIC4B, ICC clock (above 9600 bps)

ICC REGISTER

Note: For any externally clocked or direct-attached line interface, the ICC bits are meaningless.

Bits 0 through 4 of the ICC register give the transmission speed as follows:

Bit 0	1	2	3	4	Speed (bps)	Line Protocol
1	0	0	0	1	50	SDLC/BSC
1	0	0	1	0	110	SDLC/BSC
1	0	1	0	0	134.5	SDLC/BSC
1	0	1	1	1	200	SDLC/BSC
1	1	0	0	0	300	SDLC/BSC
1	1	0	1	1	600	SDLC/BSC
1	1	1	0	1	1200	SDLC/BSC
1	0	0	1	1	Special*	SDLC/BSC
1	1	1	1	0	Special*	SDLC/BSC
0	0	0	0	0	50	Start-Stop
0	1	1	1	0	75	Start-Stop
0	1	1	0	1	100	Start-Stop
0	0	0	1	1	110	Start-Stop
0	0	1	0	1	134.5	Start-Stop
0	0	1	1	0	200	Start-Stop
0	1	0	0	1	300	Start-Stop
0	1	0	1	0	600	Start-Stop
0	1	1	0	0	1200	Start-Stop
0	0	0	0	1	2400	Start-Stop
0	0	0	1	0	4800	Start-Stop
0	0	1	0	0	9600	Start-Stop
0	0	1	1	1	19200	Start-Stop
0	1	0	1	1	Special*	
0	1	1	1	1	* *	Start-Stop

* non-standard line speed
** asymetrical (75 bps from terminal, 1200 bps to terminal).

Scanner Storage MAP

SCANNER STORAGE MAP (X'8000' THROUGH X'9B1F')

X'8000'	Line Interface Buffers (Transmit) (1024 halfwords)
X'8400'	Line Interface Buffers (Receive) (1024 halfwords)
X'8800'	FES Parameter/Status Area (Transmit) (512 halfwords)
X'8900'	FES Parameter/Status Area (Receive) (256 halfwords)
X'8A00'	Line Control Blocks (256 halfwords)
X'8C00'	Interface Control/Parameter-Status Area Blocks (2048 halfwords)
X'9400'	Translate Tables (256 halfwords)
X'9500'	Scanner Control Block (512 halfwords)
X'9700'	Initialization Routine (192 halfwords)
X'97C0'	(not used) (52 halfwords)
X'97EE'	PSWs 1, 2, 3 Save Area (Halt processing) (6 halfwords)
X'97F4'	Level 1 Scan Interface Trace Save Area (4 halfwords)
X'97F8'	Level 3 Scan Interface Trace Save Area (4 halfwords)
X'97FC'	Scan Interface Trace Buffer Control Block (4 halfwords)
X'9800'	Interface Control Block Table (64 halfwords)
X'9840'	Command Table
X'9881'	Scanner Microcode
X'9B00'	Line Control Block Address Table (32 halfwords)
X'9B20'	Scanner Microcode
X'F400'	Trace Area

SCB (Part 1 of 2)

SCANNER CONTROL BLOCK (X'9500' THROUGH X'96FF')

First Scanner Interface Trace Area

X'9500'	Trace Interface Control Block (16 halfwords)
X'9510'	Trace Parameter/Status Area (16 halfwords)
X'9520'	Trace Information Block (8 halfwords)

Second Scanner Interface Trace Area

X'9528'	Trace Interface Control Block (16 halfwords)
X'9538'	Trace Parameter/Status Area (16 halfwords)
X'9548'	Trace Information Block (8 halfwords)

Third Scanner Interface Trace Area

X'9550'	Trace Interface Control Block (16 halfwords)
X'9560'	Trace Parameter/Status Area (16 halfwords)
X'9570'	Trace Information Block (8 halfwords)

Fourth Scanner Interface Trace Area

X'9578'	Trace Interface Control Block (16 halfwords)
X'9588'	Trace Parameter/Status Area (16 halfwords)
X'9598'	Trace Information Block (8 halfwords)

Rejected Scanner Interface Trace Area

X'95A0'	Trace Interface Control Block (16 halfwords)
X'95B0'	Trace Parameter/Status Area (16 halfwords)

Miscellaneous Control Areas

X'95C0'	Pointer to Line Vector Table (LNVT) (2 halfwords)
X'95C2'	Pointer to MOSS Area (2 halfwords)
X'95C4'	Scanner Status | Status Ext |Scanner Numb.
X'95C5'	Active Command Completion
X'95C6'	Stacked Command Completion

Control Program Error Area

X'95C7'	Error Status Type | Error Levels
X'95C8'	Error Status
X'95C9'	Abending Information (2 halfwords)
X'95CB'	Abending Routine Address
X'95CC'	(Not used)

MOSS Error Area

X'95CD'	Error Status Type | Error Levels
X'95CE'	Error Status
X'95CF'	Abending Information (2 halfwords)
X'95D1'	Abending Routine Address
X'95D2'	Error Command Completion
X'95D3'	CIL Stacked Save
X'95D4'	(Not used)
X'95CF'	Level 7 Pages Save Error (8 halfwords)
X'95DD'	Snapshot Table (35 halfwords)
X'9600'	Mailbox 1 MOSS/Scanner Control Block (16 halfwords)
X'9610'	MailBox 1 Parameter/Status Area (8 halfwords)
X'9618'	Mailbox 2 MOSS/Scanner Control Block (16 halfwords)
X'9628'	Mailbox 2 Parameter/Status Area (8 halfwords)

SCB (Part 2 of 2)

SCANNER CONTROL BLOCK (X'9500' THROUGH X'96FF') (CONTINUED)

Save Area

X'9630'	Local Store Save Area (or Alter/Display Buffer) (64 halfwords)
X'9670'	External Register Save Area (16 halfwords)
X'9680'	Mail Box 1 (MICB+PSA) Save Area (if program reset) or MICB3 (if snapshot address compare ending) (24 halfwords)
X'9698'	Scanner Status/Active Command Completion Save Area (2 halfwords)
X'969A'	MOSS Error Area Save Area (6 halfwords)
X'96A0'	Command Completion Stack (16 halfwords)
X'96B0'	(Not used) (80 halfwords)

The scanner control block ends at address X'96FF'.

LCB

LINE CONTROL BLOCKS (ADDRESSES X'8A00' THROUGH X'8BFF')

There are 32 line control blocks (LCB) in a scanner storage, one for each line controlled by the scanner; each LCB occupies 16 halfwords. The starting address of the control block for a line 'n' may be found from the following formula:

Address of LCB for line 'n' = 8A00 + (n x 10)

where all values are hexadecimal.

Line Address	Line Number	LCB Address
0	1	8A00
1	2	8A10
2	3	8A20
3	4	8A30
4	5	8A40
5	6	8A50
6	7	8A60
7	8	8A70
8	9	8A80
9	10	8A90
A	11	8AA0
B	12	8AB0
C	13	8AC0
D	14	8AD0
E	15	8AE0
F	16	8AF0
10	17	8B00
11	18	8B10
12	19	8B20
13	20	8B30
14	21	8B40
15	22	8B50
16	23	8B60
17	24	8B70
18	25	8B80
19	26	8B90
1A	27	8BA0
1B	28	8BB0
1C	29	8BC0
1D	30	8BD0
1E	31	8BE0
1F	32	8BF0

Each LCB has the following format:

HW X'0'	Level 2 Modem Change	
HW X'1'	Level 2 Timeout	
HW X'2'	Disable Timeout (Set Mode Data Bytes 0 and 1)	
HW X'3'	Set Mode Data Byte 2	Set Mode Data Byte 3
HW X'4'	Set Mode Data Byte 4	Set Mode Data Byte 5
HW X'5'	Set Mode Data Byte 6	Set Mode Data Byte 7
HW X'6'	Set Mode Data Byte 8	Set Mode Data Byte 9
HW X'7'	Reply Timeout	
HW X'8'	Enable/Dial Timeout	
HW X'9'	Receive Text Timeout	
HW X'A'	Line Control	LCS
HW X'B'	EP BSC Index Byte Save Area	Line Flag
HW X'C'	(Not used)	
HW X'D'	(Not used)	
HW X'E'	Level 7 Transmit Interface Timer	
HW X'F'	Level 7 Receive Interface Timer	

ICB/PSA

INTERFACE CONTROL/PARAMETER STATUS AREA BLOCKS (X'8C00' THROUGH

X'93FF') The interface control/parameter status area blocks occupy addresses from X'8C00' through X'93FF'. There are 64 of these blocks, one for each line interface (transmit or receive); each block occupies 32 halfwords. The first 16 halfwords of each block contain the interface control block (ICB), and the last 16 halfwords the parameter-status area (PSA) block.

	0	F0	F	
8C00	ICB (Transmit)	PSA (Transmit)		Line Interface 00
8C20	ICB (Receive)	PSA (Receive)		Line Interface 01
8C40				Line Interface 02
8C60				Line Interface 03
8C80				
8CA0				
up to				
9380				
93A0				
93C0	ICB (Transmit)	PSA (Transmit)		Line Interface 62
93E0	ICB (Receive)	PSA (Receive)		Line Interface 63

The starting address of the interface control block for a line 'n', may be found from the following formula:

Address of ICB for interface 'n' = 8C00 + (n x 20)

where all values are hexadecimal.

Similarly, the starting address of the parameter status area block for a line interface 'n', may be found from the following formula:

Address of PSA for interface 'n' = 8C00 + (n x 20) + 10

Each ICB has the following format:

HW X'0'	Status	Control
HW X'1'	Pointer to FES Parameter Status Area	
HW X'2'	Pointer to CSP Control Store (Cycle Steal)	
HW X'3'	Cycle Steal Type	Cycle Steal Length
HW X'4'	CCU Count	Pointer to CCU STG byte X
HW X'5'	Pointer to CCU Storage (Bytes 0 and 1)	
HW X'6'	TICB Address Byte 1	Pointer to CP PSA byte X
HW X'7'	Pointer to CP PSA (Bytes 0 and 1)	
HW X'8'	TD 0	TA 1
HW X'9'	TA 0	TD 1
HW X'A'	Link Pointer (Command Queue)	
HW X'B'	Level 3 Pointer (Normal SVC)	
HW X'C'	Level 3 Timeout	
HW X'D'	Level 2 Pointer (Normal SVC)	
HW X'E'	Line Interface Identification	
HW X'F'	Flag Extension	Cycle Steal Burst Size

Parameter Area:

HW X'0'	TCC	MODIFIERS
	Command Dependant Fields	
HW X'7'		

Status Area:

HW X'8'	STATUS SCF	CMD
		LCS
	Command Dependant Fields	
HW X'F'		

Note: Refer to the NCP/EP Handbook for a detailed description of the PSA fields.

LIB

LINE INTERFACE BUFFERS (X'8000' THROUGH X'87FF')

The line interface buffers occupy addresses X'8000' through X'87FF'. There are 64 of these buffers, one for each line interface (transmit or receive); each buffer occupies 32 halfwords. The first 32 buffers are for the transmit interfaces, and the last 32 buffers are for the receive interfaces. The starting address of the transmit interface buffer for a line interface 'n' may be found from the following formula:

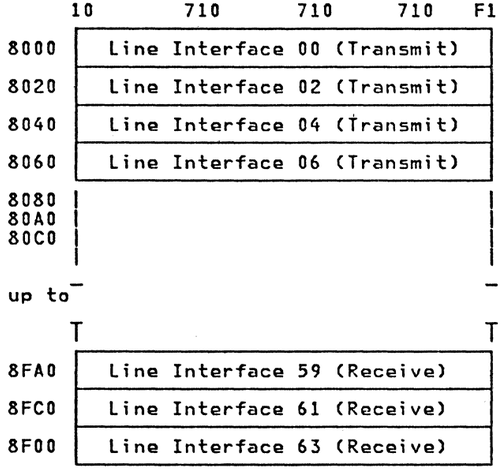
Address of transmit buffer for interface 'n' = 8000 + (n x 20)

where all values are hexadecimal.

Similarly, the starting address of the receive interface buffer for a line interface 'n' may be found from the following formula:

Address of transmit buffer for interface 'n' = 8400 + (n x 20)

where all values are hexadecimal.



Chapter 13. Transmission Subsystem

Section 2. Troubleshooting Guidelines

DC Voltages and Tolerances at Board Pin Level

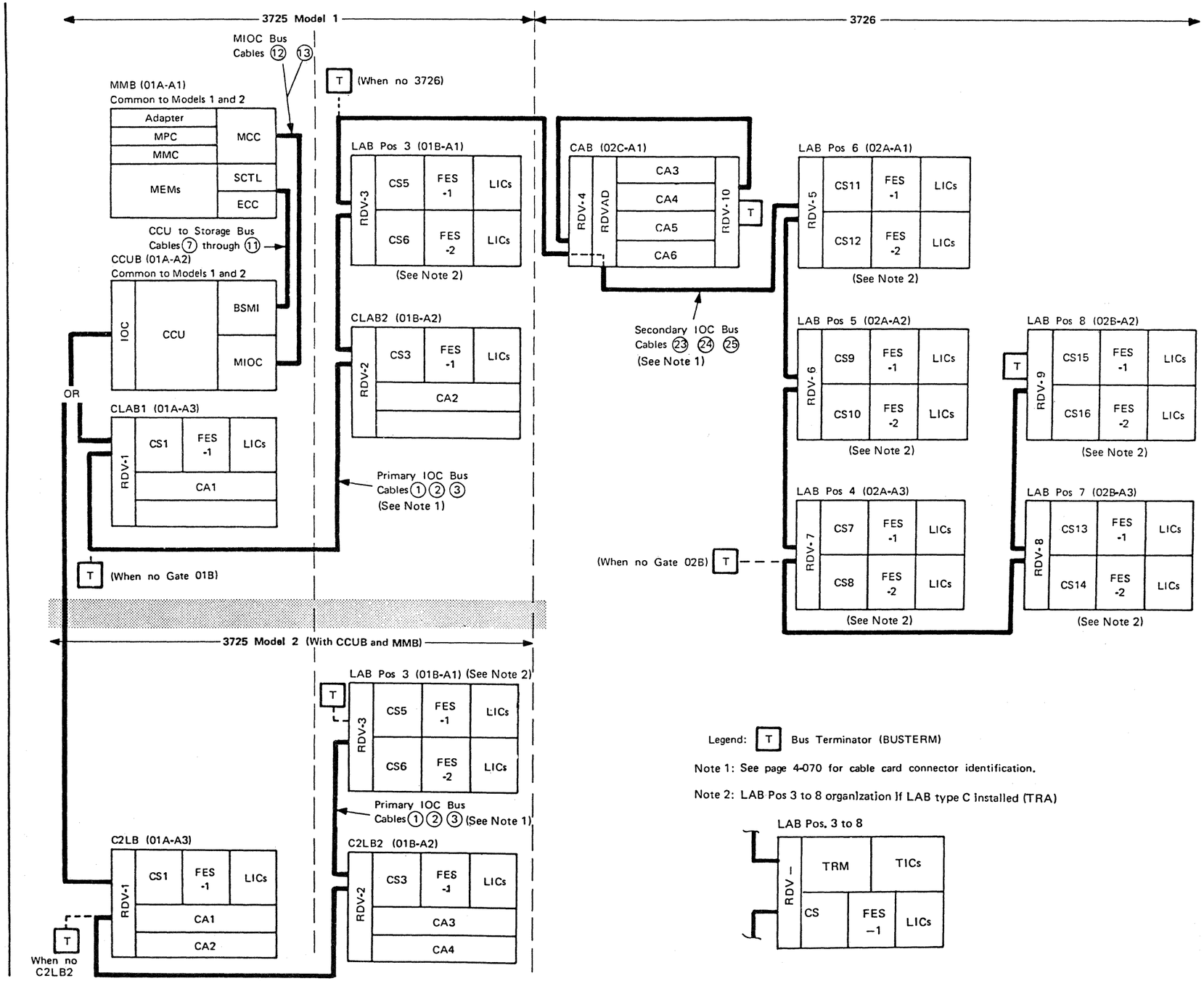
Vdc	Vmin	Vmax	Ripple (max)
-12.0	-10.92	-13.20	0.45V p-p
-8.5	-7.73	-9.35	0.25V p-p
-5.0	-4.55	-5.50	0.15V p-p
-4.3	-4.19	-4.48	0.07V p-p
-1.5	-1.48	-1.56	0.03V p-p
+5.0 (Note 1)	+4.55	+5.50	0.20V p-p
+5.0	+4.75	+5.25	0.13V p-p
+8.5	+7.73	+9.35	0.35V p-p
+12.0	+10.92	+13.20	0.40V p-p
+12.0 (Note 2)	+11.40	+13.20	0.40V p-p
+24.0	+21.00	+27.60	0.30V p-p

Notes:

1. 02-PS7 only
2. 01-PS4 only

TSS Troubleshooting Techniques

- For CSP troubleshooting, see page 13-800.
- For interrupt request to CCU, see page 13-810.
- For cycle steal request to CCU, see page 13-820.



Legend: T Bus Terminator (BUSTERM)

Note 1: See page 4-070 for cable card connector identification.

Note 2: LAB Pos 3 to 8 organization if LAB type C installed (TRA)

Transmission Subsystem Troubleshooting

TRANSMISSION SUBSYSTEM ISOLATION (3725/3726)

By depopulating the IOC bus (see page 11-803), it is possible to change the environment of the whole board (CLAB1, CLAB2, LAB type A, LAB type B, or LAB type C). This can be used when the autoselect mechanism is suspected to be failing. A redrive or its attached adapters suspected of causing a failure can be disconnected by unplugging the top connectors of that redrive (for disconnecting CLAB1 and CLAB2, or disconnecting C2LB, C2LB2, and LAB position 3 boards for Model 2, see page 11-803).

To limit a CLAB1/2 (C2LB, C2LB2 for 3725 Model 2) or a LAB type B or C to a single CSP, it is necessary to unplug the channel adapter or the other CSP or the TRA.

Note: Running 'J' requires the updating of the CDF.

TRANSMISSION SUBSYSTEM CLOCKING

The high-speed clock signals generated by the ACLK card are redriven to the CSP-1 card by the RDV card via a twisted pair (see page 5-051):

Signal name	RDV	Wire color	CSP1-1	CSP1-2
-29.4912 MHz to CSP-1 +29.4912 MHz to CSP-1	B08 B07	Red Black	G03 G02	
-29.4912 MHz to CSP-1 +29.4912 MHz to CSP-1	B10 B09	Red Black		G03 G02

The CSP1 card receives a pulse every 15 us. It is generated by the ACLK card and redriven by the RDV card (see pages 5-053 and 5-054). It is used by the refresh mechanism of the CSP storage.

The CSP2 card receives a pulse every 100 ms generated by the ACLK card and redriven by the RDV card. The transmission subsystem microcode uses it to define various timers.

The 480 Hz signal is used by the FES for the X.21 protocol. It is also used by the LIC as a local clock in the case of locally attached terminals when no ICC is present (see pages 5-053 and 5-054).

The RDV card generates the CLK1 through CLK4 signals to the FES and the ICC as shown on pages 5-053 and 5-054.

CSP TROUBLESHOOTING

CELIA Card

The CSP ROS diagnostics are started by a reset of the IOC bus which resets all the CSPs in the machine. This reset is initiated by the MOSS microcode after a Power On, a Machine Reset, a MOSS Start, or a Diagnostic Request for the IOC bus or the TSS. It can be forced manually by simulating a reset signal on the TSS for a short time by grounding CSP1 pin D02 to ground.

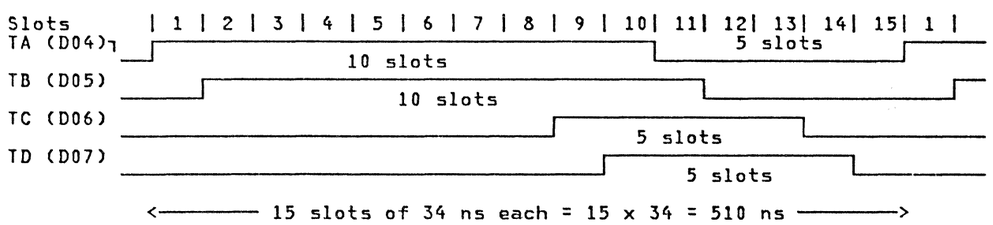
Warning: Do not force a CSP reset while the customer is using other scanners.

The result of the CSP ROS diagnostics is displayed on the indicator card (see page 3-011 for the use of the indicator card and the MIM Part 2 Scanner Troubleshooting Tool section for the analysis of the display.

After a reset, the LED set by hardware (LED 4 or LED 16) must flash (with a delay of approximately 2 seconds after removing the jumper) for a short period of time (approximately 1 second). If this does not happen, the CSP1 processor is failing.

CSP Timing Chart in Cycle Steal Mode

The four CSP1 time slots TA, TB, TC, and TD (not to be confused with the TA, TC, and TD timings of the CCU) are free-running and instruction-dependent. They are shown in the diagram below for the cycle steal mode; this mode can be forced by setting the cycle steal request from the FES (CSP1 pin D10 to ground).



* Pin numbers refer to the CSP1 card.

Legend:
+ : 3.2V
- : 0V

CSP Storage Refresh

The 3725 uses a capacitive type of storage (CMOS) which must be refreshed to maintain the data. The CSP1 card sends a refresh pulse to the storage every 15 us from CSP1 pin P04 to CSM pin B09.

Interrupt Requests to CCU (Part 1 of 2)

PRESELECTION MECHANISM

In the 3725, scanners supporting high-speed lines must be serviced with a higher priority than those supporting medium- and low-speed lines. For this reason, the TSS has a preselection mechanism for the following requests raised by the scanner CSP:

- Level 2 interrupt requests to the CCU
- Cycle steal requests to the CCU.

The preselection mechanism is partly on the CSP and partly on the redrive card.

Each scanner may be given high or low priority; this priority is common to both the level 2 interrupt requests and cycle stealing. The priority level (high or low) is assigned at IPL time by the scanner microcode which sets XR05 bit 5 on (to indicate high priority) or off (to indicate low priority).

INTERRUPT PRESELECTION

When a scanner requires service from the CCU for a specific line, it issues a Level 2 interrupt request to the CCU via the IOC bus. More than one scanner may raise an interrupt request at the same time.

The level 2 preselection mechanism continuously determines which scanner(s) has raised the highest priority level 2 interrupt request (according to the predetermined priority level set by the microcode). The scanner(s) with the highest priority is said to be 'preselected'.

INTERRUPT AUTOSELECTION

To service the level 2 interrupt request, the control program issues a 'Get Line Identification' PIO read instruction. This instruction is decoded by all scanners, starting the autoselection operation. When the instruction is issued, the preselection mechanism stops; no further level 2 requests are taken into account, regardless of their priority level. Only the preselected scanner answers the 'Get Line ID' command. If more than one scanner has interrupt requests of the same priority pending, the scanner that responds depends only on the position reached by the preselection mechanism at that moment in time.

INTERRUPT PRIORITY STRUCTURE

Priority determination is done in three different places:

- Scanner level.

  When two CSPs are plugged into the same LABB, the first CSP (even address) responds first if it has a priority higher than or equal to the priority of the second CSP (odd address).

  The mechanism works as follows (refer to LAB-3 on the diagram): if the first CSP on the board has both a level 2 interrupt request and the priority bit (XR05, bit 5) set, they are ANDed together and raise the '-L2 Priority' line to the RDV card. The '-L2 Sel Out Secondary' signal coming from the redrive card is not propagated to the second CSP.

- Board level

  In the 3725, the boards reply in the following order: CLAB-1, CLAB-2, LAB-3, CAB (frame redrive), or C2LB, C2LB2, and LAB-3 for the 3725 Model 2.

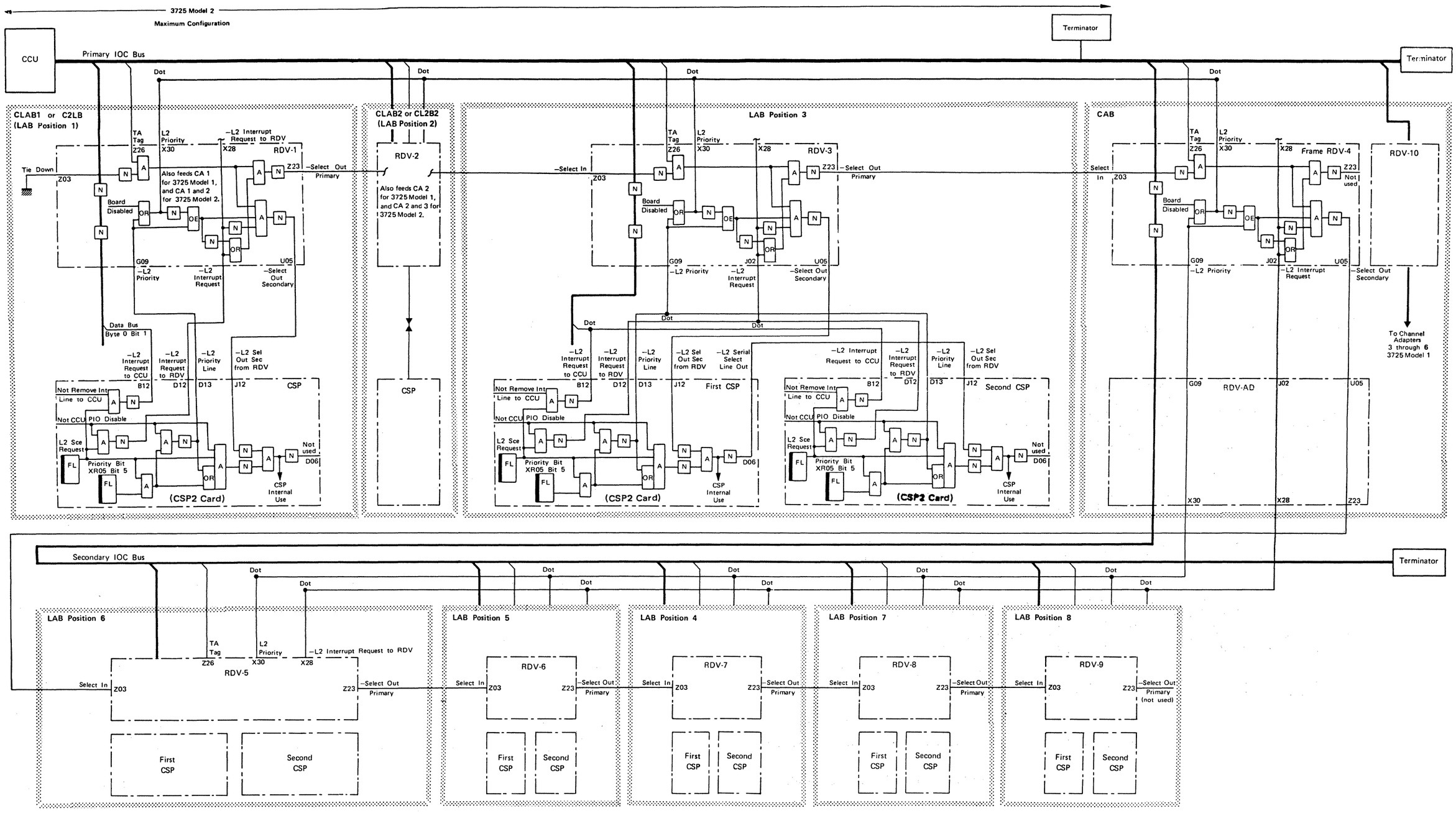
  In the 3726, if none of the boards in the 3725 have replied, the boards reply in the following order: LAB-6, LAB-5, LAB-4, LAB-7, LAB-8.

  The mechanism works as follows (refer to CLAB-1 and LAB-3 on the diagram): suppose that one of the CSPs of LAB-3 has both a level 2 interrupt request and the priority bit on, but that the single CSP of CLAB-1 has only a level 2 interrupt request. The '-L2 Priority' line from the CSPs of LAB-3 passes through the OR of the RDV-3 card to raise the 'L2 Priority' line at pin G09 of the redrive card. This pin is DOT-ORed with the other redrive cards, and causes the CLAB-1 redrive card to pass on the '-Select Out Primary' signal to LAB-3 (via CLAB-2) despite the fact that CLAB-1 itself has a low-priority interrupt pending.

- Frame level

  The frame redrive, which acts as a relay to the secondary IOC bus, can only pass on the selection signal if none of the boards in the 3725 have replied.

Interrupt Requests to CCU [Without TRA (LABC) Installed on any LAB pos 3 to 8] (Part 2 of 2)



Cycle Steal Requests to CCU (Part 1 of 2)

CYCLE STEAL PRESELECTION

When a scanner or TRA requires service from the CCU for a specific line, it issues a CS request to the CCU via the IOC bus. More than one scanner or TRA may raise a CS request at the same time.

The CS preselection mechanism continuously determines which scanner(s) or TRA(s) has raised the highest priority CS request (according to the predetermined priority level set by the microcode). The scanner(s) or TRA(s) with the highest priority is said to be 'preselected'.

CYCLE STEAL AUTOSELECTION

To service the CS request, the CCU issues a 'cycle steal grant'. This line is trapped in the preselected scanner or TRA, and the CS operation starts.

CYCLE STEAL PRIORITY STRUCTURE

Priority determination is done in three different places:

- Scanner or TRA level.

  When two adapters are plugged into the same LABB or LABC, the first adapter (even address) responds first if it has a priority higher than or equal to the priority of the second adapter (odd address).

  The mechanism works as follows (refer to LAB-3 on the diagram): if the first adapter on the board has both the CS request and the priority bit (XR05, bit 5 or TRM CTRL Reg bit 1) set, they are ANDed and raise the '-CS Request' line to the RDV card.

- Board level

  In the 3725, the boards reply in the following order: CLAB-1, CLAB-2, LAB-3, CAB (frame redrive), or C2LB, C2LB2, and LAB-3 for the 3725 Model 2.

  In the 3726, if none of the boards in the 3725 have replied, the boards reply in the following order: LAB-6, LAB-5, LAB-4, LAB-7, LAB-8.

  The mechanism works as follows (refer to CLAB-1 and LAB-3 on the diagram): suppose that one of the adapters of LAB-3 has both a CS request and the priority bit on, but that the single CSP of CLAB-1 has only a CS request. The '-CS Request' line from the adapters of LAB-3 passes through the OR of the RDV-3 card to raise the CS request line at pin M08 of the RDV card. This pin is DOT-ORed with the other redrive cards, and causes the CLAB-1 redrive card to pass on the '-CS Grant Secondary' signal to LAB-3 (via CLAB-2) despite the fact that CLAB-1 itself has a low-priority CS request pending.

- Frame level

  The frame redrive, which acts as a relay to the secondary IOC bus, can only pass on the selection signal if none of the boards in the 3725 have replied.

LIC TO DCE SCOPING

The manual intervention routines that may be used to scope the LIC to DCE interconnection are given on pages 4-140 through 4-240. They require the installation of the wrap block at the tailgate as indicated and the update of the I-field for the given port in the CDF (see pages 2-463 and 2-464).

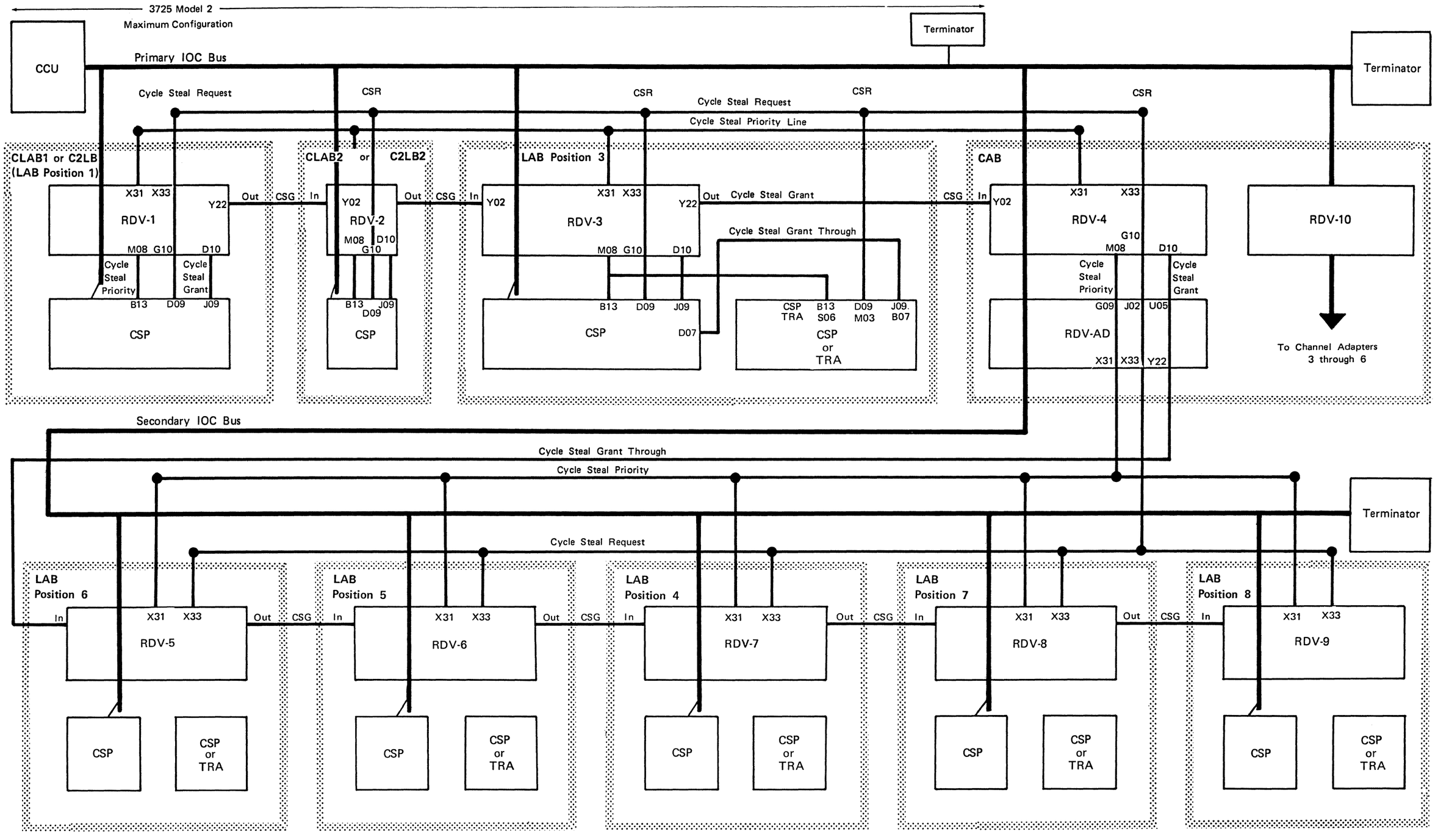
I = 1 for LIC types 1, 2, and 4.

I = 2 for LIC type 3.

Notes:

1. For the physical layout of the LIC top connector, refer to page 4-130.
2. Use the 'Modify' option with C R250 to loop on the routine.

Cycle Steal Requests to CCU (Part 2 of 2)



Contents

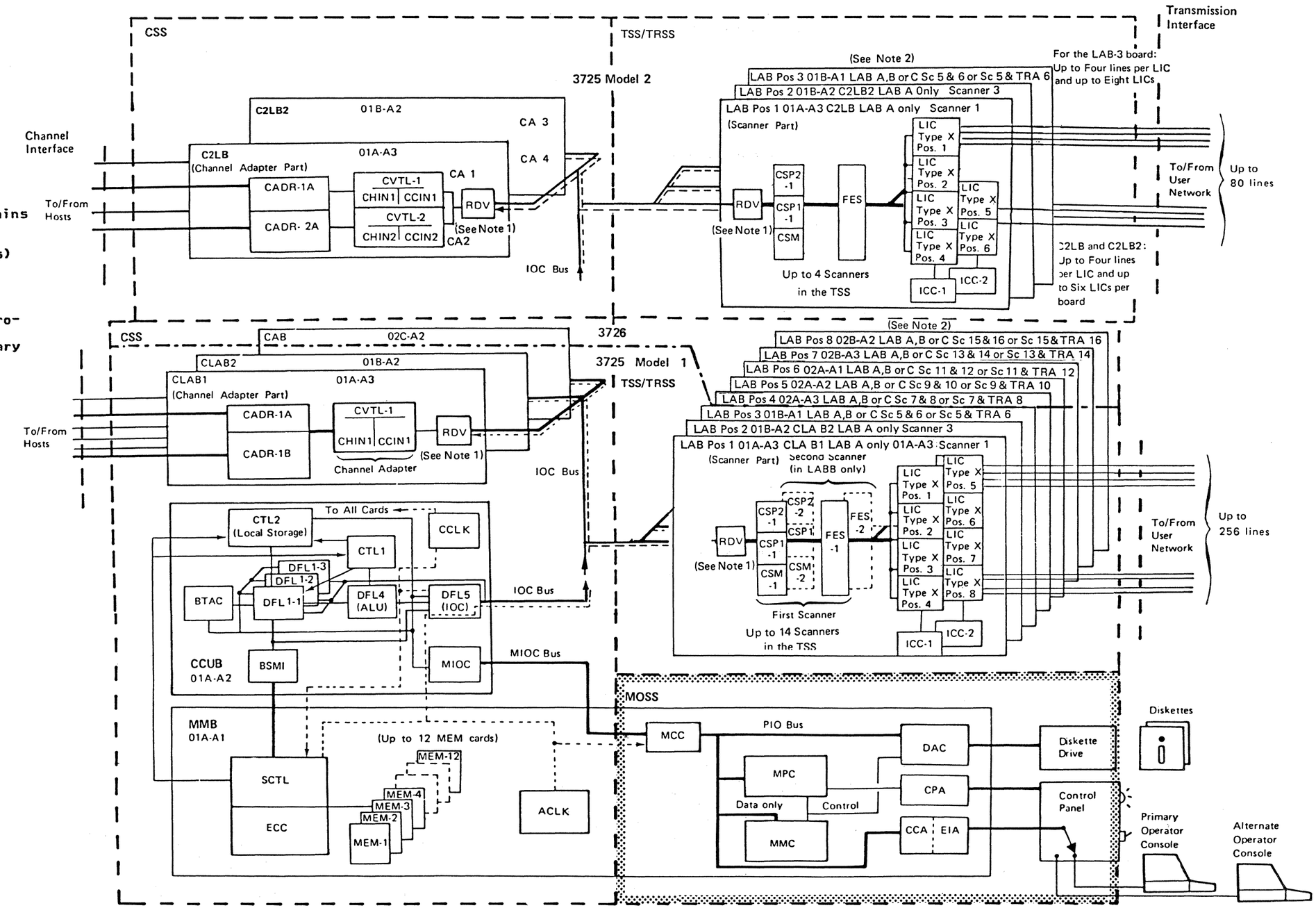
Chapter 14. Maintenance and Operator Subsystem

Section 1. Unit Description

MOSS Description (Part 1 of 3)

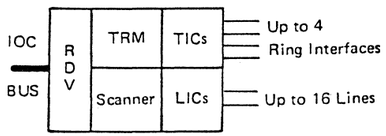
The maintenance and operator subsystem (MOSS) provides access to the 3725 and improves the maintainability of the following components:

- Control program (NCP, EP, PEP)
- Central control unit (CCU)
- Channel adapters (CAs)
- Scanners (CSPs, FESs, and scanner microcode)
- Token-ring adapters (TRA that contains TRM and TIC cards).
- Line interface cards (LICs and ICCs)
- Token-ring interface coupler cards (TICs)
- The MOSS processor itself, its microcode, and the I/O adapters used to attach the control panel, the primary or alternate console, the diskette drive, and the CCU



Note 1: As the CLAB and the CL2B boards are split into two parts in this figure, one for the channel adapter and one for the scanner, the redrive (RDV) function is shown twice for clarity. However, there is one RDV per CLAB or C2LB board.

Note 2: LAB Pos. 3 to 8 organization if LAB type C installed



Legend:

------ Clock signals

——— Data/control signals

MOSS Description (Part 2 of 3)

MOSS FUNCTIONS

Via the MOSS, the service personnel can:

Initialize the 3725 with:

- Hardware checkout diagnostics
- MOSS IML
- CCU IML (CLDP only)
- Scanner IML

Maintain the 3725:

- Run diagnostics
- Box error handling (error recording, analysis and display, alert/alarm generation)
- Line services (line wrap tests, line interface block display)
- CCU control program procedures
- Machine history files: configuration data file (CDF), machine level table (MLT), IPL port table
- Microcode utilities (dumps and file transfer to host, ZAP)

Use 3725 services:

- CCU services
- TSS services
- TRSS Services

The MOSS is composed of seven cards. Four groups of cables connect the MOSS to the 3725 communication controller. The MOSS is interrupt-driven under control of its processor.

MOSS PROCESSOR

The MOSS processor drives the MOSS. It has its own instruction set, and does the usual processing activities: instruction fetch, instruction decode, and instruction execution. The MOSS processor responds to interrupts from the CCU, its attached adapters, and its own error detection circuits. It controls access to the MOSS storage. The MOSS processor is packaged on the MPC card.

DATA FLOW

The data flow external to the MPC card is 9 bits wide, comprising 8 data bits plus parity. Parity is checked on data entering the MPC card, except on 'read MCC status reg 1' and 'diskette read sense byte', on which parity is not checked.

MOSS INTERRUPT LEVEL ORGANIZATION

The MOSS processor has eight interrupt levels, level 0 having the highest priority level.

See page 14-110 for a description of the interrupt system. The table below shows the functions of each level:

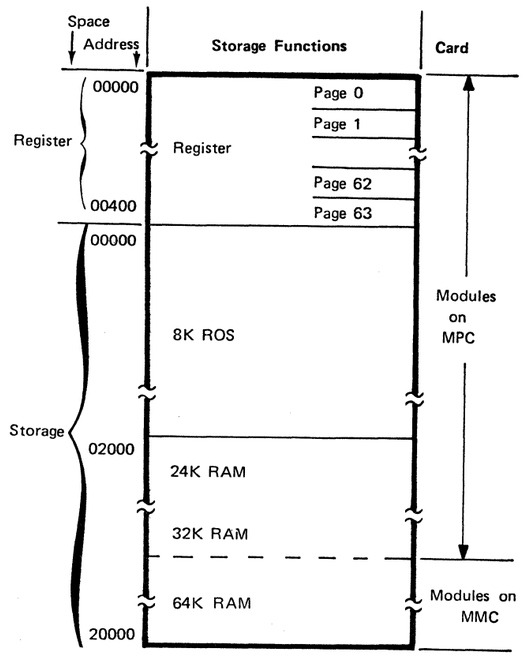
Level	Functions
0	1. MOSS IML 2. Error stored in MCPC register 3. Dump request
1	1. Error (MCPC) 2. CCU high level int. requests 3. 100-ms timer
2	(Not used)
3	Console adapter requests
4	CCU low level interrupt request (scanner interrupt request)
5	Diskette adapter requests
6	MOSS supervisor
7	MOSS tasks

MOSS STORAGES

The MOSS storage is logically divided into two parts: the register space (1K, divided into 64 'pages'), and the storage space. The storage space is physically spread over two cards, the MPC (64K, of which 8K is ROS) and the MMC.

MOSS STORAGE MAP

For details of the register and storage maps, see pages 14-120 and 14-130.



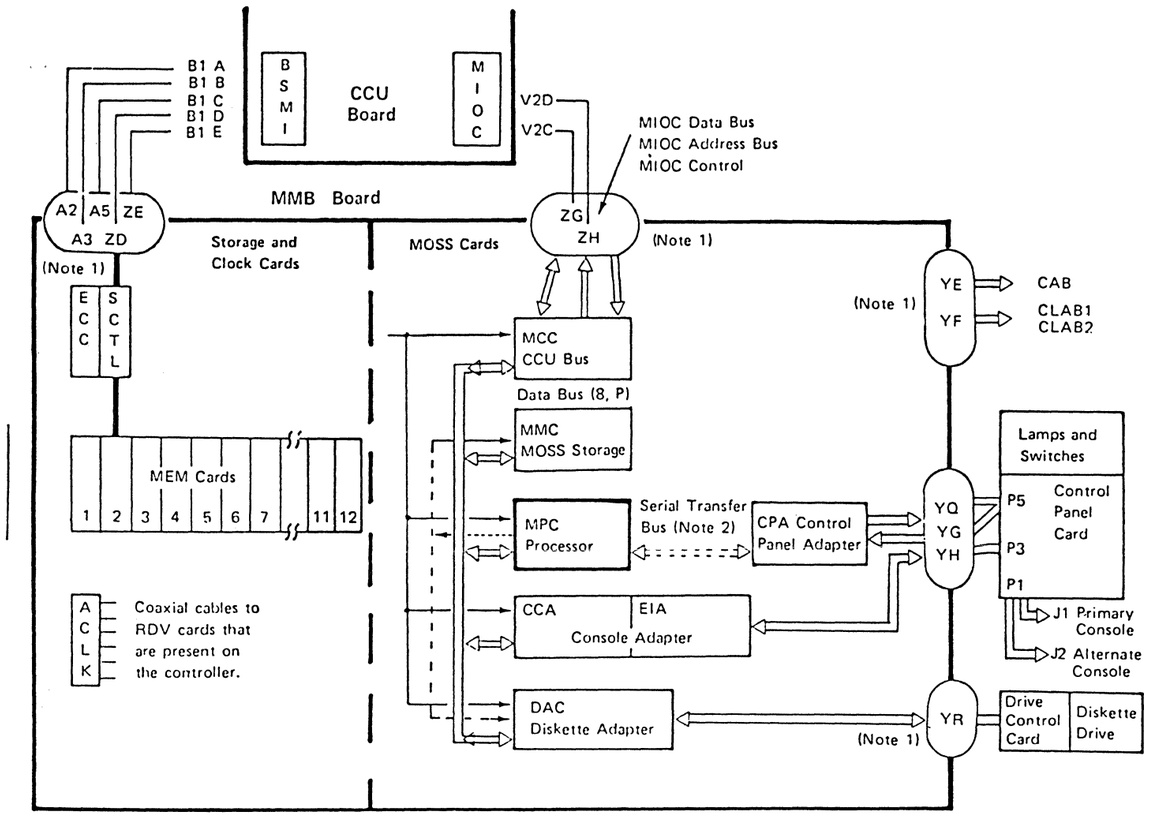
Total: 128K

MPC AND MMC CARDS

See Chapter 5 for module replacement.

MOSS Description (Part 3 of 3)

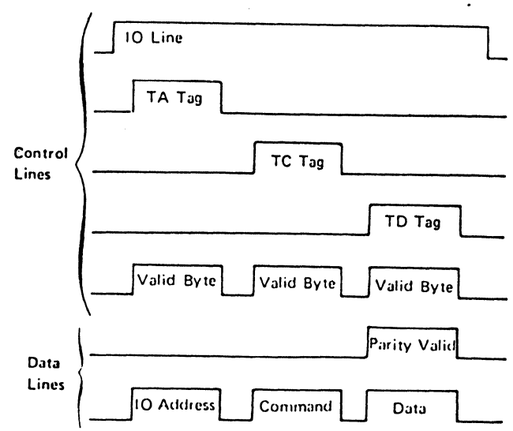
MOSS CARD INTERCONNECTION



Notes:

1. See signal tables in chapter 4.
2. The serial transfer bus and the external PIO bus are separate. During IML check-out, the external PIO bus is disabled to allow MPC and CCA card testing without interference.

PIO BUS PROTOCOL



MOSS/CCU Communication

MIOC AND MCC CARDS

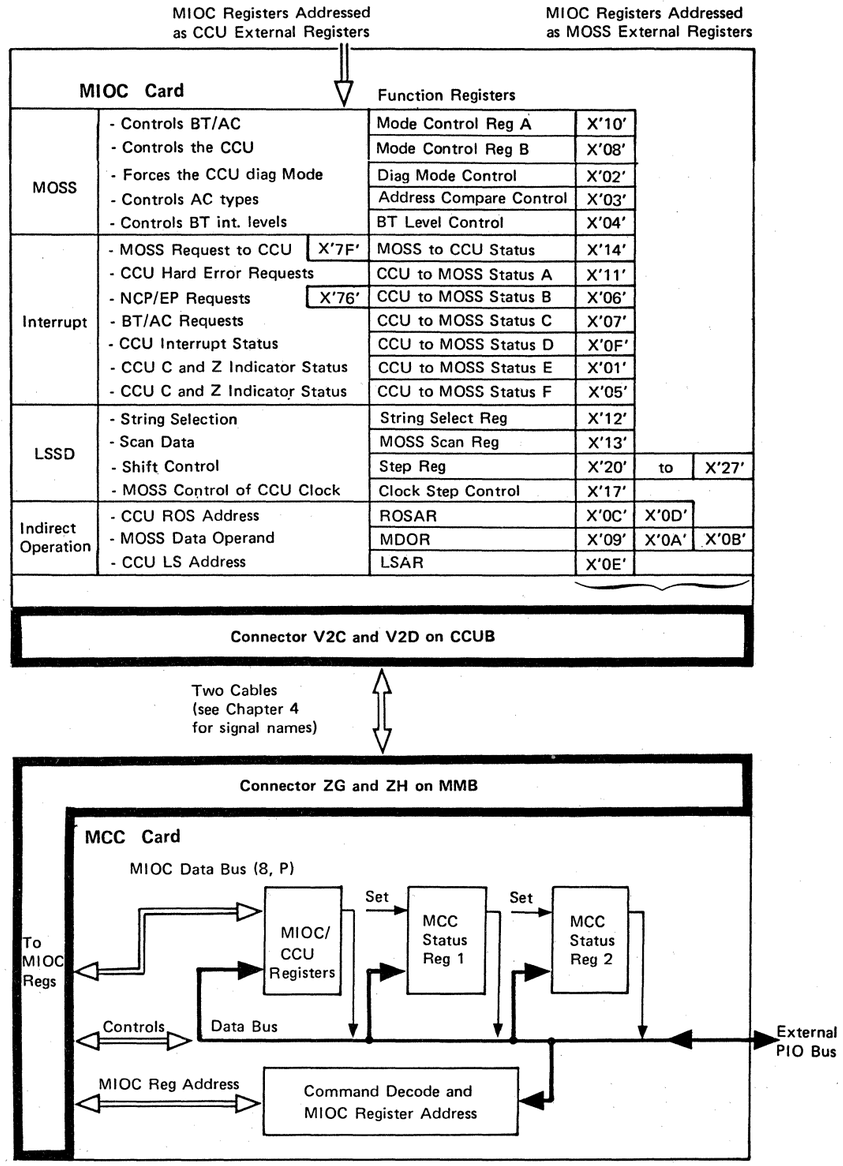
The MOSS communicates with the CCU via the MIOC card (located on the CCU board) and the MCC card (in the MOSS).

The MOSS processor selects the MCC card with its address at TA time, and places command at TC time to read or write at TD time:

- Status registers 1 and 2 (located on the MCC card)
- The MIOC/CCU registers (located on the MCC card) used to access any CCU resources
- Any MIOC register (located on the MIOC card)

Registers on the MIOC card appear as external registers to the MOSS. Some of these registers can be addressed by the CCU using its own register address definition. For example, the 'MOSS to CCU register' is addressed by the MOSS using address X'14', or by the CCU using addresses X'7E' and X'7F'.

There is no direct relationship between the MOSS and CCU register addresses.



MOSS/CCU OPERATIONS

The MOSS executes various operations via the MOSS/CCU connection (MCC/MIOC):

Direct Operations

Direct operations take one CCU cycle to run. In direct operations, the MOSS reads or writes any MIOC register to:

- Check the CCU status (carry or zero flags, current interrupt level)
- Place the CCU in the diagnostic mode
- Prepare the branch trace/address compare circuits
- Control LSSD operations
- Prepare indirect operations

Indirect Operations

In indirect operations the MOSS reads or writes any CCU data flow register, working register, local store, or CCU X'7x' register.

Indirect operations take several CCU cycles to run.

Indirect looped operations allow storage diagnostics (storage scan).

LSSD Operations

In LSSD operations, the MOSS can verify or change the status of any discrete CCU latch. Such operations can be performed after any CCU cycle, by stopping the CCU clock.

LSSD operations are powerful diagnostic tools used especially during IML, or when running CCU diagnostics.

MCC Status Register Description

The two status registers are located on the MCC card. They contain the MOSS-to-CCU status.

MCC Status Register 1

Bit	Functions
0	Enable timer (100 ms)
1	Enable CCU interrupts
2	Enable scanner interrupts
3	MOSS inoperative
4	Timer interrupt (100 ms) (L1)
5	CCU high level interrupt (L1)
6	Scanner interrupt (L4)
7	CCU low level interrupt (L4)

Bit 0: Enable Timer

This bit is set by the MOSS microcode. It is reset at power on, or by the MOSS microcode. When set, bit 0 allows a 100-ms clock to interrupt the MOSS processor on level 1.

Bit 1: Enable CCU Interrupts

This bit is set by the MOSS microcode. It is reset at power on, or by the MOSS microcode. When set, bit 1 allows the two lines from the MIOC (CCU high and low level interrupts) to interrupt the MOSS on levels 1 and 4 respectively.

Bit 2: Enable Scanner Interrupt

This bit is set by the MOSS microcode. It is reset at power on, or by the MOSS microcode. When set, bit 2 allows the scanner interrupt line from the scanners to interrupt the MOSS. This interrupt line is generated by the scanners as bit 0 in XR05. The line is then ORed between all the scanners and sent to the MOSS.

Bit 3: MOSS Inoperative

This bit is set at power on, or by the MOSS microcode. It is reset by the MOSS microcode. When set, bit 3 indicates to the CCU that the MOSS is no longer ready to work (during MOSS IML operation, for example). Signals on the MIOC bus are no longer valid. Bit 3 also turns on the MOSS inoperative lamp on the 3725 control panel.

Bit 4: Timer Interrupt

This bit generates a level 1 interrupt to the MOSS processor. It is set every 100-ms, if the timer has been enabled, by setting status register 1 bit 0. Bit 4 is reset by the MOSS microcode after each interrupt using the reset timer command. If the bit is not reset within 100 milliseconds, the next interrupt is lost.

Bit 5: CCU High-Level Interrupt

This bit corresponds to the CCU high level interrupt line from the MIOC. It generates a level 1 interrupt to the MOSS processor when the CCU interrupt has been enabled by setting the MIOC status register bit 1. The interrupt line from the MIOC is reset when the interrupt condition drops in the MIOC.

The interrupt condition from the CCU may be simulated during diagnostics by setting bit 0 of the status register 2.

Bit 6: Scanner Interrupt

This bit corresponds to the scanner interrupt line from the scanners. It generates a level 4 interrupt to the MOSS processor but only if the scanner interrupts are enabled by setting the MIOC status register bit 2. The interrupt line is reset when all the scanners have reset bit 0 of their XR05 register.

The interrupt condition from the scanners may be simulated during diagnostics by setting bit 0 of status register 2.

Bit 7: CCU Low-Level Interrupt

This bit corresponds to the CCU low level interrupt line from the MIOC. It generates a level 4 interrupt to the MOSS processor when CCU interrupts have been enabled by setting bit 1 of status register 1. The interrupt line from the MIOC is reset by resetting the interrupt condition in the MIOC.

The interrupt condition from the CCU may be simulated during diagnostics by setting bit 0 of status register 2.

MCC Status Register 2

Bit	Functions
0	Interrupt test
1	CCU response timeout
2	(not used)
3	CCU clock check
4	Adapter clock check
5	Adapter check
6	MIOC parity check
7	CCU parity check

Bit 0: Interrupt Test Bit

This bit forces the interrupt lines generated by the CCU or the scanners for testing purposes.

Bit 1: CCU Response Timeout

A timer is enabled every time an MIOC register is selected for a read or a write operation. If a timeout occurs before the completion of the operation, this bit is set on, and a level 0 interrupt is sent to the MOSS processor.

Bit 2: (not used)

Bit 3: CCU Clock Check

This bit is set if a failure occurs in the CCU clock card. It is reset by the microcode.

This bit also sets a level 1 interrupt to the MOSS processor.

Bit 4: Adapter Clock Check

This bit is set when a failure occurs in the adapter clock card. It is reset by the microcode.

This bit also sets a level 1 interrupt to the MOSS processor.

Bit 5: Adapter Check

This bit is set when the machine check line from the MPC card goes on during an I/O operation. It is also set when any other error occurs that sets bit 6 or 7. It is reset by the microcode.

This bit also sets a level 0 interrupt to the MOSS processor.

Bit 6: MIOC Parity Check

This bit is set by the adapter when a parity check is detected on the data byte received from the MIOC during a read operation. It is reset by the microcode.

Bit 7: CCU Parity Check

This bit is set by the MIOC when it detects a parity check on the address bus or the data bus. It sets a parity check line to the MOSS. It is reset by the microcode.

Interrupt Register Description

MOSS-TO-CCU STATUS REGISTER (X'14')

The MOSS raises its requests to the CCU with a Write command X'14'. Any bit on causes an interrupt to the CCU. The CCU reads the request using In X'7E' and X'7F' and resets it via Out X'77'.

MOSS X'14'

Bit	Functions
0	MOSS diag L2
1	MOSS IML L1
2	MOSS diag L3
3	MOSS request SVC L4
4	MOSS response SVC L4
5	ROS operation service request
6	Panel interrupt request L3
7	(not used)

CCU-TO-MOSS STATUS A REGISTER (X'11')

The CCU sets this register via Out X'70', Out X'79', or one of its latches. Any bit on, except bit 3, causes a high-level interrupt to the MOSS. The MOSS reads or resets this register using a MOSS Read X'11' command.

MOSS X'11'

Bit	Functions
0	IOC error during MIOH
1	CCU Out X'79' (program IPL Req)
2	Channel IPL request
3	CCU hardstop (No HLIR)
4	Out X'70'
5	Address exception
6	CCU hardware check
7	MOSS operation check

CCU-TO-MOSS STATUS B REGISTER (X'06')

The CCU sets this register with Out X'71', X'72', or X'76'. Any bit on, except bit 4, causes a low-level interrupt to the MOSS. The MOSS reads or resets this register via a MOSS In X'06'.

MOSS X'06'

Bit	Functions
0	(not used)
1	(not used)
2	CCU program request
3	CCU program response
4	CCU busy (No LLIR)
5	(not used)
6	CCU Out X'71' (pgm display 1)
7	CCU Out X'72' (pgm display 2)

CCU-TO-MOSS STATUS C REGISTER (X'07')

For details, see branch trace/address compare registers on page 14-051.

CCU-TO-MOSS STATUS D REGISTER (X'0F')

This register is set by the CCU hardware, and provides the MOSS with the status of the CCU interrupts. The MOSS reads this register using MOSS In X'0F'.

MOSS X'0F'

Bit	Functions
0	(not used)
1	CCU wait state
2	L1 entered
3	L2 entered
4	L3 entered
5	L4 entered
6	L5 entered
7	(not used)

CCU-TO-MOSS STATUS E REGISTER (X'01')

This register is set by the CCU hardware, and provides the MOSS with the carry and zero condition latches associated with each interrupt level. The MOSS reads this register using MOSS In X'01'.

MOSS X'01'

Bit	Functions
0	C condition latch - L1
1	Z condition latch - L1
2	C condition latch - L2
3	Z condition latch - L2
4	C condition latch - L3
5	Z condition latch - L3
6	C condition latch - L4
7	Z condition latch - L4

CCU-TO-MOSS STATUS F REGISTER (X'05')

This register is set by the CCU hardware, and provides the MOSS with the carry and zero condition latches associated with each interrupt level. The MOSS reads this register using MOSS In X'05'.

MOSS X'05'

Bit	Functions
0	C condition latch - L5
1	Z condition latch - L5
2	Configuration data set A
3	Configuration data set B
4	Configuration data set C
5	Configuration data set D
6	(not used)
7	(not used)

Branch Trace/Address Compare Register Description (Part 1 of 2)

BRANCH TRACE FUNCTION

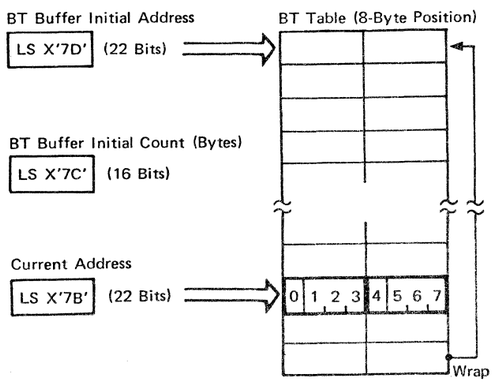
Each time a successful branch is taken, the branch trace function allows the operator to know the interrupt level and storage address from which the control program is coming and where it is going to . When requested, the MOSS loads the branch trace registers with the upper and lower branch trace limits, and with the branch trace table definition (address in main storage and length). The branch trace registers are located in CCU local storage and in the BTAC and MIOC cards.

The MOSS then requests the CCU to record the 'come from' and 'go to' information of any branch taken in the branch trace buffer. Any IAR non-sequential change is considered as a branch taken: true branch, IAR modified by an instruction (load for example), interrupt level entered, or execution of an Exit instruction.

The branch trace information is stored in two contiguous storage positions as follows:

BRANCH TRACE BUFFER

The MOSS defines the branch trace buffer with MOSS Write LS (indirect operations). When the buffer is full, either the CCU is stopped and the MOSS is interrupted, or the MOSS only is interrupted.



If wrap is specified, branch trace buffer full causes the branch trace count to be REINITIALIZED and the branch information to overlay the start of the branch trace buffer.

Branch Trace Extra Records

Under certain circumstances, the branch trace buffer may contain records showing the entry and the exit of the CCU through some program level without instruction execution in that level.

Refer to 3725 Problem Determination Guide and Extended Services Chapter 14, under "Branch Trace Extra Records" for more details.

ADDRESS COMPARE FUNCTION

Address compare allows the comparison of up to two main storage addresses (addresses used during instruction fetch, data read or write, or cycle steal operation) with the address(es) to be compared (stored in AC registers 1 and 2).

Via the MOSS, the operator can request one single, two single, or one double address compare. The type of address that is compared is defined in the AC control register.

Address Compare Control Reg (X'03')

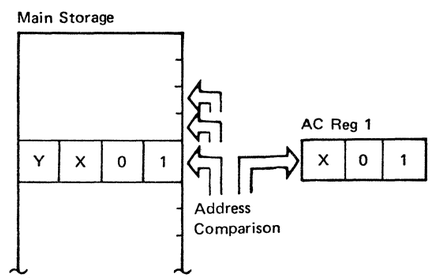
Bit	Functions
0	First AC on cycle steal
1	First AC instruction fetch
2	First AC on storage read
3	First AC on storage write
4	Second AC on cycle steal
5	Second AC instruction fetch
6	Second AC on storage read
7	Second AC on storage write

Bits 0 through 3 control the first address compare; the compare address is stored in AC register 1.

Bits 4 through 7 control the second address compare; the compare address is stored in AC register 2.

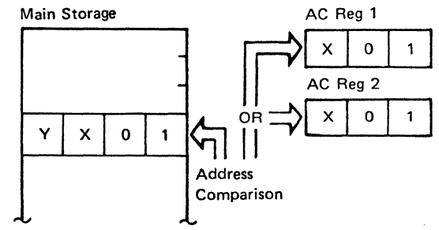
Single-Address Compare

The main storage address is compared with the contents of AC register 1, depending on bits 0 through 3 of the AC control register.



Two Single-Address Compares

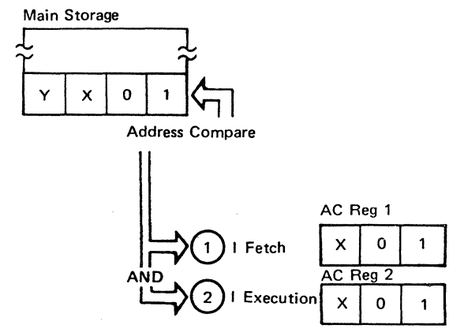
The main storage address is compared with the contents of AC registers 1 and 2, depending on bits 0 through 7 of the AC control register.



Double-Address Compare

Two storage addresses must match with the contents of AC registers 1 and 2 to cause a successful compare.

1. The main storage address used during instruction fetch must compare equal with the contents of AC register 1.
2. If the first compare is successful, and the instruction is load store, the storage address used during instruction execution is compared with the contents of AC register 2.



Branch Trace/Address Compare Register Description (Part 2 of 2)

MODE CONTROL REGISTER A (X'10')

The action to be taken on a successful address compare is defined by this register. It is not accessible by the CCU.

Bit	Functions
0	AC Compare Type bit 1
1	AC Compare Type bit 2
2	Branch trace active
3	Wrap Trace
4	Stop branch trace on AC
5	CCU Stop on AC or BT full
6	CCU L1 on AC
7	Low level interrupt to MOSS on AC or branch trace full

BRANCH TRACE LEVEL CONTROL REGISTER (X'04')

The MOSS sets bits in this register according to the level(s) to be traced. This register is not accessible by the CCU.

Bit	Functions
0	(not used)
1	Trace level 1
2	Trace level 2
3	Trace level 3
4	Trace level 4
5	Trace level 5
6	(not used)
7	(not used)

CCU-TO-MOSS STATUS REGISTER C (X'07')

This register holds the branch trace/address compare results. It is read and reset by the MOSS via In X'07'.

Bit	Functions
0	Branch trace buffer full
1	Successful address compare
2	CCU stop because BT full
3	CCU stop because successful AC
4	AC on Address 1 Reg
5	AC on Address 2 Reg
6	Program stop
7	(not used)

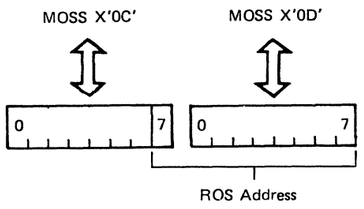
MOSS Indirect Operation Register Description

MOSS indirect operations force the CCU to execute instructions coming from the MOSS. They require the use of the ROS in the CCU. The MOSS prepares these operations by loading the necessary parameters into the MIOC registers (LSAR, MDOR, and ROSAR). The complete sequence follows.

1. The MOSS sets the ROSAR, the MDOR, and the LSAR.

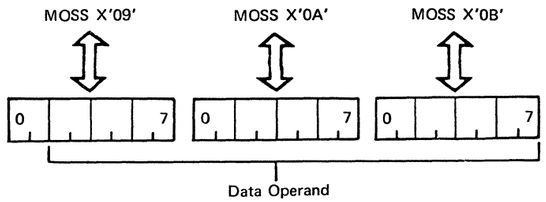
   ROSAR:

   The ROS address register holds the ROS entry address used to execute the MOSS request.



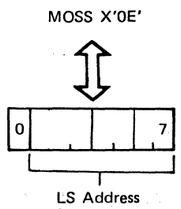
   MDOR (When Write):

   The MOSS data operand register holds the 22-bit data for CCU/MOSS exchanges.



   LSAR:

   The local store address register holds the CCU local store address to be used for MOSS read/write LS operations, or the operand identification for non-read/write LS operations (see LSAR value in the MOSS indirect register tables).



2. The MOSS sets the 'ROS operation service request' (bit 5) in the 'MOSS-to-CCU status reg' X'14'.

3. MIOC sets 'CCU Busy' (bit 4) in the 'MOSS status B reg' X'06'.

4. When the selected ROS operation is finished, the MIOC resets 'CCU Busy'.

5. The MOSS reads the CCU operation result, if any, in the MDOR (when read).

MOSS Indirect Register Read

Register Function	LSAR (Hex)	ROS Address Name	Equate In OP.
RD Op Reg	- -	MOSS RD Op Reg	
RD IAR	- 0	MOSS RD Data Flow	
RD WKR-1	- 1	MOSS RD Data Flow	
RD WKR-2	- 2	MOSS RD Data Flow	
RD WKR-3	- 3	MOSS RD Data Flow	
RD WKR-4	- 4	MOSS RD Data Flow	
RD WKR-5	- 5	MOSS RD Data Flow	
RD WKR-6	- 6	MOSS RD Data Flow	
RD WKR-7	- 7	MOSS RD Data Flow	
RD Z Reg	- 8	MOSS RD Data Flow	
RD SAR	- 9	MOSS RD Data Flow	
RD Op Reg 1.2, 1.7 and Pop 0.0, 1.7	- A	MOSS RD Data Flow MOSS RD Data Flow	
RD A Reg in IOC	- B	MOSS RD Data Flow	
RD D Reg in IOC	- C	MOSS RD Data Flow	
(not used)	- D		
RD LAR	- E	MOSS RD Data Flow	
(not used)	- F		
RD LS Address 00 to 7F	00-7F	MOSS Read LS	00, 6F, 71 72, 78, 7B 7C
RD SP/AE Key	73	MOSS Read Control	73
RD IOC CCW	75	MOSS Read Control	75
RD IOC Error Status	76	MOSS Read Control	76
RD Adap L2, L5 Int. Req	77	MOSS Read Control	77
RD Probe and C/Z	79	MOSS Read Control	79
RD High Resol Timer	7A	MOSS Read Control	7A
RD CCU Hard Checks	7D	MOSS Read Control	7D
RD L1 Int. Request	7E	MOSS Read Control	7E
RD CCU L2, L4 Int Reg	7F	MOSS Read Control	7F
RD STG Size from BSM	70	MOSS Read Control	70

MOSS Indirect Register Write

Register Function	LSAR (Hex)	ROS Address Name	Equate In OP.
Write SAR	- -	MOSS Write SAR	
Write WSDR	- -	MOSS Write WSDR	
Write BT Upper Limit	- -	MOSS Write BT/UL	
Write BT Lower Limit	- -	MOSS Write BT/LL	
Write AC Address 1	- -	MOSS Write AC/AD1	
Write AC Address 2	- -	MOSS Write AC/AD2	
Write IAR in Data Flow	-8	MOSS Write IAR	
Write WKR-1 in Data Flow	-1	MOSS Write WKR-1	
Write WKR-2 in Data Flow	-2	MOSS Write WKR-2	
Write WKR-3 in Data Flow	-3	MOSS Write WKR-3	
Write WKR-4 in Data Flow	-4	MOSS Write WKR-4	
Write WKR-5 in Data Flow	-5	MOSS Write WKR-5	
Write WKR-6 in Data Flow	-6	MOSS Write WKR-6	
Write WKR-7 in Data Flow	-7	MOSS Write WKR-7	
Set CCU Hardstop	70	MOSS Write Control	70
Miscellaneous Control	76	MOSS Write Control	76
Miscellaneous Control	77	MOSS Write Control	77
Set Force ALU Control	78	MOSS Write Control	78
Utility	79	MOSS Write Control	79
High RESOL Timer Ctrl	7A	MOSS Write Control	7A
Set PCI L2	7B	MOSS Write Control	7B
Set PCI L3	7C	MOSS Write Control	7C
Set PCI L4	7D	MOSS Write Control	7D
Set Program Int. Mask	7E	MOSS Write Control	7E
Reset Program Int. Mask	7F	MOSS Write Control	7F
SP/AE Control	73	MOSS Write Control	73
Write LS Address 00-7F	0-7F	MOSS Write LS	

LSSD Operation (Part 1 of 2)

In level scan sensitive design (LSSD) operations, the MOSS can verify or change the status of any discrete CCU latch. Such operations can be performed after any CCU cycle, by stopping the CCU clock.

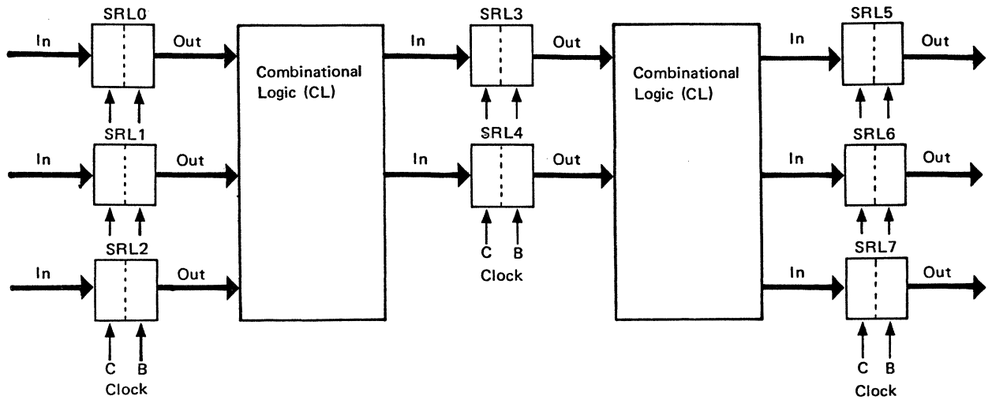
LSSD operations are powerful diagnostic tools used especially during IML, or when running CCU diagnostics.

All the circuits composing the CCU can be represented with two types of element:

1. The shift register latch (SRL), and
2. The combinational logic (AND, OR, and XOR circuits).

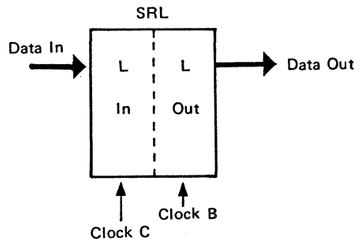
Notes:

1. The A-clock is the LSSD clock.
2. The B-clock is free-running.
3. The C-clock is the normal operation clock.
4. The A and C clocks are mutually exclusive.



SHIFT REGISTER LATCH

Two latches compose an SRL: 'latch in' and 'latch out'.



Data Flow Simplified Representation

The following is a simplified representation of the data flow. Normally, the complete CCU can be represented with 12 strings each having an average of 100 SRL statuses.

The 'SRLs out' lines present their status to the following combinatory logic at clock pulse B. The resulting combinatory logic values set the 'SRLs In' at next clock pulse C.

LSSD Testing Circuit

For testing purposes, the above 'SRLs Out' are connected to 'SRLs In' in a sequential string as follows. The 'In' and 'Out' lines go to their combinational logics. 'Scan In' is the string input; 'Scan Out' is the string output.

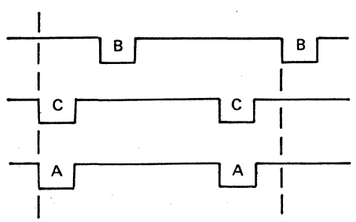
With LSSD operations the MOSS can:

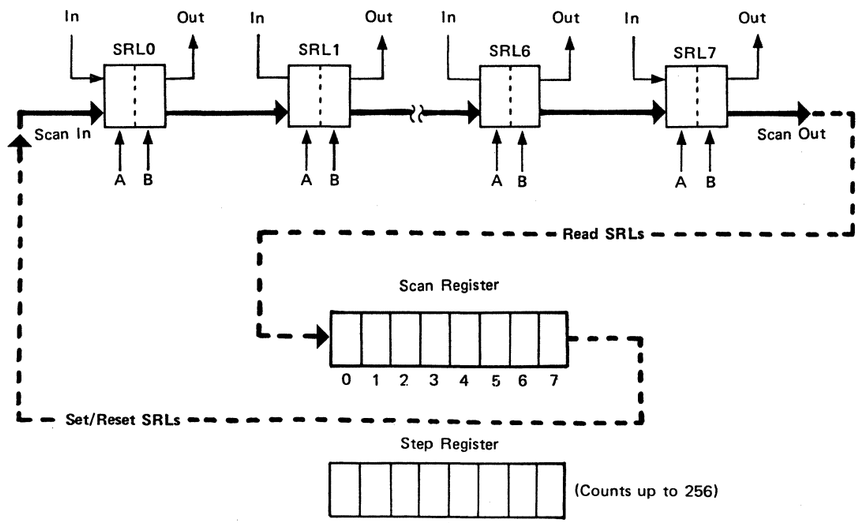
1. Read the CCU SRLs for diagnostic purposes ('scan out' to MOSS scan reg).
2. Set/reset the CCU SRLs to a preset value for initial CCU reset (scan reg to 'scan in').

SRL Operation

- 'SRL in' is set with 'data in' at clock pulse C. 'SRL out' is not affected.
- At clock pulse B, an internal shift occurs, transferring the 'SRL in' value to 'SRL out', and then to the 'data out' line.

Clock Pulse Timing (Not Displayable)





LSSD Operation (Part 2 of 2)

SRL READ OPERATION

To read the status of the eight SRLs, the MOSS:

1. Stops clock pulse C to freeze the CCU data flow
2. Advances clock pulse A (LSSD clock) eight times to shift out the contents of the SRLs onto the 'scan out' lines
3. Shifts into the scan register the SRLs value that appears on the 'scan out' line
4 Reads the scan register contents

The MOSS first places the number of shifts to perform (number of SRLs to read) in the step register (X'20' through X'27').

SRL SET/RESET OPERATION

To set SRLs to a predetermined value, the MOSS uses the same registers, and then:

1. Stops clock pulse C
2. Loads the scan register with the SRL value to be set
3. Advances clock pulse A eight times to shift out the contents of the scan register to the SRLs on the scan in line.

CCU STRINGS

The complete CCU is divided into 12 strings. The MOSS places the selected string address in the string select register.

String	Card
1	MIOC
2	BTAC and CCLK
3	CTL1
4	CTL2
5	DFL1-3 *
6	DFL1-3 *
7	DFL1-2 *
8	DFL1-2 *
9	DFL1-1 *
10	DFL1-1 *
11	DFL1-4
12	DFL1-5

* The DFL1-1, DFL1-2, and DFL1-3 cards have the same part number.

LSSD REGISTER DESCRIPTION

String Select Register (X'12')

Bit	Function
0	C-clock stop
1	(not used)
2	MIOC diagnostic
3	(not used)
4	String address (bit 0)
5	String address (bit 1)
6	String address (bit 2)
7	String address (bit 3)

Bit 0: C-Clock Stop

The MOSS sets on this bit to stop the C-clock and perform LSSD operations.

Bit 2: MIOC Diagnostic

Bits 4 through 7: String Address

The MOSS places the CCU LSSD string to scan in or scan out in these bits.

MOSS Scan Register (X'13')

Bit	Function
0-7	Scan data

This register holds the actual data bits to be scanned in, or that have been scanned out.

Clock Step Control Register (X'17')

Bit	Function
0	Clock step mode
1	Step 1 CCU cycle
2-7	(not used)

This register controls CCU clock stepping. When the CCU operates in clock step, the interval timer interrupt requests (level 3) are masked.

Bit 0: Clock Step Mode

When on, this bit stops the clock pulse C to the whole CCU data flow, except for those latches and registers involved in LSSD operation (direct MOSS/MIOC, LSSD registers, and the clock step control register).

Bit 1: Step 1 CCU Cycle

When on, this bit causes the CCU to execute a clock C pulse cycle (if the CCU is in the clock step mode). The stepped clock C cycle automatically resets this bit.

Step Register (X'20' to X'27')

Bit	Function
0-6	Shift control bit

Bits 0 through 6: Shift Control

These bits control MOSS scan register shifting during LSSD operation. The register address also selects the number of shifts to be performed:

Register Address	
X'20'	1-bit right shift
X'21'	2-bit right shift
X'22'	3-bit right shift
X'23'	4-bit right shift
X'24'	5-bit right shift
X'25'	6-bit right shift
X'26'	7-bit right shift
X'27'	8-bit right shift

Bit 7: MOSS Shift Mode

When on, the clock A pulse is activated to indicate that an LSSD operation is in progress. In this case, clock C (for normal operations) does not run. LSSD operations can also be performed in the clock step mode.

MOSS/Diskette Drive Interaction (Part 1 of 2)

The diskette drive itself is described in Chapter 7.

DISKETTE COMMANDS

The MOSS processor controls the diskette drive adapter and its attached drive via programmed I/O commands on the MOSS external PIO bus. These commands allow the MOSS to:

- Start or stop the diskette drive motor
- Set or sense the file adapter registers
- Seek head carriage
- Engage the heads (position the heads close to the diskette surface)
- Read data from the diskette
- Write data to the diskette
- Read back the data for checking purposes

The diskette adapter rejects commands that are invalid, or that have a bad parity. A command that temporarily cannot be serviced by the diskette drive is considered invalid. The command is not retried. It causes an machine check/program check end status indication.

READ/WRITE OPERATIONS

Read or write operations are initiated via PIO commands. Data is exchanged directly with the MOSS storage through the direct memory access (DMA) function of the DAC (an operation similar to cycle stealing).

A read or write operation can only begin 35-ms (head settling delay) after a previous seek operation is completed.

Read or write operations need specific parameters that are passed to the file adapter in PIO mode. These parameters, which are entirely microcode-dependent, identify:

- The diskette side (head select) and the track selected
- The addressed sector on the track
- The number of sectors to be read or written
- The selected recording mode:
  - Frequency modulation for track 0
  - Modified frequency modulation for all other tracks
- In seek operations, the number of tracks and the direction of displacement

TIMEOUT CONSIDERATIONS

Diskette heads must be engaged at least 80 ms before a read or write operation can start. To minimize diskette wear, the head engage operation is initiated at random on a track.

The heads are disengaged when not in use. They may remain engaged during seek operations.

The drive circuits automatically disengage the heads if no operation is performed for six diskette revolutions. This prevents machine checks.

The diskette drive motor is powered on and off at MOSS request. MOSS sends a motor power-off command if the diskette drive stays inactive for 7 minutes. After a power-off command, the drive motor requires a maximum of 10 seconds to reach its working speed.

DISKETTE ADAPTER STATUS REGISTER

This allows the MOSS to read the status of the diskette adapter.

Bit	Function
0	Exception
1	Busy
2	Index mark interrupt
3	Head engaged
4	5-ms timer interrupt
5	Machine check
6	Interrupt enabled
7	Operation complete interrupt

Bit 0: Exception

This bit specifies any diskette error condition. It is defined in the sense byte (see next page).

Bit 1: Busy

The adapter is performing a seek operation or a data transfer. It remains active from the start of an operation request to the rise of the operation complete interrupt.

Bit 2: Index Mark Interrupt

This condition occurs only if enabled. The interrupt becomes active when an index mark is detected during a diskette revolution. This allows the MOSS to test the diskette and ensure that it is revolving at the proper speed. This operation cannot be done while another diskette operation is being performed.

Bit 3: Head Engaged

Indicates that the heads have been engaged.

Bit 4: 5-ms Timer Interrupt

The DAC cards contain a timer that may be used by the software to count 5-ms periods. When the timer is enabled, the counter raises interrupts every 5-ms until it is disabled. This feature may be used to time the 80-ms head engage time prior to a write or read, and the 35-ms head settling time after a seek operation. The timer function is inoperative while the adapter is busy.

Bit 5: Machine Check

This results from an invalid command, a parity error on the command or data to the adapter during a PIO instruction, or from a command temporarily rejected. This condition must be cleared before an operation can start on the diskette.

Bit 6: Interrupt Enabled

This bit, when on, allows the timer or operation-complete interrupts to appear on the interrupt bus.

Bit 7: Operation-Complete Interrupt

This bit indicates the completion of an operation.

MOSS/Diskette Drive Interaction (Part 2 of 2)

DISKETTE SENSE BYTE

Bit	Function
0	Modulation mode
1	Control record or access 0
2	Overrun or access 1
3	Write current
4	I/O bus parity error
5	Data field or diskette sense
6	No record found
7	Read CRC or write control error

Bit 0: Modulation Mode

This bit specifies the type of recording on the diskette:

- Bit 0 = 0: frequency modulation mode (FM)
- Bit 0 = 1: modified frequency modulation mode (MFM)

Bit 1: Control Record or Access 0

The control record indicator is set during a read data or read back check request if the specified sector is flagged as a control record in the diskette format. Multisector operations are terminated at the detected sector.

Access 0 indicates the state of the control latch that sequences one line of the file stepping motor during a 'seek' function request.

Bit 2: Overrun or Access 1

The overrun condition occurs if the direct storage access fails to service the file during data transfer.

Access 1 is the state of the second control line to the file stepping motor.

Bit 3: Write Current

The write current bit indicates either a normal or an exception condition, depending on the time during which it is sampled. The acceptable 'window' is from the time that the write data request is accepted until the interrupt indicating the completion of the request. An error occurs when the bit is set outside of this window; the MOSS terminates the diskette operation.

Bit 4: I/O Bus Parity Error

This error occurs when bad parity is detected by the adapter for commands or data transferred to the adapter via PIO instructions. The error is also indicated for incorrect data transfers during direct access storage operations or internal register loading.

Bit 5: Data Field or Diskette Sense

The data field indicator is set if an error occurs for the data record of a specific sector. The condition may be 'no record found', CRC or write control error.

The 'diskette sense' bit may be used to determine the type of diskette present in the drive. It is 0 for a type 1 diskette and 1 for a double-sided diskette (type 2 or 2D).

Bit 6: No Record Found

This error condition occurs if the label specified in a read or write type function request is not found on the track currently under the read/write head.

The error also occurs if the correct sector is located, but the data cannot be read from the diskette; in this case, bit 5 ('data field') is also set.

Bit 7: CRC or Write Control Error

These two error conditions are mutually exclusive. A CRC error can only occur for an incorrect data field during a 'read data' or 'readback check' request.

The 'write control error' indicates an adapter hardware malfunction when attempting to write data to the diskette.

DISKETTE MAPPING

The controller and the service diskette space is allocated at diskette generation time, with header labels on cylinder 0.

Copies of the controller and the service diskettes are included as spare diskettes in the shipping group. Their updating is under the responsability of the CE using the 'diskette swap' utility function (page 2-408).

The diskette mapping is as follows:

(1 track = 6.5 Kbytes)

Controller Diskette Mapping

Mapping	Tracks
Disk Management --- Volume label, Header labels, label directory	2
Microcode Modules --- - MOSS - Diagnostics subset	 92 11
3725 Initialization Control --- - 3725 Loader dump program (CLDP) - Rolled out area of CCU storage	 2.0 2.5
Utility Data Sets --- - Catalogued CP procedures - BER file - Configuration data file - Machine load table	 3 1 0.5 0.5
Buffer Areas for Dump --- - For one MOSS or scanner dump (CHGDMP) - For four TIC dumps (CHGTRSS)	 19 3.5
Spare ---	1.0

Service Diskette Mapping

Mapping	Tracks
Disk Management --- Volume label, Header labels, label directory	2
Microcode Modules --- - MOSS subset - Diagnostics	 45 84
Utility Data Sets --- - BER file - Configuration data file - Machine load table	 1 0.5 0.5
Spare ---	2

I/O Connections

MOSS/CONSOLE CONNECTIONS

The primary and alternate consoles are attached to the processor via the CCA and EIA cards. The MOSS can control only one console at a time. One cable connects the EIA card to the control panel, where it is routed to the primary or alternate console by the console switch.

For details of the operator console, see chapter 6.

For details on the connectors and line connections, see page 14-012.

CCA Card Functions

- Buffer the transmitted/received bytes (one at a time).
- Serialize/deserialize the data.
- Control the number of bits transmitted.
- Control the transmit/receive bit timing.
- Control the DCE interface leads.
- Add/delete the start/stop bit.
- Check for received byte parity.
- Detect the 'ATTN' condition.

The CCA transmits and receives in start-stop mode, at 2400 bps, using ASCII code with a stop bit only.

CCA Basic Status Register

This register allows the MOSS to read the status of the CCA card and the attached 3727 console.

Bit	Function
0	Input request
1	Output request
2	DCE interrupt
3	Timer interrupt from adapter
4	Exception interrupt
5	Machine check
6	Adapter enabled
7	Interrupt pending

EIA Card Control

This card adapts the 3725 voltage levels (VTL) to EIA line voltages. For signal distribution, see page 6-040.

MOSS/CONTROL PANEL CONNECTIONS

The control panel adapter (CPA) card attaches the MOSS to the control panel. The connections to the processor uses a serial data transfer facility on the MPC driven by the PIO internal bus. The MOSS can disable the external PIO bus to permit testing the MOSS processor itself, the CPA card, and the control panel without interference from the external bus.

For details of the connectors and line connections, see page 14-012.

MOSS Microcode (Part 1 of 2)

GENERAL INFORMATION

The MOSS microcode provides for the functions of the different MOSS applications. These applications schedule the various tasks and communicate with the adapters controlled by the MOSS (the CCU is considered as an adapter by the MOSS) and with the devices.

Three types of microcode applications must be considered:

System Applications

System applications are resident and provide information on the system state, or record errors occurring in the system (machine status area, BER recording/alert, CCU re-IML).

Operator-Selected Applications

One application at a time is under the control of the operator to provide interactive functions.

Background Application

This application allows the host system to transfer a dump data set (MOSS or CSP) or to support REQMS/RECFMS.

MAILBOX PROTOCOL

Refer to page 14-140.

MOSS STATES

Some MOSS states are passed to the NCP or the EP ('INOP', 'offline', 'online'). Other states exist when the NCP or EP is not loaded in the CCU ('down', 'alone', 'service mode').

- 'MOSS INOP' indicates that the microcode is not running in the MOSS or that MOSS IML is in progress. In this state the 'MOSS inoperative' line is on and CCU interrupts are disabled. This state is transitory.
- 'MOSS online' indicates that the MOSS microcode is running and allows communications with the NCP/EP via the mailbox protocol. In this state, the 'MOSS inoperative' line is off and the CCU is enabled.
- 'MOSS offline' indicates that the MOSS microcode is running in the MOSS, but that communication through the mailbox protocol must not be established with the NCP/EP (except mailboxes exchange with CLDP and NCP/EP initialization, which are part of the 3725 system IPL). In this state, the 'MOSS inoperative' line is off and the CCU is enabled.
- 'MOSS alone' indicates that only the MOSS microcode is loaded and only the MOSS is operational. The 'MOSS inoperative' line is off.
- 'Service mode' indicates that the MOSS is loaded with the diagnostic control monitor. The 'MOSS inoperative' line is off, the CCU can be enabled, and diagnostic programs can be operational in the CCU.

MOSS Microcode (Part 2 of 2)

MOSS CHANGES OF STATE

The following is a description of the events and actions that cause the MOSS state to change. Numbers identify events and actions in the figure (right).

1. Power On switch. The result depends on the Function Select switch position:

   a. 'MOSS IML' with controller diskette mounted:

      The MOSS INOP state is entered for the duration of MOSS IML, then the MOSS ALONE state is entered (10) and is kept until further action.

   b. 'Normal' with controller diskette mounted:

      The MOSS INOP state is entered for the duration of MOSS IML, then the MOSS ALONE state is entered (10), followed by MOSS OFFLINE (4), then MOSS ONLINE (5).

      This is the full 3725/3726 IPL.

   c. 'Maintenance' or 'MOSS IML' with service diskette mounted.

      The MOSS INOP state is entered for the duration of MOSS IML, then the SERVICE MODE state is entered (11).

2. Function Start switch. Depending on the position of the Function select switch, the same sequence occurs as in (1) above.

3. Power Off switch. The entire controller leaves its current state, whatever this state is.

4.

   a. A controller initialization has been requested (1 or 2).

   b. A controller initialization has been requested from the keyboard service menu.

5.

   a. A controller initialization has been requested (1 or 2).

   b. A controller initialization has been requested from the keyboard service menu.

   c. The MOSS Operator entered a MOSS ONLINE command from the keyboard to terminate the MOSS maintenance, which was performed in OFFLINE state.

6.

   a. The MOSS operator entered a MOSS OFFLINE command from the keyboard for MOSS maintenance purposes.

   b. A hardware error occured on the MIOC during the processing of a mailbox (in/out).

   c. A timeout occurred during the processing of a mailbox in.

   d. MOSS re-IMLed from the control panel (Function Start switch)

7.

   a. During a CCU IML or a controller initialization, the process aborted and cannot be successfully completed.

   b. A channel IPL request or a control program IPL request was presented to the MOSS.

   c. A CCU hardcheck was presented to the MOSS.

8.

   a. A channel IPL request or a control program IPL request was presented to the MOSS.

   b. A CCU hardcheck was presented to the MOSS.

9.

   a. A MOSS abend occured: Any problem in the MOSS microcode or mailbox in is rejected as invalid (command, parameters, MOSS INOP).

   b. Function Start switch was pressed while the MOSS was in OFFLINE, ONLINE, or ALONE state.

10.

   a. If the reason for entering the MOSS INOP state was abend, after a MOSS re-IML the final state is the one that was interrupted by the abend (that is, MOSS OFFLINE, ONLINE, or ALONE).

   b. If the reason for entering the MOSS INOP state was the Function Start switch, after a MOSS re-IML the final state depends on the position of the Function Select switch. If set to MOSS IML, the final state is the same as the one which was interrupted, except for MOSS ONLINE which is forced to MOSS OFFLINE.

11. A MOSS re-IML or LOAD DIAG is performed with the service diskette.

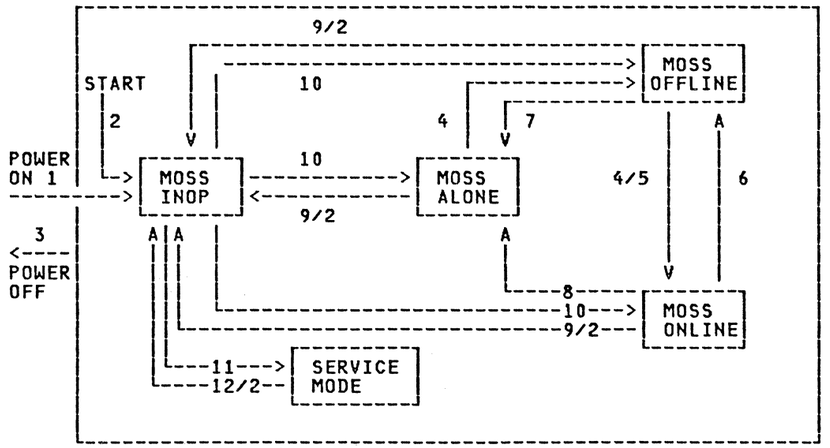
12.

   a. A MOSS abend occured because of any problem in the MOSS microcode or in the diagnostic control monitor.

   b. Function Start switch was pressed while MOSS was in SERVICE mode.

   c. An unexpected error found by the diagnostic control facilities requires a manual IML.

The following figure summarizes these states and their relationship.



Note: Numbers refer to events and actions (see following descriptions).

Interrupt Levels

The MOSS interrupt levels are organized to give the highest priority to error detection and processing, followed by the I/O processing levels, and finally the supervisory level and the task level.

LEVEL 0

This highest priority level is used initially for MOSS IML. Once the 3725 is operational, it is used for detecting errors, such as machine check/program check and abend requests.

The MOSS debugging functions (dump) also run on this level.

LEVEL 1

- This level is used to process errors, such as I/O errors, and MOSS processor errors not reported via the MCPC register. Error recoveries are initiated by this level to recover from I/O adapter errors.
- The CCU high-level interrupt is detected on this level for the following errors:
  - CCU hardcheck
  - Control program request IPL
  - Channel request IPL
  - Program output X'70' sense
  - IOC error
  - MOSS operation check
  - Addressing exception check (MOSS)
  - CCU clock check
  - Adapter clock check
- 100-ms timer interrupt is detected on this level.

LEVEL 2

This level is not used.

LEVEL 3

This level supports the communication common adapter (CCA) to which is attached the keyboard/display console(s).

LEVEL 4

This level handles CCU low-level interrupts:

- Program display 1 and 2
- CCU address compare interrupt
- CCU address compare CCU Stop
- Two single-address compares on address 1
- Two single-address compares on address 2
- CCU branch trace interrupt
- CCU branch trace CCU stop
- CCU program request
- CCU program response
- Scanner interrupt

LEVEL 5

This level supports the diskette common adapter code for physical command processing and I/O interrupt handling.

LEVEL 6

This supervisory level contains the following types of functions:

- Logical I/O support to initiate and service I/O requests for:
  - Console keyboard/display
  - Diskette drive
  - Communication with CCU control program
- Routing and task dispatching functions
- System functions such as system messages and error queuing

LEVEL 7

This level executes tasks in the following priority order:

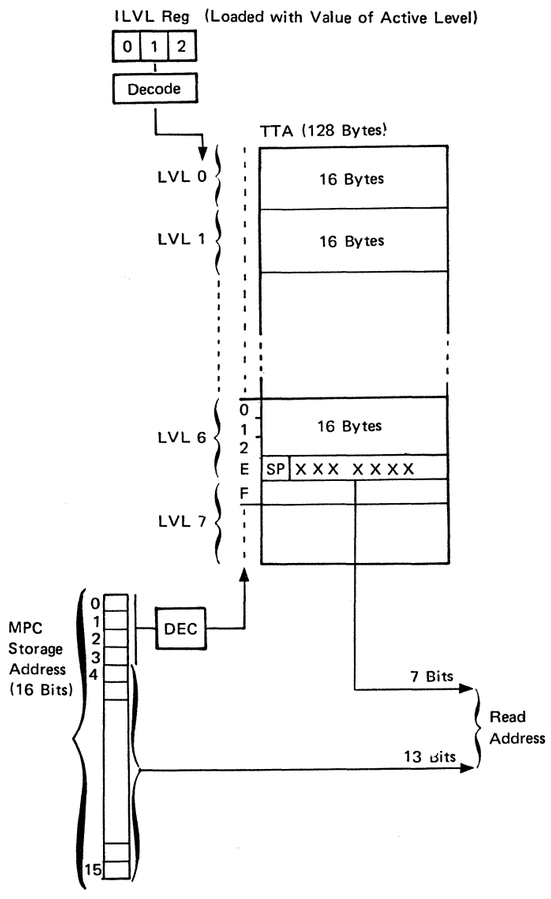
1. Machine status and system message display
2. Box error analysis and recording
3. CCU control program/host background communication tasks (dump transfer, REQMS/RECFMS support)
4. Operator transaction processing
5. CCU initialization

Register Map

The MOSS processor (MPC) address space is limited to 64K bytes because of its 16-bit address bus. To address the 64K bytes of storage on the MMC, the processor uses a translate table array (TTA) on the MCC. The TTA is 128 bytes long.

An address provided by the MOSS processor is considered as a virtual address. It is modified via the translation table array to select the real storage address.

Each MOSS interrupt level is allocated 16 bytes in the TTA, each byte allowing the selection of a 4K-byte storage block. Each interrupt level can thus extend its addressing to the 64K bytes present in the MOSS.



MOSS REGISTER SPACE ALLOCATION (1K BYTES)

PSW 0	PSW 1	PSW 2	PSW 3	Page 0
PSW 4	PSW 5	PSW 6	PSW 7	Page 1
Level 0 Primary page				Page 2
Level 0 Secondary page				Page 3
Level 1 Primary page				Page 4
Level 1 Secondary page				Page 5
Level 2 Primary page				Page 6
Level 2 Secondary page				Page 7
Level 3 Primary page				Page 8
Level 3 Secondary page				Page 9
Level 4 Primary page				Page 10
Level 4 Secondary page				Page 11
Level 5 Primary page				Page 12
Level 5 Secondary page				Page 13
Level 6 Primary page				Page 14
Level 6 Secondary page				Page 15
Level 7 Primary page (BER task)				Page 16
Level 7 Secondary page (BER task)				Page 17
Level 7 Primary page (MSD task)				Page 18
Level 7 Secondary page (MSD task)				Page 19
Level 7 Primary page (BG task)				Page 20
Level 7 Secondary page (BG task)				Page 21
Level 7 Primary page (CA monitor)				Page 22
Level 7 Secondary page (CA monitor)				Page 23
Level 7 Primary page (Op. ctl)				Page 24
Level 7 Secondary page (Op. ctl)				Page 25
Level 7 Primary page (IPL task)				Page 26
Level 7 Secondary page (IPL task)				Page 27
Level 5 Primary page (for dump)				Page 28
Level 5 Secondary page (for dump)				Page 29
Level 6 Primary page (for dump)				Page 30
Level 6 Secondary page (for dump)				Page 31

TTA Save Area (16 x 8 = 128 bytes)	Page 32 Page 33 Page 34 Page 35 Page 36 Page 37 Page 38 Page 39
Diagnostics Pages	Page 40 Page 41
Level 7 Primary Page (Loop Task)	Page 42
Level 7 Secondary Page (Loop Task)	Page 43
Diagnostics Use	Page 44 Page 45 Page 46 - - - - - Page 62 Page 63

<16 bytes>

Storage Mapping (Part 1 of 2)

(This mapping applies to the 128K MOSS storage only.)

CHGMOSS Translate Table Array

		4K Block Number														
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Level	0	0	1	2	3	4	D	E	F	18	19	1A	1B	1C	1D	1E	14
	1	0	1	2	3	4	D	E	F	18	19	1A	1B	1C	1D	1E	14
	2	0	1	2	3	4	D	E	F	18	19	1A	1B	1C	1D	1E	14
	3	0	1	2	3	4	D	E	F	18	19	1A	1B	1C	1D	1E	14
	4	0	1	2	3	4	D	E	F	18	19	1A	1B	1C	1D	1E	14
	5	0	1	2	3	4	D	E	F	18	19	1A	1B	1C	1D	1E	14
	6	0	1	2	3	4	D	E	F	18	19	1A	1B	1C	1D	1E	14
	7	0	1	2	3	4	D	E	F	18	19	1A	1B	1C	1D	1E	14
		Or after supervisor modification															
	7	B	C	2	3	4	15	16	17	18	19	1A	1B	1C	1D	1E	14

All blocks not accessible contain a storage address out of range (e.g block number 20).

CHGMOSS Storage Map

Real Addr.	Virtual Address	Contents
0000	0000	ROS : Diagnostic/IML/Dump/IH0 (1) Diskette CAC/diskette IH5
2000	2000	RAM : Common areas (not IMLed)
	2500	MOSS load module starting address ********** Common areas (IMLed) Resident data pool Loop detection
3000		MOSS re-IML bootstrap
4000		
5000	5000	3727 logical support level 6 Oper. panel logical support level 6 Diskette logical support level 6
6000		
7000	1000	Supervisor
8000	5000	Operator control
9000		
A000	7000	Operator control communication area
B000	0000	Trap address zero CCU functions (resident part)
C000	1000	Machine status display
D000	5000	Oper. panel CAC + interr. handler level 4 Interrupt handler level 1 Interrupt handler level 0 (RAM part)
E000		
F000		3727 CAC + interrupt handler level 3
10000	0000	Diskette logical/physical IOCS level 5
11000	5000	Reserved for future use (level 2)
12000		
13000	0000	Trap address zero + reserved level 6
14000	F000	CCU functions transient area
15000	5000	Alarm processing BER logging
16000		CCU background
17000		Channel adapter monitoring
18000	8000	Operator control transient area (24K)
19000		
1A000		
1B000		
1C000		
1D000		
1E000	E000	CCU background transient area
1F000	F000	(Used only by DCM)

Storage Mapping (Part 2 of 2)

DCF TRANSLATE TABLE ARRAY (TTA)

The current DCF structure allows defiming three TTA modes:
(changes from previous mode are flagged with a '*')

1. MOSS mode: the TTA mode when DCF receives the control from MOSS, and returns to it. (note that the console requires this mode)

2. DCF mode: the TTA mode required by DCM to allow
   -all the DCM overlays parts and IFT sections loading, in MOSS.
   -the activation of the DCF code previously loaded.

		4K block number															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	0	0	1	2	3	4	D	E	F	18	19	1A	1B	1C	1D	1E	14
	1	0	1	2	3	4	D	E	F	18	19	1A	1B	1C	1D	1E	14
L	2	.	.	2	3	4	11	12	.	18	19	1A	1B	1C	1D	1E	14
E	3	.	.	2	3	4	.	.	F	18	19	1A	1B	1C	1D	1E	14
V	4	B	.	2	3	4	D	E	F	18	19	1A	1B	1C	1D	1E	1F*
E	5	10	1	2	3	4	15	16	17	18	19	1A	1B	1C	1D	1E	1F*
L	6	13	7	2	3	4	5	6	A	18	19	1A	1B	1C	1D	1E	14
	7	B	C	2	3	4	15*	16*	17*	18	19	1A	1B	1C	1D	1E	1F*

3. IFT mode: the TTA mode required by IFTs when they have the control (access to panel MIOC common adapter microcode and interrupt handler microcode level 4; i.e, MOSS code).

		4K Block Number															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	0	0	1	2	3	4	D	E	F	18	19	1A	1B	1C	1D	1E	1F*
	1	0	1	2	3	4	D	E	F	18	19	1A	1B	1C	1D	1E	1F*
L	2	.	.	2	3	4	11	12	.	18	19	1A	1B	1C	1D	1E	14
E	3	.	.	2	3	4	.	.	F	18	19	1A	1B	1C	1D	1E	14
V	4	B	.	2	3	4	D	E	F	18	19	1A	1B	1C	1D	1E	1F*
E	5	10	1	2	3	4	15	16	17	18	19	1A	1B	1C	1D	1E	1F*
L	6	13	7	2	3	4	5	6	A	18	19	1A	1B	1C	1D	1E	14
	7	B	C	2	3	4	D*	E*	17	18	19	1A	1B	1C	1D	1E	1F*

DCF Loading map

15000	5000	DCM nucleus overlay 1	(overlay areas)	
16000	5900	DCM nucleus overlay 2		
17000	6600	spare for DCM microcode ==> restriction : no operator control access.		
18000	8000	DCF tables (root)		
19000	9100	DCF nucleus (root) 3727 PS nucleus (root)		
1A000	A200	3727 presentation services or DCM customization		
1B000	AB00	Diskette, MIOC, CP presentation services (overlay areas)		
1C000	B400	command processor MOSS - MOSS. (overlay area)		
1D000	D000	transient buff for CCU IFT running in MOSS.	transient buff in MOSS for IFT running in CCU.	SST tables (transient)
1E000	D600			Transient processes (CCU init, SST tables) or trans.buffer in MOSS for IFT running in TSS.
1F000	E000			

Communication Between MOSS and NCP/EP (Part 1 of 2)

The control panel functions of the 370X communication controllers are performed on the 3725 using the 3727 operator console and applications in MOSS.

Two of these MOSS applications, elementary data exchange (EDE) and CCU control program procedures, provide the communications to the NCP and EP control panel routines.

The elementary data exchange provides a high level operator communications that replaces the manual control panel operations of earlier machines:

- Setting the control panel storage address/register data switches.
- Setting the control panel display/function select switch.
- Pressing the CCU level 3 panel interrupt button as used in the 370X communications controllers.

The CCU control program procedures provide a high level operator communications to create, catalog, display, and run procedures. These procedures consist of EDE statements and control statements to control the sequencing of the EDE statements.

The MIOC support provides basic functions to communicate with the CCU and the MIOC card. These functions can be classified as follows:

- Read/write MIOC registers
- Read/write CCU LSRs
- Read/write CCU RAM
- Execute read/write IOH on IOC bus
- Execute CCU ROS operations.

This physical support is used by several logical functions:

- CCU machine status display, to show the state of the CCU.
- CCU functions, to provide CCU panel functions to the operator.
- TSS functions (all communications to or from the scanners must go on the IOC bus, through the CCU).
- CCU control program/host request (the logical protocol of mailbox exchange is based on the physical MIOC support).
- Diagnostic control facilities to communicate with the command processor loaded in the CCU or scanner.
- CCU diagnostics during CCU testing.

CCU CONTROL PROGRAM/MOSS COMMUNICATIONS

Mailbox Description

The NCP/EP and the MOSS communicate through CCU storage areas called mailboxes. These mailboxes are located in the 2K storage area reserved for MOSS use.

CCU Storage	
(192 spare bytes)	X'yxFFFF' X'yxFF40'
NCP in mailbox (MOSS to CCU) (32 bytes)	X'yxFF3F' X'yxFF20'
NCP out mailbox (CCU to MOSS) (32 bytes)	X'yxFF1F' X'yxFF00'
MOSS to scanner communication area (1536 bytes)	X'yxFEFF' X'yxF900'
Scanner mailboxes (16 x 16) (256 bytes)	X'yxF8FF' X'yxF800'
Top of non-reserved storage	X'yxF7FF'

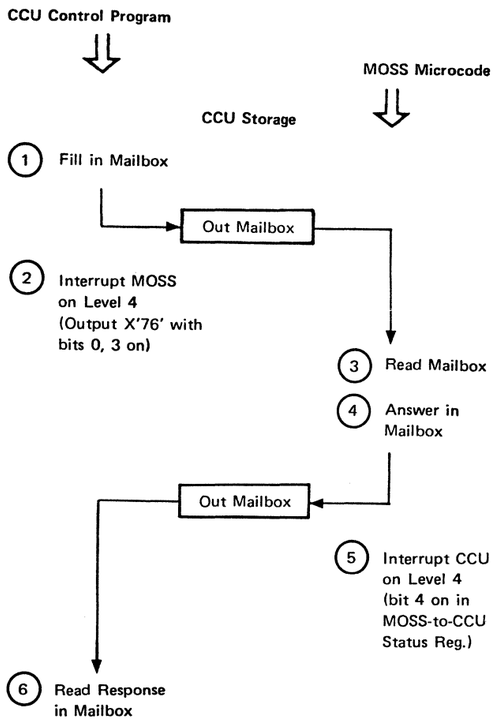
Where: yx = 07 for 512 systems
0B for 768 systems
0F for 1024 systems
17 for 1536 systems
1F for 2048 systems
27 for 2560 systems
2F for 3072 systems

For a detailed description of the mailbox contents, see 3725 Principles of Operation, GA33-0013.

CCU TO MOSS COMMUNICATION (OUT MAILBOX)

The out mailbox is used to pass requests from the NCP/EP to the MOSS and for the MOSS to post the status response.

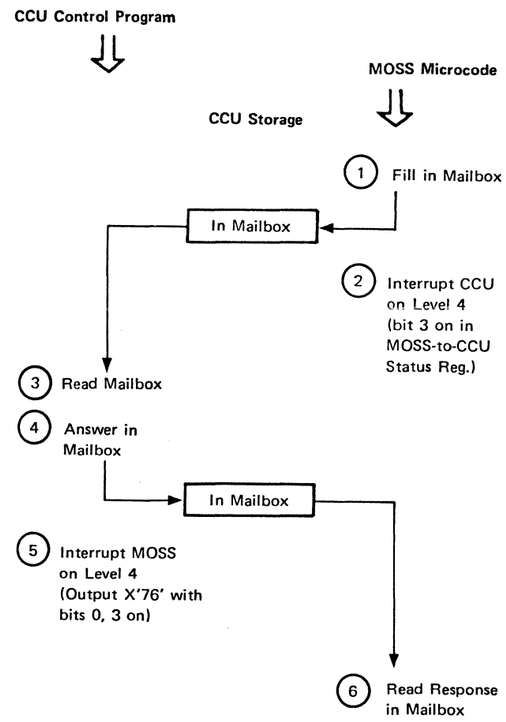
Exchange Procedure



MOSS TO CCU COMMUNICATION (IN MAILBOX)

The in mailbox is used to pass MOSS requests to the NCP/EP and for the NCP/EP for to post status response.

Exchange Procedure



Communication Between MOSS and NCP/EP (Part 2 of 2)

EXCHANGE TIMEOUTS

The requestor fills in its mailbox and then posts an interrupt to the receiver. The receiver grants the interrupt, decodes the mailbox contents, posts a status in the mailbox, and interrupts the requestor. Throughout this time, the mailbox is busy to the requestor. A timer is used to provide a protection against requests that are not answered.

On the NCP side, if a timeout occurs (3 seconds), the MOSS is considered down and an alert is sent to the host.

On the MOSS side, if a timeout occurs (3 seconds), a return code is passed to the requesting application to signal this situation. In this case NCP/EP is probably down and operator intervention may be required.

MAILBOX COMMANDS

The following table shows the commands exchanged between the MOSS and the CCU, and in which function they are involved.

Hex	Mailbox NCP Out Commands	Function	Mailbox NCP In Commands	Hex
42	Roll in saved storage	CLDP	Scanner IML complete	C1
41	Control program loaded	CLDP	Roll in complete	C2
23	Control prog parameter	Init	Ctrl prog parameter saved	A3
24	Request hardware CDS	Init	CDS information available	A4
25	Ctrl prog init complete			
06	Transfer PIU . REQMS request from NCP . Dump storage request	Normal	Transfer PIU . REQMS/RECFMS response . RECFMS unsoll. reply . Dump storage response	86
07	NCP/EP BER transfer	Normal	Wrap test request	89
08	Buffers now available	Normal	Connect scanner	8D
09	Wrap test results	Normal	Request buffer	8E
0C	Time/date valid	Normal	Free buffer	8F
			MOSS offline	90
			MOSS online	91

SENSE CODES

The following sense codes are used to return negative responses to the host:

0809 - Mode Inconsistency

The SNA request cannot be processed because the file that contains the requested data is empty. The MOSS returns this sense code on a DISPSTOR or a REQMS request when the dump file or the CIL is empty.

0812 - Insufficient Resource

This sense code is used in the following cases:

- Unable to queue SNA Request: MOSS has acknowledged negatively a PIU Transfer (06), and NCP has sent a negative response to the host.
- The MOSS resources temporarily not available. MOSS cannot process the SNA request because of a lack of resource access such as diskette or storage.

081C - Request not Executable

- MOSS down/MOSS offline: this sense code is used by NCP to respond to the host that has sent a request to the MOSS while it is not operational.
- Disk I/O error: this sense code is used by the MOSS when a not retriable diskette I/O error occurs during the processing of a SNA PIU that needs to access the MOSS diskette.

1005 - Parameter Error

- End of File encountered during the transfer of Dump File. This sense code is used by MOSS to indicate to the Host which has sent a request of DISPSTOR that the transfer of Dump File is completed.

MAILBOX STATUS

This is used to report status of request transfer, as follows:

		MOSS	NCP	EP
1...	Accepted	X	X	X
.1..	Rejected (never alone)	X	X	X
1... 1...	Keep buffer	X		
1... .1..	Free buffer	X		
.1.. .... 1...	Buffer not available		X	X
.1.. .... .1..	Function not supported			X
.1.. .... ..1.	Invalid command	X	X	X
.1.. ....1	MOSS down or offline		X	X
.1.. .... 1...	Invalid parameters	X	X	X
.1.. 1	Unable to queue SNA request	X		

Scanner/MOSS Communication

Communication between the MOSS and a scanner uses a dedicated area located in the upper part of the CCU storage, called "Mailbox".

In all circumstances the scanners are subordinate to the MOSS and therefore cannot initiate any unexpected action.

The scanner communication area contains a number of control blocks, one for each scanner.

Each control block contains entries for the command, the response, the data pointer, and the count.

1. The MOSS initializes the scanner block
2. The MOSS forces (through the MIOC) a PIO to the communication scanner processor (CSP) that indicate a MOSS operation.
3. Information is given in the PIO (R1, R2) or read from the control block via cycle steal by CSP.
4. The CSP stores the requested data in the communication area at the appropriate address via cycle steal.
5. The CSP sets the scanner control block fields and interrupts the MOSS on L4. An additional line in the IOC bus cable is used for this purpose.

SCANNER COMMUNICATION AREA COMMANDS

Here is the list of the commands the MOSS sends to a scanner.

Command in hex	Command description	
02	Dump Registers *	)
06	Address Compare Reset	) no data
08	Snap Shot Stop	)
09	Initialization *	)in mailbox
0B	Scanner Stop	)
0C	Scanner Go	)with these
0E	Checkpoint Trace On	)
0F	Checkpoint Trace Off	) commands
41	IPL *	
42	Dump Control Store *	
43	Display Immediate	)commands
44	Alter Immediate	)that use
63	Display Delayed	)mailbox
64	Alter Delayed	)
85	Adderss Compare Set	)see note
87	Snap Shot Start	)see note

* Commands initiated by ROS

Note: Commands "85" or "87" may be pending on a scanner together with commands "63" or "64".

Detection of MOSS Errors

HARDWARE CHECKING

Errors detected by the MOSS processor are reported via the MCPC register bits, or bit 0 of the IOIRR register.

Errors detected by the adapters are reported via the specific IOIRR register bits.

The error type is posted in the basic status register of each adapter.

Machine Check/Program Check Register

Bit	Functions
0	Inbound data parity error on I/O read
1	PIO timeout
2	Storage data parity error on read
3	Program check
4	
5	
6	
7	

Any bit set in the MCPC register causes a level 0 interrupt to the processor.

I/O Interrupt Request Register (IOIRR)

Bit	Functions
0	1. MOSS IML 2. Errors
1	1. Errors, 2. CCU high level requests 3. Timer
2	(not used)
3	Console and CCA 1. Operation 2. Errors
4	1. CCU low level requests 2. Scanners Requests
5	Diskette and DAC card 1. Operations 2. Errors
6	Supervisory
7	Back Level

Bit 0: MOSS Errors

This bit is set when one of the following occurs:

- Write storage violation (from MMC status register)
- Translate table array parity error (from MMC status register)
- CCU communication parity error (from MCC)
- MIOC communication parity error (from MCC)
- CCU 30-second timeout (from MCC)

SOFTWARE CHECKING

Software checking, performed by the MOSS microcode, consists of verifying the consistency of the data exchanged between the different components: applications, supervisor, or adapter processors, or within each component itself. It also consists of integrating the elementary errors on each component (counts and thresholds) before deciding that the corresponding component is out of service. These errors are reported on level 0.

Chapter 14. Maintenance and Operator Subsystem

Section 2. Troubleshooting Guidelines

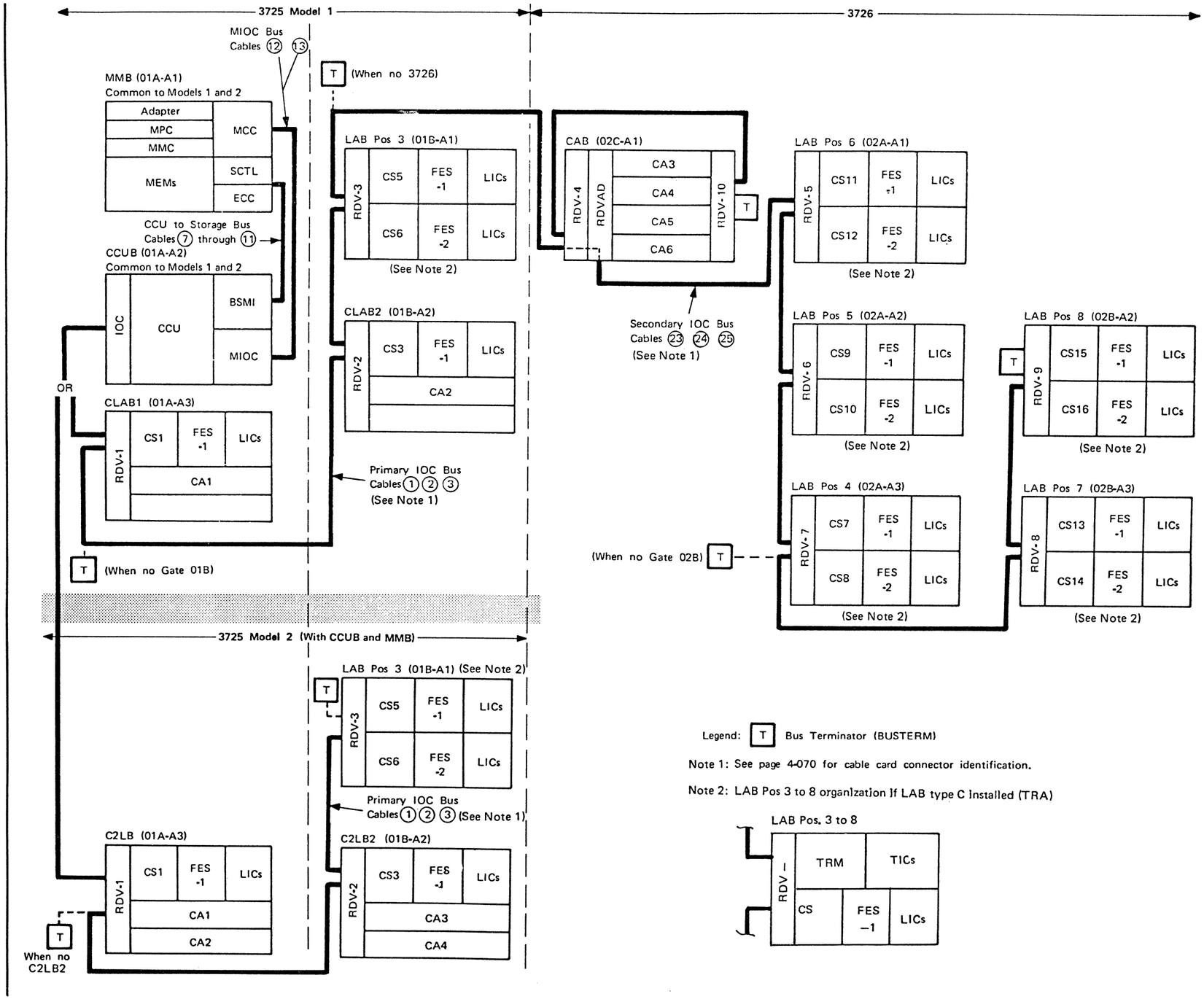
DC Voltages and Tolerances at Board Pin

Vdc	Vmin	Vmax	Ripple (max)
-12.0	-10.92	-13.20	0.45V p-p
-8.5	-7.73	-9.35	0.25V p-p
-5.0	-4.55	-5.50	0.15V p-p
-4.3	-4.19	-4.48	0.07V p-p
-1.5	-1.48	-1.56	0.03V p-p
+5.0 (Note 1)	+4.55	+5.50	0.20V p-p
+5.0	+4.75	+5.25	0.13V p-p
+8.5	+7.73	+9.35	0.35V p-p
+12.0	+10.92	+13.20	0.40V p-p
+12.0 (Note 2)	+11.40	+13.20	0.40V p-p
+24.0	+21.00	+27.60	0.30V p-p

Notes:

1. 02-PS7 only
2. 01-PS4 only

Board and Bus Organization



Legend: T Bus Terminator (BUSTERM)

Note 1: See page 4-070 for cable card connector identification.

Note 2: LAB Pos 3 to 8 organization if LAB type C installed (TRA)

MOSS Troubleshooting Techniques

INTRODUCTION

Notes:

1. Conditions outside the MOSS may cause failures which appear as MOSS failures. These outside conditions may be isolated using the MOSS isolation procedure on page 14-810.

2. For failures while running routines AAxx (CCU Diagnostics), see page 14-880 onward.

A MOSS initialization proceeds in IML steps. Each step tests certain logic areas of the MOSS and indicates successful completion by a hexa display value (displays F00 thru F0A). Some of the displays are not visible due to the speed of successful completion. See page 14-011 for a diagram of MOSS and its interconnections. Refer to page 6-070 for a description of the functional areas of the MOSS tested during each IML step. Note that the functional area must be successfully tested before the hexa display will indicate completion of that step. For example, the IML/IPL Decode and the CPA Test must successfully complete before F01 is displayed. The following areas of the MOSS troubleshooting section are available for problem determination by MOSS IML step and error codes:

IML Step	Possible Display Codes	Procedures used	Details on page:
Prior to F00	None	MOSS Clocking Power On/Machine Reset Checks Blank Display Checks Display Code is other than X'F01' (including blank)	14-810 14-820 14-824 14-821
After F00 but prior to F01	EE1 thru EEA	See CA Troubleshooting section	12-800
	E60 thru E66	Display code is other than X'F01' (including blank)	14-821
	Blank Display	Display Code is other than X'F01' (including blank) MOSS clocking	14-821 14-810
After F01 but prior to F03	E80 thru E82 Blank Display	No procedure is available. This is an internal MPC card storage test.	-
After F03 but prior to F04	EC1 thru ECF	PIO bus checking	14-830
After F04 but prior to F05	EAX thru EBX	Permanent Interrupt after reset	14-850
	E70 thru E75	Diskette drive bus errors	14-860

For errors incurred after F05 but prior to F07, use the PIO Bus Checking section page 14-830. Error codes may vary. The suspected failing condition must be determined from the MIM2 data and the IML Step description on page 6-070.

After F07 but prior to F08, use the section called 'Console Signals' page 14-831.

After F08 but prior to F0A, use the same information as for after F05 but prior to F07 (page 14-830).

MOSS Isolation and Clocking Checks

MOSS ISOLATION

The MOSS can operate as a stand-alone subsystem by disconnecting the CCU and the channel adapters. This must be done with the function select switch in the maintenance mode, using the service diskette.

Notes:

1. A hexadecimal display of FEF indicates a successful MOSS IML.
2. A hexadecimal display of E29 indicates that the function select switch was not positioned in the maintenance mode.

The CCU may be disconnected by:

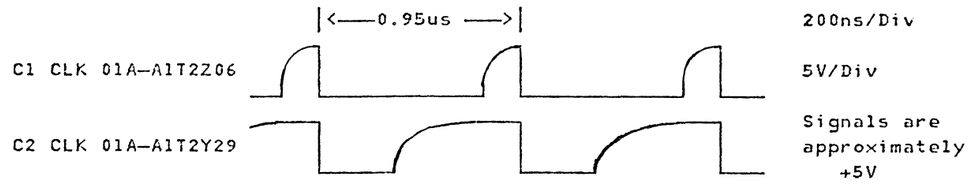
- Unplugging connector 01A-A2V2 or connectors 01A-A1ZG and 01A-A1ZH
- Inserting a jumper between 01A-A1V2J05 and ground.
- Resetting the CAs with the following jumpers:
  - 01A-A3A2M12 and ground on CLAB-1
  - 01B-A2A2M12 and ground on CLAB-2
  - 02C-A1X2M12 and ground on CAB.

The channel adapters may be disconnected by:

- Unplugging the cable at 01A-A1YE to disconnect channel adapters 3 through 6.
- Unplugging the cable at 01A-A1YF to disconnect channel adapters 1 and 2.

CLOCKING CHECKS

The MPC card contains its own 8MHz clock; it generates two clock signals: C1 CLK and C2 CLK which can be scoped at pins 01A-A1T2Z06 and 01A-A1T2Y29 respectively. The signals have the following form:



Note: Multiple traces (appearing as jitter) is normal, and is caused by clock assymmetry.

The ACLK card provides a pulse every 15 microseconds (see page 5-052). It is used by the MCC to test for a CCU timeout. If no answer is received from the CCU in reply to a 'read strobe' or a 'write strobe' request from the MOSS within 30 to 60 microseconds, the MCC card generates a level 0 interrupt to the MOSS processor. Refer to page 14-030 for the MCC status register bit 1 (CCU Interconnect Timeout).

The ACLK card also provides a pulse every 100 milliseconds (scopable at pin 01A-A1X4J12), which is used by:

- The MCC card which generates a level 1 interrupt to the MOSS processor. The MOSS microcode uses it to define various timers.
- The CPA card in conjunction with the start switch to generate the start condition.

  Note: on an operational machine, the microcode timer can be checked using the time and date feature (Request Q).

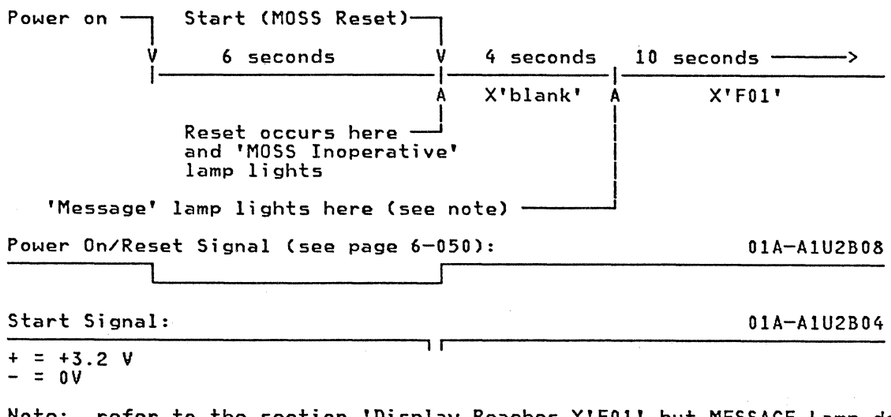
Code Other Than X'F01' (Part 1 of 2)

MOSS COMPLETELY DOWN

POWER ON/MACHINE RESET CHECKS

This section assumes that the MOSS is completely down, and that the hexa display never reaches X'F01', being either blank or displaying other digits.

After power on, or after pressing 'Start', the sequence of events should be as follows:



Note: refer to the section 'Display Reaches X'F01' but MESSAGE Lamp does not Light' for further details.

If the display never reaches the X'F01' stage (10 seconds), there are two possibilities:

1. Display is blank or has digits other than X'F01'.
2. The machine reset signals should not be permanently active. Refer to page 6-050 for the routing of this signal.

BLANK DISPLAY CHECKS

Note: for additional possible causes of a blank display, see the section 'Display is other than X'F01' (Including Blank)'.

1. Static Display Test

   Remove the crossover from the CPA card and then remove the CPA card from the MOSS board (01A-A1U2); the display should indicate 'FFF'. If the 'FFF' indication is not obtained, remove the display cable (17) from the MOSS board at position 01A-A1YQ (see page 4-070). If the 'FFF' indication is still not obtained, the YQ cable or the control panel board is faulty and must be repaired. Check also the power supplies on the MMB board, using pages 4-040 onwards. Loose connections between the MMB and the control panel may also be suspected.

2. Static Address Bus Test

   Note: the MPC, MMC, and DAC address bus pin locations are listed in this chapter under the heading 'MOSS PIO Bus and Internal Signal Routing'.

   The address bus at the MPC output is not gated as the PIO bus; when a blank display occurs, the address bus levels should be checked:

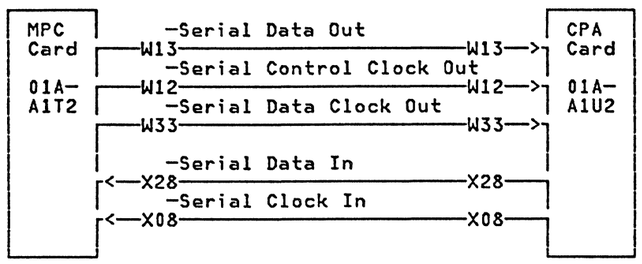
   Bits 0 through 15 may be pulsed, but should never be continuously minus (approximately ground level).

   The address bus is distributed only to the MMC and DAC cards; if a problem occurs, these cards may be unplugged (after power off) to locate the source of the fault. Check also the real address bits 3 through 6 coming from the MMC card; they should be the image of address bits 0 through 3.

Code Other Than X'F01' (Part 2 of 2)

DISPLAY IS OTHER THAN X'F01' (INCLUDING BLANK)

If the display never reaches X'F01', it is looping on a display error, and the display mechanism between cards MPC and CPA can be suspected. The display mechanism uses five lines in the W and X crossover connectors between the two cards, as follows:



Using the three output lines, the display data is sent to the CPA card in serial form, and then returned, still in serial form, to the MPC card using the two return lines. The transmitted and received patterns are compared, and the MPC loops if the two patterns are not equal.

Signals are present on all 5 lines when:

1. the display value is being updated
2. there is a loop on display caused by an unequal compare of transmitted and received patterns.
3. All five lines are at +5V when inactive, and approximately at ground level when active.

DISPLAY REACHES X'F01' BUT MESSAGE LAMP DOES NOT LIGHT

When 'Power On/Reset' is pressed, and the microcode reaches X'F01', the message lamp should light; if it does not, the crossover connection between the MPC and the CPA cards may be faulty (-Short 6 signal) or the connection to the Panel (+Message signal):

Signal 01A-A1	MPC	CPA	Connector YH*
-Short 6	---T2X13-------U2X13		
+Message	----------U2D09--------X1A08		

* Cable 16 - see page 4-083.

PIO Bus Checking (Part 1 of 2)

INTRODUCTION

This section describes three procedures for checking out the MOSS PIO bus. These are static checking, reduced configuration, and loop on error. Prior to hex display F03, the ROS code within the MOSS checks out the MOSS processor with its attached adapters isolated, that is, the MOSS PIO bus is disabled (see page 6-070). After hex display F03 these adapters (MMC, DAC, CCA, and MCC cards) are connected to the MOSS processor by activating the MOSS PIO bus (see page 14-012). The scope points referred to in this section are listed in this chapter under the heading 'MOSS PIO Bus and Internal Signal Routing' on page 14-835.

STATIC CHECKING

If the display is not blinking, the PIO Bus could be failing and causing an unrecoverable error. The PIO bus can be checked in a static condition by installing the following jumpers:

- 01A-A1U2B09 to ground to force a permanent reset.
- 01A-A1T2X11 to 01A-A1P2Y28 to enable the PIO bus.

The following signals on the bus should be at +5 V:

Address bits 0 through 15
Real address bits 3 through 6
Data 0 through 7 and P
I/O, TA, TC, and TD tags
IOIRR bits 0 through 7 (interrupt bits)

MOSS REDUCED CONFIGURATION

This approach strips down the MOSS to the minimum configuration that can work by physically removing everything else.

The minimum configuration consists of the following:

- The MPC card, 01A-A1T2 (without the pluggable module MMM24). Refer to Chapter 5 for MPC card and MMM24 module removal.
- The CPA card, 01A-A1U2.
- The MROS card, 01A-A1U4, if installed.
- The hexadecimal display indicators.

Note: top crossovers must be installed.

The following module or cards are not required in the minimum configuration and should be removed:

- The module MMM24 on the MPC card, 01A-A1T2
- The ACLK card, 01A-A1X4
- The DAC card, 01A-A1R2
- The MMC card, 01A-A1S2
- The MCC card, 01A-A1V2
- The CCA card, 01A-A1W2

To avoid a blank display when the MMC (S2) card is removed, the following jumpers must be installed on the MPC (T2) card.

01A-A1T2Z04 to Z24
01A-A1T2Z05 to Z25
01A-A1T2Z10 to Z30
01A-A1T2Z11 to Z31

Pressing 'Power On/Reset' causes the ROS checkout of the MOSS to start. The function 'Start' cannot be used because it requires that the ACLK card be installed.

The display should indicate X'E82'; if this indication is not obtained, even the minimum configuration is failing.

If the X'E82' indication is obtained, the minimum configuration may be assumed to be working. It is now possible to start rebuilding the MOSS by adding one or more cards at a time.

PIO Bus Checking (Part 2 of 2)

The chart below gives the expected hex display with specific cards removed:

Hex display	Card(s) removed
EC0	Note 1
EC1	DAC
EC2	CCA
EC3	CCA+DAC
EC6	MCC or MCC+CCA (Note 2)
EC7	MCC+DAC or MCC+DAC+CCA
EC8 *	MMC
EC9 *	MMC+DAC
ECA *	MMC+CCA
ECB *	MMC+CCA+DAC
ECE *	MMC+MCC or MMC+MCC+CCA (Note 2)
ECF *	MMC+MCC+DAC or MMC+MCC+DAC+CCA (Note 2)
Blank	Note 3

* To avoid a blank display when the MMC (S2) card is removed, the following jumpers must be installed on the MPC (T2) card.

  01A-A1T2Z04 to Z24
  01A-A1T2Z05 to Z25
  01A-A1T2Z10 to Z30
  01A-A1T2Z11 to Z31

Notes:

1. Hex display EC0 occurs when the entire configuration is failing.
2. MCC card removal gives the same hexadecimal display as MCC+CCA card removals.
3. A blank hex display is unexpected.

LOOP ON ERROR

Natural failures on the MMC, MCC, CCA, and DAC cards will produce hex displays of EC1 thru ECF. The last hex display digit is used by the moss checkout routines to indicate the suspected failing card(s). Each card has an assigned binary weight as shown in the chart below. These weights are additive to form the last hex display digit.

Card	Weight
MMC	8
MCC	4
CCA	2
DAC	1

If the display indicates X'EC1' through X'ECF', inserting a jumper between pin 01A-A1U2J10 of the CPA card and ground causes the ROS diagnostic to loop on the entire routine. (The display stops flashing and becomes steady). The whole routine is run and tests a certain number of functions as shown in the table below. Each test is preceded by an instruction which generates a negative sync pulse of 5.5us duration at pin 01A-A1T2U12.

Any error on these tests causes a machine check (-MCHK at pin 01A-A1T2U13), or an IOIRR bit 0 at pin 01A-A1T2Y27 or 01A-A1S2Y27. Counting the sync pulses and noting the position of the machine check quickly indicates the failing function.

Sync pulse	Test
1	Enable PIO bus (adapter write)
2	Read address X'FFFF' (storage address test: page 14-840)
3	Read address X'0000' (storage address test: page 14-840)
4	Sense MCC (adapter read)
5	Sense CCA (adapter read)
6	Sense DAC (adapter read)
7	Sense Communication Adapter (adapter read) (Note 4)

Notes:

1. For troubleshooting intermittent errors, a permanent error may be forced (to cause looping) by removing an adapter card (for example, the CCA card).
2. When performing a 'Start' or 'Power On/Reset' with 01A-A1U2J10 jumpered to ground, the display will show E61. The jumper must be put on after the EC1 through ECF display.
3. Refer to timing charts.
4. Communication Adapter on DFA card, not used in 3725.

CONSOLE SIGNALS

The console signals may be scoped while running the Console Link test. Install the wrap block (P/N 2667737) in position 1 as shown on page 6-040. The EIA break out box may be helpful. See page 4-083 for other pin locations.

PIO Bus Signal Routing (Part 1 of 2)

In the diagrams which follow, the symbols have the following meaning:

I = Card input signal pin
O = Card output signal pin
V = Voltage pin

Addresses 0-15 = MOSS internal storage addresses

The MOSS PIO Bus has the following signals:

- Data 0-7 and P
- IO Tag
- Parity Valid
- TA Tag
- TC Tag
- TD Tag
- Valid Byte Tag
- Write
- Reset

PIO Bus Signal Routing (Part 2 of 2)

Board Location 01A-A1

Signal Name 01A-A1	R DAC	S MMC	T MPC	U CPA	V MCC	W CCA
-ADDRESS 0	O_2Z24-	I_2Z24-	O_2Z24			
-ADDRESS 1	O_2Z25-	I_2Z25-	O_2Z25			
-ADDRESS 2	O_2Z30-	I_2Z30-	O_2Z30			
-ADDRESS 3	O_2Z31-	I_2Z31-	O_2Z31			
-ADDRESS 4	O_2Z32-	I_2Z32-	O_2Z32			
-ADDRESS 5	O_2Z12-	I_2Z12-	O_2Z12			
-ADDRESS 6	O_2Z33-	I_2Z33-	O_2Z33			
-ADDRESS 7	O_2Z13-	I_2Z13-	O_2Z13			
-ADDRESS 8	O_2Z23-	I_2Z23-	O_2Z23			
-ADDRESS 9	O_2Z03-	I_2Z03-	O_2Z03			
-ADDRESS 10	O_2Z22-	I_2Z22-	O_2Z22			
-ADDRESS 11	O_2Z02-	I_2Z02-	O_2Z02			
-ADDRESS 12	O_2Y33-	I_2Y33-	O_2Y33			
-ADDRESS 13	O_2Y13-	I_2Y13-	O_2Y13			
-ADDRESS 14	O_2Y32-	I_2Y32-	O_2Y32			
-ADDRESS 15	O_2Y12-	I_2Y12-	O_2Y12			
-B CLOCK	I_4B12-	--------	O_4B12			
-C CLOCK	I_4D12-	--------	O_4D12			
-C1 CLK	-------	-------	I_2Z06			
	-------	-------	O_2Z26			
-C2 CLK	-------	I_2Y29-	O_2Y29			
-C2 CLK WS	-------	-------	O_5D05	I_3D12		
-DATA P	O_4B07-	O_4B07-	O_4B07	------	I_5B12	
	-------	-------	------	------	O_3D13	I_2B02
	-------	-------	------	------	------	O_3B12
-DATA 0	O_4B02-	O_4B02-	O_4B02	------	I_5B02	
	-------	-------	------	------	O_3B09	I_3B02
	-------	-------	------	------	------	O_3B09
-DATA 1	O_4D02-	O_4D02-	O_4D02	------	I_5B03	
	-------	-------	------	------	O_3B10	I_3D02
	-------	-------	------	------	------	O_3D06
-DATA 2	O_4B03-	O_4B03-	O_4B03	------	I_5B04	
	-------	-------	------	------	O_3B12	I_2D10
	-------	-------	------	------	------	O_3B07
-DATA 3	O_4B04-	O_4B04-	O_4B04	------	I_5B05	
	-------	-------	------	------	O_3B13	I_3B08
	-------	-------	------	------	------	O_3D07
-DATA 4	O_4D04-	O_4D04-	O_4D04	------	I_5B07	
	-------	-------	------	------	O_3D09	I_3B10
	-------	-------	------	------	------	O_3D04
-DATA 5	O_4B05-	O_4B05-	O_4B05	------	I_5B08	
	-------	-------	------	------	O_3D10	I_2D09
	-------	-------	------	------	------	O_3D10
-DATA 6	O_4D05-	O_4D05-	O_4D05	------	I_5B09	
	-------	-------	------	------	O_3D11	I_2B09
	-------	-------	------	------	------	O_3D11
-DATA 7	O_4D06-	O_4D06-	O_4D06	------	I_5B10	
	-------	-------	------	------	O_3D12	I_2D06
	-------	-------	------	------	------	O_3D12

Board Location 01A-A1

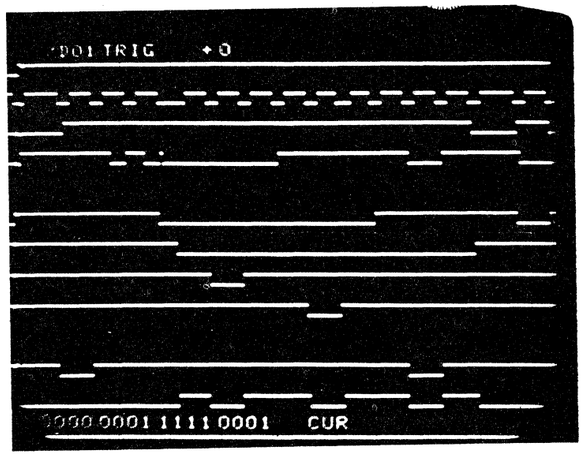
Signal Name 01A-A1	R DAC	S MMC	T MPC	U CPA	V MCC	W CCA	
+DISABLE	-------	-------	I_2X11				
	-------	-------	O_2X31				
+DISABLE OUT	I_2Y08-	-------	O_2Y08				
-DMA PARITY	I_2Y25-	-------	O_2Y25				
-IO TAG	I_4D09-	-------	O_4D09	I_2B05	I_5D02	I_2B04	
INTERRUPT LVL 0	-------	-------	I_3D12	------	O_4D11		
-IOIR0	-------	0-2Y27-	I_2Y27				
INTERRUPT LVL 1	-------	-------	I_4D11	------	O_4D12		
-IOIR2	-------	-------	I_4D13	------	------	-------	unused
-IOIR3	-------	-------	I_5B02	------	------	O_2B07-	
INTERRUPT LVL 4	-------	-------	I_5D02	------	O_4D13		
-IOIR 5	O_5D02-	-------	I_5B03				
-IOIR 6	-------	-------	I_5B04	------	------	-------	unused
-IOIR 7	-------	-------	I_5D04	------	------	-------	unused
-IRPT	-------	I_2Y03-	O_2Y03				
-MCHK	-------	-------	O_5D13	I_2B12			
-MEM CYCLE REQ	O_2Y24-	I_2Y24-	I_2Y24				
-MEM SEL	-------	-------	O_2Y23				
-OSC UNUSED	-------	-------	I_3B04				
	-------	-------	O_3D04				
-PARITY VALID	O_4B10-	-------	I_4B10	------	O_5D09	O_2D07	
-READ GATE	I_2Z07-	-------	I_2Z07				
	-------	-------	O_2Z27				
-REAL AD 3	-------	O_2Z04-	I_2Z04				
-REAL AD 4	-------	O_2Z05-	I_2Z05				
-REAL AD 5	-------	O_2Z10-	I_2Z10				
-REAL AD 6	-------	O_2Z11-	I_2Z11				
-REG SEL	-------	I_2Y02-	O_2Y02				
-RESET	I_5B05-	I_4B13					
	-------	I_3D09-	I_4B13				
	-------	-------	I_3D09				
	-------	-------	I_3D11	O_2B09	I_5D11	I_3D05	
-STOP REQST	O_2Y26-	O_2Y26-	I_2Y26				
-STOPPED	I_2Y22-	I_2Y22-	O_2Y22				
-TA TAG	I_4D07-	-------	O_4D07	------	I_5D04	I_2D05	
-TC TAG	I_4B08-	-------	O_4B08	------	I_5D05	I_2B08	
-TD TAG	I_4B09-	-------	O_4B09	------	I_5D06	I_2B05	
-V.BYTE TAG	O_4D10-	-------	I_4D10	------	O_5D10	O_3B03	
-WRITE	O_2Y05-	I_2Y05-	O_2Y05				

See page YZ 500 for the conventions used in this table.

MPC Timing Charts

During cycles with storage access, the clock C2CLK is extended to allow time for the storage to reply. For this reason, cycles without storage access (MSEL inactive) last 950ns, and cycles with storage access (MSEL active) last 1450ns; this is shown on the following diagrams. Clock -C2CLK is also extended during refresh cycles.

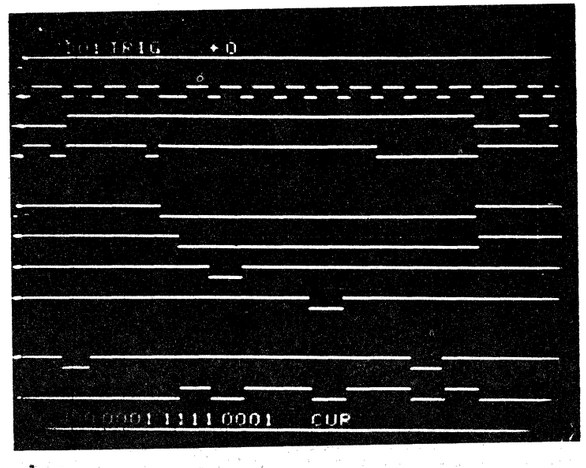
READ COMMAND TIMING CHART



01A-A1T2U12	-TS (*)
01A-A1T2Y29	-C2 CLK
01A-A1T2Y23	-MEM SEL
01A-A1T2Y05	-WRITE
01A-A1T2M02	-DATA 0
01A-A1T2P09	-IO TAG
01A-A1T2P07	-TA TAG
01A-A1T2M08	-TC TAG
01A-A1T2M09	-TD TAG
01A-A1T2P10	-V.BYTE TAG

(*) trigger scope at the end of this pulse.

WRITE COMMAND TIMING CHART



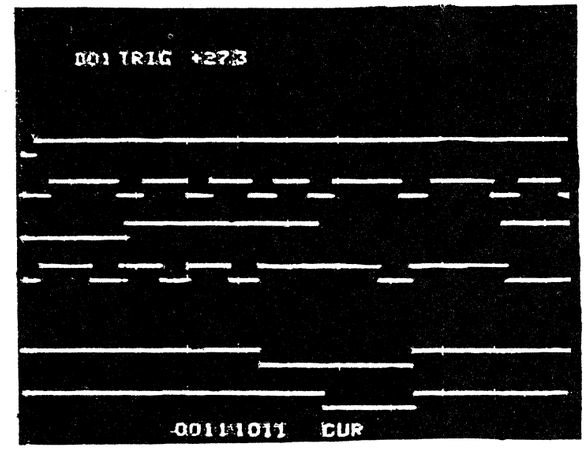
01A-A1T2U12	-TS (*)
01A-A1T2Y29	-C2 CLK
01A-A1T2Y23	-MEM SEL
01A-A1T2Y05	-WRITE
01A-A1T2M02	-DATA 0
01A-A1T2P09	-IO TAG
01A-A1T2P07	-TA TAG
01A-A1T2M08	-TC TAG
01A-A1T2M09	-TD TAG
01A-A1T2P10	-V.BYTE TAG

(*) trigger scope at the end of this pulse.

STORAGE ADDRESS TEST

The storage address test checks the address bits with the memory expansion enabled. The program runs on level 0, and the MOSS PIO bus is enabled. Two tests are performed using a Read command:

- After sync pulse 2, the virtual address FFFF of the read command is changed by the MMC to 01FFF. This means that address bits 0-15 and real address bit 6 are active.
- After sync pulse 3, the virtual address 0000 of the read command is changed by the MMC to 00000. This means that the address bits 0-15 and real address bits 3-6 are inactive.
- Refer to page 14-120 for virtual and real address information.



01A-A1T2U12	-TS (*)
01A-A1T2Y29	-C2 CLK
01A-A1T2Y23	-MEM SEL
01A-A1T2Y05	-WRITE
01A-A1T2Z31	-ADDRESS 3
01A-A1T2Z11	-REAL AD 6

* Trigger scope at the end of this pulse.

Permanent Interrupt After Reset

A permanent interrupt (the IOIRR is described on page 14-160) after reset has occurred if the hexadecimal display shows X'EAx' or X'EBx' (where x is a digit 1 through F). The 3 hexadecimal digits have the following meaning:

'E' = error

'A' = error is in IOIRR bits 0 through 3

'B' = error is in IOIRR bits 4 through 7

'x' = 4-bit combination of bits 0 through 3 or 4 through 7

Note: X'EAx' takes precedence over X'EBx'.

Example.

If the error indication is X'EBC', the error is in bits 4 and 5.

IOIR SCOPE POINTS

Display	Interrupt Level or IOIRR bit	MPC | MMC	CCA	MCC	DAC	Comments
EA8	0	T2J12----------V2P11------ T2Y27-S2Y27----------				
EA4	1	T2P11----------V2P12------				Note 2
EA2	2	T2P13----------				unused
EA1	3	T2S02-------W2B07----------				
EB8	4	T2U02----------V2P13------				
EB4	5	T2S03----------R2U02				
EB2	6	T2S04----------				unused
EB1	7	T2U04----------				unused

Notes:

1. The 'Display' column assumes a single error only.
2. If there is a permanent level 1 (EA4), check the signals from the CCLK and the ACLK at the entry to the MCC card:

   + CCU clock card check 01A-A1ZGB08 - 01A-A1V2J05
   + A clock check 01A-A1X4G07 - 01A-A1V2J04

   These two lines are not masked, and hold the permanent level 1 interrupt.

Diskette Drive Bus Errors (Part 1 of 2)

See page 14-080 for MOSS/Diskette drive interaction.

DISK ADAPTER ROS TESTS (X'E70' THROUGH X'E75')

If the MOSS stops with an X'E70' through X'E75' indication, one of the disk adapter microcode tests is failing, as follows:

Display	Meaning
X'E70'	Disk adapter phase 0 test failed
X'E71'	Disk adapter phase 1 test failed
X'E72'	Disk adapter phase 2 test failed
X'E73'	Disk adapter phase 3 test failed
X'E74'	Disk adapter phase 4 test failed
X'E75	Disk adapter phase 5 test failed

Each phase tests a small number of functions of the disk adapter. Before each function test, an instruction is executed that generates a sync pulse on the MPC card at pin 01A-A1T2U12 (a negative-going pulse 5.5us long). When the MOSS stops at one of the indications X'E70' through X'E75', inserting a jumper between pin 01A-A1U2J10 and ground causes the MOSS to execute a loop from the beginning of the phase up to the error; the sync pulses can now be scoped. Counting the sync pulses and comparing them against the expected sequences given below will indicate the error which occurred by its place in the sequence.

DISK ADAPTER ROS TEST PHASE 0 (X'E70')

Throughout this test, IOIRR Level 0 and MCHK must both be inactive. The test checks the disk adapter status register; it generates the following sequence:

Sync Pulse	Action	Timing chart (page 14-840)
1	Reset disk adapter status register	Adapter write
2	a. Read the disk adapter sense byte; the line 'Parity Valid' must be active. b. Loop if a machine check/program check occurs.	Adapter read
3	Set Machine Check bit in the status register.	Adapter write
4	a. Read Machine Check bit in the status register. b. Loop if the machine check bit is not present in the status register. c. Reset the disk adapter status register.	Adapter read Adapter write

DISK ADAPTER ROS TEST PHASE 1 (X'E71')

This test checks the level 5 interrupt; it generates the following sequence:

Sync Pulse	Action	Timing chart (page 14-840)
1	a. Set 'Enable', 'Op Complete' (to force a level 5 interrupt), and 'Head Engage' in status Reg. b. Read status register and check expected bits c. Check that IOIRR Level 5 is active. d. Loop if the expected bits are not correct, or if IOIRR Level 5 is inactive.	Adapter write Adapter read
2	a. Reset the disk adapter status register b. Read the status register and check that all bits are reset c. Check that IOIRR Level 5 is inactive. d. Loop if the status bits are not reset, or if IOIRR Level 5 is active.	Adapter write Adapter read

DISK ADAPTER ROS TEST PHASE 2 (X'E72')

This test checks the disk direct storage access storage address register and the data bus; it generates the following sequence:

Sync Pulse	Action	Timing chart (page 14-840)
1	a. Write DAC storage address register low. b. Write DAC storage address register high.	Adapter write Adapter write
2	a. Read DAC storage address register low; expected value is X'AA'. b. Loop if value is not X'AA'.	Adapter read
3	a. Read DAC storage address register high; expected value is X'AA'. b. Loop if value is not X'AA'.	Adapter read
4	a. Write DAC storage address register low. b. Write DAC storage address register high.	Adapter write Adapter write
5	a. Read DAC storage address register low; expected value is X'55'. b. Loop if value is not X'55'.	Adapter read
6	a. Read DAC storage address register high; expected value is X'55'. b. Loop if value is not X'55'. c. Reset DAC storage address register low. d. Reset DAC storage address register high.	Adapter read Adapter write Adapter write

Diskette Drive Bus Errors (Part 2 of 2)

DISK ADAPTER ROS TEST PHASE 3 (X'E73')

This test checks the 5ms timer; it generates the following sequence:

Sync Pulse	Action	Timing chart (page 14-840)
1	a. Enable 5ms timer in adapter status register. b. Start a 4ms software timer; when it times out, check that there is no IOIRR Level 5.	Adapter write
2	a. Loop if 5ms interrupt is already present (IOIRR Level 5 active; see page 14-850. b. Start a 2ms software timer; when it times out, check that IOIRR Level 5 is active. Note: this 2ms software timer, in conjunction with the 4ms software timer (1b above) gives a 6ms timer.	
3	a. Loop if 5ms interrupt is not present (IOIRR Level 5 still active). b. Reset the adapter status register.	Adapter write

DISK ADAPTER ROS TEST PHASE 4 (X'E74')

Note: refer to chapter 7 for the layout of the diskette and the scoping points.

This test checks that the diskette is ready; it generates the following sequence:

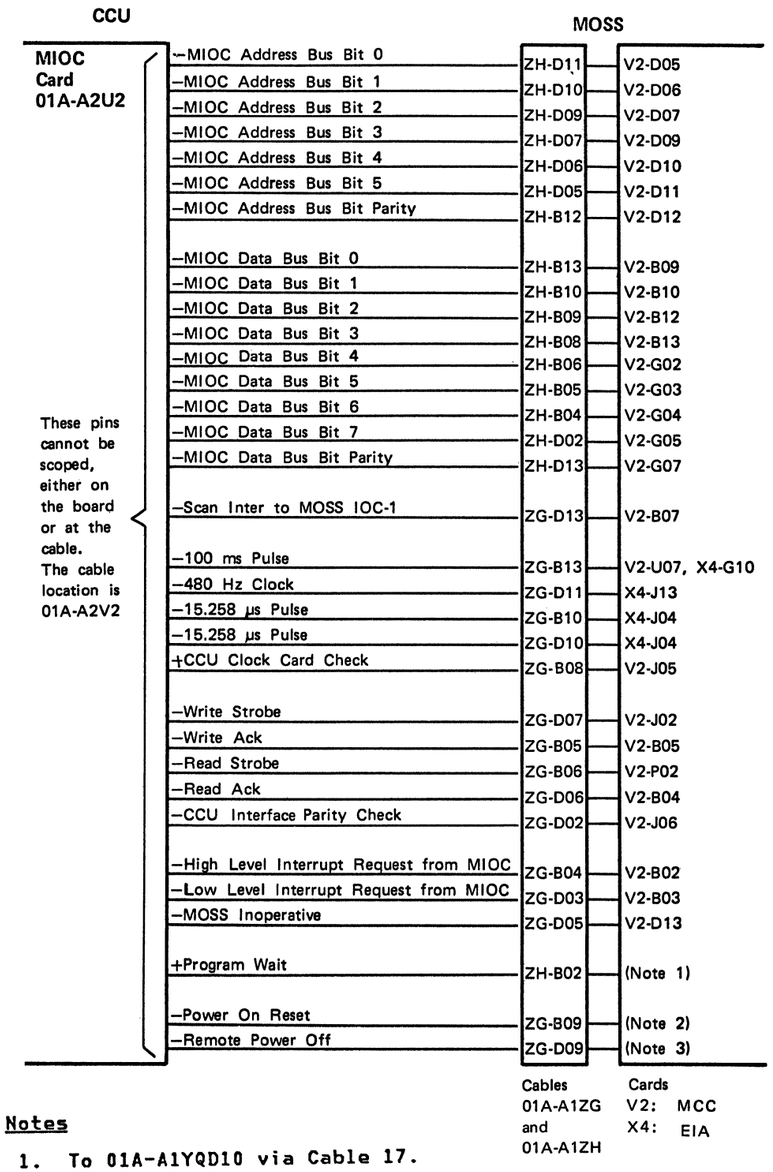
Pulse	Action	Timing chart (page 14-840)
1	a. Enable index mark interrupt. b. Reset index mark interrupt, if any. c. Send a 200 ms timer. d. IOIRR bit 5 must be on before timer times out (one revolution of the diskette = about 166ms). e. Check the index mark bit in status register after level 5. f. Reset enable and index mark bit in status register.	Adapter write Adapter write Adapter write Adapter write

DISK ADAPTER ROS TEST PHASE 5 (X'E75')

This test sets the modified frequency modulation bit (MFM) in the sector identifier register; it generates the following sequence:

Pulse	Action	Timing chart (page 14-840)
1	a. Set the MFM bit in the sector identifier register. b. Read the sense byte and verify that the MFM bit is on.). c. Loop if the MFM bit is not on.	Adapter write Adapter read

MIOC Bus Connections



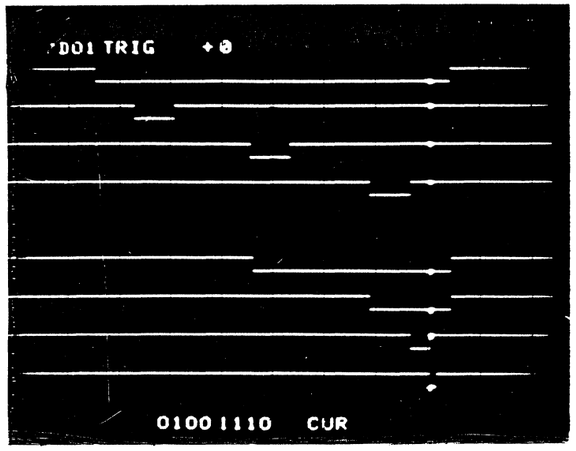
Notes

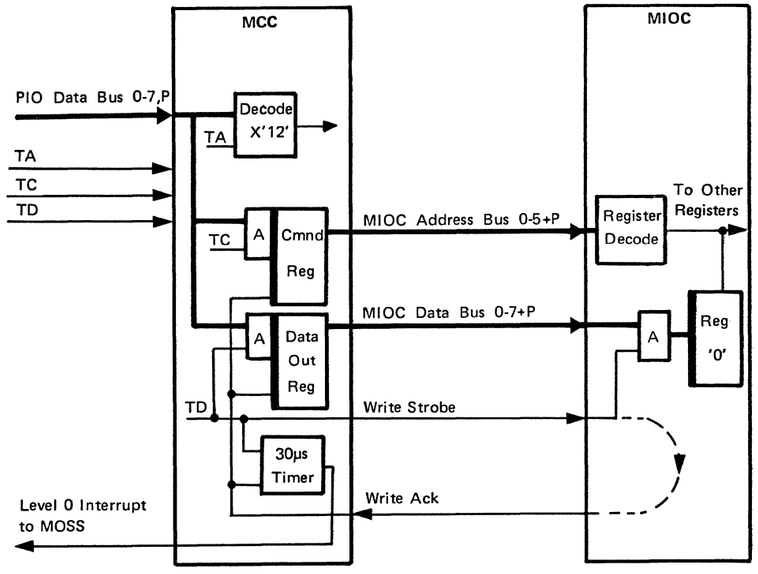
1. To 01A-A1YQD10 via Cable 17.
2. See page 6-050.
3. To 01A-A1YQB12 via Cable 18.

MIOC Bus Write Operation

01A-A1T2P09 -IO TAG
01A-A1T2P07 -TA
01A-A1T2M08 -TC
01A-A1T2M09 -TD

01A-A1V2D05 -MIOC ADDRESS BUS BIT 0
01A-A1V2B09 -MIOC DATA BUS BIT 0
01A-A1V2J02 -WRITE STROBE
01A-A1V2B05 -WRITE ACK





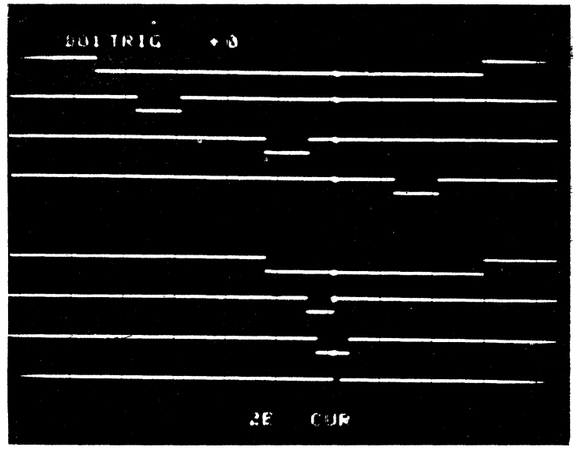
Notes:

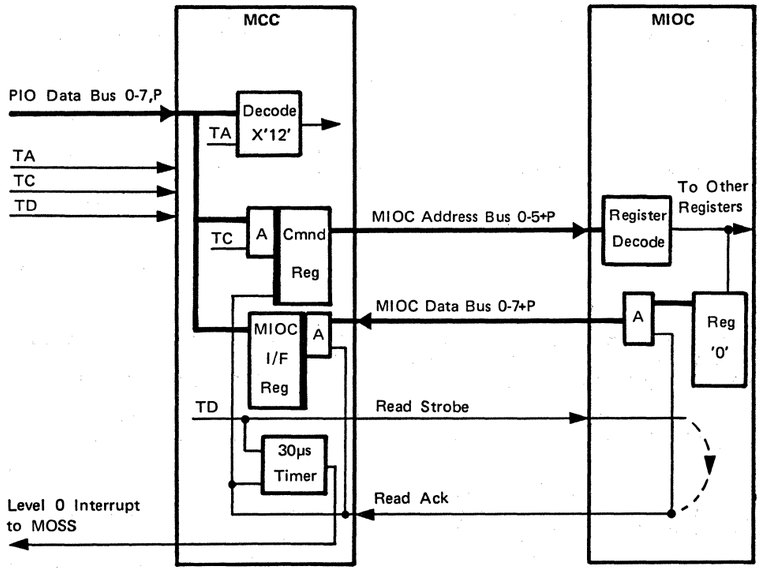
1. In the above timing chart, the active level is always negative-going.
2. If no reply is received from the CCU to 'Write Strobe' within 30 us, the timer times out and generates a level 0 interrupt to the MOSS.
3. The address on the PIO bus at TA time is always X'12' for the MCC.
4. The 6-bit address (plus parity) sent on the MIOC Address Bus at command time consists of the middle 6 bits of the command; bit 0 of the command is always ON, and bit 7 is always OFF for a write command.
5. Write Strobe and Write Ack occur after TD during the data transfer prepared on the PIO data bus during TD time.

MIOC Bus, Read Operation

01A-A1T2P09 -IO TAG
01A-A1T2P07 -TA
01A-A1T2M08 -TC
01A-A1T2M09 -TD

01A-A1V2D05 -MIOC ADDRESS BUS BIT 0
01A-A1V2P02 -READ STROBE
01A-A1V2B12 -MIOC DATA BUS BIT 2
01A-A1V2B04 -READ ACK





Notes:

1. In the above timing chart, the active level is always negative-going.
2. If no reply is received from the CCU to 'Read Strobe' within 30 us, the timer times out and generates a level 0 interrupt to the MOSS.
3. The address on the PIO bus at TA time is always X'12' for the MCC.
4. The 6-bit address (plus parity) sent on the MIOC Address Bus at command time consists of the middle 6 bits of the command; bit 0 of the command is always ON, and bit 7 is always ON for a read command.
5. Read Strobe and Read Ack occur before TD to prepare the data transfer to the PIO data bus during TD time.

MIOC Bus Scoping Routine

If the MOSS-to-CCU internal bus is suspected, the manual intervention routine AB02 provides scoping facilities for this internal bus. This routine does not isolate a FRU, but allows looping using the given command, address, and data patterns for scoping.

There is no preliminary checking phase for errors on selected lines; these errors may be detected using the following diagnostic routines:

WARNINGS

- The possible effect on the CCU of commands issued at the MOSS/CCU interconnection are not taken into account by routine AB02.
- Because routine AB02 disturbs the MOSS/CCU interconnection, any subsequent action (such as: run diagnostics, or analyze BERs) may lead to unpredictable results. Consequently, a MOSS re-IML is needed after any AB02 run.

Running the Routine

Proceed as follows:

1. Select the CCU manual intervention routine AB02. The message 'SELECT LINES TO SCOPE' is displayed.
2. Reply with 'Rxx' or 'R04yyzz',

   where

   xx is the scope request - 01 through 03, 05, or 06 (see the AB02 Request column in the following "Signal Characteristics" table).

   yy is the value (X'01' through X'3F') to be set on the address bus. The address bits 0-5 are right-justified in the address.

   zz is the value (X'00' through X'FF') to be set on the data bus.

   All other combinations cause the message 'INVALID REQUEST' (Rxx) 'PRESS SEND' or 'INVALID REQUEST' (R04yyzz) 'PRESS SEND', to be displayed. Pressing SEND causes the 'SELECT LINES TO SCOPE' message to be redisplayed.
3. Looping continues until the Attention key is pressed; the 'BREAK RECEIVED' message is then displayed. Typing G then returns to display the first request, C cancels the routine, and A aborts it.

Notes:

1. The invalid address X'00' may be exercised using request R06.
2. The 'High Level Interrupt Request' and 'Low Level Interrupt Request' lines should not be scoped using request 04 as the result is unpredictable due to the values set on the address bus.

Signal Characteristics

Signal Name	Location	Diagnostic Routine (Note 1)	AB02 Request (Note 2)	Signal Character. when looping	
				Period	Length
-MOSS inoperative	V2D13	AA01	R01	2.7ms	600 us
-Read strobe	V2P02	AA02	R03	2.0ms	600 ns
-Read ack	V2B04	AA02	R03	2.0ms	200 ns
-Write strobe	V2J02	AA03	R02	2.0ms	500 ns
-Write ack	V2B05	AA03	R02	2.0ms	200 ns
-CCU/MOSS parity chk	V2J06	AA02	R06	22.0ms	600 ns
-HLIR from MIOC	V2B02	AC05	R05	5.0ms	600 us
-LLIR from MIOC	V2B03	AC08	R05	5.0ms	600 us
-MIOC Data bus bit 0	V2B09	AA05	R04yyzz	2.0ms	2.0 us
-MIOC Data Bus bit 1	V2B10	AA05	R04yyzz	2.0ms	2.0 us
-MIOC Data Bus bit 2	V2B12	AA05	R04yyzz	2.0ms	2.0 us
-MIOC Data Bus bit 3	V2B13	AA05	R04yyzz	2.0ms	2.0 us
-MIOC Data Bus bit 4	V2G02	AA05	R04yyzz	2.0ms	2.0 us
-MIOC Data Bus bit 5	V2G03	AA05	R04yyzz	2.0ms	2.0 us
-MIOC Data Bus bit 6	V2G04	AA05	R04yyzz	2.0ms	2.0 us
-MIOC Data Bus bit 7	V2G05	AA05	R04yyzz	2.0ms	2.0 us
-MIOC Data Bus bit P	V2G07	AA05	R04yyzz	2.0ms	2.0 us
-MIOC Addr Bus bit 0	V2D05	AA07/08	R04yyzz	2.0ms	5.2 us
-MIOC Addr Bus bit 1	V2D06	AA07/08	R04yyzz	2.0ms	5.2 us
-MIOC Addr Bus bit 2	V2D07	AA07/08	R04yyzz	2.2ms	5.2 us
-MIOC Addr Bus bit 3	V2D09	AA07/08	R04yyzz	2.2ms	5.2 us
-MIOC Addr Bus bit 4	V2D10	AA07/08	R04yyzz	2.2ms	5.2 us
-MIOC Addr Bus bit 5	V2D11	AA07/08	R04yyzz	2.2ms	5.2 us
-MIOC Addr Bus bit P	V2D12	AA07/08	R04yyzz	2.2ms	5.2 us

Notes:

1. Each signal name corresponds to a routine in the diagnostic package.
2. Write operation only is used by request R04yyzz.

Contents

Chapter 15 Token-Ring Subsystem

Section 1. Unit Description

Contents

Overview of the IBM Token-Ring Network (Part 1 of 3)

PURPOSE OF THE IBM TOKEN-RING NETWORK

The IBM Token-Ring Network has been architectured to replace a large variety of cabling systems that are included in many companies' installations.

Fulfilling this need is the primary purpose of the IBM Token-Ring Network.
On the premises of a company, you may find networks made up of:

- Telephone wiring, or
- Twisted-pair communication wiring, or
- Coaxial cable, or
- Optical fibers.

These complex networks are very difficult to maintain, and the "book-keeping" chore of maintaining records on the attachments is very complex.

Any or all of these types of communication media may be included in a single network.

The IBM Token-Ring Network can replace a complex system with a very organized approach to providing a customer's communications needs at a particular company site.

The IBM Token-Ring Networks are designed specifically to provide an integrated approach to communications in a particular location.

A particular location is defined as a location through which no public thoroughfares pass.
This does not mean that Token-Ring Networks cannot be interconnected through common carrier facilities.

In cases of large companies with multiple, widely separated sites, individual Token-Ring Networks will almost always be interconnected.

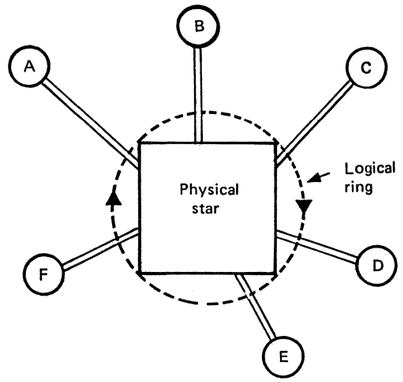
A second characteristic of a Token-Ring Network is that the data rate over the network is much higher than is possible using common carrier communications.
A Token-Ring Network can replace dial and leased telephone lines and some of the other types of media quoted above, and the resulting communications capability is increased.
The Token-Ring Network will also allow voice communications to be carried through the same cabling system that carries data communications.

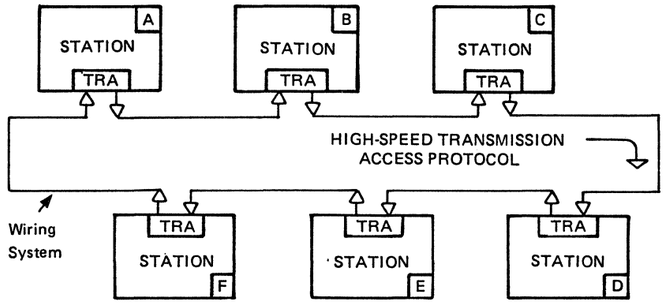
WHAT IS AN IBM TOKEN-RING NETWORK

Definition

An information transport system that provides high speed (4 Mbps) connection between users within a single building complex through the implementation of a common:

- Cabling system (called the ring)
- Communication adapter
- Access protocol





Legend:
TRA: Token–ring adapter

Station: Can be controller, display/keyboard terminal, Node etc...

CABLING SYSTEM (RING)

Transmission Media

Several transmission media may be used together.

- IBM Cabling System (Twisted-pair of copper wire).
- Telephone twisted communication wiring.

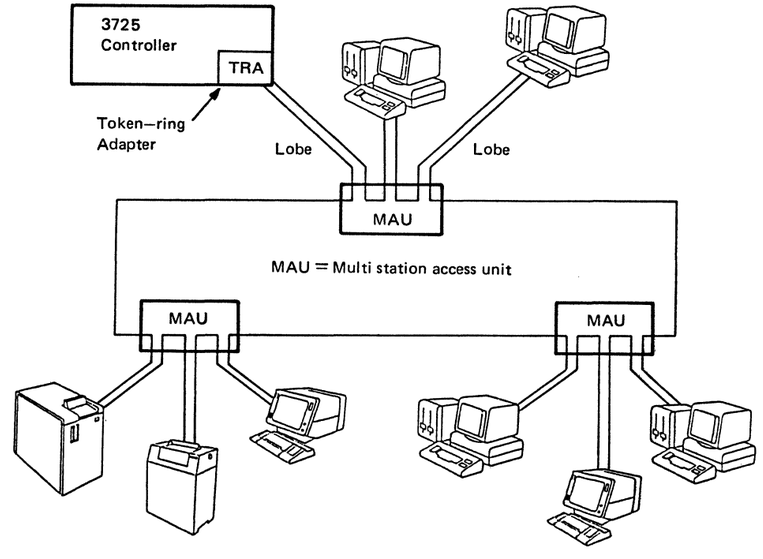
TOKEN-RING ADAPTER

The adapter provides the following functions:

- Frame and address recognition.
- Token generation.
- Error checking and logging.
- Buffering (Transmit and Receive).
- Time-out controls.
- Connect the product to the Token-Ring Network.

FOR EXAMPLE:
The 3725 is connected to the IBM Token-Ring Network thru the token-ring interface coupler card (TIC). A cable composed of two pairs of copper wires (transmit and receive pairs), connects the controller to the Multistation Access Unit (MAU) thru a wall outlet.

The cable running from the controller or a station to the MAU, is called a LOBE.

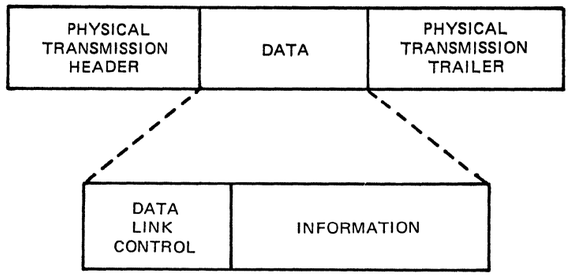


Overview of the IBM Token-Ring Network (Part 2 of 3)

ACCESS PROTOCOL

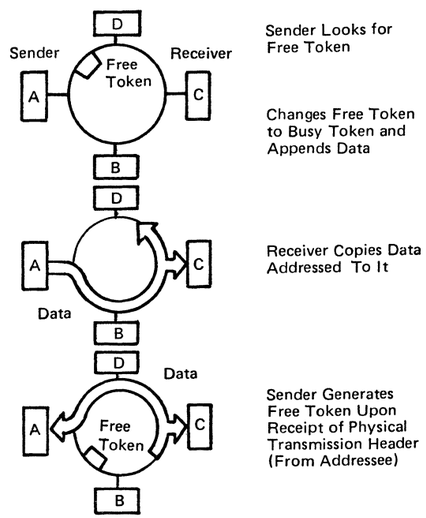
A token is a short message. It travels around a communications ring that allows attaching different systems to the communications facilities provided by that ring.
When a token is used by an attachment for transmission, it is known as a frame.
Actually, a token is a very short frame (3 bytes) that has no addressing, message, or error-checking capabilities included.
Those are added when the token becomes a true frame.

General Frame Format



After a station has been physically attached to a ring, it first synchronizes itself to the data patterns passing through it over the ring.
To allow a station to synchronize quickly at bit level, Differential Manchester encoding is used as the electrical means of encoding bits on the ring.

Token-Ring Access Control Protocol

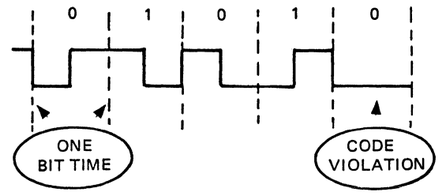


Token-Ring Encoding

DIFFERENTIAL MANCHESTER PROTOCOL:

During every bit time on the ring, there is at least one electrical level transition. This occurs in the center of the bit time.
Whether the bit being sensed is a one bit or a zero bit is determined by whether there is a level transition at the leading edge of the bit time.
If there is a transition, the bit is a zero.
If there is no transition, the bit is a one.

Because there is at least one transition in every bit time, a station that has only recently attached can gain synchronization relatively quickly.
A code violation is defined as the absence of the center bit transition and is allowed only in the starting and ending delimiter.



ADVANTAGES OF THE DIFFERENTIAL MANCHESTER CODE:

- Quick, reliable synchronization.
- Minimal DC component of token-ring signal.
- Code violation checking on all bits between starting and ending delimiter.
- Full transparency.
- Improved immunity from external interference.

MAJOR SYSTEM COMPONENTS

Nodes

A node may be one of the following machines:

- Communications controller.
- Intelligent workstation.
- Keyboard/display terminal.
- Printer.
- Terminal control unit.
- Processor.
- Facsimile device.
- Personal computer.
- Protocol converter.

Interface Adapters

Each node has its own token-ring adapter to access to the token-ring network or ring.

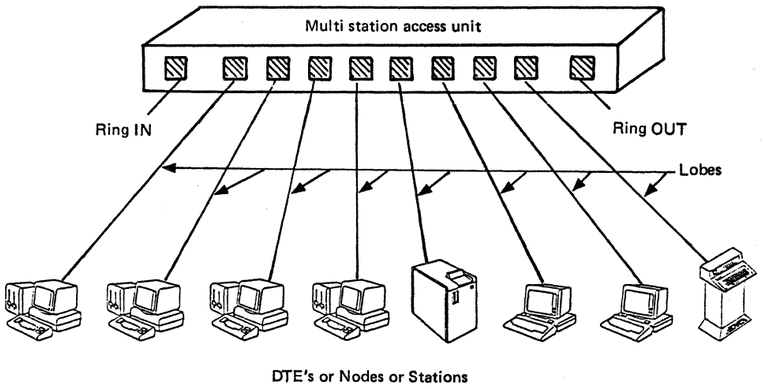
The 3725 may have several token-ring adapters called token-ring interface coupler card (TIC).

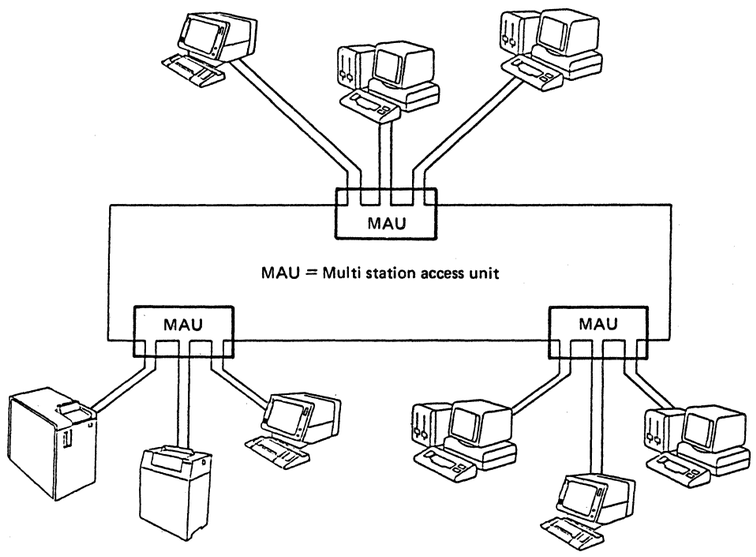
Overview of the IBM Token-Ring Network (Part 3 of 3)

Multistation Access Unit 8228

The Multistation Access Unit (MAU) provides insertion service to the main ring for its attached lobes. Nodes or stations are attached to the lobes.

The Multistation Access Units may include electronic or electro-mechanical switching elements.

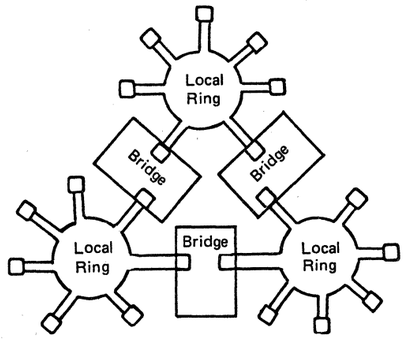


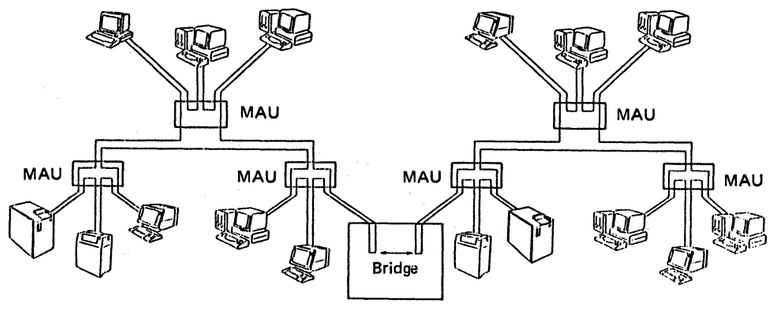


Bridges

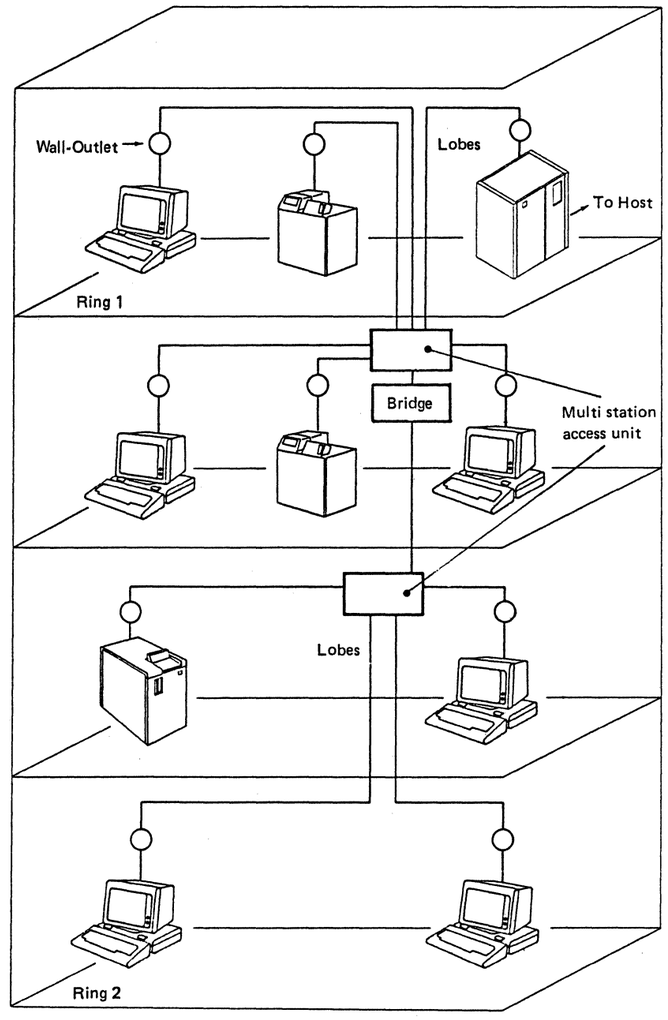
A bridge is a high speed switching device that allows the link between multiple rings and maintaining a physical ring separation.

RING TO RING VIA BRIDGE

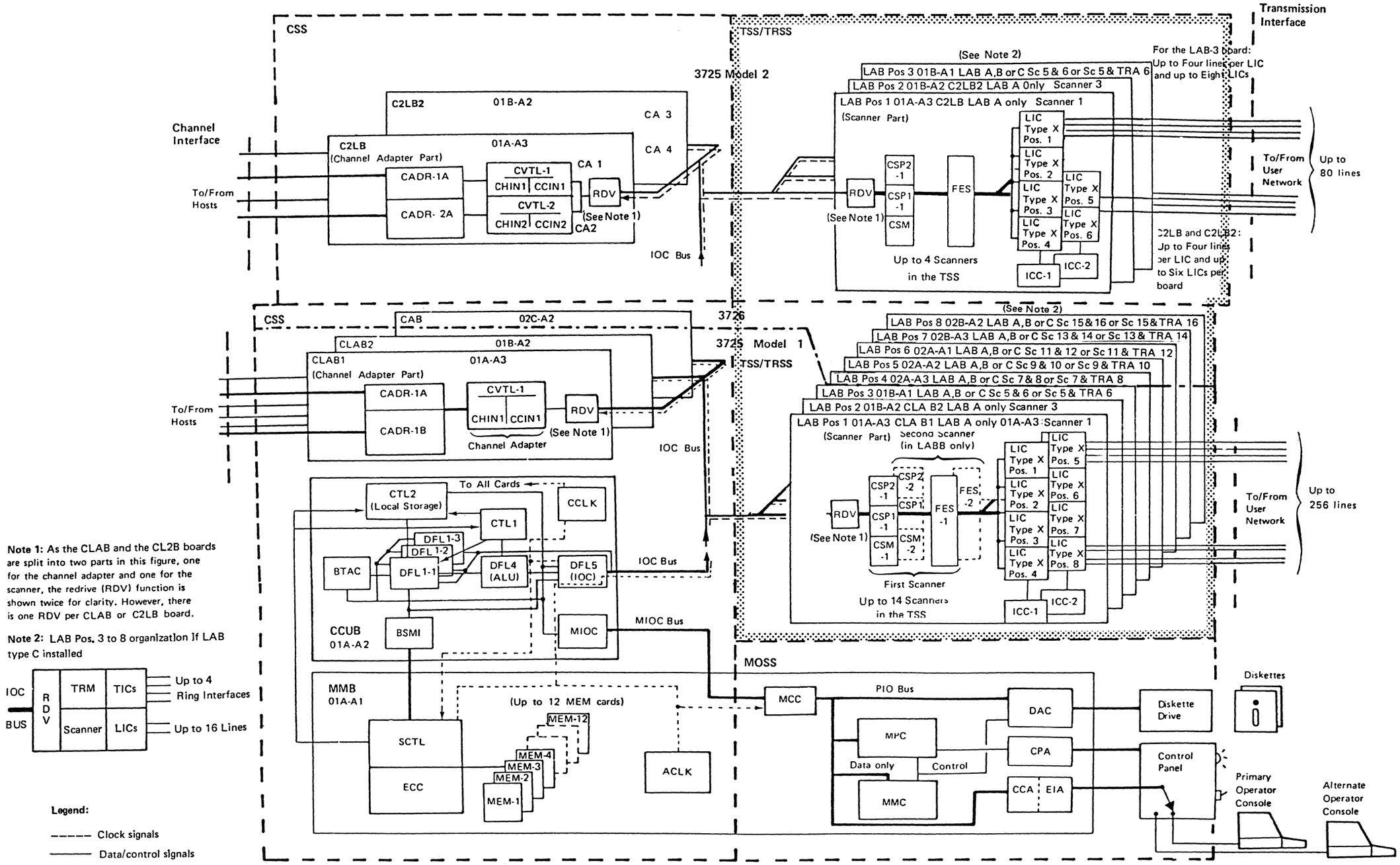




Typical Multi-Floor Wiring



The Token-Ring Subsystem Data Flow



Note 1: As the CLAB and the CL2B boards are split into two parts in this figure, one for the channel adapter and one for the scanner, the redrive (RDV) function is shown twice for clarity. However, there is one RDV per CLAB or C2LB board.

Note 2: LAB Pos. 3 to 8 organization if LAB type C installed

Legend:

----- Clock signals

——— Data/control signals

The Token-Ring Subsystem in the 3725

INTRODUCTION

The token-ring subsystem (TRSS) allows connection to an IBM Token-Ring Network which uses the token-ring protocol.

The TRSS in a communication controller, is controlled by the NCP/Token-Ring Interconnection (NTRI).

The hardware is based on a firmware driven card named token-ring interface coupler card (TIC) and the token-ring multiplexer card (TRM).

The combination of a TRM and the associated TICs (up to four) is called a token-ring adapter (TRA).

The combination of all the TRAs in a controller is called the TRSS.

One Token-Ring Network can be accessed by each TIC card.

LAB TYPE C

Each TRA is located on a new line attachment board type C (LABC).

A LAB type C has one TRM and one scanner and optionally, four TICs and one ICC.

Up to two LABs type C can be installed in the 3725 Controller and expansion frame.

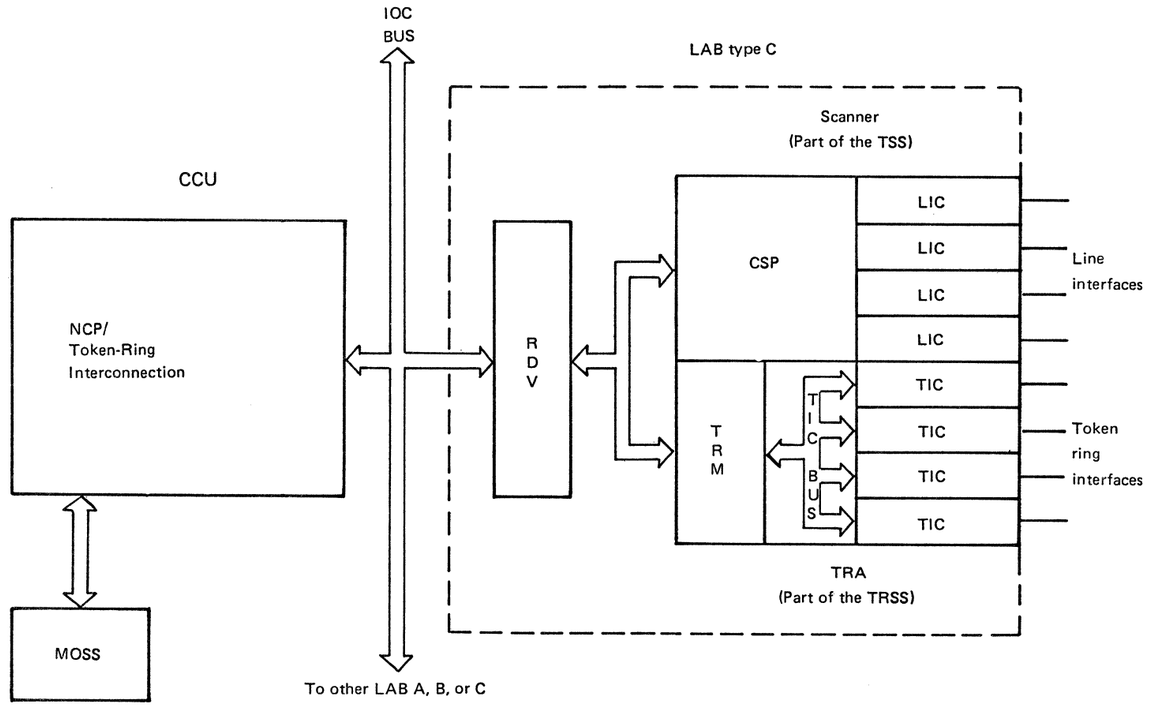
- One LABC in the base frame and one LABC in the expansion frame in any LAB position 3 to 8 can be installed.
- Or two LABCs in the expansion frame and nothing in the base frame.

Only one LABC can be installed in the 3725 Model 2.

A redrive card (RDV) and a scanner (part of the TSS) are also located on a LABC. The figure shows the general block diagram of the TRSS attachment in the 3725.

The TRM interfaces with the input/output control bus (IOC bus) and is accessed from the CCU with a unique address as for the scanner.

The TIC cards are connected to the TRM by a bidirectional bus which is called the TIC bus.



Token-Ring Interface Coupler Card (Part 1 of 3)

TOKEN-RING INTERFACE COUPLER CARD (TIC) DESCRIPTION

The TIC card is a card with the standard 96 I/O pins for mounting on a standard board. It has no top-card connectors. It has a low profile Berg connector for connection to the ring.
The card consists of four high-level functional areas.

- The front end that interfaces to the ring.
- The protocol handler.
- The message processor.
- The TIC bus interface control.

TIC DATA FLOW

The TIC adapter card can perform the following operations on the data stream that passes through it on the ring:

1. Repeat the received data without copying the data.
2. Repeat and copy the received data.
3. Change the state of single bits in the received data before retransmitting it.
4. Originate the transmission of data.
5. Remove messages from the ring that it has previously transmitted.

The diagram shows the relationship of the message processor to the other functional areas of the TIC card.

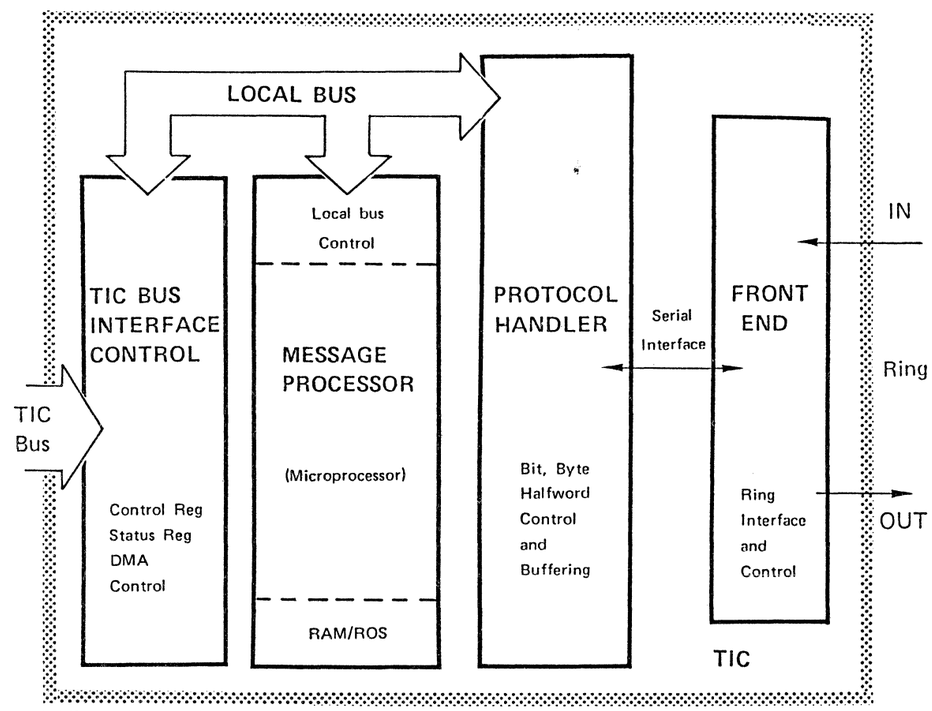
The message processor acts as the master control element for the protocol handler and the TIC bus interface control in the TIC adapter card.
Interaction with these functional areas is across an 18-bit local bus.

The ROS contains custom microcode which controls movement of data through the TIC adapter card.
It is also used to process messages that are passed on the ring at the physical layer (e.g., polls, logs, error recovery procedures, etc.).

The message processor can set up control conditions or interrogate status across the local bus using MMIO-type operations.
Data is accessed and stored in the RAM by the protocol handler and TIC interface control using direct memory access (DMA) protocols.

The TIC interface control provides a mechanism for transferring data between the TIC adapter storage (RAM) and CCU storage via DMA.
It operates halfword bus mode.

TIC Card Data Flow



TIC card data flow

Token-Ring Interface Coupler Card (Part 2 of 3)

THE FRONT END

The front end is the direct interface with the transmission line of the ring.

The front end transmits and recovers Manchester encoded data (including certain "violation" sequences) as it passes around the ring.

The front end synchronizes to the received data stream and develops a clock which is boundary aligned to the bit stream.
The bit stream, along with the derived clock, is passed to the protocol handler.
The front end requires monitoring and control by the protocol handler.

THE PROTOCOL HANDLER

The protocol handler acts as the logical interface between the front end and the local bus.

It prepares received data for processing by the microcode operating in the microprocessor of the message processor. It also prepares data that has been assembled by the microprocessor for transmission through the front end onto the ring.

It decodes addresses for recognizing received messages and strips messages from the ring that have been previously transmitted.

It checks that the ring is active and operational. Part of this checking is to identify and present to the microprocessor protocol errors for logging.

These logs are used to isolate the faulty element in the IBM Token-Ring Network.

The protocol handler controls and maintains the low-level bit and byte protocols on the ring.

THE MESSAGE PROCESSOR

The message processor is the general name for the microprocessor (microcode and the hardware).
It consists of the following items:

- A microprocessor which is based on the IBM Universal Controller (UC) architecture.
- Two static random access modules (RAM) organized as 2K X 16 bits.
- One static random access module (RAM) for parity.
- One read only storage (ROS) module.

The message processor acts as the base of the processing power of the TIC adapter.
The microprocessor executes instructions that are permanently stored in the read-only storage (ROS).
The instructions and parameters stored in the ROS are accessed by the message processor via the local bus.

The local bus is the halfword bus that physically connects all of the functional areas (excluding the front end modules) of the TIC card.

The microprocessor uses the random-access memory (RAM) as workspace when processing instructions.
The RAM is also used as buffer space for messages to be passed from the ring to the TIC bus interface and messages to be passed from the TIC bus interface to the ring.
Access to the buffer space by the protocol handler and the TIC bus control is also controlled by the microprocessor.

Transfers into and out of the RAM are performed using direct memory access (DMA) protocols.

IOC BUS INTERFACE CONTROL

The function performed by the IOC bus interface control can be visualized as a double interface DMA controller with a 128 byte hardware store-and-forward buffer. The 128-byte buffer can logically connect either to the IOC Bus interface control or to the local bus, but not both at the same time.

When DMA data transfer is occurring at the IOC bus interface control, the TIC adapter is the bus master, and the data can flow in only one direction (read or write) at a time.

In addition to its DMA function, the IOC bus interface control can also transfer data to or from TIC adapter storage.
This function is intended for adapter initialization and for obtaining TIC adapter status.

Token-Ring Interface Coupler Card (Part 3 of 3)

RECEIVE OPERATION

On receive, data is taken from the ring into the front end, where it is reshaped into distortion-free digital signaling elements.

The front end synchronizes to the received data stream and develops a clock which is boundary aligned to the bit stream.

This bit stream, along with the derived clock, is passed to the protocol handler.

The protocol handler converts the encoded data stream into coding that is usable by the adapter.

By counting received clock pulses, the protocol handler assembles the bit stream into halfwords units.
Parity is generated on the received data de-serialized from the ring to check data validity through the adapter.

During the receive sequence, cyclic redundancy check (CRC) calculation on the received data is begun.

The destination address is compared against the stored values in the protocol handler to determine if the message is to be copied by the adapter.
If so, the protocol handler conditions itself to begin the copy, and transfer into the message processor is started.

The destination and source addresses, the physical control field, the data portions, and the CRC characters of the in-progress received message are now passed to the message processor in sequence.

When the end of the CRC-protected field is received, the previously received CRC characters are compared to the calculated CRC.

If there is a match, the occupied bit is set in the ending delimiter and in physical control field extension. The receipt of the message is considered complete by the protocol handler.

If there is a mismatch on the CRC check, the message processor is signalled that the message should not be considered valid.

The message processor assembles the transfers received from the protocol handler into multi-byte segments.

When that assembly is complete, the message processor begins a transfer into the CCU memory. These segment transfers continue until the complete message has been transferred.

When the transfer has been completed, the message processor completes the receive operation by reporting the status of the transfer to the CCU.

TRANSMIT OPERATION

Data flow on transmit operations is essentially the reverse of the receive operation.
The message is accumulated by the CCU in the CCU memory, and the message processor is set up with the memory location and length information.
The message processor then does a memory-to-memory transfer of the message, including the destination and source addresses, from the CCU memory into the message processor.

The message processor signals the protocol handler to begin the transfer of the message from the message processor resident memory into the hardware buffers of the protocol handler.

When the protocol handler senses that a transmission is pending, it begins the transfer into its buffers.
After enough characters have been buffered, the protocol handler searches for a free token on the ring.
When one is found, the token is changed to a busy token.

The control characters are generated, and transmitted.

Sequential transmission out of the buffers continues with the destination and source addresses.

When the entire information field has been transferred, the protocol handler inserts the CRC characters that have been accumulated on the message.

The protocol handler begins to remove (strip) any data from the ring that is being received.

The incoming data stream is searched for a match of the source address with the address of the TIC adapter.

After a match is found and transmission is complete, the TIC adapter encodes and transmits a free token on the ring.

The transmission of the free token is followed by a continuous transmission of idle characters.

The TIC adapter begins also repeating the received data stream, and the transmit operation is considered completed.

Token-Ring Multiplexor Card (Part 1 of 2)

TRM CARD DESCRIPTION

The token-ring multiplexor card (TRM) handles the operations between the CCU and the token-ring interface adapters (TICs).
In one side it interfaces the CCU and the MOSS thru the redrive card (RDV) and the IOC bus, and in the other side, the token-ring cards (TICs) thru the TIC bus interface.
The TRM functions are the following:

- Convert PIO operations into MMIO operations.
- Convert DMA operations into cycle style operations.
- Can generate three different types of interrupt.
- Check the validity of the transactions between the CCU or the MOSS and the TIC adapters.

To perform these functions, the several registers are housed in the TRM:

- Buffer and the extended buffer
- TIC control register
- Interrupt request and Bus request register
- Diagnostic register
- Level 1 error status register
- Level 2 error status registers
- MOSS error status register.

IOC Bus Interface

The IOC bus interface allows communication between the TRSS and the control program (NCP/NTRI) or between the TRSS and MOSS.

See Chapter 11, "IOC Bus and Redrive" for more information.

Note: On the IOC interface, the CSR and the CSG are duplicated. They are CSR/CSG low for channel adapter AIO control and CSR/CSG high for TSS/TRSS AIO control. The contention between CSRH and CSRL is handled by the IOC.

SUMMARY OF THE IOC BUS INTERFACE SIGNAL LINES

Signal Line Names	Mnemonic	Initiated by	PIO/ TRM	AIO	Int
Input output	- IO	CCU	X	X	
Interrupt request removed	- IRR	TRM	X	X	
Halt	- HALT	CCU	X	X	
TA	- TA	CCU	X		
TD	- TD	CCU	X	X	
Cycle steal request high	- CSRH	TRM		X	
Cycle steal grant high	- CSGH	CCU		X	
Valid half-word	- VH	TRM	X	X	X
End of chain	- EOC	TRM		X	
Valid byte	- VB	TRM		X	
Modifier	- M	TRM		X	
Parity valid	- PV	TRM	X	X	
Data	-D0-D15	TRM,CCU	X	X	X
Parities	-B0,B1	TRM,CCU	X	X	X
Interrupt to MOSS	-INT. MOSS	TRM	X	X	X
Reset	- RESET	CCU			
CJ,CK,CL Clocks	-CJ,CK,CL	RDV	X	X	X
Select out secondary	SOS	RDV			X
L2 priority line		RDV,TRM			X
Cycle steal priority line		RDV,TRM		X	
CSG thru tag		TRM		X	
Select out primary tag	SOP	TRM			X

Token-Ring Multiplexor Card (Part 2 of 2)

TIC BUS INTERFACE

The TIC bus is a bidirectional bus which connects the TRM card to the TIC cards.

Three types of operation are used on this interface: DMA, MMIO, or IACK operation.

DMA operation allows the transfer of a burst of data between the TRM and the TIC.

During DMA operations, the TIC is the master of the bus and the TRM is the slave unit.

MMIO allows access to the registers of the IOC interface control of the TIC.

IACK cycles are used to access a vector in the TIC when the TIC requests an interrupt to the TRM. The vector indicates the cause of the interrupt.

TIC BUS SIGNAL LINES

- The bus requests (BR) (one per TIC) request a DMA operation.
- The interrupt request (IRQ) (one per TIC) alerts the TRM that the TIC needs service from the program in CCU or MOSS.
- The bus grants (BGR) (one per TIC) are used by the TRM to start the DMA operation on the TIC bus.
- The interrupt acknowledges (IACK) (one per TIC) are used to fetch the interrupt vector from the TIC card.
- The card selects (CS) (one per TIC) are used during the MMIO operation.
- The resets (RESET) (one per TIC) reset the associated TIC hardware.
- The read not write (RNW) indicates the direction of data transfer.
- The bus busy (BBS) is activated by TIC during a DMA operation. The TRM never activates this tag.
- The bus release (BRLS) is not used by the TRM.
- The bus error (BERR) aborts a DMA data transfer and generates a retry of the operation.
- The lower data strobe (LDS) controls the data on D7-D0. This tag is activated by the TRM during MMIO and interrupt operations and by the TIC during DMA operations.
- The upper data strobe (UDS) controls the data on D15-D8. This tag is controlled by the TRM during MMIO and interrupt operations and by the TIC during DMA operations.
- The address strobe (AS) indicates the 3 bytes are valid on the address bus during a TIC DMA operation. The TRM controls this tag during MMIO and interrupt operations.
- The address bus (An) is used as follows: Three bytes of address are given in DMA. In MMIO only 2 register select bits (address bus bits 1 and 2) are used to select the type of operation. These bits select specific registers in the TIC.
- The data transfer acknowledge (DTACK) is used with LDS/UDS during MMIO/DMA.
- The data bus (Dn) is a halfword with odd byte parity. Odd parity means an odd number of data plus parity bits are active.
- Activation of the system last transfer (SLT) signal informs the TRM that the DMA operation is at or near completion.
- The system clock (BCLK) is generated by TRM.

SUMMARY OF THE TIC BUS SIGNAL LINES

Signal Line Names	Mnemonic	Initiated by	PIO/ MMIO	DMA	Int
Address bus	+A1-+A2; +A3-+A23, +APL,+APH +APX,+A0	TRM, TIC TIC	X	X	
Data bus	+D0-+D15 +DPL, +DPH	TRM, TIC	X	X	X
Address strobe	- AS	TRM, TIC	X	X	X
Card select(4)	- CS	TRM	X		
Read / write	+ RNW	TRM, TIC	X	X	X
Upper data strobe	- UDS	TRM, TIC	X	X	X
Lower data strobe	- LDS	TRM, TIC	X	X	X
Data transfer acknowledge	- DTACK	TRM, TIC	X	X	X
Bus error	- BERR	TRM		X	
Bus request(4)	- BR	TIC		X	
Bus grant(4)	- BGR	TRM		X	
Bus busy	- BBSY	TIC		X	
Bus release	- BRLS			X	
Interrupt request (4)	- IRQ	TIC			X
Interrupt acknowledge(4)	- IACK	TRM			X
Reset (4)	- RESET	TRM			
System clock	+ BCLK	TRM	X	X	X
System last Xfer	- SLT	TIC		X	

TRM ARBITRATION MECHANISM

Since four TIC cards can be attached to the TRM, the TRM can simultaneously receive up to four bus requests for DMA and up to four interrupt requests from the TIC cards.

The arbitration logic comprises two scan wheels, one for bus requests (BR), the other for interrupt requests (IR). Each wheel in turn can point to either of the four attached TIC adapters.

Machine Internal Communications

The NTRI running in the central control unit (CCU) is able to communicate with the TRM or TIC cards by 3 main operations:

- Programed input/output or mapped memory input/output operations
- Cycle steal or direct memory access operations
- Interrupt operations.

PIO-MMIO OPERATIONS

Direct communications are made at the program level by use of input output halfword (IOH) instructions. These instructions are always passed on the IOC bus through the program input or output (PIO) operation.

The PIO is decoded by the TRM and is either responded to directly if it addresses the TRM or is transformed into a memory mapped input output (MMIO) if it addresses an associated TIC card. (See page 15-115 for details)

CS-DMA OPERATIONS

There are pointers in the TIC which enable the TIC to indirectly communicate with the CCU by use of CCU memory blocks.

Each TIC card issues direct memory access (DMA) requests to the TRM to fill or empty these memory blocks. These requests are passed to the IOC as cycle steal requests.

A cycle steal operation (AIO) which transfers the data between the main memory and the TIC is then performed. The program product uses IOHs to alert the TIC of the availability of new memory blocks. (See page 15-130 for details)

INTERRUPT OPERATIONS

In the other direction, the TIC can alert the NTRI by use of interrupt requests that are presented to the TRM.

The TRM determines what kind of interrupt has been raised by fetching an interrupt vector in the TIC.

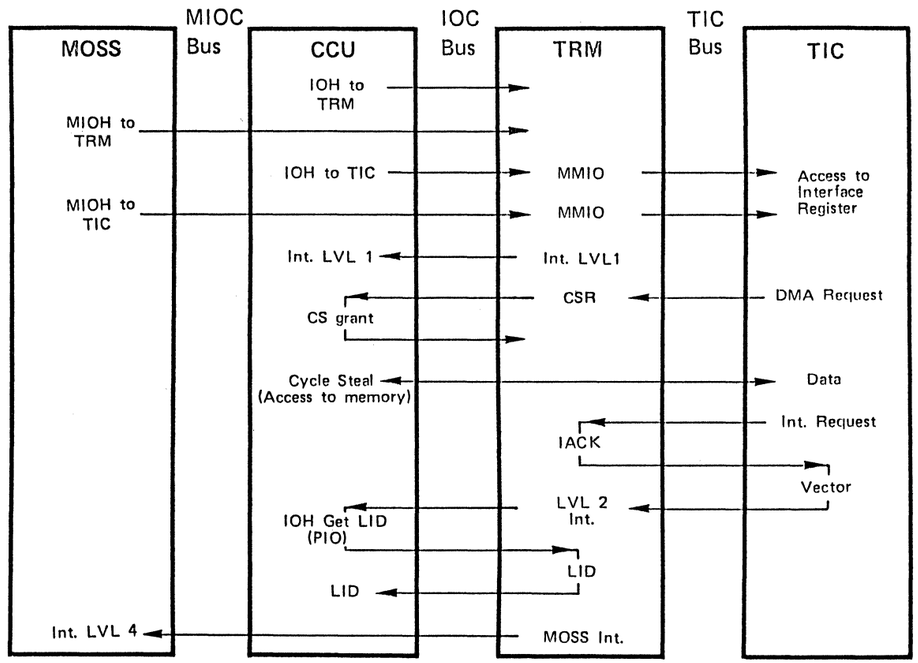
The TRM then presents a level 2 interrupt request to the CCU
The CCU then issues an IOH 'get line ID', which is responded to by the TRM with a line identifier determined by the value of the interrupt vector.

The TRM is capable of alerting the MOSS by use of the 'Interrupt to MOSS'. The MOSS is able to generate PIOs on the IOC bus by use of MIOH. (See page 15-135 for details)

IMPACT OF TRSS ON NCP LEVEL 1 CODE

Level 1 interrupt occurs in two cases:

1. When a TRA disconnect command is requested by the MOSS, a level 1 request is sent on the IOC bus.
2. When an error is detected on the IOC bus during an AIO or PIO operation.



Internal communications

Programmed Input/Output Operations

GENERAL PIO OPERATIONS

PIO operations are halfword operations driven either by the CCU or the MOSS.

The CCU program execution generates IOH instructions which are transformed into PIO operations in order to control the attached TIC adapters through the IOC interface.

The MOSS can generate MIOH operations which are treated the same way as IOHs.

Support operations with the TRM.

Support operations with a TIC through the TRM by converting PIO operations into MMIO operations.

The IOC bus control controls the IOC tags to synchronize the CCU and TRM exchanges.

The PIO decode and execute is the logic interface between the CCU and the different areas accessed in PIO mode.

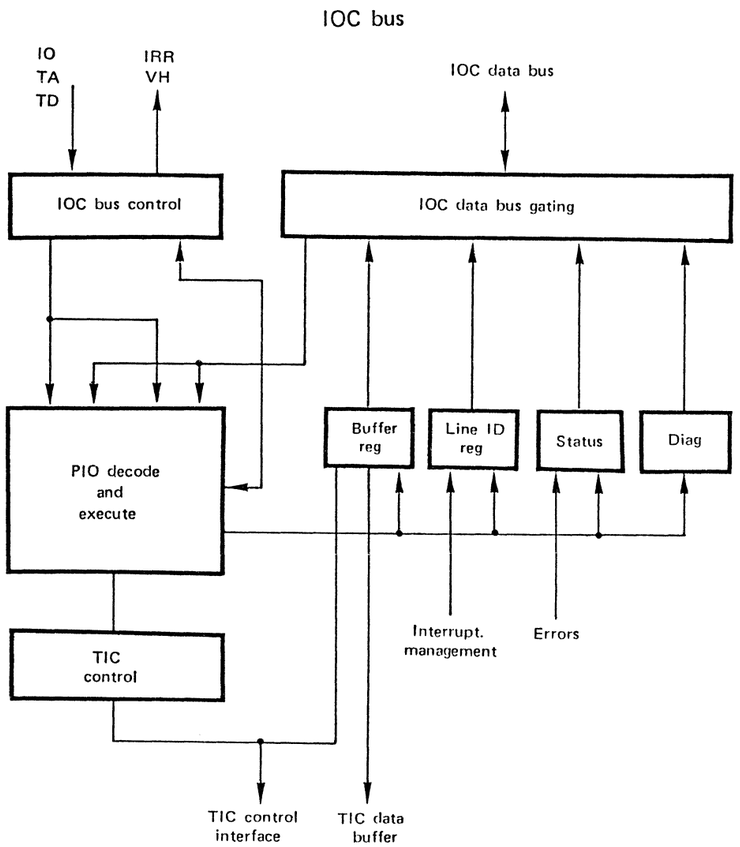
The line ID register is a halfword register which stores the type A line identification for the first TIC. The other seven line IDs are calculated using the contents of this register.

The status block is a set of registers used to log hardware detected errors. This set of registers include a level 1 error status register, four level 2 error status registers and a MOSS error status register.

The diagnostic register is used to control diagnostic operations in the TRM.

The TIC control is used to control the operations of each TIC.

TRM PIO OPERATION



PIO Format and Types (Part 1 of 4)

PIO FORMAT AT TA TIME

The PIO function is used to address registers in the TRM and in the TIC.

The format follows the general rules of the machine to allow compatibility with the other adapters.

The group address selects the token-ring subsystem (TRSS).

The subgroup selects a board in the machine. Thus, the first byte is used to address only one board in the machine with a TRA installed.

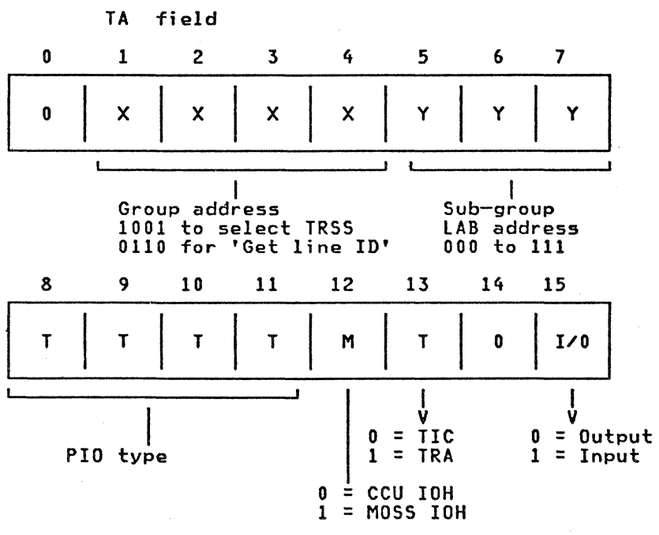
In the second byte, the least significant bit (LSB) (bit 15) indicates whether the operation is an input or an output to the CCU. The second LSB (bit 14) must be an 0, while the third LSB (bit 13) differentiates between operations intended for TIC or the TRM. The next bit (bit 12) indicates whether the operation was initiated by MOSS or the CCU. Finally, the four most significant bits of this byte define the PIO type.

The figure below describes the PIO format at TA time. Most figures in this section are shown in IOC bus format. Bit 0 is the most significant bit and bit 15 is the least significant.

The 'get line ID' PIO is also used which has a special format.

It uses the B '0110' group address and byte 1 must be x'01'.

There are two kinds of PIO operations, one type for the TRM and another for TIC.



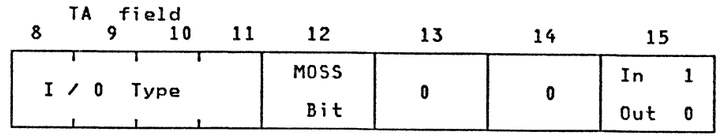
LIST OF THE PIO TYPES

PIO Types	Read	Write
0 0 0 0	Get line ID (CCU)	
0 0 0 0	Get command completion (MOSS)	Start/Stop/Mask/Unmask from MOSS
0 0 0 1	Get TRM control reg.	Set TRM control reg.
0 0 1 0	Get TIC control reg.	Set TIC control reg.
0 0 1 1	Read line ID base	Load line ID base
0 1 0 0	Read IR/BR reg.	Write IR/BR reg.
0 1 0 1	Read diag. reg.	Write diag. reg.
0 1 1 0	Read buffer reg.	Write buffer reg.
0 1 1 1	Read extend buffer reg.	Write extend. buffer reg.
1 0 0 0	Read computed line by MOSS	Programmed reset TRM
1 0 0 1	Get level 2 error status (1)	
1 0 1 0	Get level 2 error status (2)	
1 0 1 1	Get level 2 error status (3)	
1 1 0 0	Get level 2 error status (4)	
1 1 0 1	Get level 1 error status	
1 1 1 0	Get MOSS error status reg read	
1 1 1 1	CSCW	

Note: Usage of unused codes will be detected as an invalid IOH. To get the programmed reset TRM format, the NCP level 1 routine finds the get level 1 error status format in the CDS table and exclusive or's it with the pattern X'0051'.

PIO TYPES FOR TIC

The purpose of this section is to show the encoding of the TA field for TIC PIO/MMIO operations. The format of byte 0 is the same as PIO format at TA time.



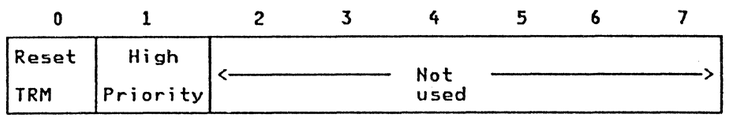
I/O Types	R e a d or W r i t e		
0 0 X X	Data register	TIC	X X
0 1 X X	Data register (with increment)	"	"
1 0 X X	Address register	"	"
1 1 X X	Interrupt register	"	"

X X	TIC number
0 0	1
0 1	2
1 0	3
1 1	4

PIO Format and Types (Part 2 of 4)

SET/GET TRM CONTROL REGISTER

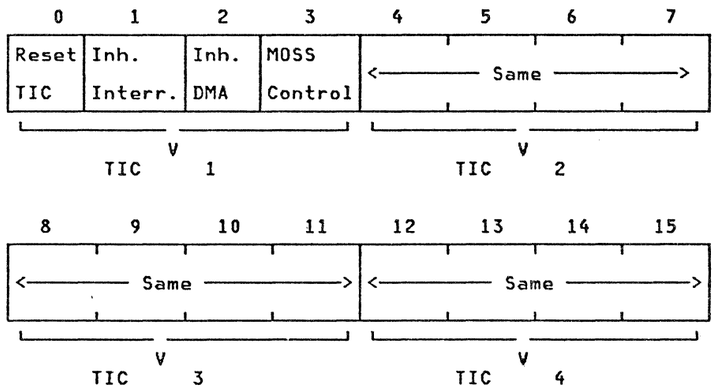
TD field byte 0 (byte 1 not used)



- RESET TRM
  - Set by the program, using the IOH programmed reset TRM (PIO type 1000), or by the IOC reset tag.
  - Reset by the program via the set TRM control register with TD bit 0 off.
  - The reset TRM bit is only an indicator of a previous reset. Setting this bit via a set TRM control reg IOH does not perform any reset function.
- HIGH PRIORITY
  - Set by the program to give this TRM high priority in the level 2 interrupt and cycle steal request mechanisms.

SET/GET TIC CONTROL REGISTER

TD field



- RESET TIC
  - Controls the reset line of each TIC
  - The reset TIC bit is set by the IOC reset tag or a set TIC control register PIO. The IOC reset tag is activated during machine power on reset and when MOSS performs a reset IOC function. A programmed reset TRM does not affect this bit.
- INHIBIT INTERRUPT
  - TIC interrupt request is ignored by the TRM
- INHIBIT DMA
  - TIC DMA request is ignored by the TRM
- MOSS CONTROL
  - A TIC interrupt is switched from a level 2 interrupt to a MOSS interrupt. The MOSS error status register is used instead of the level 2 error status register to log errors.

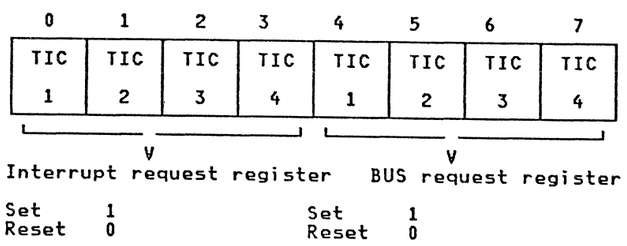
WRITE/READ INTERRUPT REQUEST AND BUS REQUEST REGISTER

This register is used by diagnostics to simulate interrupt requests and bus requests in order to check the interrupt and DMA mechanisms.

The read command is used for isolation in case of errors reported in level 3.

During wrap mode, the value set by a write IR/BR PIO will remain until another write IR/BR PIO is performed. However, during normal operation, the value will follow the tags coming from the TIC bus.

TD field byte 0 (byte 1 not used)



TRM BUFFER AND EXTENDED BUFFER

The buffer register is a two byte register.
The extended buffer is a single byte register which is accessed on byte 1 of the IOC bus.
These buffers are the interface between the IOC bus and the TIC bus. All data and addresses are passed from one bus to the other via these buffers.

TRM Buffer and Extended Buffer

The buffer register is a two byte register.
The extended buffer is a single byte register which is accessed on byte 1 of the IOC bus.
These buffers are the interface between the IOC bus and the TIC bus. All data and addresses are passed from one bus to the other via these buffers.

PIO Format and Types (Part 3 of 4)

TRM LEVEL 1 ERROR STATUS REGISTER

TD field

0	1	2	3	4	5	6	7
R/W	IO check	CSG	I/O	Halt	TA	TD	Disconn

8	9	10	11	12	13	14	15
TA Select	CS Select	TIC Bus Idle	CCU PIO Disable	TRM/ TIC	V		Invalid IOH

TIC number for DMA or interrupt

Level 1 interrupts can be generated by the:

- TRM on disconnect
- IOC on time-out, bus-in parity, CSCW, storage protect or address exception errors.

Error reporting using IOC level 1 interrupts is limited to errors occurring on the IOC bus interface and errors detected by the IOC.

TRM internal errors and the TIC interface errors found during PIO/MMIO, DMA/CS, or read interrupt vector operations, are reported by a level 2 interrupt for the appropriate TIC.

The pattern in the Level 1 status register for a disconnect level 1 interrupt is B'1000000100000000'.
This is compatible with the TSS to minimize the impact to the NCP level 1 interrupt routines.

When the IOC detects an error, it raises the halt tag. When the TRM detects the halt tag, it sets the level 1 status register with the exception of bits 7 and 11. Bits 7 and 11 are the state of the indicated latches when the get level 1 error status register PIO is executed. During an IOC level 1 interrupt bits 0-3, 5-6, 8-10 and 12-15 are valid only if bit 4 is on.

- The R/W bit (bit 0) is on if a read PIO/AIO is being executed. It is off if a write PIO/AIO is being executed. The R/W bit is always on for a disconnect level 1 interrupt. It is valid only if bit 8 or 9 is on for an IOC level 1 interrupt.
- The IO check bit (bit 1) is on if the TRM detects a parity error on the IOC bus when the halt tag is activated.
- The CSG bit (bit 2) is on if the CSG tag into the TRM is active when the halt tag is activated.
- The I/O bit (bit 3) is on if the I/O tag into the TRM is active when the halt tag is activated.
- The halt bit (bit 4) is on if the TRM has detected the activation of the halt tag on the IOC interface.
- The TA bit (bit 5) is on if the TA tag into the TRM is active when the halt tag is activated.
- The TD bit (bit 6) is on if the TD tag into the TRM is active when the halt tag is activated.
- The disconnect bit (bit 7) is on if the TRM is in disconnect mode when the read level 1 error status register PIO is executed.
- The TA select bit (bit 8) is on if the TRM has recognized its address (with good parity) and the PIO has not completed when the halt tag is activated.
- The CS select bit (bit 9) is on if the TRM has trapped CSG for an AIO and the AIO has not completed when the halt tag is activated.
- The TIC bus idle bit (bit 10) is on if the TRM detects that the TIC bus is idle when the halt tag is activated.
- The CCU PIO disable bit (bit 11) is on if the TRM is disabled for CCU PIOs when the read level 1 error status register is executed.
  If the TRM is CCU PIO disabled, the level 1 error status register can be successfully read by an MIOH only.
- The TRM/TIC bit (bit 12) is on if an AIO or a get line ID PIO is being executed by the TRM when the halt tag is activated.
  If this bit is on, it means that bits 13 and 14 are valid.
  If this bit is off, it means that the error cannot be associated with any particular TIC.
- Bits 13 and 14 are used to encode the TIC number for some errors.
  During a cycle steal operation, bits 13 and 14 are used to identify the TIC that was doing the DMA operation when the halt tag was received.
  They also indicate the originator of the level 2 interrupt for an error during a get line ID PIO. These bits are valid only if bit 12 is on.

  0 0 = TIC 1
  0 1 = TIC 2
  1 0 = TIC 3
  1 1 = TIC 4

- The invalid IOH bit (bit 15) is on if the TRM decodes an invalid PIO type when the halt tag is activated.
  This bit is valid only if the bit 5 is on.

Note: The level 1 error status register is compatible with the TSS error status type one to minimize the impact to the NCP level 1 interrupt routines for IOC detected errors.

TRM LEVEL 2 ERROR STATUS REGISTERS

The TRM provides 4 level 2 status registers - 1 per TIC.

The level 2 status information is set by the TRM when an error is found in the TRM (not on the IOC bus interface), an error is detected on the TIC interface or the TIC requests an interrupt.

This information can be read by the program thru a 'Get TRM level 2 error status' command. The error or status is reset if the PIO read operation completes with no errors. After the status is successfully read, the TRM is able to log another error or status.

FORMAT 1 - ERROR DETECTED BY TRM

TD field byte 0 (byte 1 not used)

0	1	2	3	4	5	6	7
1	Error	Encoding		Error	Encoding		0

TRM error found in MMIO — V

Error found in DMA or Interrupt vector input. — V

- Error encoding

  0 0 0 No error
  0 0 1 TRM internal
  0 1 0 TIC interface type 1
  0 1 1 TIC interface type 2

Internal The TRM is suspected.

Type 1 The working TIC is suspected.

Type 2 All TICs can be suspected (idle state error).

This error encoding is performed by the TRM using the checkers in the TRM (See page 15-160, "TRM Error Detection and Reporting").

The status provides 2 independent error fields:

- PIO/MMIO operations which require program retry
- Operations which do not require program retries

  DMA (TIC retries automatically)

  Input interrupt vector (TRM automatically retries)

Both fields can be significant at a given time.

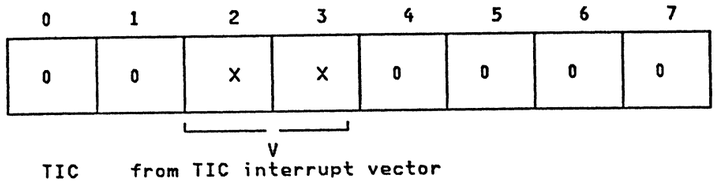
When a field is set, another error will not be set in the same field until it is reset by a successful 'get TRM level 2 error status'.

A DMA or input interrupt vector operation is retried automatically when the error in the level 2 error status register is reset by a successful 'get TRM level 2 error status' PIO.

PIO Format and Types (Part 4 of 4)

FORMAT 2 - INTERRUPT REQUEST BY THE TIC

TD field byte 0 (byte 1 not used)



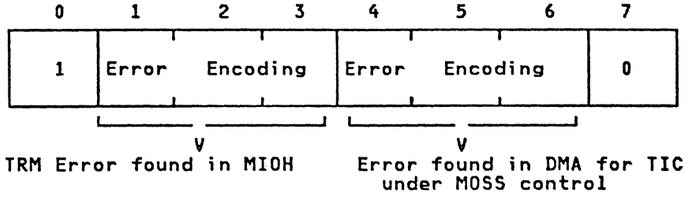
Format 2 is used when a TIC requests an interrupt.

- The contents of this register will be:
  - B'00100000' for the SCB clear vector
  - B'00010000' for the adapter check vector.
  - B'00000000' or B'00110000' for the other TIC vectors.
- NTRI initializes the TIC interrupt vectors to
  - B'XX10XXXX' for SCB clear
  - B'XX01XXXX' for adapter check
  - B'XX00XXXX' or B'XX11XXXX' for the remainder
- A format 1 entry will cause the type B line ID to be used.
  A format 2 entry with bits 2 and 3 = B'10' or B'01' will also use the type B line ID. All other cases use the type A line ID.
- If the level 2 error status register contains a format 2 entry and the TRM detects an error that requires logging a format 1 entry, the TRM will overlay the format 2 entry.
  After the format 1 entry is read and reset, the interrupt vector will again be fetched from the TIC and the format 2 entry set into the register again.

MOSS Error Status Register

- Format 1 - Error detected by TRM in MOSS PIO/MMIO or DMA

  TD field byte 0 (byte 1 not used)



  - Error encoding

    0 0 0 No error
    0 0 1 TRM internal
    0 1 0 TIC interface type 1
    0 1 1 TIC interface type 2

    Internal The TRM is suspected.

    Type 1 The working TIC is suspected

    Type 2 All TICs are suspected

- MOSS error status register is used to log errors associated with MOSS operations.
  If a MOSS control bit in the TIC control register is on, all MMIO and DMA errors for that TIC will be logged in the MOSS error status register.
  If the TRM is connected, all MIOH/MMIO errors will be logged in the MOSS error status register.
  If the TRM is disconnected and the associated MOSS control bit is off, all MMIO and DMA errors for a TIC are logged in the normal level 2 error status register.
- Format 2 is not used for the MOSS error status register.

TRM PIO Functional Description

TRM PIO Initialization

The TRM detects a request from the CCU when it detects that the IO tag is active. At this time, the TRM can be in one of the following states:

IDLE: Ready to enter a PIO sequence

INTERRUPT REQUEST PENDING: Since interrupt requests are presented to the CCU on the data bus, the TRM and all other adapters on the IOC bus must remove their interrupt requests.

CYCLE STEAL REQUEST PENDING: Since cycle steal request (CSR) is presented to the CCU on a dedicated line, the TRM does not deactivate the request while performing the PIO operation.

TRM Selection

DIRECT SELECTION: The CCU knows the TRM address and sends it at TA time.
The TRM decodes its address by comparing the sub-group and IOC bus number of the address/command halfword to the board wired address.
It also compares the group address to B'1001' and bit 14 to B'0'.
If all compares are equal, the TRM becomes selected and sends VH to the CCU.

INDIRECT SELECTION: The CCU issues a PIO 'Get line ID' when it honors a level 2 interrupt because it does not know which adapter requested service.
In this case, the TRM becomes selected when it recognizes the 'get line ID' command and it is preselected.

RESTRICTIONS

- If the TRM is in CCU PIO disable mode, it can be selected only if the MOSS bit = 1.
- When a CCU operation is initiated the IOC bus is busy until the end of the operation.
  This operation cannot be overlapped with another operation such as an AIO.

TRM MOSS SELECTION: When a PIO is executed by MOSS through the IOC interface, the MOSS bit is active in the command field.

TRM PIO MANAGEMENT

The TRM decodes and executes the PIO operations issued by the CCU.

PIO Write Sequence

CCU		TRM
Deactivates TD, then activates IO.	——————>	Receives IO active, deactivates IR if any.
		Activates IRR and
Receives IRR active and VH inactive, places address and command on data bus and	<——————	deactivates VH.
activates TA.	——————>	Receives TA active, decodes address – If selected, latches command and
Receives VH active, deactivates TA.	<——————	activates VH.
		V
		Receives TA inactive,
Receives VH inactive, places data on data bus and activates TD.	<——————	deactivates VH.
	——————>	Receives TD active, latches data and
Receives VH active, deactivates TD and	<——————	activates VH.
clears data bus.	——————>	Receives TD inactive,
Receives VH inactive,	<——————	deactivates VH.
deactivates IO.	——————>	Receives IO inactive, activates VH, deactivates IRR and
Receives VH active and IRR inactive,	<——————	activates IR if any.
activates TD.	——————>	Receives TD active and TRM logic is ready for next operation.

End of operation

PIO Read Sequence:

Same as the write sequence except during TD time the data is sent from the TRM to the CCU.

TRM Mapping of PIO to MMIO

Some PIOs are used to access the TIC cards. During these operations, (the protocol is the same on the IOC bus) MMIO operations are performed by the TRM on the TIC bus.
Bits at TA time are used to generate the address on the TIC bus, and the halfword data at TD time is transferred to the TIC data bus.

TIC INTERFACE MANAGEMENT BY THE TRM

To perform an MMIO operation, the TRM activates the following control tags of the TIC bus:

CS. Card select to select the addressed TIC.

RNW. Read not write to indicate the direction of the exchange.

RS. Register select (TIC address bus bits A1 and A2) to specify the addressed TIC register.

UDS. Upper data strobe.

LDS. Lower data strobe.

AS. Address strobe.

The TIC waits for at least one data strobe to activate its MMIO timing.

UDS controls the transfer of the upper data byte of the bus, and LDS, the lower data byte.

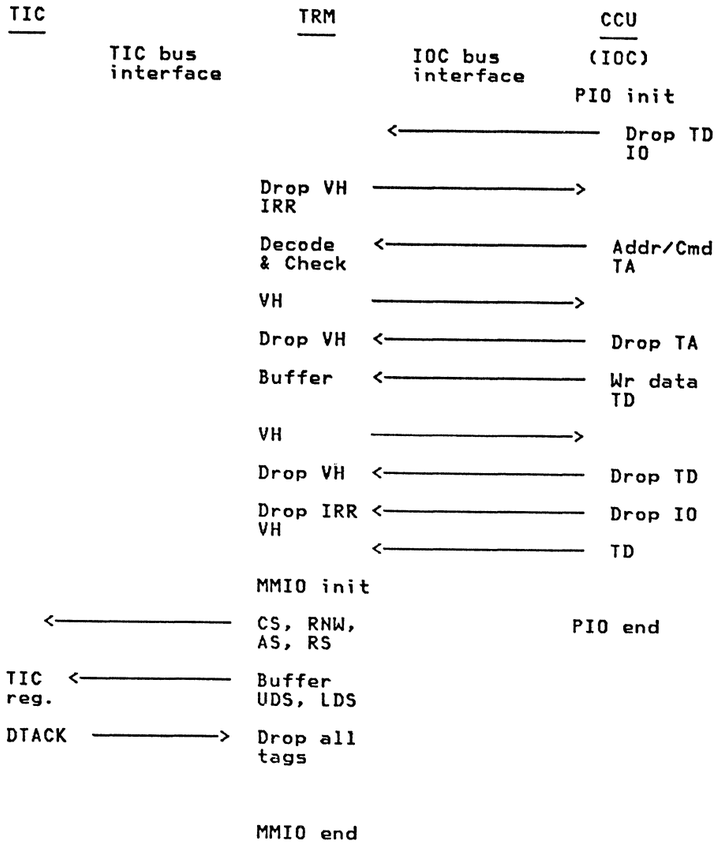
The response of the TIC to an MMIO is data transfer acknowledge (DTACK), to notify the TRM either that the data has been loaded into the selected register (MMIO write), or that the data is available on the bus (MMIO read).

On receipt of DTACK, the TRM releases the MMIO control tags, and the operation ends on the TIC bus.

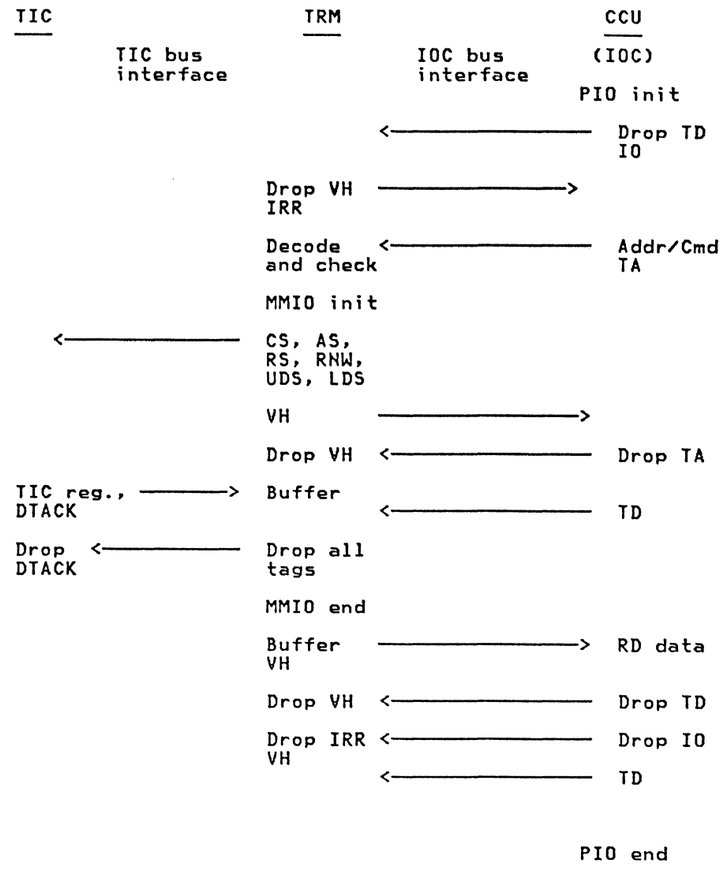
If the TRM detects an error condition on the IOC bus during an MMIO write, it does not even initiate the MMIO operation on the TIC bus.
Once the TRM starts an MMIO operation, it will complete it normally unless a hardware error is detected on the TIC bus.

PIO/MMIO HAND-SHAKING MECHANISM

PIO/MMIO WRITE



PIO/MMIO READ



TRM PIO Command Description

GET LINE IDENTIFICATION

This is the command by which the program running in the CCU is able to find the origin of the level 2 interrupt.
This command is a broadcast command which is answered by the first adapter in the daisy chain with a level 2 interrupt pending.

A level 2 interrupt can be originated by the TRM for two reasons; either to report errors detected by the TRM or to pass TIC interrupts to the control program.

The line identification (line ID) is computed by the TRM using the following algorithm:

```
Base + (8 * (TIC number - 1)) + (4 * Line ID type)
Base          = Data loaded by the PIO "load line ID base"
TIC number    = 1 - 4
Line ID type= 0 for type A
              1 for type B
```

(See page 15-135 "Get Line ID Generation")

The level 2 interrupt is reset in the TRM upon successful completion of the 'get line ID' PIO.

READ COMPUTED LINE ID BY MOSS

The function of this PIO is the get line ID except that it is not a broadcast command and is always answered by the addressed adapters.
This command is issued by the MOSS.

READ/LOAD LINE ID BASE

This function is used for loading the first line ID in the halfword register during TRM initialization.
The use of this information is described in the get line ID command.

GET COMMAND COMPLETION

TD field

0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0

8	9	10	11	12	13	14	15
MOSS Int.	0	0	0	Level 2 Int.	MOSS Status Int.	MOSS Direct Int.	MOSS Int. Mask

Get command completion is used by MOSS to answer an interrupt to MOSS.

Bit 8 MOSS interrupt allows MOSS to verify that the addressed adapter is the originator of the interrupt.

Bit 8 is automatically reset by the TRM at the end of the get command completion unless the operation is abnormally terminated by a halt.

Bit 8 will stay active if the get command completion is terminated by a halt.

Bit 12 indicates that a status is pending in one of the level 2 error status registers.

Bit 13 indicates that a status is pending in the MOSS error status register.

Bit 14 indicates that the interrupt request from a TIC under MOSS control is active.

Bit 15 reflects the current state of the MOSS interrupt mask.

SET COMMAND

Four commands are defined, based on the data sent at TD time.

STOP command: TD=X'0B00'
is used by MOSS to start the disconnect process.

START command: TD=X'0C00'
is used by MOSS to reconnect a TRA.

MASK command: TD=X'8000'
will mask all the interrupts to MOSS except the "disconnect end of operation" MOSS interrupt request. The state of the MASK can be checked by the get command completion.

UNMASK command: TD=X'4000'
will unmask the interrupts to MOSS.

All other TD values cause no action to take place. The PIO will complete without error, however, if the other TD values are used.

PROGRAMMED RESET TRM

The programmed reset TRM resets the TRM but does not force a reset to the attaching TICs.
This command will set the reset bit in the TRM control register.
The disconnect and CCU PIO disabled state of the TRM is not changed by a programmed reset.
The TD field is not used by this PIO.

The next PIO issued to the TRM after a programmed reset must be a get TRM control register PIO.

READ CSCW

TD Field

0	1	2	3	4	5	6	7
0	0	0	0	0	1	0	0

8	9	10	11	12	13	14	15
1	1	MOSS	TRA	Identifier			0

The read CSCW is a diagnostic function to read the CSCW.
(See "Cycle Steal Operations")

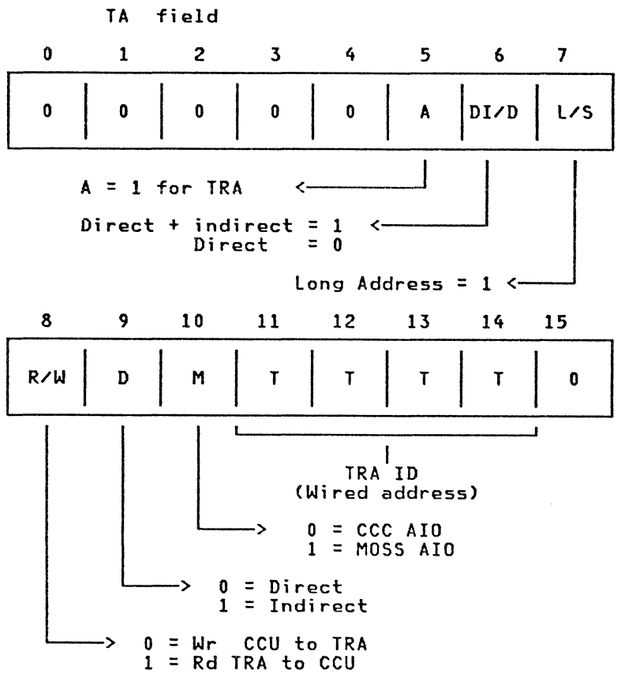
The MOSS bit is set to 1 if the TRM is under MOSS control (CCU PIO disabled).
See "Cycle Steal Operations" for a description of the TRA identifier.

TRM Cycle Steal Operations

In the 3725 controller family, the IOC bus is managed by the CCU.
In order for an adapter to access the main memory without program intervention, it must use CCU resources.
The adapter access of main memory is therefore called a "cycle steal" operation since CCU cycles are used (stolen) to perform the access.

Cycle steal is also called an adapter initiated operation (AIO) because it is initiated by an adapter when the adapter wants to transfer data to or from the main memory without the use of the control program.

Different types of cycle steal operations are identified by a control halfword.
This control halfword is called the cycle steal control word (CSCW).
The CSCW is sent by the adapter as the first data halfword during a cycle steal operation.
The format of the CSCW is shown in the following figure.



DIRECT LONG AND INDIRECT OPERATION FOR NORMAL CYCLE STEAL

The TSS and TRSS use a shared pointer register (X'F') for cycle steal operations.
For this reason the direct/indirect transfer mode is used as follows:

- The first 2 halfwords of data after the CSCW (address high, address low) are used to load the pointer register X'F'. Since it is a 22 bit register, 2 halfwords are needed.
  Therefore, LONG is specified in CSCW bit 7.

- The next halfword of data will be stored into the CCU storage or will be read from the CCU storage (depending on the R/W bit in the CSCW) according to the address contained in the pointer register.
  For each halfword transmitted, the pointer register will be advanced by 2.

Since the pointer register is fixed (X'F') and known, there is no need to include it in the CSCW. Therefore, bits 10 to 14 are used as an identification for the TRA.
If a level 1 interrupt occurs for an AIO, the CCU can read the following information by executing an input '75':

CSCW bit 5	1 for TRSS or TSS
CSCW bits 11-14	TRA ID for TRA

The TRA identification number is composed of the 3 bit subgroup address (board address) in bits 11 - 13 and bit 14 which is a 1. This TRM ID enables the program to determine which adapter was performing an AIO when an error occurs.

The pattern for the normal CSCW is B'0000 0111 R0MT TTT0'.

R is 1 for read and 0 for write
M is 1 for MOSS and 0 for CCU
TTTT is the TRA identifier.

DIRECT SHORT OPERATION FOR ERROR CYCLE STEAL OPERATION

This operation is used in case an error is detected at the beginning of a TIC DMA operation.
In this case, the TRM modifies the CSCW to direct and short mode which allows the TRM to stop the operation after sending only one address halfword to the IOC pointer register.
The address halfword is sent as all zeros.

The pattern for the error CSCW is B'0000 0100 R1MT TTT0'.

CYCLE STEAL SEQUENCE DESCRIPTION

TRM MAPPING OF DMA TO CYCLE STEAL

Exchanges of control blocks or data buffers between the CCU memory and the TIC RAM are initiated by the TIC.

The DMA operations are mapped by the TRM to cycle steal operations on the IOC bus.

The TRM manages the bus-to-bus timings and acts as the vehicle by which addresses and the data are transferred between the CCU and the TIC.

The CCU addresses are provided directly by the TIC. They have been previously loaded into the TIC memory by the NTRI.

The TIC interface is a 16-bit interface, the DMA is optimized when halfwords are exchanged on this interface.

A single byte can be presented on the IOC bus only if it is the last byte transferred.

The TRM transfers over the IOC bus in 64-bytes bursts.

TRM Interrupt Operations (Part 1 of 4)

The TRM can generate 3 different types of interrupts:

LEVEL 1 INTERRUPT: For a disconnect function initialized by the MOSS.
This interrupt is put on the IOC bus byte 0 bit 5 between PIO and AIO operations.
The IOC will also generate a level 1 interrupt for errors it detects during a PIO or AIO with the TRM.

LEVEL 2 INTERRUPT: When the TRM passes an interrupt initiated by a TIC or when an interrupt is generated as the result of an error detected by the TRM.
This interrupt is put on the IOC bus byte 0 bit 1 between I/O operations.

INTERRUPT TO MOSS: When the TRM has been put into disconnect mode, or a TIC is under MOSS control or there is an MIOH error.
This interrupt is sent to the MOSS on a dedicated line.

Note: For the level 1, contention between adapters is treated by the redrive card in such a manner that the first adapter on the IOC bus will be serviced first.: For the level 2 a priority mechanism at 3 different levels (machine, frame, board) is used to resolve contention between the different adapters.
Each adapter can be set to either high or low priority.
For adapters set to the same priority, the first adapter on the IOC bus will be serviced first.

Since the MOSS works with only one adapter at a time, no contention is possible.

LEVEL 1 INTERRUPT

General Operation

A level 1 interrupt can happen for two reasons:

- The IOC has detected an error and raised a level 1 interrupt that the program is able to read in the external register X'76'.
- A disconnect operation has been sent by the MOSS.

Scenario for IOC Level 1 Error Recovery

Origin			Operation	TA	TD
IOC	TRM	NCP			
X			Send halt tag		
X			Raise level 1 interrupt		
	X		Load level 1 error status reg		
		X	Read ext. regs. 7E,76,75		
		X	Reset IOC level 1		
		X	Get level 1 error status	X'ccD5'	X'rrrr'

cc TRM Address B'01001sss'
sss is the sub-group address (board address)
rrrr is the level 1 error register contents
(see note below and get level 1 error status register)

Note: For compatibility with the TSS and existing NCP error recovery code, the response to a get level 1 error status for an error on a get line ID is B'1xx11xxx1xxxxxxx'.

Disconnect Operation Scenario

Origin			Operation	TA	TD
MOSS	TRM	NCP			
X			Disconnect (TRSS stop)	X'cc04'	X'0B00'
	X		Level 1 to NCP		
	X		Load level 1 error status reg.		
		X	Poll command to redrives	X'4001'	X'aadd'
		X	Get level 1 error status CSP if scanner installed on LAB	X'xxxx'	X'yyyy'
		X	Get level 1 error status TRM	X'ccD5'	X'8100'

cc TRM address B'01001sss'
sss is the sub-group address (board address)
aa RDV address B'11pppsss'
ppp is the redrive primary address
dd RDV error register
xxxx is CSP get level 1 error reg
yyyy is CSP response to get level 1 status

LEVEL 2 INTERRUPT

Line Identification (Line ID) Generation

When the CCU receives a level 2 interrupt, it places a PIO get line ID command on the IOC bus.

This command is decoded by all adapters. However only one, according to the level 2 interrupt priority mechanism, will answer with a line ID. The line ID is information which allows the program to identify the requesting adapter and to process the interrupt.

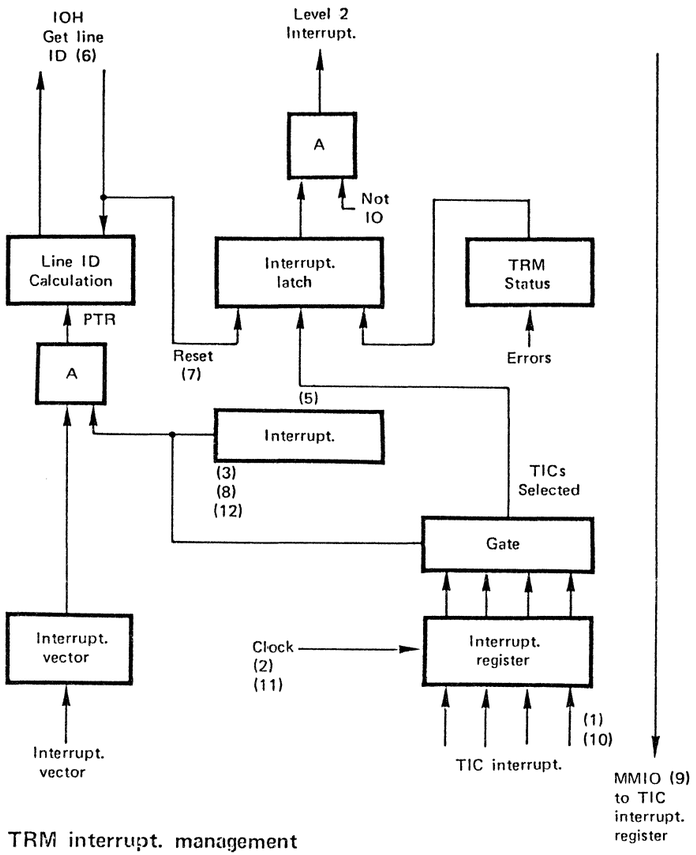
- There are 2 different line IDs per TIC:
  - Type A. For the TIC interrupts for which the system status block (SSB) is cycle-stolen to the CCU.
  - Type B. For the TIC interrupts for which the SSB is not cycle-stolen to the CCU (SCB clear or adapter check) and interrupts generated for TRM detected errors.

General Description

- The TRM requests a level 2 interrupt to the CCU for two reasons:
  - The TRM sends an interrupt initiated by a TIC.
  - The TRM generates an interrupt in case of a TRM detected error.
- The general data flow for the interrupt operation is shown on page 15-136, "TIC Interrupts".
- In both cases the response from the CCU is a get line ID. The TRM answers by presenting a line ID type A or B.
- Receiving the line ID allows the program to process the corresponding interrupt.
- The get line ID command resets the level 2 interrupt and the TRM is now free for another operation including an interrupt (IR) from another TIC.
- The CCU interrupt code sequence ends with a reset interrupt PIO/MMIO to the concerned TIC.
The TIC will lower its interrupt line and this will allow the TRM to treat another interrupt coming from the same TIC.
- In its turn the TIC is free for other operations.

TRM Interrupt Operations (Part 2 of 4)

TIC INTERRUPTS



TRM interrupt. management

(1) TIC n raises an interrupt.
(2) Interrupt is stored in TRM IR/BR register.
(3) TIC n on selected.
(4) TRM inputs interrupt vector.
(5) TRM raises a level 2 interrupt.
(6) CCU issues get line ID.
(7) Get line ID resets TRM level 2 interrupts.
(8) Another TIC is selected if another request is stored in the TRM interrupt register
(9) CCU resets the TIC interrupt.
(10) TIC n interrupt falls.
(11) TIC n interrupt is reset in the TRM IR/BR register.
(12) Interrupt from TIC n will be now honoured. TIC n.

TIC INTERRUPT SCENARIO

1. The TIC interrupts the CCU when the TIC adapter or ring status has changed.
2. The TIC will cycle steal (DMA) the system status block (SSB), if any, to the CCU and then send an interrupt request to the TRM.
3. The TRM raises interrupt acknowlegment (IACK).
4. Upon receiving IACK, the TIC gates the interrupt vector onto the lower data bus.
   Two bits of the interrupt vector are then read into the TRM level 2 error status register.
   - The interrupt vector identifies the cause of the interrupt.

     Six interrupt vectors are loaded into the TIC during initialization.
     Each vector consists of one byte.
     Each byte contains a value which identifies the cause of the interrupt:
     - Command status
     - Transmit command status
     - Receive command status
     - Ring status
     - SCB clear
     - Adapter check
5. Upon receiving a vector, the TRM deactivates IACK and sends a level 2 interrupt to the CCU.
   When it receives a get line ID from the CCU, it places the proper line ID on the IOC bus at TD time.
   The line ID is based on the line ID base, TIC number and received vector (See page 15-120 for details).
   - The TRM is bus master as in an MMIO read operation. The vector data is validated when the TIC activates data transfer acknowledge (DTACK).
6. The TIC then waits for a reset interrupt MMIO operation or a retry.
7. If the TRM detects an error during a TIC interrupt, it will log an error in the level 2 error status register and request a level 2 interrupt.
   When the error is reset by the error recovery procedures, the TIC interrupt acknowledge will be retried.

Origin			Operation	TA	TD
TIC	TRM	NCP			
X			DMA SSB 8 bytes (not adapter check or SCB clear interrupt)		
X			Interrupt request		
	X		Send IACK		
	X		Read interrupt vector 8 bits		
	X		Level 2 to NCP		
		X	Get line ID	X'3001'	
	X		Send line ID A or B to answer the get line ID		X'1111'
	X		Reset level 2 interrupt if get line ID is terminated normally		
		X	Reset TIC interrupt	X'nnnn'	X'A000'

llll is the line ID type A or B
nnnn is B'01001sss11rr0000' where:
sss is secondary group address (LAB address)
rr is TIC number

TRM Interrupt Operations (Part 3 of 4)

TRM INTERRUPT SCENARIO

- The TRM interrupts the CCU only in case of an error. When the TRM detects an error during an operation with a TIC, it logs the error into the TRM level 2 error status register associated with the TIC and activates a level 2 interrupt request to the CCU.
  Then it presents a type B line ID and the program reads the TRM error status register.

Origin			Operation	TA	TD
TIC	TRM	NCP			
	X		Load level 2 error status reg.		
	X		Level 2 interrupt request sent to NTRI		
		X	Get line ID	X'3001'	
	X		Send line ID B to answer the get line ID		X'bbbb'
	X		Reset level 2 int. if the get line ID is terminated normally		
		X	Get level 2 error status	X'pppp'	X'vvvv'

bbbb is the line ID type B
vvvv is level 2 error status value
pppp is B'01001sssskkkk0000' where:
sss is secondary group address (LAB address)
kkkk is the PIO code according to the TIC number

LINE ID LOADING

- Line ID loading is part of the TRM initialization process.
- With a PIO operation the program sets the line ID base which corresponds to the type A line ID for the first TIC.
- The line ID base is initialized to X'0000' after a TRM reset.

TIC ADAPTER CHECK REGISTER

The adapter check interrupt is generated when the TIC adapter encountered an uncoverable hardware or microcode error.
In this case the TIC adapter is waiting for a reset.
The reason for the TIC adapter check is located in 8 bytes beginning at address X'05E0'.
The TIC adapter check status is defined as follows:

X '05E0'
TIC adapter memory

Bytes 0-1	Adapter check
Bytes 2-3	Parameter 0
Bytes 4-5	Parameter 1
Bytes 6-7	Parameter 2

Each bit of the TIC adapter check gives the error type and the contents of the parameter 0 to 2 give additional information.
(See page 15-138 for more details)

TRM Interrupt Operations (Part 4 of 4)

TIC ADAPTER CHECK REGISTER DECODING (TWO BYTES)

Bit	Error Type	Error Description	Parm 0-2 Contents
0	MMIO parity error	Data parity error between CCU and TIC	Contents is ignored
1	DMA abort - read	DMA read operation abort for timeout or parity error or bus error	Parm 0 = 0000 for timeout 0001 for parity error 0002 for bus error Parm 1-2 contains the failing CCU address plus or minus 6 bytes
2	DMA abort - write	DMA write operation abort	Same as for DMA abort-read
3	Illegal OP code	Illegal OP code detected	Parm 0 = TIC reg. 13 Parm 1 = TIC reg. 14 Parm 2 = TIC reg. 15
4	Parity error	Local bus parity error detected by TIC processor	Same as for illegal OP code
5	Parity error - Ext	Local bus parity error detected by TIC during operation with CCU	Same as for illegal OP code
6	Parity error - IOC bus interf.	Local bus parity error detected by TIC during operation with CCU	Same as for illegal OP code
7	Parity error - PH	Local bus parity error detected by TIC during operation with the protocol handler	Same as for illegal OP code
8	Parity error - XMIT	Local bus parity error detected by TIC during ring receive operation	Parm 0 = buffer address
9	Parity error - RECV	Local bus parity error detected by TIC during ring transmit operation	Parm 0 = buffer address
10	Ring underrun	DMA underrun detected during ring transmit	Parm 0-2 are ignored
11	Ring overrun	DMA overrun detected during ring receive operation	Parm 0-2 are ignored
12	Invalid interrupt	Unrecognized error interrupt was generated	Parm 0 = TIC reg. 13 Parm 1 = TIC reg. 14 Parm 2 = TIC reg. 15
13	Invalid error interrupt	Unrecognized error interrupt was generated	Parm 0 = TIC reg. 13 Parm 1 = TIC reg. 14 Parm 2 = TIC reg. 15
14	Invalid XOP	Unrecognized XOP request was generated	Parm 0 = TIC reg. 13 Parm 1 = TIC reg. 14 Parm 2 = TIC reg. 15
15	Program check	TIC processor program check detected	Parm 0 = Abend code Parm 1 = Address location that detected the error

INTERRUPT TO MOSS

General Operation

- The TRM sends an interrupt to MOSS instead of a level 2 interrupt to the CCU when a TIC is under MOSS control, when the TRM is disconnected from NCP or when an error is detected during an MIOH.

Scenario for Interrupt to MOSS

Origin			Operation	TA	TD
TRM	RDV	MOSS			
X			Raise interrupt to MOSS		
		X	Get command completion	X'ccOD'	X'yyyy'
X			Clear interrupt to MOSS		

cc TRM address B'01001sss'
sss is the sub-group address (board address)
yyyy is the command completion value in the TRM

TRSS Disconnect/Connect Function

These functions are initiated by the MOSS.

A level 1 interrupt to the CCU is raised by the TRM after it receives the stop PIO command.

DISCONNECT FUNCTION

The disconnect function disables the TRM for all PIO operations from NCP. PIOs from MOSS will still be accepted.

This function also prevents the TRM from sending any level 2 interrupt to NCP. Errors and TIC interrupts will be reported to MOSS when the TRM is in the disconnect state.

As soon as the CCU PIO disable latch is set, an IOC timeout will occur in case of an IOH sent to the TRM from the NCP.

CONNECT FUNCTION

The connect function allows the reconnection of the TRA to NCP. The MOSS also sends a mailbox to the control program.

The origin of a PIO is determined to be from MOSS if IOC data bus bit 12 is ON during TA time.

Origin			Operation	TA	TD
MOSS	TRM	NCP			
X			Disconnect (TRSS stop)	X'cc0C'	X'0B00'
	X		Set latch disconnect		
	X		Level 1 to NCP		
		X	Get level 1 error status reg	X'ccD5'	
	X		Set latch CCU PIO disable		
	X		Send interrupt MOSS		
X			Get command completion	X'cc0D'	X'zzzz'
X			Connect (TRSS start)	X'cc0C'	X'0C00'
	X		Reset latch disconnect		
X			Send mailbox to NCP		

cc TRM address B'01001sss'
sss is the sub-group address (board address)
zzzz is the command completion value in the TRM

TRA Reset Operations

The purpose of this topic is to describe all the mechanisms which allow the TRA or the subset of the TRA to be put in a defined state.

There are three functions which are implemented to reset the TRA or a part of the TRA:

The hardware reset function

The programmed reset function

The TIC reset function.

HARDWARE RESET FUNCTION

This function is activated by the reset tag on the IOC bus.

It completely resets the TRA:

- All requests on the IOC bus are deactivated.
- The TRM reset bit is forced ON in the TRM control register (which indicates a reset has been done).
- The TIC reset bits are forced ON (and the reset tags on the TIC bus) in the TIC control register.
- All the interrupts and their associated statuses are reset.
- The disconnect bit and the CCU PIO disable bit are reset.
- The entire control logic is forced to idle state.
- More generally, all other functions and all other registers are forced in their inactive state.

PROGRAMMED RESET FUNCTION

This function is initiated by a PIO (IOH or MIOH). It allows the program to reset the TRM without resetting the attached TIC and the connect/disconnect state of the TRM.

The result is the same as the hardware reset except the TIC reset bits, the disconnect mode latch and the CCU PIO disabled latch are left in their current state.

RESET TIC FUNCTION

This function is activated by writing to the TIC control register with the bit reset TIC ON.

The reset bits are connected to the TIC reset tags on the TIC bus.

The reset tag is active as long as the reset bit is on in the TIC control register.

The TIC reset tag causes the TIC to disconnect itself from the ring and perform a reset.

TRM Error Detection and Reporting (Part 1 of 2)

Parity Checker to the IOC Bus (1)

This parity checker is provided to check the validity of the data to/from the IOC bus and from/to the TRM.
If a parity error is detected, the TRM can report the error to the CCU in two different fashions:

Parity error detected by the adapter

The first way is that the TRM will not answer valid halfword (VH) to the CCU at TD time and cause a time out on the IOC bus.

Parity error detected by the IOC

The TRM will answer VH and letting the IOC bus detect the parity error with its bus parity checker.

For either way, once the TRM receives halt from the IOC, the error is logged in the level 1 error status register. The IOC then alerts the CCU by a level 1 interrupt.

Parity Checker to the TIC Bus (2)

A parity error caused by the data or addresses transferred between the TRM and the TIC is detected by this checker.
An error is reported by logging it into the level 2 error status register and raising a level 2 interrupt to the CCU.
For a DMA operation, the TRM will send bus error (BERR) to the TIC so that the TIC will retry the DMA operation.

Internal Parity Checker (3)

The internal parity checker validates the data or address in the TRM internal buffer registers.
Errors detected by this checker will be logged in the level 2 error status register.
The TRM then raises a level 2 interrupt to the CCU.
The TRM will signal bus error to the TIC in case of an error during a DMA operation.

DTACK Checker (Timer) (4)

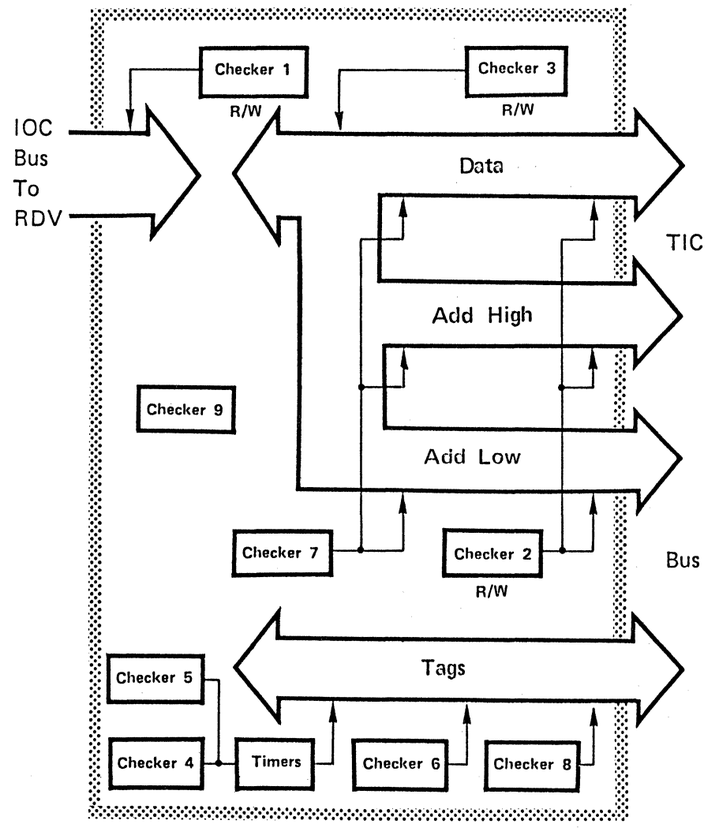
This checker is used to detect a time out error during an MMIO operation in case the TIC does not answer data acknowledge (DTACK).

If a time-out occurs, the TRM will release all the control tags to the TIC to terminate the operation.
An error will be logged in the level 2 error status register and a level 2 interrupt is requested.

AS/DS Checker (Timer) (5)

This timer is provided to detect a time out error during a DMA operation.

TRM Hardware Checkers Placement



TRM hardware checkers placement

Idle Checker of the TIC Bus Tags (6)

This checker checks that all the TIC interface tags are inactive during the idle state.
An error is reported by logging it into the level 2 error status register and raising a level 2 interrupt to the CCU.

Idle Checker of the TIC Address/Data Bus (7)

This checker checks the TIC interface address/data bus during the idle state.
Any existing error is reported by logging it in the level 2 error status register and alerting the CCU with a level 2 interrupt.

Data Strobe Checker (8)

The data strobe checker is used in DMA operation to check the consistency between the starting address of the CCU memory (even or odd) and the activation of the data strobes (UDS/LDS).

If any data strobe error is detected, it is logged in the level 2 error status register and a level 2 interrupt is sent to the CCU.

Invalid PIO Checker (9)

Any invalid PIO command is detected by this checker.
The error is logged in the level 1 error status register after receiving halt from the IOC bus.

TRM Error Detection and Reporting (Part 2 of 2)

No.	Function	Time	Checker	TRM Action	TIC	IOC Action
1.	PIO/TRM write	TA	Checker 1 or checker 9	.No VH raise .Wait for halt .Log error in level 1 error register		.IOC timeout .Level 1 int. .Halt .Get level 1 error status
2.	PIO/TRM write	TD	Checker 1	.Same as case no.1		
3.	PIO/TRM read	TA	Checker 1 or checker 9	.Same as case no.1		
4.	PIO/TRM read	TD	Checker 1	.VH Raise; .Wait for halt .Log error in level 1 error register		.Find bad parity .Halt .Get level 1 error status
5.	MMIO Wr. MMIO Rd.	TA	Checker 1 or checker 9	.Same as case no.1		
6.	MMIO Wr.	TD	Checker 1	.Same as case no.1		
7.	MMIO Rd.	TD	Checker 1	.VH raise .Wait for halt .Log error in level 1 error register		
8.	MMIO Wr.		Checker 2 or checker 3 or checker 6 or checker 7	.Log error in Level 2 error register		.Level 2 interrupt
9.	MMIO Wr.		Checker 4	.Log error in level 2 error register .Drop CS .Drop AS/DS	.Stop Oper.	.Level 2 interrupt
10.	MMIO Rd.	TD	Checker 2 or checker 3 or checker 6 or checker 7	.Log error in level 2 error register .Send valid pattern		.Level 2 interrupt
11.	MMIO Rd.	TD	Checker 4	.Log error in level 2 error register .Send valid pattern .Drop CS .Drop AS/DS	.Stop Oper.	.Level 2 interrupt

No.	Function	Time	Checker	TRM Action	TIC	IOC Action
12.	DMA Rd. DMA Wr.	Building of CSCW in TRM	Checker 2 or checker 3 or checker 5 or checker 6 or checker 7	.Parity error in ADDH, ADDL xferred from TIC .CSCW short/dir. .Raise VH .Send valid pattern w/EOC .BERR to TIC .Log err.in level 2 error register	.Stop Oper. .Retry whole oper.	.Level 2 interrupt .Error on TIC bus .No error detected IOC
13.	DMA Rd. DMA Wr.	CSCW UC gate	Checker 1	.Raise VH .Wait for halt .Log error in level 1 error register .BERR to TIC	.Stop Oper. .Retry whole oper.	.IOC detected parity error .Halt .Get level 1 error status
14.	DMA Rd.	Data	Checker 1	.Same as Case no.1 .BERR to TIC		
15.	DMA Wr.	Data	Checker 1	.Same as Case no.4 .BERR to TIC		
16.	DMA Wr.	Data UC gate	Checker 2 or checker 3 or checker 5 or checker 8	.Detect parity error or T.O. .Send valid pattern .Raise VB + M .BERR to TIC	.Stop Oper.	.Level 2 interrupt
17.	DMA Rd.	Sata UC gate	Checker 2 or checker 3 or Checker 5 or checker 8	.Detect parity error or T.O. .Raise VB + M .BERR to TIC	.Stop Oper. .Retry	.Level 2 interrupt
18.	Inter-rupt	IACK	Checker 2 or checker 3 or checker 4 or checker 6 or checker 7	.Log error in level 2 error register .Send line ID B		.Level 2 interrupt
19.	Inter-rupt	GLID	Checker 1	.No answer Get line ID .Wait for halt .Log error in level 1 error reg. (same as case 4) .Level 2 not reset		.IOC timeout .Halt .Get level 1 error status

TRA Interaction with Control Program (Part 1 of 2)

TIC INITIALIZATION

The NCP Token-Ring Interconnection (NTRI) program initializes a TIC as follows:

1. The NTRI resets the TIC by activating the reset bit in the TIC control register and then deactivating the reset bit.
2. NTRI issues read interrupt register MMIO operations until the initialize bit is a 1 and the error and test bits are 0.
3. NTRI writes X'0200' into the TIC address register using an MMIO operation.
4. NTRI loads 22 bytes of initialization parameters using MMIO write operations to the TIC data with autoincrement register.
5. NTRI writes X'9080' to the TIC interrupt register.
6. NTRI issues read interrupt register MMIO operations until the initialize, error and test bits are all 0. The initialization is then completed.

In order to begin normal operation, NTRI initiates communication by placing an open command in the system control block (SCB) and specifying the required parameters. When the open command status is set into the system status block (SSB) and the command status interrupt is received, NTRI issues a receive command so that the TIC can receive frames.

TIC Read Interrupt Register (Initialize)

Bit Name	Bit Description
Bit 0 — Adapter interrupt	Bit ignored
Bit 1–7 — Interrupt request	Bit ignored
Bit 8 — Interrupt system	Always "0"
Bit 9 — Initialize	Bit 9 = 1 Bring up successful and TIC adapter ready for initialization. Bit 9 = 0 Initialization completed with or without error.
Bit 10 — Test	Bit 10 = 1 Bring up diagnostics following hardware reset. Bit 10 = 0 When initialize is set to 1.
Bit 11 — Error	Bit 11 = 1 Error during bring up diagnostics or initialization sequence. Bit 12–15 specify the error condition.
Bit 12–15 — Error code (bring–up or initialization)	Contain bring up error code if test = 1. Contain bring up initialization error code if test = 0.

Bits 12-15 - Bring-Up Error Code (Test = 1)

Bit 12–15	Error Code Description
0000	Initial test error
0001	ROS CRC error
0010	RAM error
0011	Instruction test error
0100	XOP test error, interrupt test error
0101	PH hardware error
0110	IOC interface control register error

Bits 12-15 - Initialization Error Code (Test = 0)

Initialization error codes are as follows:

Bit 12–15	Error Code Type	Description
0001	Invalid parameter length	This code will be set if 22 bytes are not passed.
0010	Invalid options	
0011	Invalid receive burst size	The receive burst size is odd.
0100	Invalid transmit burst size	The transmit burst size is odd.
0101	Invalid DMA abort thresholds	The bus error or parity error count is zero.
0110	Invalid SCB address	The SCB address is odd.
0111	Invalid SSB address	The SSB address is odd.
1000	MMIO parity error	TIC adapter detects a parity error during a system MMIO write operation
1001	DMA timeout	TIC adapter times out (10 seconds) waiting for a test DMA transfer to complete.
1010	DMA parity error	TIC adapter detects bad parity from the CCU during one of the test DMA transfers.
1011	DMA bus error	The CCU asserts bus error during one of the test DMA transfers.
1100	DMA data error	The initialize DMA test fails due to a data compare error.
1101	Adapter check	The TIC adapter has encountered an unrecoverable hardware error (for details see page 15–171).

TRA Interaction with Control Program (Part 2 of 2)

COMMANDS TO TIC

The following commands are initiated in the TIC adapter by loading the system control block and interrupting the TIC adapter to cycle steal the command from the SCB. A maximum of 3 commands (receive, transmit and one other command) can be executed at the same time.

- Open command

  Used to begin communications with the TIC. The TIC is enabled to receive frames. This command also inserts the TIC into the ring.

- Transmit command

  Used to transmit frames from the CCU to other nodes.

- Transmit halt

  Used to interrupt a transmit operation.

- Receive command

  Used to receive frames from other nodes. This command is normally issued only once after the open command is issued.

- Close command

  Used to terminate communication on the ring. It removes the TIC from the ring.

- Set group address

  Used to change the TIC adapter group address after the open command has been issued.

- Set functional address

  Used to set or reset the TIC adapter functional address after the open command has been issued.

- Read error log

  Used to read and reset the TIC adapter error log

- Read adapter

  Used to transfer TIC adapter storage to the CCU.

Chapter 15 Token-Ring Subsystem

Section 2. Troubleshooting Guidelines

DC Voltages and Tolerances at Board Pin Level

Vdc	Vmin	Vmax	Ripple (max)
−12.0	−10.92	−13.20	0.45V p−p
−8.5	−7.73	−9.35	0.25V p−p
−5.0	−4.55	−5.50	0.15V p−p
−4.3	−4.19	−4.48	0.07V p−p
−1.5	−1.48	−1.56	0.03V p−p
+5.0 (Note 1) (Note 3)	+4.55	+5.50	0.20V p−p
+5.0	+4.75	+5.25	0.13V p−p
+8.5	+7.73	+9.35	0.35V p−p
+12.0	+10.92	+13.20	0.40V p−p
+12.0 (Note 2)	+11.40	+13.20	0.40V p−p
+24.0	+21.00	+27.60	0.30V p−p

Notes:

1. 02-PS7 only (3726)
2. 01-PS4 only (3725)
3. The TRA uses only this "+ 5.0 volts" voltage

RING INTERFACE ELECTRICAL CHARACTERISTICS

Ring Receiver

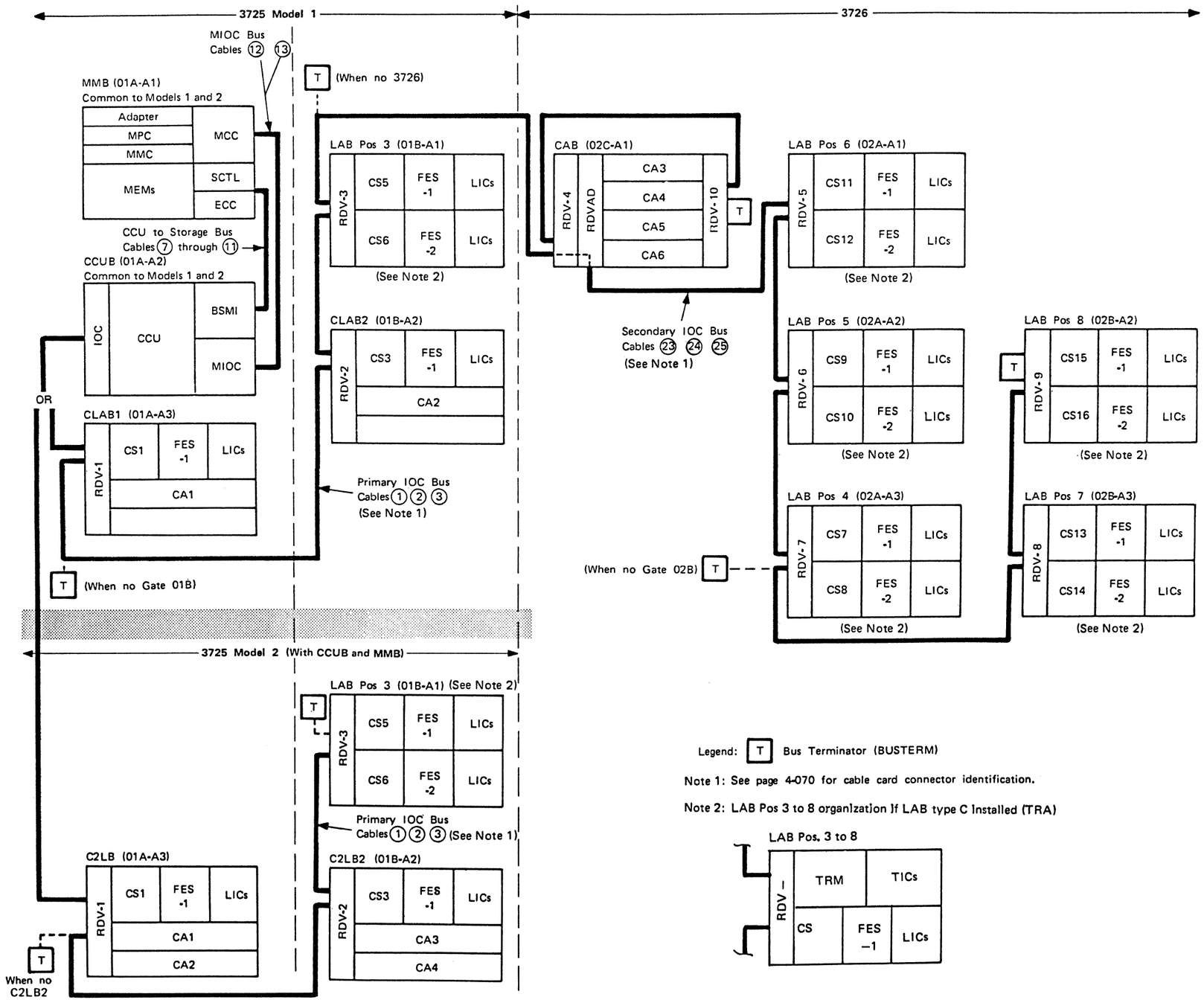
Minimum input signal: 50 millivolts peak-to-peak.

Ring Transmitter

Output signal: between 3.0 and 4.5 Volts peak-to-peak.
Output load: 150 Ohms ±10%.

TRSS Extended Troubleshooting Techniques

- For TRA troubleshooting, see page 15-800.
- For interrupt request to CCU, see page 15-810.
- For cycle steal request to CCU, see page 15-820.



Legend: T Bus Terminator (BUSTERM)

Note 1: See page 4-070 for cable card connector identification.

Note 2: LAB Pos 3 to 8 organization if LAB type C installed (TRA)

Token-Ring Subsystem Troubleshooting

TOKEN-RING ADAPTER ISOLATION (3725/3726)

By depopulating the IOC bus (see page 11-803), it is possible to change the running environment of the diagnostics. A redrive or its attached adapters, suspected of causing a failure, can be disconnected by unplugging the top connectors of that redrive.

To isolate a TRA, it is necessary to unplug the scanner on the same LAB. Any or all of the TICs may be unplugged from the TRA. Similarly, a scanner can be isolated on a LAB C by unplugging the TRA.

Note: Any configuration changes require the updating of the CDF. Sections 'F' to 'H' of the TRSS diagnostics cannot be run without at least one TIC installed.

TRSS CLOCKING

The TRMs free running latch clocks are generated by the RDV. They are called CLK1, CLK2 and CLK3 as shown on pages 5-053 and 5-054 (same as those used by FES).
The TRM generates the 7.35 MHz "System B clock" for the TICs. It is only an interface clock for the interface between the TRM and the TIC. It is on the TRM card at pin J04.

TRSS DIAGNOSTICS ORGANIZATION

Sections A-E of the TRSS diagnostics test only the TRM card of the TRA. These sections will run regardless of the TIC configuration. Sections F, G, and H test the TICs and the interface between the TICs and the TRM. These last three sections cannot be run unless at least one TIC is installed.

Routines TF01 and TG01 invoke the TIC internal tests and verify all types of communication (interrupt, cycle steal, MMIO) between the CCU and the TIC. Routine TH01 tests the management of errors on the TIC bus. These routines are designed to run on each installed TIC, then correlate and display error information. Isolation of errors is therefore greatest when no specific TIC is requested for the diagnostic run. An error reference code (ERC) is given in the ADDITIONAL INFO field for each TIC tested.

When errors are detected for more than one TIC and the ERCs given are different, it may be helpful to invoke sections TF to TH for each TIC independently using the LINE==> field in the diagnostic request menu. The request will be run using only the TIC selected and the remaining TICs will be kept in their reset state. In this manner, some failures which cause all TICs to be suspected may be isolated to the TRM and one TIC.

Interrupt Requests to CCU (Part 1 of 2) [with TRA (LABC) Installed on any LAB pos 3 to 8]

PRESELECTION MECHANISM

The TRA supports the preselection mechanism used by the scanners for level 2 interrupts and cycle steal requests to the CCU. Each TRA may be given high or low priority; this priority is common to both the level 2 interrupt requests and cycle stealing. The priority level is assigned by the Set TRM control register PIO operation; bit 1 on indicates high priority, off indicates low priority.

INTERRUPT PRESELECTION

When an adapter requires service from the CCU for a specific line, it issues a level 2 interrupt request to the CCU via the IOC bus. More than one adapter may raise an interrupt request at the same time.

The level 2 preselection mechanism continuously determines which scanner(s)/TRM(s) has raised the highest priority level 2 interrupt request (according to the predetermined priority level set by the microcode). The scanner(s)/TRM(s) with the highest priority is said to be 'preselected'.

INTERRUPT AUTOSELECTION

To service the level 2 interrupt request, the control program issues a 'get line identification' PIO read instruction. This instruction is decoded by all scanners/TRAs, starting the autoselection operation. When the instruction is issued, the preselection mechanism stops; no further level 2 requests are taken into account, regardless of their priority level. Only the preselected scanner/TRM answers the 'get line ID' command. If more than one adapter has interrupt requests of the same priority pending, the adapter/TRA that responds depends only on the position reached by the preselection mechanism at that moment in time.

INTERRUPT PRIORITY STRUCTURE

Priority determination is done in three different places:

- Scanner/TRA level

  When two adapters are plugged into the same LABB or LABC, the first adapter (even address) responds first if it has a priority higher than or equal to the priority of the second adapter (odd address).

  The mechanism works as follows (refer to LAB-3 on the diagram): if the first adapter on the board has both a level 2 interrupt request <u>and</u> the priority bit (CSP XR05, bit 5 or TRM control register, bit 2) set, they are ANDed together and raise the '-level 2 priority' line to the RDV card. The '-level 2 select out secondary' signal coming from the redrive card is not propagated to the second adapter.

- Board level

  In the 3725, the boards reply in the following order: CLAB-1, CLAB-2, LAB-3, CAB (frame redrive), or C2LB, C2LB2, and LAB-3 for the 3725 model 2.

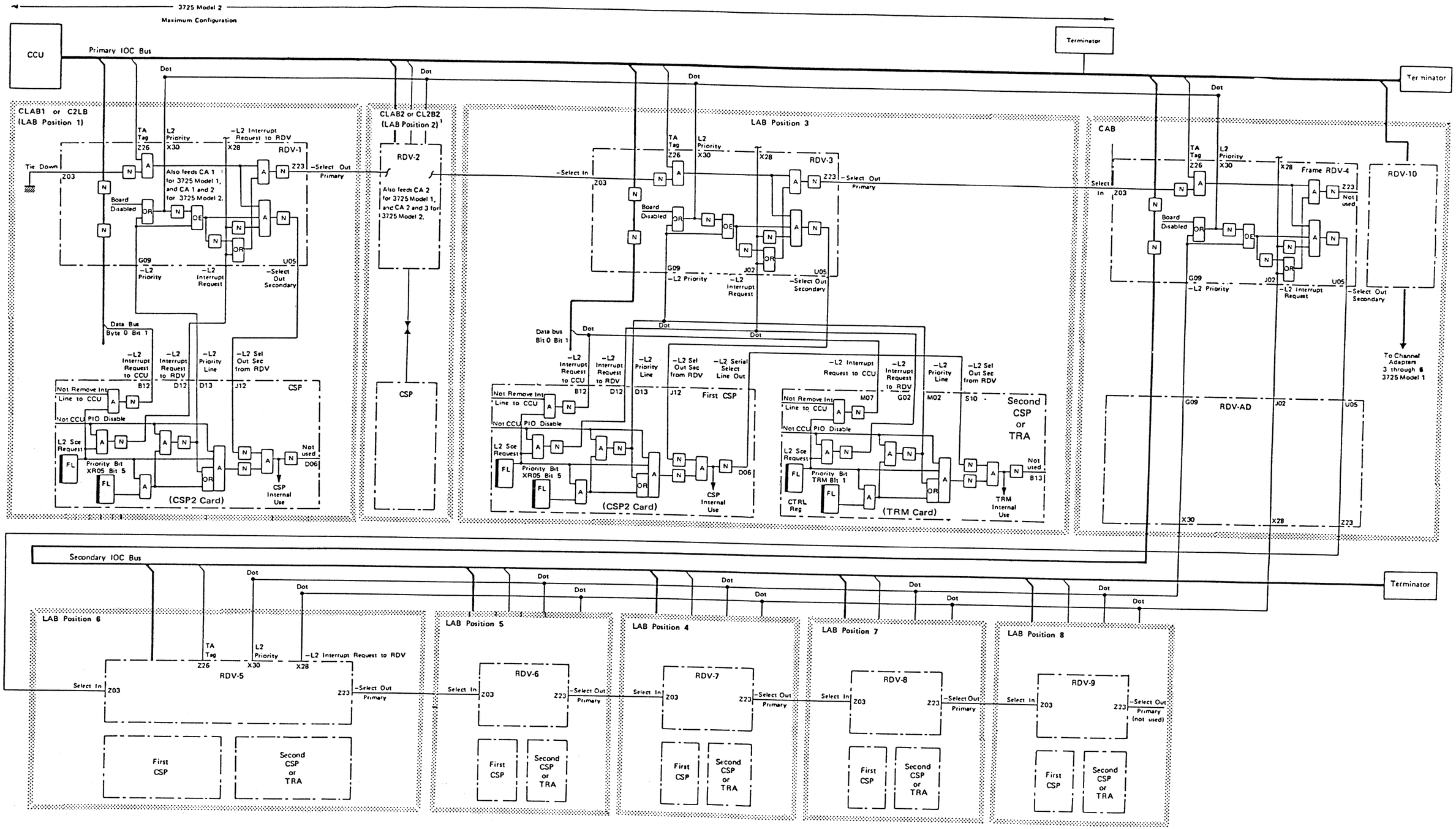
  In the 3726, if none of the boards in the 3725 have replied, the boards reply in the following order: LAB-6, LAB-5, LAB-4, LAB-7, LAB-8.

  The mechanism works as follows (refer to CLAB-1 and LAB-3 on the diagram): suppose that one of the adapters of LAB-3 has both a level 2 interrupt request and the priority bit on, but that the single CSP of CLAB-1 has only a level 2 interrupt request. The '-level 2 priority' line from the CSPs of LAB-3 passes through the OR of the RDV-3 card to raise the 'level 2 priority' line at pin G09 of the redrive card. This pin is DOT-ORed with the other redrive cards, and causes the CLAB-1 redrive card to pass on the '-Select out primary' signal to LAB-3 (via CLAB-2) <u>despite the fact that CLAB-1 itself has a low-priority interrupt pending.</u>

- Frame level

  The frame redrive, which acts as a relay to the secondary IOC bus, can only pass on the selection signal if none of the boards in the 3725 have replied.

Interrupt Requests to CCU (Part 2 of 2)
[with TRA (LABC) Installed on any LAB pos 3 to 8]



Cycle Steal Requests to CCU (Part 1 of 2)

CYCLE STEAL PRESELECTION

When a scanner or TRA requires service from the CCU for a specific line, it issues a CS request to the CCU via the IOC bus. More than one scanner or TRA may raise a CS request at the same time.

The CS preselection mechanism continuously determines which scanner(s) or TRA(s) has raised the highest priority CS request (according to the predetermined priority level set by the microcode). The scanner(s) or TRA(s) with the highest priority is said to be 'preselected'.

CYCLE STEAL AUTOSELECTION

To service the CS request, the CCU issues a 'cycle steal grant'. This line is trapped in the preselected scanner or TRA, and the CS operation starts.

CYCLE STEAL PRIORITY STRUCTURE

Priority determination is done in three different places:

- Scanner or TRA level.

  When two adapters are plugged into the same LABB or LABC, the first adapter (even address) responds first if it has a priority higher than or equal to the priority of the second adapter (odd address).

  The mechanism works as follows (refer to LAB-3 on the diagram): if the first adapter on the board has both the CS request and the priority bit (XR05, bit 5 or TRM Control register bit 1) set, they are ANDed and raise the '-CS request' line to the RDV card.

- Board level

  In the 3725, the boards reply in the following order: CLAB-1, CLAB-2, LAB-3, CAB (frame redrive), or C2LB, C2LB2, and LAB-3 for the 3725 model 2.

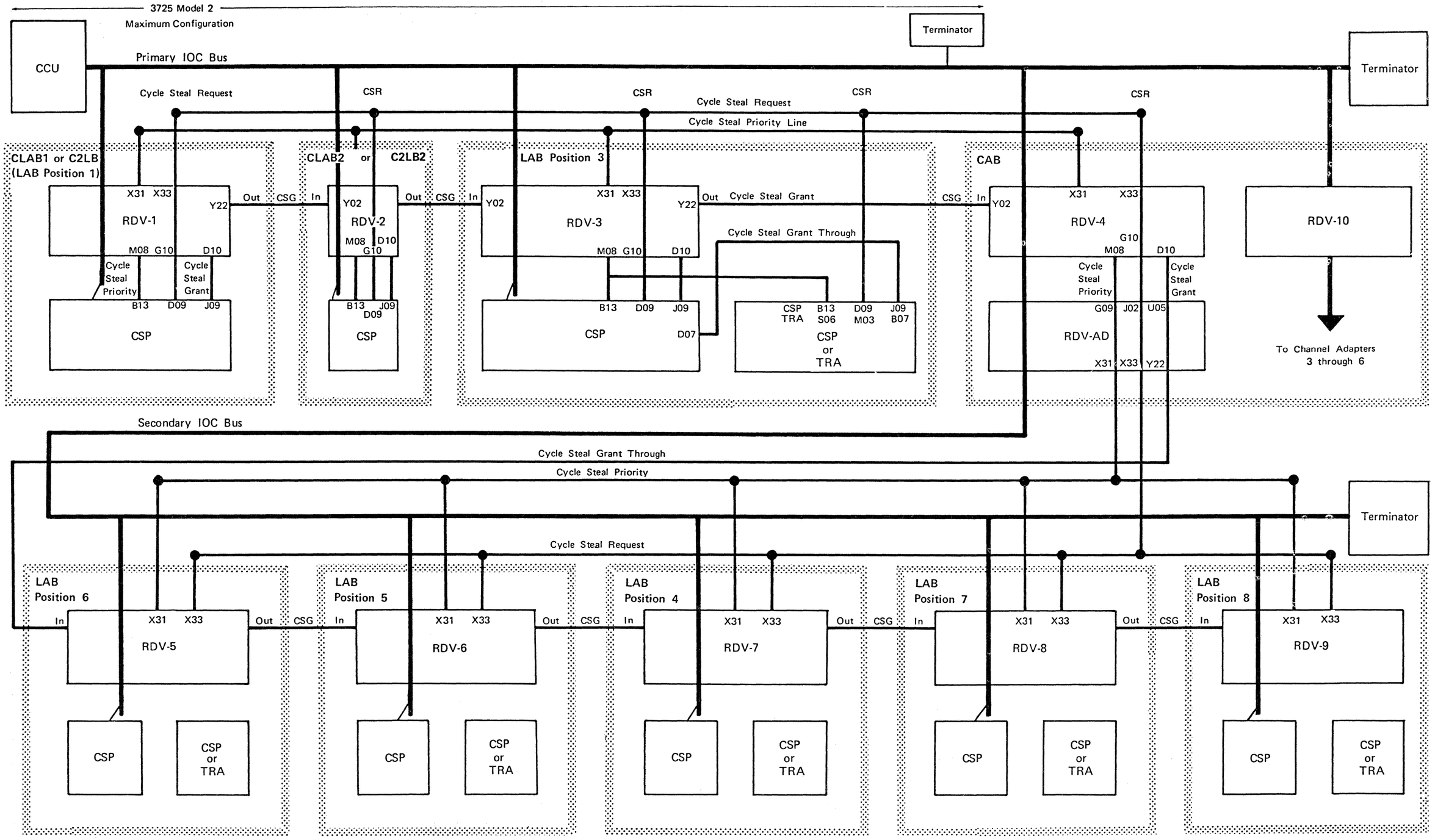
  In the 3726, if none of the boards in the 3725 have replied, the boards reply in the following order: LAB-6, LAB-5, LAB-4, LAB-7, LAB-8.

  The mechanism works as follows (refer to CLAB-1 and LAB-3 on the diagram): suppose that one of the adapters of LAB-3 has both a CS request and the priority bit on, but that the single CSP of CLAB-1 has only a CS request. The '-CS request' line from the CSPs of LAB-3 passes through the OR of the RDV-3 card to raise the CS request line at pin M08 of the RDV card. This pin is DOT-ORed with the other redrive cards, and causes the CLAB-1 redrive card to pass on the '-CS grant secondary' signal to LAB-3 (via CLAB-2) despite the fact that CLAB-1 itself has a low-priority CS request pending.

- Frame level

  The frame redrive, which acts as a relay to the secondary IOC bus, can only pass on the selection signal if none of the boards in the 3725 have replied.

Cycle Steal Requests to CCU (Part 2 of 2)



Abbreviations and Glossary

Abbreviations and Glossary

A	ampere
ABEND	abnormal end of task
ac, AC	(1) alternating current (2) abandon call (signal) (3) address compare
ACB	adapter control block
ACF	Advanced Communication Function
ACK	affirmative acknowledgment (BSC)
ACLK	adapter clock (card)
ACR	add character register (instruction)
ACU	automatic calling unit
AE	address exception
AEK	address exception key
AGC	automatic gain control (signal)
AHR	add halfword register (instruction)
AIO	adapter-initiated operation
AIT	average instruction time
ALU	arithmetic and logic unit
AR	(1) add register (instruction) (2) amplifier
ARC1	auto-restart card in unit 01
ARC2	auto-restart card in unit 02
ARI	add register immediate (instruction)
ASCII	American National Standard Code for Information Interchange
ATTN	attention (3727 operator console key)
B	branch (instruction)
BAL	branch and link (instruction)
BALR	branch and link register (instruction)
BB	branch on bit (instruction)
BCC	block check character (BSC)
BCCW	bit clock control word
BCL	branch on C latch (instruction)
BCLE	
BCT	branch on count (instruction)
BER	box error record
BG	background
B/M	bill of material
bps	bits per second
BSC	binary synchronous communication
BSM	basic storage module
BSMI	basic storage module interconnection (card)
BT	branch trace
BTAC	branch trace/address compare (card)
BTAM	basic telecommunication access method
BUSTERM	IOC bus terminator (card)
BZL	branch on Z latch (instruction)
C	(1) Celsius (2) control (X.21 signal)
CA	channel adapter
CAB	channel adapter board
CAC	common adapter code
CADR	channel adapter driver receiver (card)
CADRUK	channel adapter driver receiver type UK (card)
CB	circuit breaker
CCA	communication common adapter (card)
CCIN	channel-to-CCU interface (card)
CCITT	Comite Consultatif International Telegraphe et Telephone
CCLK	CCU clock (card)
CCMD	current command (storage)
CCN	communications controller node
CCR	compare character register (instruction)
CCU	central control unit
CCUB	CCU board
CCW	channel control word
CDF	configuration data file
CDS	configuration data set (NCP/EP)
CE	(1) customer engineer (WTC term for FE) (2) channel end (channel status)
CELIA	CE latched indicator analytic (card)
CHCW	channel control word
CHIN	channel interface (card)
CHR	compare halfword register (instruction)
CLAB	channel and line attachment board
CLDP	controller load/dump program
CNM	communication network management
CNMI	communication network management interface
CNSL	console
CP	command processor, control program
CPA	control panel adapter (card)
CPIT	control progam information table
CPM	connection point manager
CPT	checkpoint trace
CR	(1) compare register (instruction) (2) call request (signal)
CRC	cyclic redundancy check
CRI	compare register immediate (instruction)
CRP	check record pool
CS	(1) cycle steal (2) communication scanner
CSCW	cycle steal control word
CSG	cycle steal grant
CSGH	cycle steal grant high
CSGL	cycle steal grant low
CSM	communication scanner memory (card)
CSP	communication scanner processor
CSP1	communication scanner processor (card) type 1
CSP2	communication scanner processor (card) type 2
CSR	cycle steal request
CSRH	cycle steal request high
CSRL	cycle steal request low
CSS	control subsystem
CSW	channel status word
CTS	clear to send (signal)
CTL1	control type 1 (card)
CTL2	control type 2 (card)
CVTL	card vendor transistor logic
CZ	Carry/zero (latch)
C2LB	CLAB in 3725 Model 2
DAC	diskette adapter card
DAF	destination address field (SNA)
DB	data byte (signal)
dc, DC	(1) direct current (2) data chaining (channel status)
DCE	data circuit-terminating equipment
DCF	diagnostic control facility
DCM	diagnostic control monitor
DE	device end (channel status)
DFLx	data flow type x (card) (where x = 1, 4, or 5)
DIFF	differentiator
DLO	data line occupied (signal)
DMA	direct memory access
DP	digit present (signal)
DSx	digit signal 2 to power x
DSC	distant station connected
DSR	data set ready (signal)
DSRS	data signaling rate selection (signal)
DTE	data terminal equipment
DTR	data terminal ready (signal)
DVB	asynchronous-devices (SNA)
DX	duplex (full-duplex)
EBCDIC	extended binary-coded decimal interchange code
EC	engineering change
ECC	error checking and correction (card)
EDE	elementary data exchange
EIA	Electronic Industries Association
ENQ	enquiry (BSC)
EOT	end of transmission (BSC)
EP	emulation program
EPO	emergency power off
ERC	error reference code
EREP	environmental recording, editing, and printing (program)
ERP	error recovery procedure
ESC	emulation subchannel
ESCH	emulation subchannel high
ESCL	emulation subchannel low
ESD	(1) electrostatic discharge (2) external symbol dictionary
ETB	end-of-transmission block character (BSC)
ETX	end-of-text character (BSC)
EXP	expected
FAC	flag address control
FCC	Federal Communications Commission
FCPS	final call progress signals (X.21)
FES	front-end scanner (card)
FIC	FRU isolation code
FM	frequency modulation
FNCTN	function (CCU FNCTN) (3727 operator console key)
FPS	FES parameter/status
FRU	field-replaceable unit
ft	foot
GCF	graphic configuration file
GPR	general purpose register
GPT	generalized PIU trace
GTF	generalized trace facility
HDX	half-duplex
HDR1	header 1 (diskette)
HLIR	high level interface request
HSC	high speed channel
HW	hardware
Hz	Hertz
IAR	instruction address register
IC	insert character (instruction)
ICA	integrated communication adapter
ICB	interface control block (storage)
ICC	internal clock control
ICT	insert character and count (instruction)
ICW	interface control word
ID	identifier (diskette)
IFT	internal function test
IMB	in mailbox (MOSS)
IML	initial microcode load(er)
in.	inch
IN	input (instruction)
INN	intermediate network node
INOP	inoperative (line, modem, or terminal)
IOC	input/output control
IOCB	input/output control bus
IOCS	input/output control system
IOH	adapter input/output halfword (instruction)
IOHI	adapter input/output halfword immediate (instruction)
IOIRR	input/output interrupt request register
IML	initial microcode load
INN	intermediate network node
IPF	instruction pre-fetch
IPL	initial program load(er)
IPR	isolated pacing response (SNA)
IRR	interrupt request removed
ITB	intermediate text block (BSC)
IVT	isolation verification tests
K	1024 (bytes or words)
KBD	keyboard
kbps	kilobits per second
kg	kilogram
kHz	kilohertz
ko	not ok
L	load (instruction)
LA	load address (instruction)
LAB	line attachment board
LABA	line attachment board type A
LABB	line attachment board type B
LABC	line attachment board type C
LAN	local area network
LAR	lagging address register
LCB	line control block (storage)
LCD	line control definer (storage)
LCOR	load character with offset register (instruction)
LCR	load character register (instruction)
LCS	line communication status (storage)
LDF	line description file
LED	light-emitting diode
LH	load halfword (instruction)
LHOR	load halfword with offset register (instruction)
LHR	load halfword register (instruction)
LIB	line interface buffer
LIC	line interface coupler
LIC1	line interface coupler type 1 (card)
LIC2	line interface coupler type 2 (card)
LIC3	line interface coupler type 3 (card)
LIC4A	line interface coupler type 4A (card)
LIC4B	line interface coupler type 4B (card)
LID	line identification
LLIR	low level interrupt request
LL2	link level 2 test
LNVT	line vector table (storage)
LOGREC	error logging program of access method
LOR	load with offset register (instruction)
LPDA	link problem determination aid
LR	load register (instruction)
LRC	longitudinal redundancy check
LRI	load register immediate (instruction) local storage

LSAR	local storage address register
LSI	large scale integration
LSR	local storage register (CSP)
LSSD	level sensitive scan design
m	meter
MAC	medium access control
MAP	maintenance analysis-procedure
MCPC	machine check/program check
MCC	MOSS control card
MCT	machine configuration table
MDOR	MOSS data operand register
MDR	miscellaneous data recorder
MEM	memory (card)
MES	miscellaneous equipment specifications
MFM	modified frequency modulation
MHz	megahertz
MICB	MOSS interface control block
MIM	maintenance information manual
MIO	MOSS input/output
MIOC	MOSS input/output control (card)
MIOH	MOSS input/output halfword
MIOHI	MOSS input/output halfword immediate
MLC	machine level control
MLT	machine load table (diskette)
mm	millimeter
MMB	memory and MOSS board
MMC	MOSS memory card
MMM8	MOSS memory module 8K
MMM24	MOSS memory module 24K
MMM32	MOSS memory module 32K
mn	minute
MOD	modifier
MOSS	maintenance and operator subsystem
MPC	MOSS processor card
ms	millisecond
MSA	machine status area (console)
MSD	machine status display
mV	millivolt
NAK	negative acknowledgment (BSC)
NCCF	network communication control facility (CNM)
NCP	network control program
NCR	AND character register (instruction)
NEO	network expansion option
NHR	AND halfword register (instruction)
NLDM	network logical data management
NOSP	network operation support program (VTAM)
NPDA	network problem determination application (CNM)
NR	AND register (instruction)
NRI	AND register immediate (instruction)
NRZI	non return-to-zero inverted
ns	nanosecond
NSC	native subchannel
NTO	network terminal option
NTRI	NCP Token Ring Interconnection
NTT	Nippon Telegraph Telephone
oc	overcurrent
OCR	OR character register (instruction)
OEM	original equipment manufacturer
OEMI	original equipment manufacturer's information
OHR	OR halfword register (instruction)
OLTEP	online test execution program
OLTSEP	online test standalone execution program
OLTS	online test system
OLTT	online terminal test
OMB	out mailbox
OP	operation decode
OR	OR register (instruction)
ORI	OR register immediate (instruction)
OUT	output (instruction)
ov	overvoltage
PCF	primary control field (storage)
PCI	program-controlled interrupt
PCR	power check reset
PCW	processor control word
PDF	parallel data field (storage)
PEP	partitioned emulation program
PF	program function (3727 operator console keys)
PFAR	prefetch address register
PH1-x	phase control power block x unit 01
PH2-x	phase control power block x in unit 02
PH4-x	phase control power block x in unit 02
PIO	program initiated operation
PIRR	program interrupt request register
PN	part number
POPR	prefetch operation register
POR	power-on reset
PS	power supply
PSA	parameter/status area (storage)
PSW	program status word
PTT	post, telephone and telegraph (agency)
PTX	phototransistor
PV	parity valid (signal)
PWB1	power board unit 01
PWB2	power board in unit 02
PWCA1	power-control analog in unit 01 (card)
PWCA2	power-control analog in unit 02 (card)
PWCL1	power-control logic in unit 01 (card)
PWCL2	power-control logic in unit 02 (card)
PWRC	power resistor card
RA	register to immediate address (instruction)
RAC	repair action code
RAM	random access memory
RAS	reliability, availability, and serviceability
RCAM	RC access method
RCV	receive
RD	receive data (signal)
RDV	redrive (card)
RDVAD	redrive adapter (card) register external (instruction)
RECFMS	record formatted maintenance statistics
RECMS	record maintenance statistics
REQMS	request for maintenance statistics
RFS	ready for sending (signal) (or clear to send CTS)
RI	register to immediate operand (instruction)
RIM	request initialization mode
ROK	read-only key
ROS	read-only storage
ROSAR	read-only storage address register
rpm	revolutions per minute
RPO	remote power off
RPQ	request for price quotation
RR	register to register (instruction)
RS	register to storage (instruction)
RSA	register to storage with addition (instruction)
RT	branch (instruction)
RTC	retry count (X.21)
RTM	retry timer (X.21)
RTS	request to send
RVI	reverse interrupt (BSC)
R/W	read/write
s	second
SALT	stand-alone link test
SAR	storage address register
SCB	(1) scanner control block (storage) (2) system control block
SCF	secondary control field (storage)
SCR	(1) subtract character register (instruction) (2) silicon-controlled rectifier
SCTL	storage control (card)
SDF	serial data field (storage)
SDLC	synchronous data link control (SNA)
SE	system engineer
SELN	selection (3727 operator console key)
SES	secondary status (storage)
SHR	subtract halfword register (instruction)
SIM	set initialization mode
SIO	start input/output
SIT	scanner interface trace
SKA	storage key address
SKDR	storage-protect key data register
SNRM	set normal response mode
SOH	start of heading (BSC)
SP	storage protect
SPAE	storage protect/ address exception
SPK	storage protect key
SR	subtract register (instruction)
SRI	subtract register immediate (instruction)
SRL	shift left register
SS	start-stop
SSB	system status block
SSP	system support programs
ST	store (instruction)
STC	store character (instruction)
STCT	store character and count (instruction)
STH	store halfword (instruction)
STG	storage
STX	start of text (BSC)
SVC	supervisor call
SYN	synchronous idle (BSC)
SYSGEN	system generation
TA	time address
TAP	trace analysis program
TAR	temporary address register
TC	time command
TCAM	telecommunications access method
TCB	task control block
TCC	trace correlation counter (storage)
TCP	test connector pin
TCS	two channel switch (see TPS)
TD	time data
TERMA1	terminator type A in unit 01 (card)
TERMB1	terminator type B in unit 01 (card)
TERM2	terminator in unit 02 (card)
TG	transmission group (NCP line trace)
TIC	token ring interface coupler card
TICB	trace interface control block
TIO	test I/O
TPS	two-processor switch (feature) (also referred to as TCS)
TPSA	trace parameter status area
TRA	token ring interface adapter
TRM	test register under mask (instruction)
TRM	token ring interface multiplexor card
TRSS	token ring subsystem
TRU	trace record unit
TSET	transmitter signal element timing
TSS	transmission subsystem
TTA	translate table area
TTD	temporary text delay (BSC)
UA	unumbered acknowledgment
UCW	unit control word
UE	unit exception (channel status)
UEPO	unit emergency power off
UK	United Kingdom
UKA	user key address
UKP	user key program
UKDR	user key data register
UKL	user key level interrupt
USASCII	(see ASCII)
us	microsecond
uv	undervoltage
V	volt
VB	valid byte (signal)
Vac	volts, alternating current
Vdc	volts, direct current
VFO	variable frequency oscillator
VH	valid halfword (signal)
VRC	vertical redundancy check
VTAM	virtual telecommunication access method
V.24	CCITT V.24 recommendation
V.25	CCITT V.25 recommendation
V.28	CCITT V.28 recommendation
V.35	CCITT V.35 recommendation

W	watt
WACK	wait before transmit positive acknowledgment (BSC)
WB	wrap back (signal)
WKR	working register
WSDR	working storage data register
XID	exchange identification
XCR	exclusive OR character register (instruction)
XHR	exclusive OR halfword register (instruction)
XOR	exclusive OR
XR	exclusive OR register (instruction)
XREG	external registers
XRI	exclusive OR register immediate (instruction)
X.21	CCITT X.21 recommendation
YZxxx	Wiring diagram
ZAP	control program modifier function
ZI	zero insert
ZREG	Z register
50PH1	50-Hz phase control in unit 01 (card)
50PH2	50-Hz phase control in unit 02 (card)
60PH1	60-Hz phase control in unit 01 (card)
60PH2	60-Hz phase control in unit 02 (card)

This glossary defines all new terms used in this manual. It also includes terms and definitions from the IBM Vocabulary for Data Processing Telecommunications, and Office Systems, GC20-1699.

A

access line:

In the diskette drive, a line that transmits pulses to turn the stepper motor.

adapter-initiated operation (AIO):

A transfer of up to 256 bytes between an adapter (channel or scanner) and the CCU storage. The transfer is initiated by an IOH/IOHI instruction, and is performed in cycle stealing via the IOC bus.

addressing:

A technique where the control station selects, among the DTEs that share a transmission line, the DTE to which it is going to send a message.

alcohol pad:

A pad soaked with iso-propyl alcohol.

alternate cylinder:

In the diskette drive, the area containing sectors that can be assigned in place of sectors that are not usable.

alternate track:

In the diskette drive, a track designated to contain data in place of a defective primary track.

asynchronous transmission:

Transmission in which each character is individually synchronized, usually by the use of start and stop elements. The start-stop link protocol, for example, uses asynchronous transmission contrast with 'synchronous transmission.'

auto-answer:

A machine feature that allows a DCE to respond automatically to a call that it receives over a switched line.

auto-call:

A machine feature that allows a DCE to initiate a call automatically over a switched line.

availability:

The degree to which a system or resource is ready when needed to process data.

B

bail assembly:

In the diskette drive, a mechanical arm that operates under control of the head load-solenoid to load or release the read/write head load arm.

belt clearance slots:

In the diskette drive, grooves in the fan enclosure that permit the ac motor belt to turn without rubbing against the fan enclosure.

binary synchronous communication (BSC):

A uniform procedure, using standardized set of control characters and character sequences, for synchronous transmission of binary-coded data between stations.

box error record (BER):

Information about an error detected by the controller. It is recorded on the diskette and can be displayed on the operator console for error analysis.

C

carriage:

In the diskette drive, the part that carries the read/write head under control of the stepper motor drive.

central control unit (CCU):

In the 3725, the controller hardware unit that contains the circuits and data flow paths needed to execute instructions and to control its storage and the attached adapters.

channel adapter (CA):

A communication controller hardware unit used to attach the controller to a host processor.

channel interface:

The interface between the controller and the host processors.

channel and line attachment base (CLAB):

A board that includes the first CAB and LAB of the controller.

collet:

In the diskette drive, the part that centers and holds the diskette to the drive hub.

common carrier:

In the USA and Canada, a government regulated private company that furnishes the general public with telecommunication service facilities. For example, a telephone or telegraph company (see also "post telephone and telegraph" for countries outside the USA and Canada).

communication controller:

A communication control unit that is controlled by a program stored and executed in the unit. Examples are the IBM 3705 and IBM 3725/3726.

Communication Network Management (CNM):

An IBM product program that assists the user in identifying network problems from a control point. It is stored in the host processor and comprises the network problem determination application (NPDA) and the network communication control facility (NCCF).

communication scanner:

See 'scanner'.

communication scanner processor (CSP):

The processor of a scanner.

configuration data file (CDF):

A file of the diskette that contains a description of all the hardware features (presence, type, address, and characteristics).

control panel:

A panel on the 3725 that contains switches and indicators for the use of the customer's operator and service personnel.

control subsystem (CSS):

The part of the controller that stores and executes the control program, and monitors the data transfers over the channel and transmission interfaces.

customer engineer (CE):

An individual who provides field services for IBM products.

cooling fan:

In the diskette drive, a fan that cools the stepper motor.

crosstalk:

In the diskette drive, data bits sensed from one track of the diskette while the read/write head is reading another track.

cyclic redundancy check (CRC):

A method of error checking performed at the receiving station after a block check character has been received.

D

data circuit-terminating equipment (DCE):

The equipment installed at the user's premises that provides all the functions required to establish, maintain, and terminate a connection, and the signal conversion and coding between the data terminal equipment (DTE) and the line. For example, a modem is a DCE (see "modem".)

Note: The DCE may be separate equipment or an integral part of other equipment.

data terminal equipment (DTE):

That part of a data station that serves as a data source, data sink, or both, and provides for the data communication control function according to protocols. In the 3725/3726, the DTE function is achieved by the FES with the associated LIC.

differentiator-amplifier:

An electronic circuit whose output signal is a function of the time rate of change of the input signal.

direct attachment:

The attachment of a DTE to the controller without a DCE.

diskette:

A thin, flexible magnetic disk, and its protective jacket, that records the 3725 microcode, diagnostics, error logs, and monitored data.

diskette 2D:

A diskette used for storing data on both surfaces with twice the usual bit density.

diskette drive:

A mechanism that reads and writes diskettes.

drive band:

In the diskette drive, a metal band connected to the stepper motor pulley and the head carriage assembly.

drive hub:

In the diskette drive, a continuously running part that turns the diskette at 360 rpm.

duplex transmission:

Data transmission in both directions at the same time. Contrast with 'half duplex.'

E

emulation program (EP):

The function of a network control program to perform activities equivalent to those of an IBM 2701 Data Adapter Unit, an IBM 2702 Transmission Control, or an IBM 2703 Transmission Control.

enclosure:

The diskette drive motor cooling fan safety cover.

error recovery procedure (ERP):

A procedure designed to help isolate and, where possible, to recover from errors in equipment. The procedures are often used in conjunction with programs that record the statistics of machine malfunctions.

F

front-end scanner (FES):

A circuit that scans the transmission lines, serializes and deserializes the transmitted characters, and manages the line services. It is part of the scanner.

H

half-duplex:

Data transmission in either direction, one direction at a time. Contrast with 'duplex.'

Note: The functional unit using the data circuit determines the choice of direction.

head/carriage:

In the diskette drive, the unit that contains the read/write head.

host processor:

(1) A processor that controls all or part of a user application network. (2) In a network, the processing unit in which the access method for the network resides. (3) In an SNA network, the processing unit that contains a system services control point (SSCP). (4) A processing unit that executes the access method for attached communication controllers. Also called 'host'

I

identifier:

In the diskette drive, a character or group of characters used to identify or name an item of data and possibly used to indicate some properties of that data.

initial microcode load (IML):

The process of loading the microcode into a scanner or into MOSS.

initial program load (IPL):

The initialization procedure that causes 3725 control program to commence operation.

input/output control (IOC):

The circuit that controls the input/output from/to the channel adapters and scanners via the IOC bus.

internal clock circuit (ICC):

An optional circuit that provides, through the LICs, the clock control to the DCEs or DTEs that need it.

internal function test (IFT):

A set of diagnostic programs designed and organized to detect and isolate a malfunction.

J

jacket:

A permanently attached cover that protects the diskette surface.

L

line:

See 'transmission line'.

line attachment base (LAB):

The unit of modularity of the transmission subsystem. It corresponds to one board and includes mainly the scanners and the line interface couplers.

line interface coupler (LIC):

A circuit that attaches up to four transmission cables to the controller.

Link Problem Determination Aid (LPDA):

A set of test facilities resident in the IBM 386X modems and activated from the control program in the controller.

link protocol:

The set of rules by which a logical data link is established, maintained, and terminated, and by which data is transferred across the link.

longitudinal redundancy check (LRC):

A system of error checking performed at the receiving station after a block check character has been accumulated.

M

maintenance and operator subsystem (MOSS):

The part of the controller that provides operating and servicing facilities to the customer's operator and customer engineer.

microcode:

A program, that is loaded in a processor (for example, the MOSS processor) to replace a hardware function. The microcode is not accessible to the customer.

modem (MOdulator-DEModulator):

A functional unit that transforms logical signals from a DTE into analog signals suitable for transmission over telephone lines (modulation), and conversely (demodulation). A modem is a DCE. It may be integrated in the DTE.

MOSS input/output control (MIOC):

The circuit that controls the input/output from/to the MOSS.

multiplexing:

The division of a transmission facility into two or more channels by allotting the common channel to several different channels, one at a time.

multipoint connection:

A connection established among more than two data stations for data transmission. The connection may include switching facilities.

N

network:

See 'user application network'.

Network Control Program (NCP):

A program, generated by the user from a library of IBM-supplied modules, that controls the operation of a communication controller.

nonswitched line:

A permanent dedicated transmission line that connects two or more DTEs. The connection can be point-to-point or multipoint. The line can be leased or private. Contrast with 'switched line.'

O

online tests:

Testing of a remote data station concurrently with the execution of the user's programs (that is, with only minimal effect on the user's normal operation).

operator console:

The IBM 3727 Operator Console that is used to operate and service the 3725 through the MOSS. A primary operator console must be located within 5 m (16 ft) of the 3725. Optionally an alternate operator console may be installed up to 150 m (492 ft) from the 3725.

P

partitioned emulation programming (PEP):

A feature of NCP that permits some lines to operate in network control mode while simultaneously operating others in emulation mode.

phototransistor:

An electronic part used to sense the light of a light-emitting diode.

point-to-point connection:

A connection established between two data stations for data transmission. The connection may include switching facilities.

polling:

The process whereby stations are invited, one at a time, to transmit.

post telephone and telegraph (PTT):

A generic term for the government-operated common carriers in countries other than the USA and Canada. Examples of the PTT are the Post Office Corporation in the United Kingdom, the Deutsche Bundespost in Germany, and the Nippon Telephone and Telegraph Public Corporation in Japan.

program-initiated operation (PIO):

A transfer of four bytes between a general register in the CCU and an adapter (channel or scanner). The transfer is initiated by IOH/IOHI instruction and is executed via the IOC bus.

R

redrive card:

A card that repowers the IOC bus signals at board entry. It also has logical and checking functions.

reliability:

The ability of a functional unit to perform its intended function under stated conditions, for a stated period of time.

S

scanner:

A device that scans and controls the transmission lines. It is composed of one communication scanner processor (CSP) and one front-end scanner (FES).

services:

A set of functions designed to facilitate the maintenance of a device or system.

serviceability:

The capability to perform effective problem determination, diagnosis, and repair on a data processing system.

solenoid plunger:

In the diskette drive, a moving part of the solenoid that operates the bail assembly to load and release the read/write head load arm.

start-stop:

A data transmission system in which each character is preceded by a start signal and is followed by a stop signal.

stepper motor:

In the diskette drive, the motor that steps the head carriage assembly from track to track.

switched line:

A transmission line with which the connections are established by dialing, only when data transmission is needed. The connection is point-to-point and uses a different transmission line each time it is established. Contrast with 'nonswitched line.'

synchronous data link control (SDLC):

A discipline for managing synchronous, code-transparent, serial-by-bit information transfer over a link connection. Transmission exchanges may be duplex or half-duplex over switched or nonswitched links. The configuration of the link connection may be point-to-point, multipoint, or loop. SLDC conforms to subsets of the Advanced Data Communication Control Procedures of the American National Standards Institute and High-level Data Link Control (HDLC) of the International Standards Organization.

synchronous transmission:

Data transmission in which the sending and receiving instruments are operating continuously at substantially the same frequency and are maintained, by means of correction, in a desired phase relationship. Contrast with 'asynchronous transmission.'

systems network architecture (SNA):

The description of the logical structure, formats, protocols, and operational sequences for transmitting information through a user application network. The structure of SNA allows the users to be independent of specific telecommunication facilities.

T

timeout:

The time interval allotted for certain operations to occur.

transmission interface:

The interface between the controller and the user application network.

transmission line:

The physical means for connecting two or more DTEs (via DCEs). It can be nonswitched or switched. Also called a 'line.'

transmission subsystem (TSS):

The part of the controller that controls the data transfers over the transmission interface.

tunnel erase circuit:

In the diskette drive, an electronic circuit that is used to erase the edge of the track just recorded during a write operation. This erasing prevents crosstalk between track during later read operations.

two-processor switch (TPS):

A feature of the channel adapter that connects a second channel to the same adapter.

U

user application network:

A configuration of data processing products, such as processors, controllers, and terminals, for the purpose of data processing and information exchange. This configuration may use circuit-switched, packet-switched, and leased-circuit services provided by carriers or PTT. Also called a 'user network.'

V

variable frequency oscillator:

An electronic circuit that is used to synchronize the MOSS reading circuits with the diskette drive when it is performing a read operation.

vertical redundancy check (VRC):

An odd parity check performed on each character of a block as the block is received.

W

write/erase:

Writing data to and erasing from a diskette.

Index

D

E

F

G

H

I

J

K

L

M

T

U

V

W

X

Z

0

3

5

Note: Staples can cause problems with automated mail sorting equipment.
Please use pressure sensitive or other gummed tape to seal this form.

READER'S COMMENT FORM

3725/3726 Communication Controller and Expansion
Maintenance Information Manual (MIM) Part 1 Volume 2
Order No. SY33-2018-7

This manual is part of a library that serves as a reference source for customer engineers. You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

Your comments will be sent to the author's department for whatever review and action, if any, are deemed appropriate. Comments may be written in your own language: English is not required.

Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the branch office serving your locality.*

Possible topics for comments are:

Clarity Accuracy Completeness Organization Coding Retrieval Legibility

If you wish a reply, give your name, company, mailing address, and date:

.....
.....
.....
.....

What is your occupation? .....

Number of latest Newsletter associated with this publication: .....

Thank you for your cooperation. No postage stamp necessary if mailed in the USA. (Elsewhere, an IBM office or representative will be happy to forward your comments or you may mail directly to the address in the Edition Notice on the back of the title page.)

Note: Staples can cause problems with automated mail sorting equipment.
Please use pressure sensitive or other gummed tape to seal this form.

READER'S COMMENT FORM

3725/3726 Communication Controller and Expansion
Maintenance Information Manual (MIM) Part 1 Volume 2
Order No. SY33-2018-7

This manual is part of a library that serves as a reference source for customer engineers. You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

Your comments will be sent to the author's department for whatever review and action, if any, are deemed appropriate. Comments may be written in your own language: English is not required.

Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the branch office serving your locality.*

Possible topics for comments are:

Clarity Accuracy Completeness Organization Coding Retrieval Legibility

If you wish a reply, give your name, company, mailing address, and date:

.....
.....
.....
.....

What is your occupation? .....

Number of latest Newsletter associated with this publication: .....

Thank you for your cooperation. No postage stamp necessary if mailed in the USA. (Elsewhere, an IBM office or representative will be happy to forward your comments or you may mail directly to the address in the Edition Notice on the back of the title page.)

Reader's Comment Form

Fold and tape | Please Do Not Staple | Fold and tape

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 40 ARMONK, N.Y.

POSTAGE WILL BE PAID BY ADDRESSEE:

International Business Machines Corporation
Department 6R1LG
180 Kost Road
Mechanicsburg
PA 17055

Fold and tape | Please Do Not Staple | Fold and tape

IBM

Cut or Fold Along Line

SY33-2018-7

Reader's Comment Form

Fold and tape | Please Do Not Staple | Fold and tape

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 40 ARMONK, N.Y.

POSTAGE WILL BE PAID BY ADDRESSEE:

International Business Machines Corporation
Department 6R1LG
180 Kost Road
Mechanicsburg
PA 17055

Fold and tape | Please Do Not Staple | Fold and tape

IBM

Cut or Fold Along Line

IBM