File Number 360/20-13
Form Number A26-5622-0

IBM Systems Reference Library

IBM System/360 Model 20 Universal Code Synchronous Transmit-Receive (RPQ)

This publication describes the universal code synchronous transmit-receive (UCSTR) custom communications feature. The units and features of the IBM System/360 Model 20 are described briefly, and a more complete description is provided for the UCSTR feature.

PREFACE

This publication describes the universal code synchronous transmit-receive (UCSTR) custom communications feature. This feature is available on a request for price quotation from IBM basis (RPQ M24798). Other RPQ features can be used in conjunction with the UCSTR. The following is a list of features and their respective RPQ number. For a complete description of each feature, refer to UCSTR Special Features.

Feature	RPQ Number
High speed	M24418
Six-bit and seven-bit level code	M24421
Longitudinal redundancy check (LRC)	M25019
Vertical redundancy check (VRC)	834533
Terminal address recognition	M34070

This publication includes sufficient information about the IBM System/360 Model 20 to provide a foundation for the special devices and operations included with the UCSTR feature. For more detailed information about the IBM System/360 Model 20; refer to the Systems Reference Library publication IBM System/360 Model 20 Functional Characteristics Form A26-5847.

First Edition

Specifications contained herein are subject to change from time to time. Any such change will be reported in subsequent revisions or Technical Newsletters.

The illustrations in this manual have a code number in the lower corner. This is a publishing control number and is not related to the subject matter.

Copies of this and other IBM publications can be obtained through IBM Branch Offices.

A form is provided at the back of this publication for your comments.

This manual was prepared by the IBM Systems Development Division, Product Publications, Dept. 455, Bldg. 014, San Jose, California 95114. Send comments concerning the contents of this manual to this address.

The universal code synchronous transmit-receive (UCSTR) feature provides a means of communication that is independent of the transmission code used. The UCSTR is designed for flexibility, allowing communication in several modes of operation.

COMMUNICATION FACILITIES

Private line or switched (dial-up) communications facilities may be used to implement UCSTR operation. These facilities can be obtained from communications common carriers or, if equivalent to the common carrier facilities, they may be customer provided. The communication facilities must provide for either half-duplex message transmission (alternate transmission in either direction, over one pair of wires, on a non-simultaneous basis), or four-wire half-duplex facilities may be used, reducing the normal turn-around delays associated with two-wire half-duplex message transmission. Message transmission for either two-wire or four-wire facilities is in only one direction at a time.

For UCSTR operations, the communications configuration can be designed for either a point-to-point or multipoint operation, depending on the requirements of the user. The specific data set equipment used at each channel termination point (station) is determined by the type of communications channel as well as the operating speed of the terminal equipment located at each station.

Point-to-Point Configurations

When operating on a point-to-point basis (Figure 1) all message exchanges are between only two stations. If the point-to-point configuration is a permanent installation (e.g. leased line or private line), message exchanges are always between the same two stations. However, if the facilities are the switched type, any station on the network can transmit to or receive from any other station on the network.

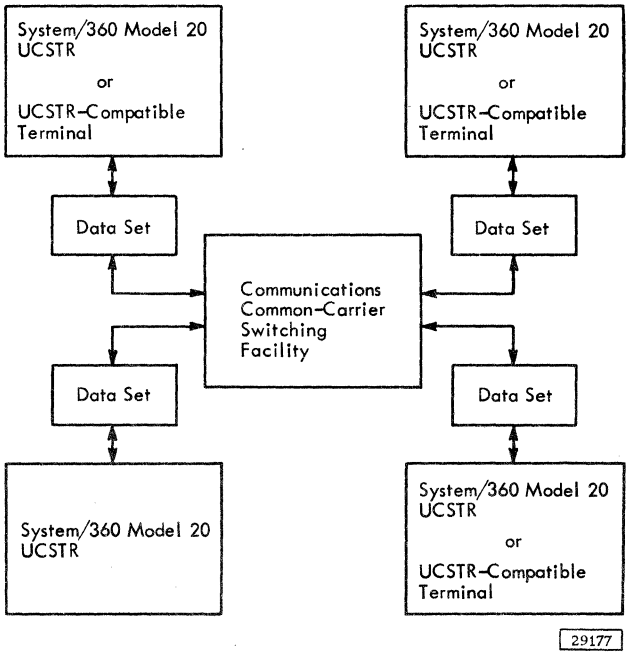


Figure 1. Point-to-Point Configuration

Multipoint Configurations

When operating on a multipoint basis (Figure 2), all message exchanges are between a master station and one or more remote stations. Control of message exchanges is performed by the master station. All message exchanges are initiated by a remote station selection operation (polling) that is performed by the master station.

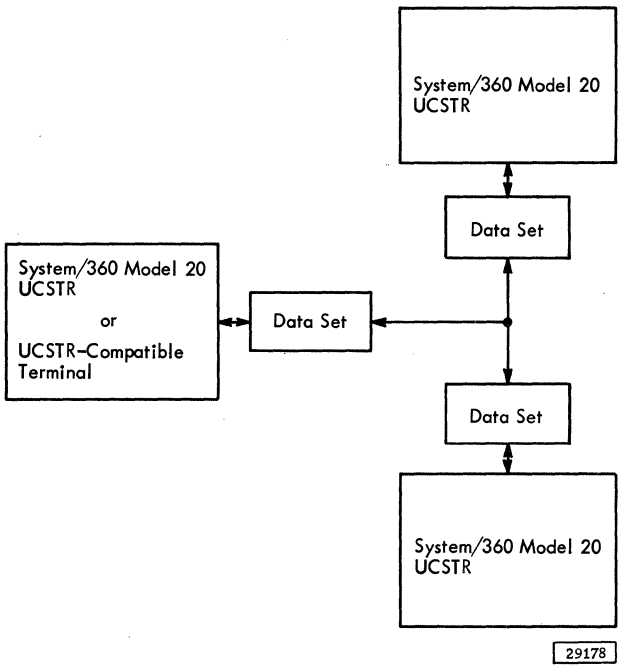


Figure 2. Multipoint Configuration

MODES OF OPERATION

This section provides a brief description of UCSTR modes of operation. For a more complete description of each mode, refer to UCSTR Feature.

Basic Mode

The basic mode of operation is the full binary minus three mode of operation. This term means that the 256 possible characters (eight-bit level) can be transmitted or received as data, with the exception of three bit combinations (characters) that are defined as control characters by the user's program. The three control characters are: end-of-block (ETB), end-of-transmission (EOT) and synchronization(sync). The bit combination used for each character is selected by the user. See UCSTR Control Field.

The seven-bit level code (special feature) can, in the basic mode, transmit or receive, as data, the 128 possible bit combinations (characters) minus the three control characters defined by the user's program.

The six-bit level code (special feature) can, in the basic mode, transmit or receive, as data, the 64 possible bit combinations (characters) minus the three control characters defined by the user's program.

Full Transparent Mode

The full transparent mode provides the capability of transmitting or receiving all 256 bit combinations (128 combinations in seven-bit level and 64 combinations in six-bit level).

Response Mode

The UCSTR has the capability of automatically setting to receive mode (after a transmit XIO instruction is completed) to receive a response from the terminal that received the message block. The automatic response operation can be in conjunction with either the basic mode or transparent mode. For a complete description of response mode, refer to UCSTR Feature.

TRANSMISSION CODES

Messages are transmitted and received in any eight-bit transmission code. However, an additional feature is available that allows the UCSTR to transmit and receive in any six-bit code or any seven-bit code.

TRANSMISSION ERROR CHECKING

The cyclic redundancy check (CRC) is the basic transmission error checking method of the UCSTR. The cyclic redundancy check is accomplished by an arithmetic accumulation of the message characters at both the transmitting and receiving stations. The transmitting UCSTR automatically sends the accumulated check character (CRC character) immediately following the message ending character (ETB or EOT). The receiving station automatically subtracts the transmitting station's check character from its own accumulated check character. If the result is zero, the message was correct. If the result is not zero, the CRC error indicator in the receiving UCSTR is set on.

CRC checking is used for eight-bit level, seven-bit level, or six-bit level operations.

With the addition of a special feature, the UCSTR can perform longitudinal redundancy checking (LRC) or vertical redundancy checking (VRC). See UCSTR Special Features.

TRANSMISSION SPEEDS

Transmission speeds from 600 to 50,000 bits per second (baud) are available.

The basic UCSTR operates at speeds of 600 to 4,000 baud with timing control provided by the UCSTR (internal clocking). The basic UCSTR can operate at any speed (600 to 4,800 baud) with the timing control provided by the appropriate data set (external clocking).

With the addition of a special feature, the UCSTR can operate at speeds up to 50,000 baud, with the timing control provided by the appropriate data set (external clocking). However, time sharing is lost at speeds above 7200 bps.

COMMUNICATIONS WITH IBM EQUIPMENT

The UCSTR can communicate with another System/360 Model 20 with the UCSTR feature, or the UCSTR can communicate with another model of System/360 via the IBM 2701 Data Adapter Unit with the inter-processor communications adapter feature (RPQ M24802).

COMMUNICATIONS WITH NON-IBM EQUIPMENT

Flexibility of the UCSTR allows it to communicate in a variety of line formats.

The UCSTR communicates in an eight-bit level line code, but with the appropriate special feature, it communicates in eight-bit level, seven-bit level, or six-bit level line codes.

Checking is accomplished by a cyclic redundancy check (CRC), but the addition of the appropriate special feature, provides longitudinal redundancy checking (LRC) and/or vertical redundancy checking (VRC). In order for a non-IBM device to communicate with the UCSTR, it must meet the following criteria:

1. It must be capable of sending at least two synchronization (sync) characters preceding each data message or control sequence. The sync character can be any eight-, seven-, or six-bit combination.
2. It must be capable of synchronizing its timing circuitry with the UCSTR by receiving from two to seven sync characters.
3. It must end all control sequences and message blocks with an ending character. No more than two unique end-of-block or end-of-message characters can be used. The UCSTR can recognize only two ending characters. The ending character can be any eight-, seven- or six- bit combination.
4. It must send the CRC (or LRC) character immediately following the ending character.
5. It must place the low-order bit of each character on the transmission line first.
6. If the system uses CRC checking, the message polynomial must agree with the UCSTR message polynomial:

Eight-Bit	$X^{16} + X^{15} + X^2 + 1$
Seven-Bit	$X^{14} + X^{13} + X^5 + X^3 + X^2 + 1$
Six-Bit	$X^{12} + X^{11} + X^3 + X^2 + X + 1$

29222

7. The UCSTR uses a "mark" (down-level) to and from the data set as a logical one (up-level) in the UCSTR, and the UCSTR uses a "space" (up-level) as a logical zero (down-level) in the UCSTR. A non-IBM terminal must be compatible with this operation.
8. The terminals must use the same clocking. If the non-IBM terminal uses data set clocking, the UCSTR must also use data set clocking.
9. If the data set provides the clocking, the UCSTR uses the positive going clock transition as a negative transition in UCSTR logic. The non-IBM terminal must be compatible with this operation.
10. The UCSTR does not operate in full transparency when the LRC check feature is used.
11. The UCSTR full transparent feature is not the same as the full transparent operation defined by the American Standard Association (ASA). As an example, the UCSTR uses a DLE-sync as an idle character rather than a sync sync.

IBM SYSTEM/360 MODEL 20 UNITS

The UCSTR feature can be installed on an IBM 2020 Central Processing Unit that has the prerequisite feature: communications adapter special feature. The communications adapter special feature is altered by the UCSTR feature so that normal System/360 Model 20 synchronous transmit-receive (STR) operations are no longer possible.

All other units and features of the IBM System/360 Model 20 are available for a system that contains the UCSTR feature. Figure 3 illustrates the variety of card reading, card punching, magnetic tape, disk storage, and printing units that can be attached to the IBM 2020 Central Processing Unit. For a more complete illustration of IBM System/360 Model 20 units and features, refer to the Systems Reference Library publication IBM System/360 Model 20 Configurator Form A26-3572.

IBM 2020 CENTRAL PROCESSING UNIT

The IBM 2020 Central Processing Unit (CPU) operation is unchanged by the addition of the UCSTR feature. The UCSTR feature uses the same interrupt priority level as the standard communications adapter. Input/output instructions used by the UCSTR are described in the section for UCSTR Feature.

GENERAL INFORMATION

Main Storage

Main storage consists of 4,096, 8,192, 12,288, and 16,384 positions of magnetic core storage. Each position has an address and contains an eight-bit unit of information referred to as a byte.

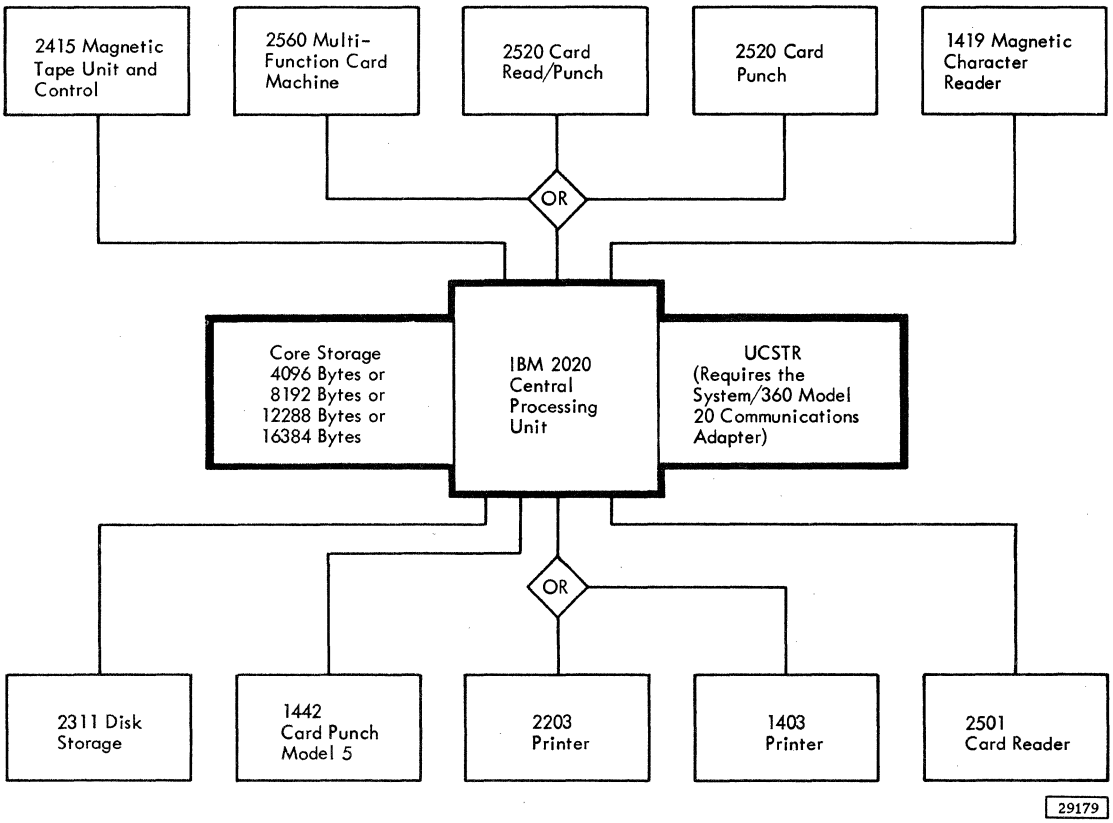
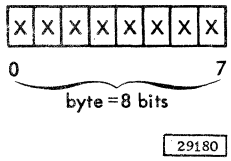


Figure 3. IBM System/360 Model 20 Units

Coded combinations of bits within a byte can represent alphabetic, numeric, binary, or logical data.



Main storage is used to hold all the data that is to be operated upon, or processed, at a given time. It also holds the instructions, or program, which control the operation of the system. For each individual job performed by the system, certain portions of main storage are assigned to store instructions, and certain portions to store data to be processed.

General Registers

Eight general purpose registers, small auxiliary storage units, are provided for temporary storage of small amounts of data. Each register is the equivalent of two bytes (one halfword) and is loaded or unloaded under the control of the stored program. Information may flow from register to register, from main storage to register, or from register to main storage.

The registers are numbered 8-15 and are selected by the four-bit R- or B-field of an instruction.

Parity Checking

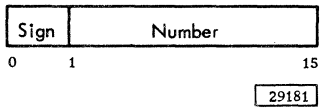
To ensure the accurate transfer of data, an extra (parity) bit is generated for each four bits transferred to or from main storage or register storage. The parity bit is added during transfer, if it is needed, to maintain an odd number of bits. The bit count is monitored continuously, and any missing or extra bits which result in an even number of bits cause a CPU parity error.

Data Formats

The basic unit of addressable data is an eight-bit byte. Each address contains eight bits of data and can be considered a byte boundary.

The byte is divided into two sections of four bits each. A check (parity) bit is provided for each four bits.

Halfword Binary Number

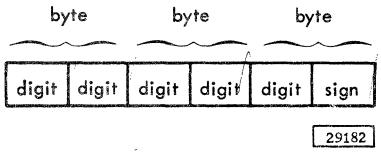


A halfword binary number has a fixed length of two bytes (16 bits). The leftmost bit is reserved for sign (+ or -) control.

Decimal Number

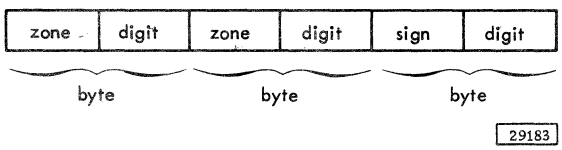
A decimal number may be in either of two forms: packed decimal or zoned decimal.

Packed Decimal: This format allows two numeric digits to be stored within one eight-bit byte.



An eight-bit byte may contain two numeric digits, except in the case of the rightmost byte, which has a sign to the right. Variable field length in this format allows for fields up to 16 bytes in length.

Zoned Decimal: This format contains one digit in the rightmost four positions of each byte.

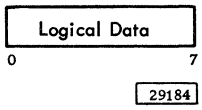


The left four bits of each byte in this format are called zone bits, and are not used except for the low-order (rightmost) byte, which uses them for the sign. Zone bits do not affect the value of the numeric digit contained in the right four bits of the byte.

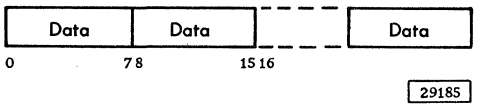
Logical Data

Logical data may be contained in the instruction or it may reside in core storage as an operand (An operand is data contained in or addressed by the instruction and used to execute the instruction.) Logical data may have a fixed length of one byte or may be variable in length up to a maximum of 256 bytes.

Fixed-Length Logical Data



Variable-Length Logical Data



Instruction Format

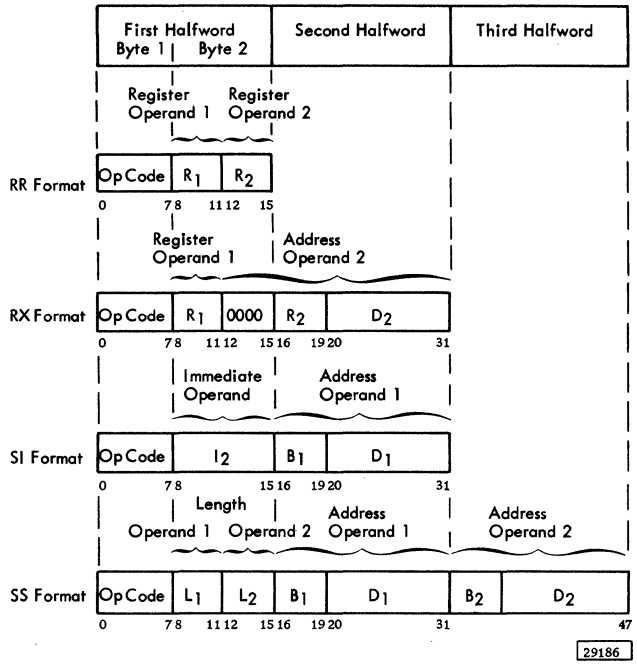
The instruction format specifies the length of the instruction and the type of operation to be performed. The length of the instruction can be one, two, or three halfwords. The types of instruction formats are RR, RX, SI, and SS.

RR Format: Denotes a register-to-register operation.

RX Format: Denotes a register-to-storage or a storage-to-register operation. In this format, bits 12 through 15 must be zero.

SI Format: Denotes a storage-immediate operation. In this format the I2 field of the instruction is the second operand.

SS Format: Denotes a storage-to-storage operation.



In each format, the first instruction halfword consists of two parts. The first byte contains the operation code (op code). The length and format of an instruction are specified by the first two bits of the operation code. The second byte may be used to contain data, specify operand lengths, or specify registers to be used by the operation.

The field names refer to the manner in which the operands participate. The operand to which a field in an instruction format applies is generally denoted by the number following the code name of the field, for example, R1, B1, L2, D2. Normally, the operation of the CPU is controlled by instructions taken in sequence. An instruction is fetched from a location specified by the current instruction address. The current instruction address itself is located in the program status word (PSW). After the fetch operation, the current instruction address is increased by the number of bytes in the fetched instruction to enable addressing of the next instruction in sequence.

The instruction is then executed by adding, subtracting, multiplying etc., both operands with each other. The result that is thus obtained usually

replaces operand 1. Upon the execution of certain instructions, a condition code, which reflects the nature of the result, is set into the PSW.

Subsequently, the updated address in the PSW is used to read out the next instruction from the main storage, and the processing continues.

Information Positioning

Byte locations in storage are consecutively numbered from 0, each number being considered the address of the corresponding byte. Bytes may be handled singly or strung together in fields. A group of two consecutive bytes is called a "halfword". The location of any field or group of bytes is specified by the address of the leftmost byte.

Information positioned in storage may be in fixed-length format or variable-length format. The length of fields is either implied by the operation to be performed or stated explicitly as part of the instruction. When the length is implied, the information is said to have a fixed length, which can be one, two, or four bytes.

Fixed-length fields must be located in main storage on an integral (halfword) boundary (Figure 4) for that unit of information. A boundary is called integral for a unit of information when its storage address is a multiple of the length of the unit in bytes. For example, a halfword (two bytes) must have an address that is a multiple of the number 2.

In the 2020, all instructions and all data of fixed word length (operands addressed by RX-format instructions) must begin on a halfword boundary. An instruction (or data with fixed word length) is properly located at a halfword boundary when its address is even or, in other words, when the low order bit of the address is zero. An improperly placed instruction causes an error stop. When the length of a field is not implied by the operation code, but is stated explicitly, the information is said to have a variable-length field.

Within any instruction format or any fixed-length operand format, the bits are consecutively numbered from left to right, starting with bit number 0.

Variable-length fields are not restricted to half-word boundaries and may contain up to 256 bytes. Length is variable in increments of one byte.

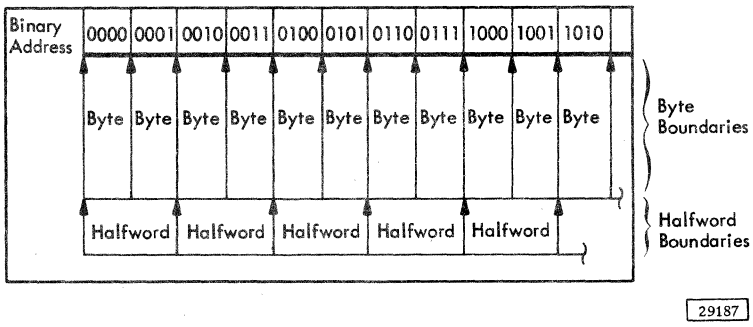


Figure 4. Integral Boundaries

UCSTR Information Positioning

The UCSTR can transmit a record of 4,095 bytes, and it can receive a record 1,023 bytes. Protection boundaries are provided by the UCSTR to protect data and programs if, for any reason, a message ending character is not recognized by the UCSTR. The storage address protection boundaries are located at core storage location 1023, 2047, 3071, 4095, 5119, etc. A receive field must be contained between two boundary locations (e.g. core storage location 1024 through core storage location 2047). The receive field cannot be located so that it ends at the highest core storage location installed (4,095, 8,191, 12,287, or 16,383).

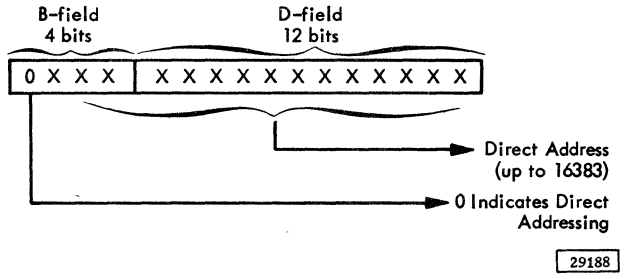
If the receive field is improperly located, the record check indicator is turned on, the receive error TIOB indicator is set on, and the audible alarm sounds when the receive field overruns a protection boundary.

ADDRESSING

Byte locations in storage are expressed in binary form and consecutively numbered from 0000 to the upper limit of available storage. The first 144 bytes (bytes 0000-0143) are reserved for internal CPU control and are not available to the program. The location of any field or group of bytes is specified by the address of the left most byte.

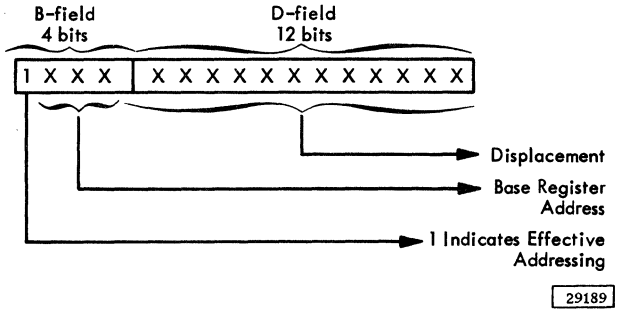
An address used to refer to main storage may be specified by either of two methods: direct addressing or effective address generation.

Direct Addressing: Direct addressing is used when the high-order bit in the B-field of an instruction is zero.



When the direct addressing method is employed, the low-order 14 bits of the combined B- and D-fields are used to refer directly to byte locations in main storage. The twelve binary bits in the D-field allow an address specification of up to 4,095. To address additional (optional) storage, the adjacent two bits in the low-order position of the B-field are used, allowing address specification of up to 16,383.

Effective Addressing: Effective addressing is used when the high-order bit in the B-field of an instruction is one.



In the effective-address generation method, the contents of the general register specified by the B-field of an instruction, are added to the contents of the D-field of the instruction to form the effective address. The content of the general register specified by the B-field is referred to as the "base address." The content of the D-field is referred to as the "displacement." (This type of address modification is commonly referred to as indexing.) Effective addressing may be either in a positive or a negative direction, depending on the sign of the base address. Caution should be exercised because a resultant effective address that is negative or that refers to the first 144 bytes of main storage results in an error condition.

Any of the eight general registers, 8 through 15, may be specified in the B-field as the location of the base address for effective-address generation.

If there are 0s in either the general register specified, or in the displacement field of the instruction when effective-address generation is specified, the effective address generated is the same as direct addressing with the nonzero component.

Effective addressing is useful for program routines which require address modification.

OPERANDS

For addressing purposes, operands can be grouped in three classes: explicitly addressed operands in main storage,

immediate operands placed in main storage as part of the instruction stream, and operands located in the general registers.

Explicitly Addressed Operands

An explicitly addressed operand is selected from a main-storage location not related to the location of the instruction referring to it. It is always specified by means of a storage address. When the operand contains more than one byte, the address gives the location of the first byte of the field, subsequent bytes being located in higher addresses. Both the first and second operands of an instruction can be explicitly addressed.

Explicitly addressed operands can be of fixed length or variable length. The length of variable-field-length operands is specified in the L-field of the instruction. The L-field, either four or eight bits long, specifies the length in terms of the number of bytes to the right of the addressed byte, and thus can specify a maximum field length of 256 bytes.

Immediate Operands

An immediate operand consists of one eight-bit byte of data which is located in the instruction itself. Only the instructions in the SI format contain immediate data. The immediate data is always the second operand; the first operand is located in the main-storage location specified by the B1-D1 field.

Operands in Registers

Information referred to by an instruction may be located in one of eight general registers. The registers are identified by numbers 8-15 and are selected by the four-bit R-or B-field of an instruction. The registers are not designated by main-storage addresses. An operand located in a register has a fixed length of one halfword, or 16 bits.

TIME-SHARING

The 2020 has the ability to operate in a mode referred to as "time-sharing." Time-sharing is a means of overlapping input/output operations with each other and with processing. Time-sharing is based on a system of monitoring the operation of input/output devices and sequencing the transfer of data to or from the I/O devices so as to make the most efficient use of processing time.

Processing operations in the CPU are time shared with the transfer of data between main storage and the I/O devices. When an I/O device requests service, processing is suspended only for the time required to send or to accept the input/output data.

Time-sharing allows the CPU to perform useful processing functions while card or forms movement is taking place. A system of signaling that the I/O device is finished with the data transfer to or from the CPU is referred to as an "interruption" system.

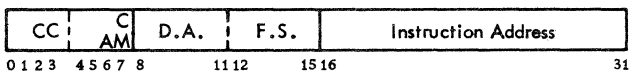
UCSTR Time-Sharing

When operating at speeds below 4800 bits per second, the UCSTR time-shares its operation with all System/360 Model 20 input/output devices. When operating at speeds above 4800 bits per second the UCSTR time-shares only with the 2020 Central Processing Unit and the 1403 buffered printer.

change to 7200

Program Status Word

The program status word (PSW) contains the information necessary for proper program execution. It is located in an internal register in the CPU and is not directly addressable. The programmer can change the PSW by means of a set PSW instruction. The PSW has a fixed-length format of two halfwords.



0-1 Not used
2-3 Condition Code
4-5 Not used
6 *ASCII Mode Bit
7 Channel Mask
8-11 Device Address
12-15 Function Specification
16-31 Instruction Address

* American Standard Code for Information Interchange

29190

The PSW is used to control instruction sequencing and to hold and indicate the status of the system in relation to the program being executed. The active or controlling PSW is called the "current PSW."

By storing the current PSW during an interrupt, the status of the CPU can be preserved for subsequent inspection. By loading a new PSW or part of a PSW, the state of the CPU can be initialized or changed.

When the current PSW is stored during an input/output interruption, the status of the CPU and the next sequential address are preserved for use after the interruption is serviced. An instruction to load a new PSW is equivalent to an unconditional branch to the instruction address contained in the new PSW.

Operation of the program load key or system reset key causes the condition code (bits 2-3), the ASCII mode bit (bit 6), and the channel mask bit (bit 7) of the current PSW to be reset to 0.

INTERRUPTION

In the 2020, an automatic interrupt system is provided to make optimum use of the I/O devices and available processing time. The I/O devices signal the CPU to store an end condition when a data transfer has been terminated. Upon completion of the execution of each instruction, during the time that the CPU is in an interruptible mode, the CPU checks the various end conditions according to a built-in priority sequence. The first end condition thus found causes the actual interrupt. The interrupt is an automatic branch from the main program to a subroutine. This subroutine may be designed to test the received data for validity or to perform some other action. Since the interrupt occurs at the earliest moment possible after a data transfer has been completed, it is the primary means of controlling I/O operations.

The branch to the subroutine is accomplished by replacing the current PSW with a new PSW which contains the start address of the subroutine in its instruction address portion. The former current PSW is stored into a particular core-storage location and thus becomes the old PSW. During the transfer, the old PSW is furnished with the device address and the function specification of the I/O unit that caused the interrupt. If the last instruction in the subroutine is a branch with the address of the old PSW specified as the branch address, the normal course of the main program is resumed because the next sequential instruction of the main program is specified in the old PSW. The entire exchange of the program status words (PSWs) is fully automatic; however, the channel mask bit in the PSW, which determines whether the CPU is interruptible or not, may be set or reset under program control. The CPU is interruptible when the channel mask bit is set to 1; it is not interruptible if the channel mask bit is 0. The channel mask bit is 0 when the CPU is in a reset state.

GENERAL INFORMATION

The universal code synchronous transmit-receive (UCSTR) feature is a flexible system of communications that allows the user to define the control characters, determine the message format and establish the line-control procedure best suited for a specific application.

The IBM System/360 Model 20 communications adapter is a prerequisite feature for the UCSTR feature. However, the communications adapter is altered to accomodate the UCSTR so that normal System/360 Model 20 synchronous transmitter receiver (STR) operations are not possible.

UCSTR operation can be with two-wire or four-wire half duplex communications facilities. Half duplex means that transmission can be in only one direction at a time. Two-wire facilities require approximately 250 milliseconds (ms) to condition the facility for transmission in one direction. This conditioning of the facility is commonly referred to as turn around delay. Each time the direction of transmission is changed a turn around delay occurs. Depending on the specific application, the turn around delay time can be used for System/360 Model 20 internal processing or input/output operations.

If the application requires less turn around delay, a four-wire facility can reduce turn around delay to 3.3 ms to 26.0 ms depending on the line speed selected. At higher speeds on four-wire facilities, turn around delay can practically be disregarded in connection with long records.

Transmission speeds can be from 600 bits per second to 50,000 bits per second. (See Speed Select.)

Transmission codes can be eight-bit level on the basic UCSTR, or with the addition of a special feature, they can be eight-bit level, seven-bit level, or six-bit level (See UCSTR Special Features.)

Control Characters

The basic mode of UCSTR operation uses three control characters. The three control characters can be any three bit combinations selected by the user. The three control characters are stored in the UCSTR control field, and they can be changed to another bit combination under program control. The three control characters are as follows.

Sync: The UCSTR (or any synchronous communications device) must have a method of obtaining synchronization of the timing circuitry in the receiving station and the transmitting station. For this purpose, one character (bit combination) must be defined as a synchronization (sync) character. The sync character is stored in the control field by the program. (See Control Field.)

ETB: The end-of-block (ETB) character is one of two ending characters used by the UCSTR. This character is used to define the end of a message block. When the transmitting station sends the ETB character, it ends the transmission after sending the CRC character. When the receiving station receives the ETB, it ends the receiving operation after it subtracts the incoming CRC characters. The ETB character ends a message transfer but does not define the end of transmission.

EOT: The end-of-transmission (EOT) character is one of two ending characters used by the UCSTR. The UCSTR functions the same as when sending an ETB character except that the receiving station sets the receive EOT test-and-branch indicator. The receive EOT test-and-branch indicator is tested by the program to determine when the last record has been received.

Transparent Control Characters

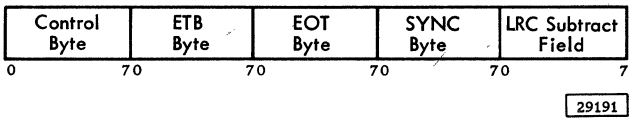
When the UCSTR is operating in the transparent mode, one additional control character is used. The data link escape (DLE) character has a bit configuration of 00010000, and it cannot be changed by the program. The ending characters in transparent mode are two characters that form a 16-bit ending character. The end-of-block character is DLE-ETB, and the end-of-transmission character is DLE-EOT. The use of the DLE character is described in the section for Transparent Mode.

Control Field

The UCSTR control field is any five consecutive bytes of storage used as a link between the System/360 Model 20 program and the UCSTR feature. The control field is defined by the user, and can be altered by the program according to the requirements of the application.

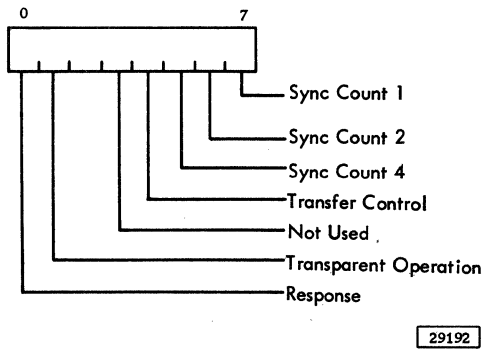
The program transfers the control field or the control byte (see Control Byte) to the UCSTR feature when a set-to-transmit or set-to-receive instruction is executed.

The program addresses the control field by addressing the high-order byte.



Control Byte

The control byte is used to control UCSTR mode of operation and to indicate the number of sync characters that will be emitted by the transmitting unit. The following illustration shows the purpose of each bit in the control byte.



Response Bit: The response bit (bit 0) places the UCSTR in response mode. Response mode allows the UCSTR to automatically set to receive after a message block has been transmitted. Thus, a single XIO transmit record instruction initiates the transmission of a message block and the receiving of a response. See Program Instructions.

Transparent Operation: The transparent operation bit (bit 1) places the UCSTR in the transparent mode of operation. See Transparent Mode.

Transfer Control: The transfer control bit (bit 4) provides the program the means to transfer the control byte or the entire control field (five bytes) to the UCSTR feature. If the transfer control bit is a one the entire five-byte control field is transferred when a set-to-transmit or set-to-receive instruction is executed. If the transfer control bit is a 0, only the control byte is transferred when a set-to-transmit or set-to-receive instruction is executed. Thus, for the first set-to-transmit or set-to-receive instruction, the transfer control bit must be set to 1 so that the complete control field is transferred. The complete control field must be transferred when the system has been "powered up" or if the system reset key has been pressed. For subsequent set-to-transmit or set-to-receive instructions, the transfer control bit can be set to 0. If a control character is to be redefined (different bit configuration) by the program, the entire control field must be transferred to the UCSTR feature.

Sync Count: These three bits (bits 5-7) indicate the number of sync characters (2-7) that the UCSTR will automatically send preceding each message block.

ETB Byte

This byte contains the bit configuration defined by the user as the ETB character. The ETB byte defined for the transmitting and receiving stations must be the same bit configuration.

EOT Byte

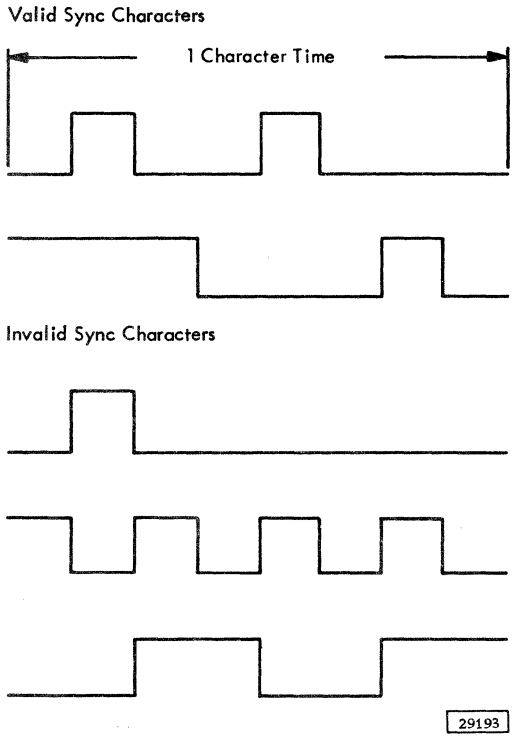
This byte contains the bit configuration defined by the user as the EOT character. The EOT character defined for the transmitting and receiving stations must be the same bit configuration.

Sync Byte

This byte contains the bit configuration defined by the user as the sync character. The sync character defined for the transmitting and receiving stations must be the same bit configuration.

The sync character selected must have four transitions from space to mark or mark to space. If the 0-bit position contains a bit, the 7-bit position must contain a no-bit. The sync character should not be a repeating pattern of

bits (e.g., 0,2,4,6, or 2,3,6,7.) The following shows an example of a valid and an invalid sync character.



LRC Subtract Field

The LRC subtract field (byte) is used in conjunction with the LRC special feature. When the UCSTR communicates with a non-IBM communications device that uses LRC checking, the UCSTR accumulates an LRC character from all characters placed on the transmission line except the beginning and ending sync characters. If the non-IBM devices excludes certain control characters from its LRC accumulation, the sum of the characters not accumulated is entered into the LRC subtract field (byte). The LRC subtract field is subtracted from the accumulated LRC in the UCSTR prior to transmitting (or receiving) the LRC character.

Synchronization

Before a message exchange can occur, the transmitting station and the receiving station must have their timing circuits synchronized with each other. The receiving UCSTR can synchronize its timing circuits with the timing circuits of the transmitting station by receiving two consecutive sync characters. The transmitting UCSTR automatically sends the number of sync characters specified in the control byte (two to seven) preceding each transmission.

Record Length

Normal System/360 Model 20 byte counting for byte length termination is not used by the UCSTR. The UCSTR depends on the ending character (ETB or EOT) to end the block transfer. In the transparent mode the ending character is a DLE-ETB or DLE-EOT. If the DLE-ETB or DLE-EOT is to be sent as data in the transparent mode, a byte count can be used to control the record length. If a byte count is used to control record length, the automatic response operation cannot be used. (See Program Instructions.) If the byte count is not used, any DLE-ETB or DLE-EOT in the message area of core storage will end the transfer. This happens only in response mode.

Maximum record length for a transmit operation is 4,095 bytes, and maximum record length for a receive operation is 1,023 bytes.

Error Checking

Transmission error detection in the UCSTR is performed by cyclic redundancy checking (CRC). This check is performed automatically during normal UCSTR operation. Both the transmitting and receiving stations develop a CRC character which is an arithmetic accumulation of each character placed on the line excluding the beginning sync characters and the ending sync character (pad). The transmitting station sends the CRC character immediately after it sends the ending character or sequence. The receiving station subtracts the CRC character from the CRC character accumulated in the receiving station. If the longitudinal redundancy check (LRC) feature is installed, the UCSTR can perform either a CRC or an LRC check under control of a switch provided by the LRC feature (See UCSTR Special Features.)

LRC or VRC checking is not required or possible when the CRC checking is used or when transparent mode is used.

A CRC or LRC error turns on a UCSTR error indicator that can be tested by a test I/O and branch (TIOB) instruction. The error indicator is turned on when the CRC (or LRC) check is made at the end of a receive operation.

If the vertical redundancy check (VRC) feature (see UCSTR Special Features) is installed, the error indicator can be turned on for a VRC error on either transmit or receive. A VRC error indicates that a character with an even number of bits was contained in the previous message. A UCSTR error condition set during a transmit operation indicates only a VRC error. A UCSTR error condition during a receive operation indicates either a VRC or LRC error (or both).

When the UCSTR is transmitting with the response bit on, the UCSTR error indicator can be set because of an error under any or all of the following conditions:

1. VRC error in the transmitted message.
2. VRC error in the response message received.
3. LRC error in the response message received.

UCSTR Time-Sharing

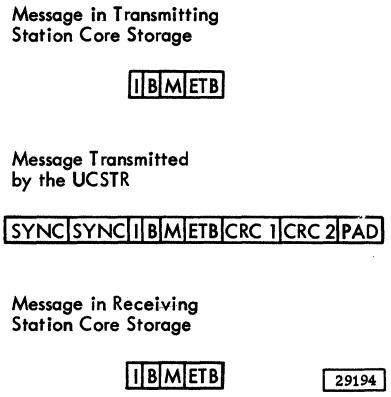
At speeds of 4,800 bits per second and below, the UCSTR time-shares its operation with all System/360 Model 20 input/output units. At speeds above 4,800 bits per second, the UCSTR time-shares its operation only with the CPU and the 1403 buffered printer.

MODES OF OPERATION

The UCSTR has two modes of operation, either of which can be used in conjunction with the automatic response mode.

Basic Mode

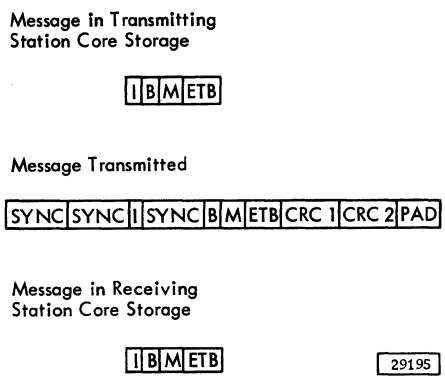
The basic mode of UCSTR operation is the full binary minus three mode. The UCSTR can send or receive all 256 possible binary combinations (characters) minus the three control characters (sync, ETB and EOT) defined by the users program. The following illustrates a message format for basic mode. If this message is the last message to be transmitted, the ETB character could be replaced by an EOT character.



The preceding illustration shows a message block consisting of three message characters and an ending character. The UCSTR precedes the message with two (up to seven) sync characters. Immediately following the ending character, the UCSTR automatically sends the 16-bit CRC character as two eight-bit characters. (In seven-bit level, the CRC character is 14 bits, and in six-bit level, the CRC character is 12 bits.) The UCSTR sends a "pad" character after the CRC character. The pad character is made up of the defined sync bit combination. The pad character is used to ensure that the last meaningful bits in a transmission are properly sent by the data set. It is not required to be received by the UCSTR.

The transmitting station can send, as data, the bit combination that has been defined as a sync character, but the receiving station does not enter the sync character in core storage. If sync characters must be sent as data, transparent mode should be used.

The capability of the UCSTR to receive but not store the sync character allows the UCSTR to communicate with equipment that sends a sync character (as an idle character) when, for any reason, it cannot send the next data character. The following illustration shows a three-character data message (with a sync character inserted) as placed on the transmission line, and it shows the portion of the message that is stored in System/360 Model 20 core storage by the UCSTR.



The receiving station response to a message depends upon the user's program. Flexibility of the UCSTR allows the following types of response:

1. A response (acknowledgment) for each message received.
2. A response message (may or may not contain an acknowledgment) for each message received.
3. After receiving several messages, sending one response message that defines any messages received in error. For specific response operations see UCSTR Operation.

Transparent Mode

Transparent mode provides the UCSTR with the capability of transmitting and receiving, as data, all 256 possible bit combinations (128 combinations in seven-bit level or 64 combinations in six-bit level).

For the control required for transparent operation, the UCSTR uses a data link escape (DLE) character. The DLE character has a fixed bit configuration of 00010000. The DLE character for seven-bit mode is 0010000, and the DLE character for six-bit mode is 010000.

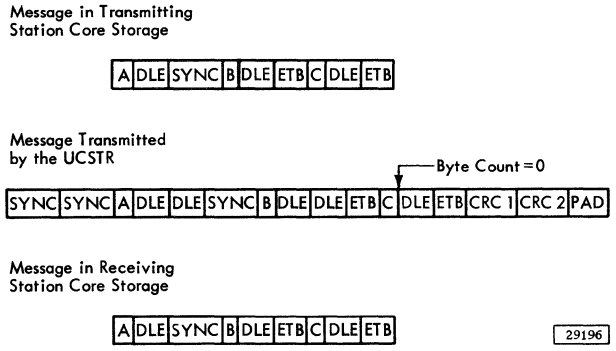
The DLE character is used in combination with the ETB or EOT to form a 16-bit ending sequence. Since the ending sequence in transparent mode is DLE-ETB or DLE-EOT, any of the 256 possible bit combinations can be transmitted individually, but an ETB or EOT character cannot be transmitted, as data, following a DLE character.

In transparent mode the transmitting station automatically inserts an extra DLE character for each DLE character in the message. The extra DLE is used to control UCSTR operations in transparent mode, but the extra DLE is not entered into core storage at the receiving station. (See Transparent without Response and Transparent with Response.)

Transparent Without Response

Transparent mode without automatic response is an important function of the UCSTR when used in applications that require transmitting programs or binary data that can possibly contain message ending characters or sequences.

Transparent mode without automatic response allows the UCSTR to send all 256 (128 or 64) bit combinations including all control characters and ending characters (sync, ETB, EOT, DLE, DLE-ETB, and DLE-EOT). This type of operation is accomplished by using a byte count in the B2-D2 fields of the XIO transmit-record instruction (See Program Instructions.) With response off, the UCSTR transfer ending operation is controlled by the byte count in the XIO transmit-record instruction. When operating in this mode, the UCSTR transmits the number of bytes specified in the byte count before checking for an ending sequence (DLE-ETB or DLE-EOT). Thus any combination of bits can be transmitted, and characters can be in any sequence. The following illustration shows a message format for transparent mode without an automatic response.



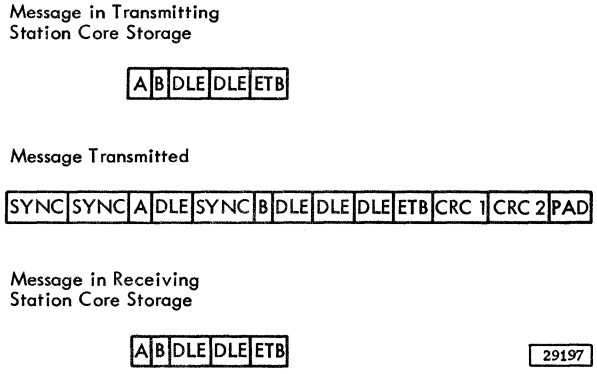
The preceding illustration shows a nine byte message in the transmitting station core storage. The beginning sync, CRC1, CRC2, and ending sync characters are transmitted in the same manner as basic mode. The byte count in the B2-D2 fields of the XIO transmit-record instruction should be seven. The transmitting station will not test for an ending sequence until the byte count has been reduced to zero. After the byte count is reduced to zero, the transmitting UCSTR starts testing each character for an ending sequence (DLE-ETB or DLE-EOT). The eighth and ninth bytes contain the ending sequence that will end the transfer operation.

The transmitting UCSTR inserts an extra DLE character for each DLE character in the message while in transparent mode. The byte count reaching zero takes the UCSTR out of full transparent mode so that the extra DLE is not inserted for the ending sequence. The preceding illustration shows that a DLE-sync combination (normally an idle sequence in transparent mode) and a DLE-ETB combination (normally an ending sequence in transparent mode) have been transmitted as data.

The receiving UCSTR does not require a byte count to control recognition of the ending sequence. The extra DLE inserted (by the transmitting station) for each DLE in the message controls the receiving station ending transfer operation. The receiving station stores the first of each pair of DLE characters. The second DLE of each pair is "stripped out" by the UCSTR. The second DLE of each pair also sets up a control in the UCSTR so that the next character (following the two DLE characters) is not tested for an idle sequence or ending sequence (sync, ETB or EOT). The transmitting station ending sequence is a single DLE (instead of a pair) followed by an ETB or EOT. The receiving station recognizes the single DLE followed by an ETB or EOT as the ending sequence.

The UCSTR recognizes a DLE-sync combination as an idle sequence and does not place this combination into core storage.

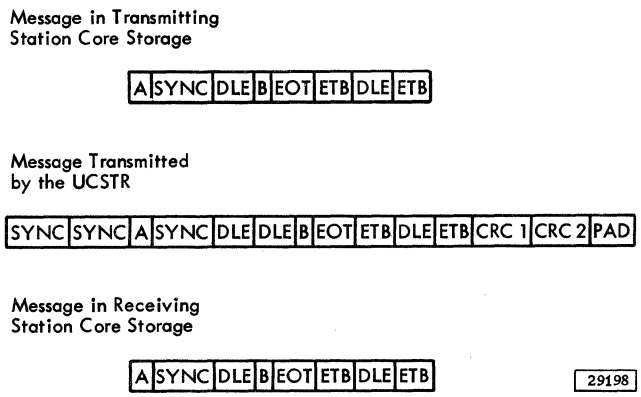
The capability of the UCSTR to recognize the DLE-sync combination as an idle sequence allows the UCSTR to communicate with equipment that sends an idle sequence when, for any reason, it cannot send the next data character. The following illustration shows a three-character data message (with an idle sequence inserted) as placed on the transmission line, and it shows the portion that is stored in System/360 Model 20 core storage by the UCSTR.



Transparent with Response

Transparent mode with automatic response operations cannot attain full transparency. Any character can be transmitted in this mode, but the 16-bit ending sequence (DLE-ETB or DLE-EOT) will end the transmission. Thus, if these combinations exist within the message, the program must search for them and split the combination before transmission. The ending character combinations (DLE-ETB or DLE-EOT) can be transmitted in transparent mode without response. (See Transparent without Response.) The transparent with response operation is similar to the transparent without response except that the transparent with response operation ends with any DLE-ETB or DLE-EOT combination rather than the first DLE-ETB or DLE-EOT after the byte count reaches zero.

The B2-D2 fields of the XIO transmit-record instruction contain the starting core storage address where the response message will be stored. The following illustration shows a message format for transparent mode with the response bit on.

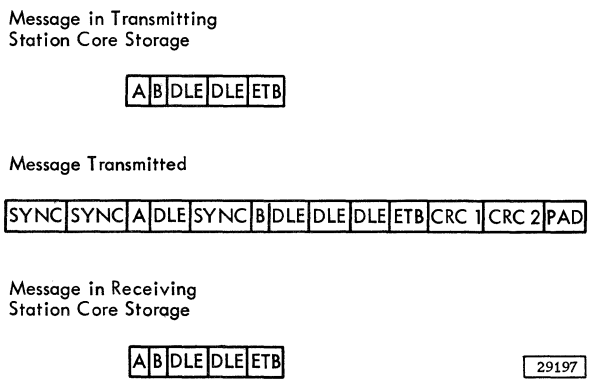


The preceding illustration shows that a sync, DLE, EOT, and ETB can be sent as individual characters, but a DLE-ETB (or DLE-EOT) combination ends the transfer. The idle sequence (DLE-sync) for transparent with response is the same as the idle sequence for transparent without response. The UCSTR receives the idle sequence but does not transfer the characters to core storage. The capability of the UCSTR to receive but not store the idle sequence allows the UCSTR to operate with equipment that sends an idle sequence when, for any reason, it cannot send the next data character. The following illustration

shows a message (with an idle sequence inserted), and it shows the portion that is stored in System/360 Model 20 core storage.

Normally, response mode allows a message to be received on a four-wire service without issuing a set-to-receive-mode instruction and receive-record instruction.

In non-response mode, there is a possibility of an I/O device interrupting before a set-to-receive-mode instruction and a receive-record instruction can be executed. To allow non-response mode to operate on a four-wire service additional sync characters can be sent by the transmitting station to allow the System/360 Model 20 to service an interrupt.



Interruption

Interruption is the general term applied to an automatic branch in the CPU program. The branch is automatic in the sense that it occurs when the condition exists, and is independent of a programmed branch instruction. In the System/360 Model 20, interruption is provided only for the channel end input/output condition. Channel end condition of an input/output device is defined as the time (in the mechanical cycle of the device) at which the data transfer has been completed.

In the time-shared mode of operation of the CPU and the input/output devices, the transfer of data between the input/output devices and main storage in the CPU is asynchronous with respect to processing operations. Thus, the channel end condition of an input/output data transfer operation may occur at any time in the instruction or execution phase of any processing operation. It is also possible that more than one input/output device reaches the channel end condition during a specific processing operation. The channel end conditions are stored in the form of interrupt bits. These bits are reset when the respective interrupts occur or are reset by a system reset or load operation.

The CPU is in an interruptible state when the channel mask bit in the program status word (PSW) is 1, and is in a non-interruptible state when this bit is 0. The channel mask bit is reset to 0 by operation of the system reset or program load key. The channel mask bit may be altered by a set PSW instruction or by an interruption.

When the CPU is in the interruptible state (channel mask bit is 1), a test is performed by the CPU at the end of each processing operation to determine whether or not a channel end condition exists for any input/output device. (The instruction and start phases of input/output operations are also considered as processing operations.)

The test for a channel end condition is performed in an established priority sequence. The program continues with the next sequential instruction if no channel end conditions exist. When one or more end conditions exist, the first one encountered in the priority sequence causes an interruption to occur. The channel end condition which causes the interruption is reset. The interruption is performed by storing the PSW in fixed main-storage location 144 and obtaining a new PSW from another fixed main-storage location (148) before the program continues. Since the new PSW contains the address of the next sequential instruction, the interrupt is equivalent to a branch operation. The PSW stored in main storage location 144 is referred to as the old PSW.

The old PSW (stored at main storage location 144 when an interrupt occurs) contains the device address of the I/O device which caused the interruption (bits 8-11), the primary function which the device was performing (bits 12-15), the address of the next sequential instruction, and the condition code. Primary functions are read, punch, or print; not included are details such as which feed was in use for a read operation on the MFCM.

The following is a list of device addresses and function specifications contained in bits 8 to 15 of the old PSW stored at main-storage location 144 when an interruption due to the corresponding channel end condition occurs. The list is in the priority sequence for these interruptions.

Channel End Condition	DA	FS
1419 Read	0110	0110
2501 Read Card	0001	0010
2520 or 2560 Read Card	0010	0010
1403 or 2203 Print	0100	0000
UCSTR Receive or Transmit	0101	0110
2560 Punch Card	0010	0100
1442 Punch Card	0011	0100
2560 Write Card	0010	0000
2520 Punch Card	0010	0100
Input/Output Channel	0111	0000

The channel mask bit in the new PSW (obtained from main-storage location 148) may be used to disable further interruptions in the routine which begins at the next sequential instruction address specified in the new PSW. As a means of returning to the point in the program at which the interruption occurred, a set PSW instruction, in which the specified address is 144, may be used as the last instruction in the routine.

UCSTR Interrupt Conditions

The following functions cause a UCSTR interrupt condition:

1. An end condition when an XIO transmit-record instruction has been executed with the response bit off.
2. An end condition when an XIO receive-record instruction has been executed.
3. An end condition when a response-message ending character or sequence is received as a result of an XIO transmit-record instruction being executed with the response bit on. (The interrupt condition is not set after the transmit operation if the response bit is on.)
4. A response time-out has occurred. The response time-out interrupt occurs when a receive-record instruction has been executed but no data is received within three seconds, or a transmit-record instruction has been executed (with the response bit on) but no response is received within three seconds.

PROGRAM INSTRUCTIONS

Transfers of information to main storage from the UCSTR and from main storage to the UCSTR are referred to as input/output operations. There are three types of instructions for input/output operations: transfer, control, and test I/O and branch instructions.

A transfer instruction (XIO) controls the transfer of data between main storage and the UCSTR.

A control instruction (CIO) directs the UCSTR to perform a specified function; e.g., set-to-transmit, or set-to-receive.

A test I/O and branch instruction (TIOB) causes an inquiry to the UCSTR for a particular condition (e.g., busy, response time-out, etc); if the tested indicator is on, the program branches to the specified address.

If the time-sharing switch is on, processing operations in the CPU are time-shared with the transfer of data between main storage and the input/output devices. When an input/output device requests service, processing is suspended only for the time required to send or accept the input/output data. See UCSTR time-sharing.

Data Format

Input/output data is located in eight-bit bytes in main storage in variable-length fields. Data for the seven-bit level operation is positioned in the low-order seven bits (1-7) of each core storage byte. Data for the six-bit level operation is positioned in the low order six bits (2-7) of the core storage byte. Input/output data may be in the zoned, binary, or packed format.

No code translation is required for UCSTR operation. The UCSTR transmits the bit configuration contained each byte of the transfer field, and when receiving; the transfer field receives the same bit configuration that is received by the UCSTR.

Condition Code

The status of an I/O device addressed by a transfer I/O instruction, and under certain conditions, a control I/O instruction, is used to set the condition

code of the PSW at the time the execution of the instruction is completed. The condition code indicates whether or not the I/O device has initiated the operation specified, and if not, the reason for the rejection. The condition code can be used for decision-making by subsequent branching operations.

The UCSTR sets a condition code only when an XIO instruction is issued. The condition code is set to 00,01, 10, or 11 by an instruction to indicate the status of the I/O device addressed.

Condition Code	Status
00	Available (A)
01	Working (W)
10	High-Speed UCSTR
11	Not Operational (N)

Available: Indicates that the addressed I/O device is operational, does not contain data or error check conditions, and is not busy with a previously initiated operation.

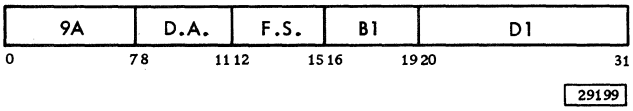
Working: Indicates that the addressed I/O device is executing a previously initiated operation.

High-Speed UCSTR: Indicates that the high-speed UCSTR feature is installed and operating when another I/O device is addressed, or it indicates that an I/O device is operating when the UCSTR is addressed. High-speed UCSTR time-shares only with the 1403 buffered printer; thus, the 1403 is the only device that can operate at the same time as the high-speed UCSTR.

Not Operational: Indicates that the addressed I/O device is in a not ready status, or an error or a data check condition exists on the device.

The operation specified by an XIO instruction is initiated only when the addressed I/O device is in the available state. If an I/O device which is not a part of the system is specified, a no-operation occurs, and the condition code is not changed.

Test I/O and Branch



The device address (DA) specifies the I/O device in which a condition is to be tested.

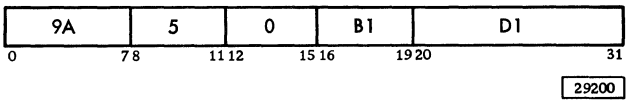
The function specification (FS) specifies the particular condition or indicator to be tested in the I/O device addressed.

If the condition tested in the addressed I/O device is on, the updated instruction address is replaced by the branch address derived from the B1-D1 fields; otherwise, normal instruction sequencing continues with the updated instruction address.

UCSTR Test I/O and Branch Instructions

UCSTR Busy

The UCSTR busy TIOB instruction is available after a transmit or receive record XIO instruction has been issued. If the UCSTR feature is in a busy condition when the TIOB instruction is issued, the updated instruction address is replaced by the branch address in the B1-D1 field of the TIOB instruction. If the UCSTR feature is not in a busy condition when the TIOB instruction is issued, normal instruction sequencing proceeds.

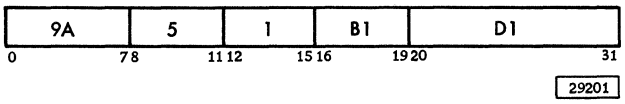


The UCSTR becomes busy when the transmit-record or receive-record XIO instruction is executed, and the UCSTR remains busy until an end-condition interrupt occurs, a response time-out interrupt occurs, or a nullify transfer instruction is executed.

CRC or LRC Error

The UCSTR error TIOB instruction is available after a receive record XIO is issued, or it is available after a transmit record XIO with the response bit (see Control Word) on is issued. If the UCSTR feature is in a busy condition when the TIOB instruction is issued, the CPU is interlocked, and the CPU performs the UCSTR error test after the UCSTR interrupt condition is set.

If the UCSTR error condition exists at the time the test is performed, the updated instruction address is replaced by the branch address in the B1-D1 fields of the TIOB instruction. If no error exists when the test is performed, normal program sequencing continues.



A UCSTR error condition indicates that an error occurred during the preceding receive operation. The error is detected by the CRC, LRC or VRC check. (See Error Checking.)

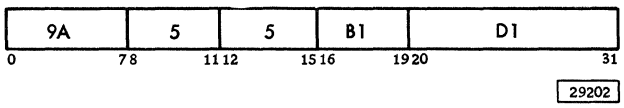
If the vertical redundancy check (VRC) feature is installed, the UCSTR error TIOB instruction is available after a receive or transmit XIO instruction (See Error Checking.)

The UCSTR error indicator is reset when a receive-record or transmit-record XIO instruction is executed.

Receive EOT

The receive EOT TIOB instruction is available after a UCSTR interrupt condition is set. The indicator is turned on when the UCSTR receives an EOT character (DLE-EOT in transparent mode).

If this indicator is on when tested, the updated instruction address is replaced by the branch address in the B1-D1 fields of the TIOB instruction. If the indicator is not on, normal instruction sequencing proceeds with the updated address.



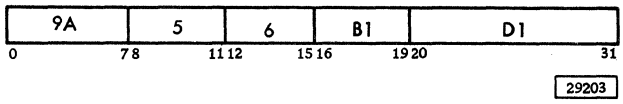
The receive EOT indicator is reset when it is tested by a TIOB instruction. If the receive EOT indicator is not tested by the program, the EOT indicator light will turn on and the audible alarm will sound until the operator presses the stop key.

Response Time-Out

The response time-out TIOB instruction is available after a UCSTR interrupt condition is set. The response time-out indicator is turned on by the following conditions.

1. A receive-record XIO instruction has been executed, but no message block has been received within three seconds.
2. A transmit-record instruction is executed with the response bit on (see Control Byte), but no acknowledgment or message has been received within three seconds of sending the ending character or sequence.

If the response time-out indicator is on when tested by a TIOB instruction, the updated instruction address is replaced by the branch address in the B1-D1 field of the TIOB instruction. If the indicator is off, normal instruction sequencing proceeds with the updated instruction address.



The response time-out indicator is turned off when one of the following occurs.

1. When the indicator is tested by a TIOB instruction.
2. When a message block or response message has been received.

Ring

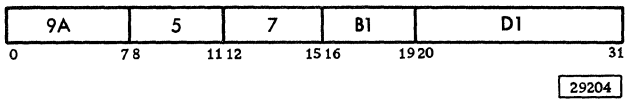
The ring indicator indicates that a ringing signal is being received from a remote terminal. The ring TIOB instruction is available after a set-to-receive-mode CIO instruction, but prior to issuing a receive-record XIO instruction.

The ring indicator is used in conjunction with a data set that has the unattended answer feature installed.

The indicator is turned on by a ring signal from the data set when the UCSTR auto answer/disconnect switch is in the on position, the UCSTR start key has been pressed, and a set-to-receive mode CIO instruction has been executed.

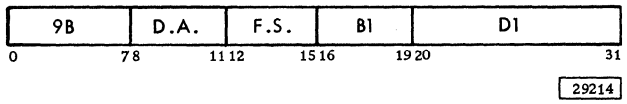
The ring indicator being on indicates that a remote station is trying to call.

If the ring indicator is on when it is tested by a TIOB instruction, the updated instruction address is replaced by the branch address in the B1-D1 field of the TIOB instruction.



The ring indicator is turned off by an automatic disconnect CIO instruction.

Control Input/Output Instructions



The device address (DA) specifies the I/O device in which a control function is to be performed.

The function specification (FS) specifies the particular component (it may also specify the primary function of that component) in the I/O device addressed.

A detailed specification of the control function to be performed is derived from the contents of the B1-D1 fields, according to the rules for direct or effective address generation. If the detailed specification derived from the B1-D1 field is all zero, a no-operation occurs.

The B1-D1 field of the set-to-receive and set-to-transmit instructions contains the address of the core storage location that contains the first byte of the five-byte control field.

The B1-D1 field is not used during the other UCSTR control I/O instructions, but the B1-D1 fields must contain some bits to prevent a program error condition.

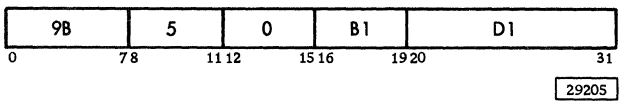
None of the UCSTR control I/O instructions will cause the UCSTR to be in a busy condition.

UCSTR Control I/O

The following control input/output (CIO) instructions are used by the UCSTR.

Set to Receive Mode

The set-to-receive-mode instruction conditions the UCSTR for a receive operation.



This instruction is issued prior to an XIO receive-record instruction to set the direction of transmission for the receiving UCSTR.

The UCSTR remains in receive mode until a set-to-transmit-mode instruction is executed. Thus, the set-to-receive-mode instruction is issued only when needed to set the initial direction of transmission, when the direction of transmission is to be changed from transmit to receive, or when a new control field byte is to be defined during a continuous receive mode operation. A continuous receive mode operation is when the UCSTR is receiving multiple message blocks without sending any response messages.

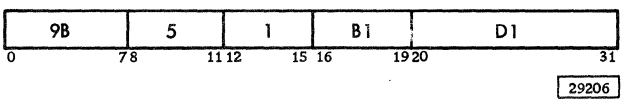
A data transfer will not occur until an XIO receive-record instruction is executed. (See Transfer Input/Output Instruction.)

The set-to-receive-mode instruction is available under the following conditions.

1. Prior to the execution of an XIO receive-record instruction to set the initial direction of transmission for the receiving UCSTR.
2. After the end-condition interrupt of an XIO transmit-record instruction to set the direction of transmission for receiving a response or message in reply to the message transmitted.
3. After the end-condition interrupt of an XIO receive-record instruction when defining a new control field byte (sync, ETB, EOT, or LRC subtract field) during a continuous receive operation.

Set to Transmit Mode

The set-to-transmit-mode instruction conditions the UCSTR for a transmit operation.



This instruction is issued prior to an XIO transmit-record instruction to set the direction of transmission for the transmitting UCSTR.

The UCSTR remains in transmit mode until a set-to-receive-mode instruction is executed, or a transmit-record operation has been completed with the response bit on. Thus, the set-to-transmit-mode instruction is issued only when needed to set the initial direction of transmission, when the direction of transmission is to be changed from receive to transmit, when the response bit is on and the UCSTR is ready to send the next message after receiving a response, or when a new control field byte is to be defined during a continuous transmit mode operation. A continuous transmit operation is an operation in which the UCSTR is transmitting multiple message blocks without receiving any responses or messages.

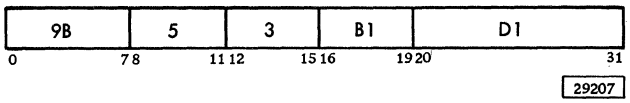
A data transfer will not occur until an XIO transmit-record instruction is executed. (See Transfer Input/Output Instruction.)

The set-to-transmit-mode instruction is available under the following conditions.

1. Prior to the execution of an XIO transmit-record instruction to set the initial direction of transmission for the transmitting UCSTR.
2. After the end-condition interrupt of an XIO receive-record instruction to set the direction to send the next message after receiving a response or message in reply to the last message transmitted.
3. After the end-condition interrupt of an XIO transmit-record instruction when defining a new control field byte (sync, ETB, EOT or LRC subtract field) during a continuous transmit operation.
4. After the end-condition interrupt of an XIO transmit-record instruction with the response bit on. (The end-condition interrupt occurs after the response is received.)

Inhibit Audible Alarm

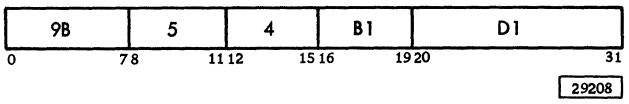
The inhibit-audible-alarm instruction suppresses the three-second processor timer. If this timer is allowed to time out, the audible alarm will sound. The timer is started by pressing the UCSTR start key, or by a not busy condition. (Not busy exists after an end-condition interrupt following a previous XIO.) The timer resets with an XIO transfer instruction.



The inhibit-audible-alarm instruction is available prior to issuing a UCSTR transmit or receive-record instruction. The suppression is cancelled when the next UCSTR transmit or receive-record instruction is executed.

Nullify Transfer

The nullify-transfer instruction cancels a previously issued XIO transfer instruction and turns off the busy condition TIOB indicator.

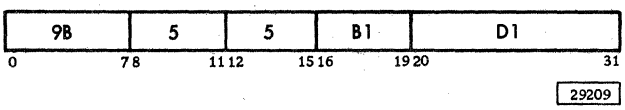


This instruction provides a means to nullify a previous XIO receive-record instruction. If the receiving station has previously executed an inhibit-time-out-response instruction and subsequently issues an XIO receive-record instruction, there is a possibility of a "hang-up" in an executed receive-record instruction. This hang-up may be caused by a communications line disconnect or the failure of the transmitting station to send a message. Since the UCSTR would not receive an ending character or sequence, an end-condition interrupt would not occur. After a reasonable length of time the program should nullify the receive-record instruction.

No UCSTR interrupt follows the nullify-transfer instruction.

Set Audible Alarm

The audible alarm automatically sounds when operator attention is required by the CPU because of a UCSTR operation that cannot take care of itself. The set-audible-alarm instruction provides for the program to sound the audible alarm when operator attention is required.

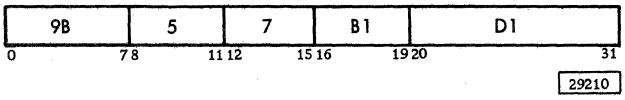


The audible alarm is turned off by pressing the UCSTR stop key, or if the alarm is sounded because of a processor condition, it can be turned off by executing a transfer instruction.

Automatic Disconnect

If the unattended-answer feature is installed on the data set being used with the UCSTR, the execution of the automatic-disconnect instruction will terminate the data call.

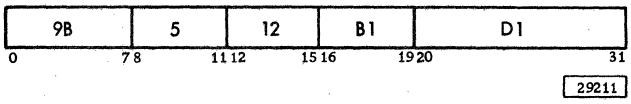
The automatic-disconnect instruction is available anytime the UCSTR is not in a busy condition.



Inhibit Response Time-Out

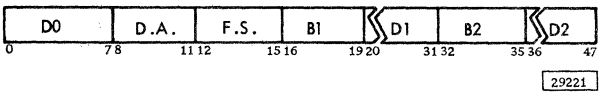
The inhibit-response-time-out instruction suppresses the automatic time-out that occurs when a message or response has not been received within three seconds after:

1. Executing an XIO receive-record instruction.
2. "Turning around" to receive a response message after transmitting a message with the response bit on.



The suppression of the response time-out is cancelled when the next transmit- or receive-record transfer instruction is executed.

Transfer Input/Output Instructions



The device address (DA) specifies the I/O device to which output data is to be transmitted, or from which input data is to be received.

The function specification (FS) specifies the input or output function to be performed on the I/O device addressed, and also the particular component of the addressed device (when required).

The main-storage location of the first byte in the input or output data field is derived from the contents of the B1-D1 fields according to the rules for direct or effective address generation.

THE B2-D2 fields of the XIO instruction are used as described in Transmit Record and Receive Record.

THE UCSTR becomes busy (UCSTR busy indicator is on) when the XIO instruction is executed, and the busy indicator remains on until an end-condition or a response time-out condition occurs.

UCSTR Transfer Input/Output Instructions

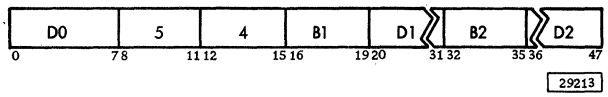
Two transfer input/output (XIO) instructions are used by the UCSTR.

Transmit Record

The transmit-record instruction initiates the transfer of a message from core storage to the UCSTR. After the operation is initiated by the transmit-record instruction, the actual transfer of data is done, one byte at a time, on a time-sharing basis with other I/O devices.

The UCSTR is available for a transmit-record instruction when:

1. A set-to-transmit-mode instruction has been previously issued.
2. A transmit-record instruction has previously been executed with the response bit off and a UCSTR end-condition interrupt has occurred.



The UCSTR is not available for a transfer instruction when the UCSTR busy indicator is on. The B1-D1 fields contain the core storage address of the first byte of the message to be transmitted.

The B2-D2 fields are used in the following ways:

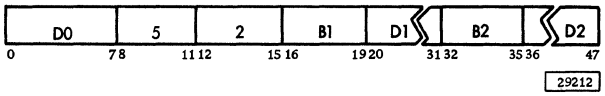
1. For transmit-record operations with the response bit on, the B2-D2 fields contain the core storage address where the first byte of the response message will be stored.
2. For transmit-record operations with the response bit off and the transparent bit off, the B2-D2 fields are not used, but they must contain some bits to prevent a program error.
3. For transmit-record operations with the response bit off and the transparent bit on, the B2-D2 fields contain the transparent byte count (See Transparent without Response.)
4. For transmit-record operations with the response bit on and the transparent bit on, the B2-D2 fields contain the core storage address where the first byte of the response message will be stored.

Receive Record

The receive-record instruction initiates the transfer of a message from the UCSTR to core storage. After the operation is initiated by the receive-record instruction, the actual transfer is done, one byte at a time, on a time-sharing basis with other I/O devices.

The UCSTR is available for a receive-record instruction when:

1. A set-to-receive-mode instruction has previously been issued.
2. A receive-record instruction has previously been executed and a UCSTR end condition interrupt has occurred.
3. A response time-out has occurred after the execution of a previous receive-record instruction.
4. A receive-record instruction has previously been executed and subsequently nullified by a nullify-transfer instruction.



The UCSTR is not available for a transfer instruction when the UCSTR busy indicator is on.

The B1-D1 fields contain the core storage address where the first byte of the message will be stored. The B2-D2 fields are not used for receive record operations, but these fields must contain some bits to prevent a program error.

KEYS AND SWITCHES

The following is a description of the keys and switches used by the UCSTR. Figure 5 illustrates the UCSTR keys, lights, and switches.

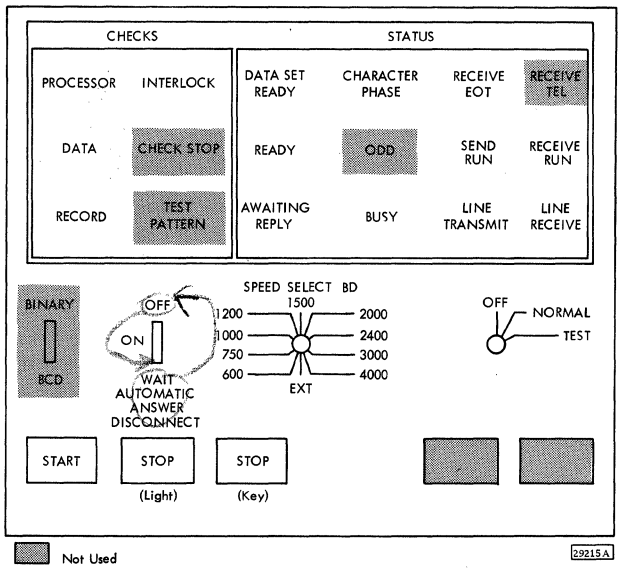


Figure 5. Keys, Lights, and Switches

Keys

Start: Pressing the start key places the UCSTR in a ready status. Pressing the start key will also turn off the audible alarm if the alarm was sounded as a result of a transmit-record or receive-record instruction being executed prior to pressing the start key.

Stop: Pressing the stop key removes the UCSTR from a ready status. If a transfer operation is in progress when the stop key is pressed it will be completed. Pressing the stop key will also turn off the audible alarm if the alarm was sounded as a result of one of the following conditions:

1. A set-audible-alarm instruction.
2. The data set is not ready.
3. An EOT has been received as the ending character of a received message block, but the receive EOT indicator was not tested by a TIOB instruction.
4. An interlock check stop.

Switches

Operation: The operation switch has three positions, which function as follows:

1. OFF - places the UCSTR in an "out-of-use" mode and resets UCSTR functional conditions.
2. NORMAL-places the UCSTR in an "in use" mode.
3. TEST-is a service feature of the UCSTR. The test position is used by the IBM Customer Engineer with a special diagnostic program routine to check UCSTR operation.

Speed Select: This switch is used to set the UCSTR to the proper transmission speed. The switch has a position for 600, 750, 1000, 1,200, 1,500, 2,000, 2,400, 3,000, and 4,000 bits per second (baud). For speeds in excess of 4,000 baud or where the data-set provides the clock, the speed select switch has an external (EXT) position. In the EXT position, the data set must provide the timing pulses to the UCSTR.

The basic UCSTR can operate at speeds up to ~~4,800~~ baud. 7200 bps

For speeds greater than ~~4,800 baud~~ 7200 bps (up to 50,000 baud) the high-speed UCSTR special feature is required.

The character transmission rate depends upon the selected speed and the line level code used. For example, when transmitting at 2,000 baud, the character transmission rate is as follows:

1. Eight-bit line level: 250 characters per second.
2. Seven-bit line level: 285 characters per second.
3. Six-bit line level: 333 characters per second.

NOTE: If data-set clocking (external) is used by one station, the other must also use data-set clocking. The same transmission speed must be selected for both (all) stations.

Auto-Answer/Wait: This switch has three positions that function as follows:

1. OFF - Disables the auto-answer-disconnect feature of the UCSTR.
2. ON - With the switch in this position, automatic answering of incoming calls and automatic disconnection of the UCSTR from the communications line can be under program control.
3. WAIT - With the switch in the wait position, the operation is the same as when the switch is in the on position except the program does not continue with the next sequential instruction after executing a receive-record instruction.

Binary/BCD: Not Used

INDICATOR LIGHTS

The following describes the status and check indicators used by the UCSTR. Figure 6 illustrates the status and check indicators on the UCSTR operator panel.

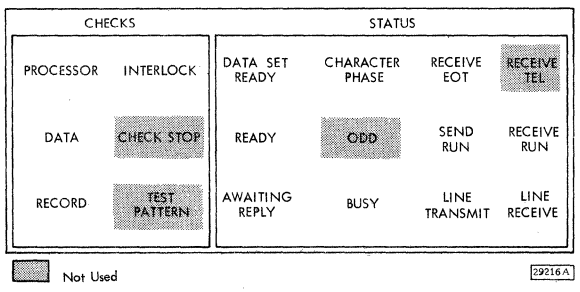


Figure 6. Status and Check Indicators.

Status Indicators

Data Set Ready: When this indicator is on, it indicates:

1. Power to the data set is on.
2. The communications line is connected.

On some machines, the data set ready indicator may be on when the operation switch is in the test position.

Character Phase: The character phase indicator being on indicates that a synchronized condition exists between the receiving and transmitting stations. Character phase is established for each individual message. Character phase is established at the beginning of a message, and it is turned off at the end of a message.

In a multipoint operation or if a message is being received but a receive-record instruction has not been executed, a momentary character phase condition can occur without the receiving station executing a receive-record instruction. If a receiving station receives two consecutive sync characters, the character phase indicator is turned on, but it remains on only as long as the UCSTR is receiving consecutive sync characters. (See UCSTR Special Features.)

When the character phase indicator is on it indicates the following conditions:

1. In a transmit operation, it indicates that an XIO transmit-record instruction has been executed and the UCSTR is transferring sync characters, a message block, and the CRC or LRC character to the remote station. The indicator turns on when the transmit-record instruction is executed, and it turns off after the sync (pad) character is transmitted at the end of the message.
2. In a receive operation, the character phase indicator turns on when a receive-record instruction has been executed and two (or more) consecutive sync characters have been received by the UCSTR. The two consecutive sync characters must be recognized by the UCSTR as the same sync character defined in its control field.

The character phase indicator being on indicates that the receiving UCSTR timing circuits are in synchronism with the timing circuits of the transmitting station.

The character phase indicator turns off after the UCSTR receives the CRC character at the end of a message.

Awaiting Reply: The awaiting-reply indicator being on indicates that the UCSTR is awaiting a response or a message because of one of the following conditions:

1. A receive-record instruction has been executed, and the UCSTR is waiting to receive a message from the remote station.
2. A transmit-record instruction has been executed with the response bit on. After the message block is transmitted the UCSTR automatically turns around (switches to receive) to await a response or message.

The awaiting-reply indicator is turned off when:

1. The UCSTR goes into synchronization (character phase) with the remote transmitting station.
2. A response time-out has occurred.
3. A nullify-transfer instruction has been executed.

Ready: When the ready indicator is on, it indicates:

1. Power is being supplied to the System/360 Model 20.
2. The data set is in a ready condition.
3. The UCSTR start key has been pressed.

The ready indicator is turned off when the stop key is pressed, but the transfer operation that is in progress is completed.

Busy: The busy indicator being on indicates:

1. A transmit-record instruction has been executed, but the transfer of data is not complete.
2. A receive-record instruction has been executed, but the transfer of data is not complete.

The busy indicator is turned off under the following conditions:

1. A receive message transfer is completed.
2. A transmit message transfer is completed.
3. A time-out response occurs.
4. A nullify-transfer instruction is executed.

Receive EOT: The receive EOT indicator being on indicates that an end-of-transmission ending character (EOT) or ending sequence (DLE-EOT) has been received.

The receive EOT indicator is turned off by either executing a test-receive EOT TIOB instruction or by pressing the UCSTR stop key.

Send Run: The send-run indicator being on indicates that the UCSTR is transmitting a message block to a remote station.

The send-run indicator is turned off when the message transfer is complete.

Receive Run: The receive-run indicator being on indicates that the UCSTR is receiving a message block from a remote station. The receive-run indicator is turned off when the message transfer is complete.

Line Transmit: The line-transmit indicator being on indicates that the data set is conditioned to send. If the half-duplex/full-duplex switch on the CE panel is in the FD (full-duplex) position, the line-transmit indicator remains on. With the switch in the HD (half-duplex) position, the indicator will be on only when the data set is conditioned to send.

Line Receive: The line-receive indicator being on indicates that the data set is conditioned to receive. If the half-duplex/full-duplex switch on the CE panel is in the FD (full-duplex) position, the line receive indicator remains on. With the switch in the HD (half-duplex) position, the indicator remains on only when the data set is not conditioned to send.

Odd: Not used.

Receive TEL: Not used.

Check Indicators

The check indicators are shown in Figure 6. A summary of check conditions and the indicators turned on by each condition is provided by Figure 7.

Problem	Indicators				Action Required to Cancel the Indicator	Interrupt Bit Set
	Processor	Interlock	Data	Record		
Failed to Issue an XIO Instruction within 3 seconds	X				Issue XIO Instruction or Press Stop Key	No
Storage Wrap-Around	X	X			Press System Reset or Press UCSTR Reset	Yes
Receive CRC or LRC Error			X	X	Test Receive Error TIOB Indicator	Yes
Program Stop Error		X			Press System Reset or Press UCSTR Reset	No
UCSTR Address Boundry Overrun				X	Test Receive Error TIOB Indicator	Yes

29217

Figure 7. Check Indicator Summary

The audible alarm sounds when any check indicator or combination of check indicators is on.

The alarm will sound for the period of time the indicator is on.

The audible alarm will stop sounding when the check indicator is reset or when the stop key is pressed.

Processor: The processor indicator being on by itself indicates one of the following:

1. The UCSTR start key has been pressed, but a transmit or receive record instruction has not been executed within three seconds of pressing the start key.
2. A transmit or receive record instruction has been executed, and an interrupt bit has been set.

   This condition can be the result of the completion of a transmit or receive operation, or it can be the result of a response time-out, but a subsequent transmit or receive record instruction has not been executed within three seconds.

For the above conditions, the processor indicator can be reset by executing a transmit or receive record instruction or by pressing the stop key.

The processor and interlock indicators being on indicates a storage wraparound error after the execution of the previous receive of transmit record operation. This is a programming error stop.

The processor and interlock indicators are turned off by pressing the system reset key on the CPU console or by switching the UCSTR operation switch to the off position (Figure 5).

Interlock: The interlock indicator being on indicates a program error stop condition. The condition that causes the processor and interlock indicators to turn on was described previously.

The interlock indicator being on by itself indicates one of the following conditions:

1. A receive-record instruction is encountered by the program while the UCSTR is in a send condition. (The direction of transmission was not changed by a set-to-receive-mode instruction.)
2. A transmit-record instruction is encountered by the program while the UCSTR is in a receive condition. (The direction of transmission was not changed by a set-to-transmit-mode instruction).
3. A set-to-transmit-mode or set-to-receive-mode instruction is encountered by the program while the UCSTR is busy executing a previous transmit-record or receive-record operation.

When a programming error stop occurs (interlock indicator on), the I-register display on the CPU console indicates 0000 (the same as a normal CPU stop condition).

The interlock indicator is turned off by pressing the system reset key or by switching the UCSTR operation switch to the off position.

Data: The data and record indicators being on indicates a receive CRC or LRC error occurred in the last message received.

The data and record indicators are turned off when the receive record TIOB indicator is tested by a receive error TIOB instruction.

Record: The record indicator being on by itself indicates that a storage address boundary overrun has occurred because of one of the following conditions:

1. An ending character or sequence was not recognized by the UCSTR (the UCSTR continues to store data characters until a storage address boundary is reached).
2. The message received exceeded the maximum field length (1,023 bytes) for a receive field.

Check Stop: Not used

Test Pattern: Not used

UCSTR SPECIAL FEATURES

The UCSTR feature and its associated special features are available on a request price quotation (RPQ) basis from IBM.

This section provides a description of the special features associated with the UCSTR.

High Speed (RPQ M24418)

This feature modifies the UCSTR to operate with broadband communications facilities at speeds up to 50,000 bits per second (full duplex).

The high-speed UCSTR time shares with only the CPU and the 1403 buffered printer. To accomplish the high-speed UCSTR time sharing function, an additional condition code 10 is used when the high-speed feature is installed.

If an I/O device (other than the 1403) is operating (has not yet reached an end condition) when the high-speed UCSTR is addressed by an XIO or CIO instruction, a condition code 10 is set in the UCSTR program status word (PSW). The UCSTR XIO or CIO instruction will not be executed until the other I/O device reaches an end condition.

If the high-speed UCSTR is operating (has not reached an end condition) when an I/O device (other than the 1403) is addressed by an XIO or CIO instruction, a condition code 10 is set in that device program status word. The device XIO or CIO instruction will not be executed

until the high-speed UCSTR reaches an end condition. The high-speed UCSTR cannot be operated at normal medium speeds of 75 to 600 characters per second.

The high-speed UCSTR does not provide the high-speed clocking pulses required for synchronization. When the high-speed feature is installed, the UCSTR must operate with a data set that provides the clocking pulses.

Six- and Seven-Bit Level Code (RPQ M24421)

This feature provides the UCSTR with the ability to transmit or receive any transmission code comprised of six bits or any transmission code comprised of seven bits as well as the basic UCSTR capability of transmitting or receiving any transmission code comprised of eight bits. The six-, seven-, or eight-bit level is selected by means of a switch.

This feature provides CRC checking for six-bit and seven-bit codes. With the addition of the LRC feature, the LRC check can also accomodate six- and seven-bit codes.

A character in a six- or seven-bit code is stored in the low-order six bits (2-7) or seven bits (1-7) of a core storage byte. The characters must be stored in the bit configuration that is to be transmitted. (No code translation is provided by the UCSTR.)

Sync characters defined by the program in the control field must be in the same level code that is being transmitted or received. Sync characters must meet the requirements of the basic UCSTR sync character. (See Sync Byte).

LRC Check (RPQ M25019)

This feature provides the UCSTR with the ability to perform a longitudinal redundancy check (LRC) on the data received. Either CRC or LRC checking can be selected by a switch provided with the LRC feature.

The LRC character is transmitted as a single character in six-, seven-, or eight-bit codes. See error checking for a description of the checking operation.

The UCSTR accumulates the LRC character from all characters placed on the transmission line except the beginning and ending sync characters.

An LRC subtract field (byte) is provided by the LRC feature for operation with equipment that excludes some of its control characters from the LRC accumulation.

When such equipment is operating with the UCSTR, the accumulation, for those characters that are not accumulated by the LRC, must be placed in the LRC subtract field to adjust the LRC accumulation in the UCSTR before the LRC check is made.

VRC Check (RPQ 834533)

This feature provides the UCSTR with the ability to check for an odd number of bits in each character. If the UCSTR with the VRC check feature senses an even number of bits in a character, the error TIOB indicator is turned on. (See Test Input/Output and Branch Instructions.)

VRC checking is controlled by a switch that is provided by the VRC feature.

Terminal Address Recognition (RPQ M34070)

This feature provides the UCSTR with the capability of being selected by an addressing (polling) operation performed by a master station. The terminal address recognition feature is used in a multidrop configuration (multiple System/360 Model 20 systems with the UCSTR feature).

Each system ordered with this feature installed is wired at the factory to recognize a unique eight-bit address that is specified by the customer. Selection of a particular station is accomplished by the master station transmitting the eight-bit code assigned to the desired station as the first data byte of a message block.

Each UCSTR in the multidrop configuration executes a set-to-receive-mode (CIO) instruction and a receive-record (XIO) instruction, and each UCSTR monitors every message that appears on the communications line. The sync characters transmitted prior to each message will cause all the UCSTR features to become synchronized with the master station.

Each UCSTR tests the first character after a sync character to determine if the first byte of the message contains its terminal address. The terminal with the address corresponding to the address in the first data byte is selected and receives the remainder of the message block. All other terminals connected to the communications line ignore the remainder of the message block.

The terminals that are not selected do not interrupt, but they continue to monitor the line for the next address selection operation. The terminal that

was selected receives the message in the normal manner, interrupting after receiving the ending character or sequence.

The terminal address recognition feature operates in eight-bit level code only.

Since the terminal address recognition feature tests the first character after a sync character for an address, a sync character within a data message can cause a false terminal selection. A false terminal selection can be prevented by a line-control procedure that begins each message with a unique start-of-message character.

If an inhibit-response-time-out instruction is not executed prior to issuing a receive-record instruction, a response time-out will occur at each terminal not selected within three seconds.

Special Feature Summary

The above special features and their prerequisites are summarized in Figure 8.

IBM SYSTEM/360 MODEL 20 INPUT/OUTPUT UNITS

IBM 1442 CARD PUNCH, MODEL 5

The IBM 1442 Card Punch, Model 5 provides punched card output for the System/360 Model 20. The 1442-5 consists of a card hopper, a serial punch station, and one radial stacker. Card punching is done serially at a maximum rate of 160 columns/second. The card punching rate depends upon the number of columns specified in the punch instruction. Punching speed may range from 265 cards/minute for punching columns 1-10, to 91 cards/minute for punching columns 1-80.

IBM 2501 CARD READER, MODELS A1 and A2

The 2501 Card Reader consists of one hopper, a serial read station, and one stacker.

Feature	RPQ Number	Description	Prerequisites
Basic UCSTR	M24798	Universal Code Synchronous Transmit-Receive that can transmit and receive in any eight-bit line code.	System/360 Model 20 Communications Adapter (FC 2073)
High-Speed	M24418	Allows the UCSTR to transmit or receive at speeds up to 50,000 bits per second (baud).	1. FC 2073 2. M24798
Six and Seven bit level code	M24421	Allows the UCSTR to transmit and receive in any six or seven bit code.	1. FC 2073 2. M24798
LRC Check	M25019	Allows the UCSTR to communicate with equipment that accumulates a longitudinal redundancy check (LRC) character instead of a cyclic redundancy check (CRC) character.	1. FC 2073 2. M24798
VRC Check	834533	Allows the UCSTR to check each character for an even parity error condition.	1. FC 2073 2. M24798 3. M25019
Terminal Address Recognition	M34070	Allows the UCSTR to automatically select one of several terminals in a multi-drop configuration.	1. FC 2073 2. M24798

29218

Figure 8. Special Feature Summary

The 2501 Card Reader, Model A1 has a maximum rate of 600 cards per minute, and the 2501 Card Reader, Model A2 has a maximum rate of 1,000 cards per minute.

IBM 2520 CARD READ PUNCH AND CARD PUNCH

The 2520 Card Read Punch, Model A1 reads cards serially at a maximum speed of 500 cards per minute and punches parallel at the same rate. Reading and punching may be overlapped to provide reading, computing, and punching at the maximum throughput speed. Reading without punching and punching without reading are also possible with the 2520-A1. All 2520s have two stackers. The 2520 Card Punch, Models A2 and A3 consist of the same basic mechanical unit as the 2520-A1 but have no read unit. Maximum punching rates are 500 cards per minute, or 300 cards per minute, depending on the model.

2520 Card Read Punch Model A1:	500 cpm read/punch.
2520 Card Punch, Model A2:	500 cpm punch.
2520 Card Punch, Model A3:	300 cpm punch.

IBM 2560 MULTI-FUNCTION CARD MACHINE

The 2560 provides full card-file maintenance abilities as well as an optional six-line card document printing feature. Cards are fed from either of two hoppers through a serial read station, a serial punch station, and a serial print station, after which they may be directed to any one of five stackers. The reading speed is 500 cards per minute, punching speed is 160 columns per second, and card printing speed is 140 characters per second.

Unit record functions such as reproduce, gang punch, summary punch, collate and decollate may all be performed on the 2560. The optional printing feature may consist of two, four, or six print heads, which may be positioned by the operator to print on any of the 25 lines of a card. Up to 64 characters may be printed on each line.

IBM 2203 PRINTER

The 2203 Printer provides output for the System/360 Model 20 at up to 750 lines per minute (lpm). Interchangeable typebars allow the operator to select a type style and character set for a specific printing job. Four character sets are available for the 2203 Printer. The 13-character set has 10 numeric and 3 special characters; the character sets with 39, 52, and 62 characters have 26 alphabetic, 10 numeric, and 3, 16, and 26 special characters, respectively.

The printing speed for any one application depends on the total number of lines printed; the amount of processing required for each printed line; and the character set used.

The complete range of speeds available with the IBM 2203 Printer is shown in the table below.

Character Set	lpm	Cycle Time (ms)
13	750	80
39	425	141
52	350	171
62	300	200

A 120-character line, at 10 characters to the inch, is standard. An additional 24 positions are available as a special feature. Vertical spacing of six or eight lines per inch can be manually selected by the operator. Single, double, and triple spacing of lines, plus skipping to a predetermined point, are performed by the tape-controlled carriage, directed by the CPU. The sequence and arrangement of data printed are also controlled by the stored program; a line to be printed is assembled in core storage in exactly the same sequence it is to appear as output.

The dual-feed carriage special feature permits independent and simultaneous control of two sets of forms.

IBM 1403 PRINTER, MODELS 2, 7, AND N1

The IBM 1403 Printer provides output for the System/360 Model 20 at a rate of 600 lines per minute for the 1403-2 and 1403-7, and 1,100 lines per minute for the 1403-N1. If the 1403-N1 is equipped with the universal character set (UCS) special feature, the print speed is 1,400 lines per minute. If the 1403-2 is equipped with UCS, the print speed is 750 lines per minute.

The 1403-2 and 1403-N1 have a print line width of 132 characters, and the 1403-7 has a print line width of 120 characters. Vertical spacing of six or eight lines to the inch can be manually selected by the operator. Single, double, and triple spacing of lines, plus skipping to a predetermined point are performed by the tape-controlled carriage, under control of the CPU stored program. The 1403-2 has a dual-speed carriage that permits high-speed skipping at approximately 75 inches per second on skips over eight lines.

Each print position can print 48 different characters; however, with the UCS special feature, each print position can print up to 240 characters. The printing format is controlled by the stored program.

IBM 2415 MAGNETIC TAPE DRIVE AND CONTROL UNIT

The IBM 2415 Magnetic Tape Drive and Control Unit is designed to meet the low-cost magnetic tape requirements of System/360. Six models of the IBM 2415 are available with the System/360 Model 20. The 2415-1, -2, and -3 use the non-return-to-zero-IBM (NRZI) method of recording, and the 2415-4, -5, and -6 use the phase encoding (PE) method. The IBM System/360 Model 20 operates with either NRZI or PE type tape drives through the input/output channel (IOC).

The user can select from three different models of either NRZI or PE type tape drives. The three different models consist of modules containing two, four, or six tape drives that operate independently but share a common control unit. The IOC can operate one control unit.

Each tape drive can be addressed by the program, and has its own keys and indicator lights for manual control.

The following illustration shows the performance of the various models. The tape drives are normally equipped with nine-track read/write heads; however, seven-track heads can be installed. For example, a 2415-2 may be equipped with two nine-track heads and two seven-track heads.

	Model	Size	Density	Bytes/Sec	Tape Speed
NRZI (9-Track)	1	two tape drives	800 Bpi	15000	18.75 inch/sec
	2	four tape drives	800 Bpi	15000	18.75
	3	six tape drives	800 Bpi	15000	18.75
PE (9-Track)	4	two tape drives	1600 Bpi	30000	18.75
	5	four tape drives	1600 Bpi	30000	18.75
	6	six tape drives	1600 Bpi	30000	18.75
The 7-Track Compat. Feature for all Models allows densities of 200, 556, and 800 Bpi NRZI					
The 9-Track Compat. Feature for Models 4, 5, 6 allows 800 Bpi NRZI Operation					

29219

IBM 2311 DISK STORAGE, MODELS 11 AND 12

The IBM 2311 Disk Storage, with easily interchanged IBM 1316 Disk Pack, affords the user the programming flexibility of large on-line storage capacity and virtually unlimited off-line storage capacity. As with magnetic tape, a virtually unlimited volume of data can be written on 1316 disk packs, separated from the 2311, and stored off-line until needed.

Two models of the IBM 2311 Disk Storage (model 11 and model 12) are available for use with the System/360 Model 20. Up to two 2311 Disk Storage Drives, (either 2311-11 or 2311-12) can be attached and controlled by the storage control special feature; however, both models cannot be attached to the same system.

Disk packs written on a 2311 model 11 may be read on a 2311 model 11 attached to any System/360 Model 20. Disk packs written by a 2311 model 12 can also be read by a 2311 model 12 attached to any System/360 Model 20. Compatibility between 2311 model 11 and model 12 is limited to the extent that the 2311-11 reads and writes over 200 cylinders on the disk pack and the 2311-12 reads and writes on only the outer 100 cylinders.

The 2311-11 and the 2311-12 differ only in the number of track positions

available. Each track position comprises a ten-high stack of disk surfaces and read/write heads, forming a cylinder of data. The 2311-11 has 200 cylinders of data available; the 2311-12 has 100 cylinders available (Figure 9).

	2311 Model 11	2311 Model 12
Storage Capacity	5.4 Million Bytes	2.7 Million Bytes
Data Transfer Rate	156,000 Bytes/Sec	156,000 Bytes/Sec
Data Bytes/Sector	270 Bytes	270 Bytes
Sectors/Track	10	10
Data Bytes/Track	2700 Bytes	2700 Bytes
Data Bytes/Cylinder	27,000 Bytes	27,000 Bytes
Data Cylinders	200	100
Alternate Cylinders	3	3
Maximum Access Time	135 ms	90 ms
Average Access Time	75 ms	60 ms
Minimum Access Time	25 ms	25 ms
Disk Rotation Time	25 ms	25 ms

29220

Figure 9. IBM 2311 Disk Storage Characteristics

Storage Capacity

Storage capacity of the 2311 model 11 is 5.4 million bytes. The storage capacity tables are based on the use of 200 of the total 203 cylinders for data and 3 cylinders for use as alternates (spares for use in the event of surface damage to a data track). Storage capacity for the 2311 model 12 is 2.7 million bytes. It is based on the use of 100 of the total 103 cylinders for data and 3 cylinders for alternates.

IBM 1419 MAGNETIC CHARACTER READER

The 1419 Magnetic Character Reader can be attached to a System/360 Model 20 through a serial input/output channel (SIOC). The 1419 reads into the system the magnetically inscribed information on checks and other banking documents at speeds as high as 1,600 documents per minute. Documents can be sorted into as many as 13 classifications as they are read. All magnetic inscriptions can be checked for validity.

Documents read by the 1419 Magnetic Character reader may be of intermixed sizes and thicknesses, as typically encountered in check-handling operations. The standard minimum length is six inches; shorter documents, such as the 51-column postal money order, can be read into the System/360 model 20 at a maximum rate of 1,960 documents per minute. Shorter documents (which may be intermixed with standard-length documents) can be sorted if the optional (no charge) feature for this purpose is installed.

Many special features are available for the 1419, including an endorser that prints a full endorsement on the back of each document at no reduction in operating speed.

The universal characteristics of the UCSTR allow the user to determine the line code, message format, control characters, and line-control procedure that best fulfill the needs of his application.

The considerations for selecting the line code, message format, and control characters are all interrelated to the line-control procedure selected by the user.

LINE-CONTROL PROCEDURES

A line-control procedure is the method used to control the message exchanges between stations.

The UCSTR flexibility places the burden of line control on the System/360 Model 20 program. The remainder of this section contains some of the many possible approaches to developing a line-control procedure. The user should not consider the following as the only approaches to a line-control procedure.

Line-control procedures for the UCSTR can be completely under System/360 Model 20 program control when operating on a leased or private line. When operating on a switched network (dial-up), the operator manually establishes and terminates the "communications connection", and the System/360 Model 20 program accomplishes the remainder of the line control tasks. A line-control procedure should:

1. Provide a method of establishing a communications connection between two stations.
2. Provide a method to determine when each station is ready to perform its respective function (receive or transmit).
3. Provide a method of record identification so that each station can refer to a particular message by a common identifier.
4. Provide a method of acknowledging that a message(s) was received, including a designation of a good message (no CRC, LRC or VRC error occurred) or an error message (a CRC, LRC or VRC error occurred).
5. Provide a method of retransmitting messages in which an error occurred.
6. Provide a method of terminating the communications connection between the two stations.

Establishing Communications

The method of establishing a communications connection between two stations depends upon the type of communications facility and the configuration of each station. The following describes the various methods and the configuration that will use each method.

1. Stations using a switched network (dial-up) facility without the unattended-answer feature can establish communications only by manually initiating a telephone call to the desired station. The operators at the two stations verbally agree upon which station will transmit and which station will receive. After placing the data set in a data condition (if required), each operator readies his station for its respective function (transmit or receive).
2. Stations using a switched network (dial-up) facility with the unattended-answer feature can establish communications by manually initiating a telephone call to the desired station. However, there is no need for a verbal agreement between the operators. The System/360 Model 20 program at the station receiving the call must automatically prepare to receive a message. When the operator initiating the call is signaled by the data set that a connection has been established, he prepares his station to transmit the first message.

   For this type of operation, the receiving System/360 Model 20 must be in a ready condition, the UCSTR must be in a ready condition, the UCSTR must be in receive mode (a set-to-receive- mode instruction has been executed), and the System/360 Model 20 must be periodically testing the ring TIOB indicator. The data set receiving the telephone call sends a ring signal to the UCSTR, setting the ring TIOB indicator on. When the ring TIOB indicator is tested by the program, the program branches to the address in the TIOB instruction. This address must contain a program subroutine that issues a receive-record instruction, preparing the UCSTR to receive a message.

   The message received can be either a control message or a data message.

After the first message is transmitted, the remainder of the message exchanges is under program control.

After all message exchanges are complete (an EOT has been received) the program terminates the call by executing an automatic-disconnect instruction (CIO instruction).

3. Stations using a leased line or private line can establish communications manually, or they can establish communications completely under program control.

Under complete program control, one station can be designated as the master station that controls the initiation of all message exchanges. The other station is designated as the remote station that responds to the master station.

Another form of complete program control is accomplished by initially placing both stations in a ready-to-receive condition. This means that the System/360 Model 20 is in a ready condition, the UCSTR is in a ready condition, a set-to-receive-mode instruction has been executed, and a receive-record instruction has been executed. (An inhibit-response-time-out instruction must be executed to prevent a response time-out.)

The communications line, under these conditions, is in an idle state.

When one of the stations wants to transmit (determined by the program sensing a condition requiring transmission), it issues a set-to-transmit-mode instruction to gain control of the transmission line.

After the required message exchange has occurred, the program at each station returns its respective station to the receive mode. The transmission line remains in an idle state until the program at one station again sets its station to the transmit mode.

To avoid the problem of simultaneous transmission requests, each station should be assigned a priority, allowing the station with the highest priority to maintain control of the transmission line.

Two methods can be used to establish a communications connection in multipoint configurations.

1. Multipoint configurations without the terminal address recognition special feature require a programmed recognition of the station selection message sent by the master station. Each remote station is set to a receive condition, and each station receives all messages placed on the transmission line by the master station. The master station should be programmed to send an address selection message following a message ending with an EOT character. Thus, the initial message should be a "dummy" EOT message that will set the receive EOT indicator in all remote stations. The program in each remote station senses that the receive EOT indicator is on, and it conditions the program to analyze the next message to determine which station will be selected.

   The master station sends a message containing a unique character assigned to the desired remote station. All remote stations receive and analyze the selection message, but only the one station's program recognizes the unique station selection character.

   The selected station's program prepares to receive all of the following messages and make the necessary responses until the master station concludes the message exchanges with an EOT ending character. All stations receive all of the messages, but only the selected station is programmed to accept and use the messages.

   After receiving the EOT character, the station selection operation is repeated, and the next remote station is selected by a station selection message.

2. Multipoint configurations with the terminal address recognition special feature can be selected by the master station with a minimum of programming effort.

   Each remote station in the multipoint configuration is set to a receive condition.

   The master station sends a unique station selection character (each station is assigned a unique character) as the first data character of a message.

   Only the station that is assigned the particular character sent is selected. All other stations ignore the message. After the message exchange is complete, the master station sends the next message containing the next station selection character.

Sending the First Message

In each of the methods of establishing communications, a decision was made

(either by the operators or by the programs) to determine which station will transmit and which station will receive. Even after this determination, there may be some delay before a message exchange is possible, requiring that some form of an inquiry message be sent by the transmitting station. This message, in effect, asks the receiving station if it is ready for the first data message.

The receiving station, after receiving the inquiry, answers the transmitting station with an acknowledgment. The acknowledgment can be either positive (yes, the receiving station is ready to receive) or negative (no, the receiving station is not ready to receive).

In some cases, the inquiry and acknowledgment exchange can be omitted. However, some program operating efficiency can be gained by the inquiry and acknowledgment exchange.

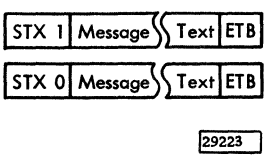
Identifying Records

To establish positive communication between the transmitting and receiving stations, records can be identified so that both stations have a common identifier for referencing a particular record. Odd-numbered records can be assigned a unique character, and a different character can be assigned to even numbered records. For example, the transmitting station program assigns two bit combinations (characters) as start of text (STX) characters. One character is an STX 0 character, and the other character is an STX 1 character.

The transmitting station inserts the STX 0 preceding the text of the message for all even-numbered messages, and it inserts the STX 1 preceding all odd-numbered messages.

The receiving station receives a message, and the program analyzes the STX character to determine if it is an odd or even STX character.

By preceding each message with an STX character that is alternated (odd or even), the receiving station has an additional checkpoint to ensure that a complete record was not lost during transmission. The following illustrates two messages as they appear in the transmitting station core storage, using the above procedure.

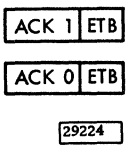


The STX characters are not confined to any particular bit combination. The bit combinations selected by the user for STX characters can be any bit combinations except the bit combinations selected as UCSTR control characters.

Responses to Messages

UCSTR operations do not require the receiving station to send a response message for each message it receives. However, a good line-control procedure should include enough response messages to ensure that a "line disconnect" will be detected before a large volume of data is transmitted.

If the transmitting station is using an odd/even record identification, the receiving station should also send responses that identify each correct message as an odd or even record. The odd/even record identification provides a more positive method for the transmitting and receiving stations to reference a particular record. The following illustrates two response messages as assembled in the receiving station core storage.



The ACK 1 and ACK 0 characters selected by the user can be any bit combinations except the bit combinations used as UCSTR control characters.

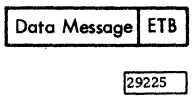
A character should be assigned as a negative response (NAK). This character is used by the receiving station to inform the transmitting station (via a response message) that an error (CRC,LRC or VRC error) occurred during a record transmission.

The character selected by the user as a NAK character can be any combination of bits except the bit combinations used as UCSTR control characters.

A response message can take one of several forms. It can be a simple acknowledgment, a message returned for the message received, or a more complex message acknowledging the receipt of multiple-message blocks.

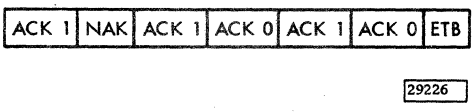
Simple acknowledgments or message-for-message responses are sent by the receiving station on a one-for-one basis. This means that the receiving station sends a response message to the

transmitting station for each message received. The following illustrates illustrates the message-for-message response.



The more complex response message that acknowledges the receipt of multiple-message blocks is a message comprised of an acknowledgment (good or bad) for each message received. After the receiving station receives a message and makes the CRC or LRC check, the receiving station program must start to "build" a response message that will indicate the status of each message received.

After all message blocks have been received, the receiving station sends the response message to the transmitting station. The transmitting station program must analyze the message to determine which message blocks (if any) were not received correctly at the receiving station. Any message blocks that were not received correctly must be retransmitted. The following illustrates a response message containing six bytes that acknowledge the receipt of six message blocks.



The response message, as assembled in core storage and shown above, shows that all but the second message block was received correctly. When the transmitting station program receives this response message and analyzes its contents, the transmitting station is informed by the NAK character that the second message block must be retransmitted.

Error Retransmission

The line-control procedure must provide for retransmission of messages in which an error occurred during transmission.

When the receiving station receives an error message (CRC, LRC, or VRC error), the receiving station sends a NAK response message to the transmitting station. The transmitting station receives the response message and transmits the previous message a second time. The receiving station continues to reply with a negative response (NAK) until the message is received without an error.

To prevent the two stations from remaining in a retransmission "loop", the program at one of the stations should count the errors. Either the receiving station or the transmitting station program can count the errors, but normally the receiving station should count the number of times the error indicator is turned on.

When a predetermined error count (determined by the user) is reached, the program sounds the audible alarm.

Terminating Communications

The line-control procedure should provide some means of terminating the data call or a means of informing the operator that all data has been transmitted.

Depending on the line-control procedure selected, the end-of-transmission can be signaled to the receiving station by receiving an EOT character, or a unique character can be assigned for use as an end-of-transmission signal. In the latter case the program must analyze each message to determine when it has received the end-of-transmission character.

If the communications facility has the unattended answer feature, the data call can be terminated by the program executing an automatic-disconnect instruction.

CONTROL CHARACTERS

The UCSTR requires three control characters (sync, ETB and EOT) that are assigned by the user. In transparent mode, another control character (DLE) with a fixed bit combination is used.

In selecting the bit combinations for the sync, ETB, and EOT characters, the user must assign bit combinations that will not occur as data when the basic UCSTR mode (full binary minus three) is used. The sync character must be selected as described in Control Field.

When transmitting in transparent mode the user must consider the possibility of a DLE-ETB or DLE-EOT combination occurring as data. If these combinations can occur as a part of the data message, they must be either split apart by the program or transmitted in transparent mode with the response bit off, thereby controlling the ending sequence by a transparent byte count.

In addition to the UCSTR control characters, the user can assign other characters as program control characters. Since the UCSTR recognizes only its three control characters, any other character designated as a control character must be recognized under program control.

LINE CODES

The UCSTR can transmit and receive in any eight-bit, seven-bit, or six-bit line code. THE UCSTR does not provide any code translation. Any code translation that is required must be done by the program.

If the UCSTR is operating with another station that is code insensitive, any line code can be used, but if the UCSTR is operating with a station that is sensitive to only one code, a translation to the proper code must be made by the System/360 Model 20 program.

If UCSTR operation is being "tailored" to a specific line format that includes control characters or sequences, the control characters or sequences must be accumulated as part of the CRC or LRC character unless they can be ignored or LRC subtracted.

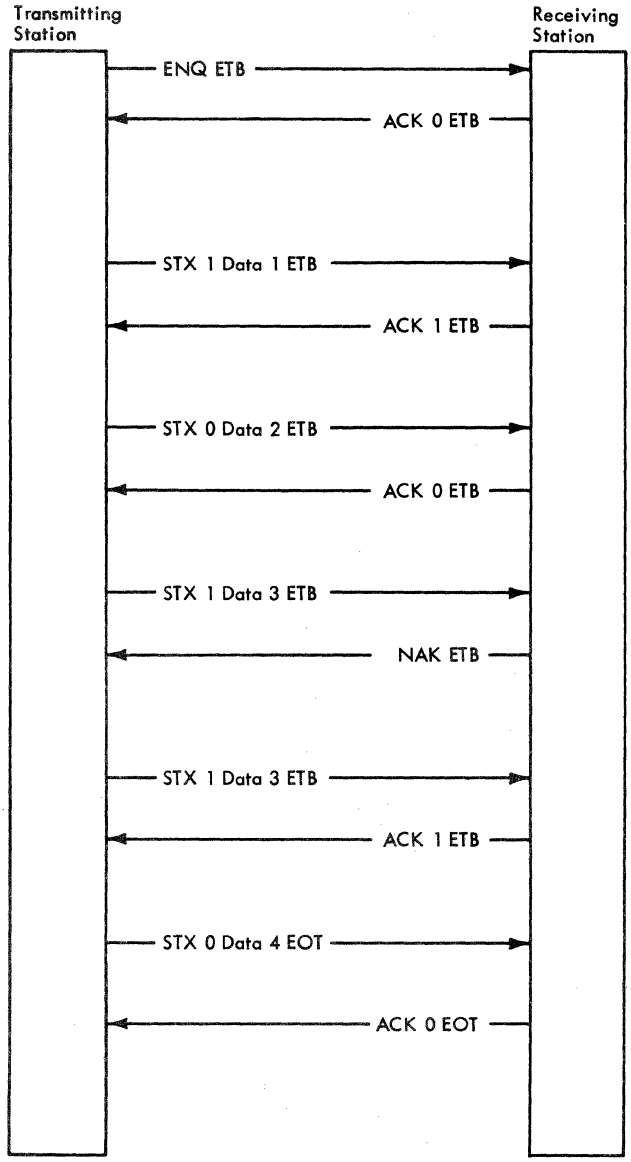
When a large volume of numeric data is to be transmitted, the user should consider the line efficiency that can be gained by using the packed decimal format.

MESSAGE FORMATS

This section provides some example message formats that can be used with the UCSTR.

Message and Acknowledgment Response

Figure 10 illustrates a message format for a message exchange including a response acknowledgment message for each message transmitted.



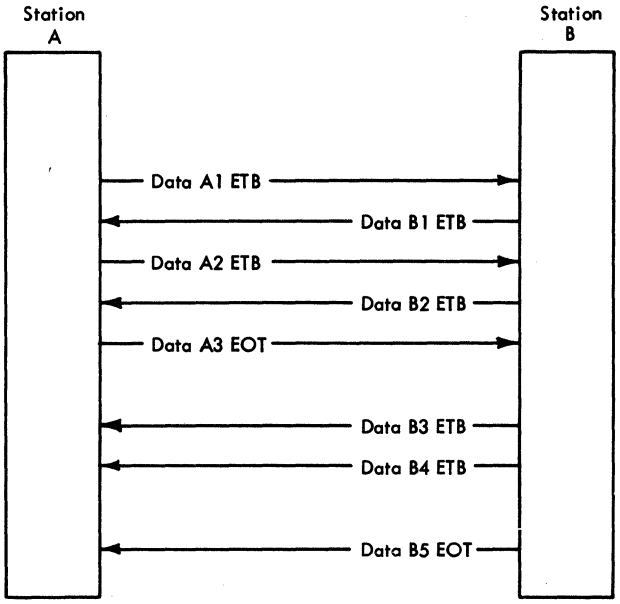
1. The transmitting station sends an inquiry (ENQ) message to the receiving station.
2. The receiving station program analyzes the message, determines that the message is an ENQ, and sends an ACK 0 message to the transmitting station. The ACK 0 message informs the transmitting station that the receiving station is ready to receive the first data message. (If the receiving station is not ready to receive data, the response message would be NAK ETB.)
3. The transmitting station sends the first data message. The first data message is preceded by an STX 1, identification character.
4. The receiving station receives the first data message, determines that it is correct, and sends an ACK 1 message to the transmitting station. The ACK 1 informs the transmitting station that the message was received correctly.
5. The transmitting station sends the second data message. The data message is preceded by an STX 0, identification character.
6. The receiving station receives the second data message, determines that it is correct, and sends an ACK 0 message to the transmitting station. The ACK 0 message informs the transmitting station that the message was received correctly.
7. The transmitting station sends the third data message. The data message is preceded by an STX 1, identification character.
8. The receiving station receives the third data message, determines that an error occurred, and sends a NAK message to the transmitting station. The NAK message informs the transmitting station that an error occurred during the last message, and the message must be retransmitted.
9. The transmitting station retransmits the third data message. The message is again preceded by an STX 1 character.
10. The receiving station receives the third data message (for the second time), determines that it is correct, and sends an ACK 1 message to the transmitting station. The ACK 1 message informs the transmitting station that the message was received correctly.
11. The transmitting station sends the last data message (EOT ending character). The message is preceded by an STX 0, identification character.
12. The receiving station receives the fourth data message, determines that is was correct, determines that an EOT was received, and sends an ACK 0 followed by an EOT. Receiving the EOT informs the receiving station that the last message has been received. When the transmitting station receives the message, it is informed that the last message was received correctly and that the operation can be terminated.

29227

Figure 10. Message and Acknowledgment

Message and Message Response

Figure 11 illustrates a message format for a message exchange including a message response for each message transmitted. The message from each station may or may not contain an acknowledgment of the previous message.



1. This example shows only the data message exchange between two stations operating on a message-for-message basis. This type of message exchange can also include inquiry sequences, message identifying characters, and acknowledgment characters as described in the section for Line-Control Procedures.
2. Station A transmits the first data message to station B.
3. Station B transmits the first data message to station A.
4. Station A transmits the second data message to station B.
5. Station B transmits the second data message to station A.
6. Station A transmits the last data message to station B. Station A informs station B that this is the last data message by sending the EOT character as the ending character.
7. Station B transmits the third data message to station A.
8. Station B transmits the fourth data message to station A. (Station B continues to send messages to station A until all messages have been transmitted.)
9. Station B transmits the last data message to station A. Station B informs Station A that this is the last data message by sending the EOT character as the ending character.

29228

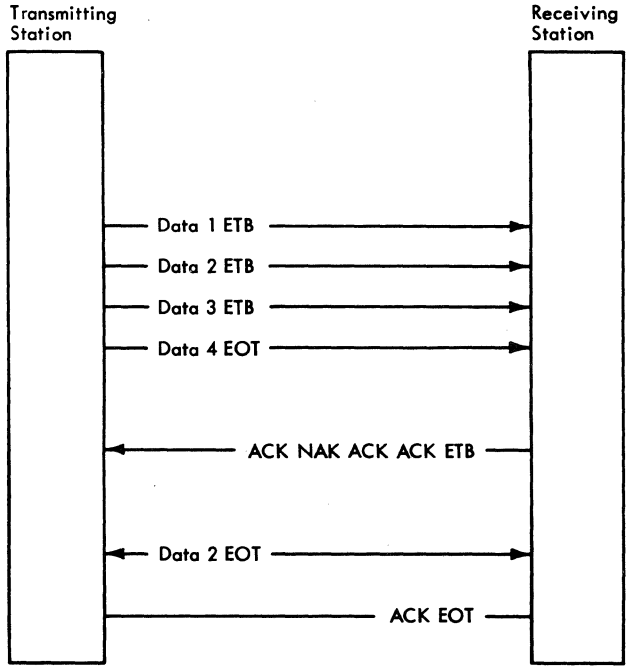
Figure 11. Message and Message Response

Multiple Messages and One Response

Figure 12 illustrates a message format for a message exchange including several messages sent by the transmitting station and one response message sent by the receiving station.

Multipoint Operation

Multipoint operation can be with or without the terminal address recognition feature. Without the terminal address recognition feature, the program must provide for terminal address recongition (See Multipoint Facilities). The format of the message exchange can be any of those previously described.



1. This example shows only the data messages and response messages for a UCSTR operation that transmits multiple message blocks and receives a single response message indicating the status (good or bad) of each message transmitted. This type of message exchange can also include inquiry sequences, message identifying characters, and acknowledgment characters as described in the section for Line-Control Procedures.
2. Transmitting station sends the first data message.
3. Transmitting station sends the second data message.
4. Transmitting station sends the third data message.
5. Transmitting station sends the fourth data message. The EOT ending character informs the receiving station that all messages have been transmitted. The receiving station records the status of each message received.
6. The receiving station sends a response message to the transmitting station. All messages except the second message were received correctly. The response message informs the transmitting station that the second message must be retransmitted.
7. Transmitting station retransmits the second message. The EOT ending character informs the receiving station that this is the last message.
8. The receiving station receives the message correctly and sends the response message to the transmitting station. The operation is ended by the transmitting station receiving an EOT ending character.

29229

Figure 12. Multiple Messages and Response

APPENDIX A. UCSTR TIMING

The following is the average CPU time, in microseconds (usec), required to execute the instructions associated with the UCSTR feature.

Instruction	Time Required
Transmit-record XIO	200usec
Receive-record XIO	205usec
Test busy TIOB	120usec
Test error TIOB	120usec
Test receive EOT TIOB	120usec
Test time-out response TIOB	120usec
Test ring indicator TIOB	120usec
Set-to-receive-mode (transfer complete control field) CIO	180usec
Set-to-receive-mode (transfer control byte only) CIO	145usec
Set-to-transmit-mode (transfer complete control field) CIO	170usec
Set-to-transmit-mode (transfer control byte only) CIO	135usec
Inhibit-time-out-response CIO	120usec
Inhibit-audible-alarm CIO	120usec
Set-audible alarm CIO	120usec
Nullify-transfer-instruction CIO	120usec
Automatic-disconnect CIO	120usec

The following is the average CPU time required in microseconds (usec), to service a UCSTR service request. This information can be used to determine system timings in the time-shared mode.

UCSTR Function	Time Required
Transmit service request (not transparent with CRC checking)	105usec
Transmit service request (not transparent with LRC checking)	115usec
Transmit service request (not transparent with VRC checking)	100usec
Transmit service request (not transparent with LRC/VRC checking)	125usec
Transmit service request (transparent with response on)	95usec
Transmit service request (transparent without response on)	95usec
Receive service request (not transparent with CRC checking)	115usec
Receive service request (not transparent with LRC checking)	130usec
Receive service request (not transparent with VRC checking)	125usec
Receive service request (not transparent with LRC/VRC checking)	135usec
Receive service request (transparent)	95usec

Printed in U.S.A. A26-5622-0

International Business Machines Corporation
Data Processing Division
112 East Post Road, White Plains, N.Y. 10601
[USA Only]

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
[International]