XEROX

PALO ALTO RESEARCH CENTER
Computer Science Laboratory
April, 1973 - CSL Archieve # 3AL-038
retyped November 7, 1978

To: Alto Group

From: C. Thacker

Subject: Alto Interfaces

This memo is in response to the request for a description of the Alto I/O system. Familiarity with the Alto is assumed[1].

There are two basic ways in which I/O devices may be connected to the Alto. A device may appear to a program as one or more memory locations in the range 177000-177777, in which case it responds to addresses from MAR, for its selection, and transfers data on the memory data bus, MD(00)'-MD(15)'. Alternatively, a device may connect directly to the processor bus, and be selected by task-specific F1 or F2 signals. The latter interface will usually have its own task microcode, and will be used only for high speed devices (> 1K words/sec). The memory bus interface will usually be used for slow peripherals which can be driven by a "Nova" program, although nothing prohibits mixing the interface methods, i.e., a memory bus device can have its own microcode task.

Memory Bus

The memory bus signals are:

MD(00)'-MD(15)'	Memory data. These lines are low true, bidirectional, open collector. They should be driven by 74H01 or 7438 gates, and should be received with onto TTL load, to minimize total capacitance.
MAR(07)-MAR(14),XMAR(15)	Memory address. These lines are the outputs of the memory address register. They are high true, and should be only lightly loaded.
XIOREF	This signal indicates that an I/O reference (address >177000) is in progress. It is the AND of MAR(00)-MAR(06). The processor does not check parity on I/O references.
XMT2	This signal indicates the transfer cycle (170ns) of a memory reference is in progress. On a fetch, the processor strobes data during the last 25ns of XMT2. During a store, data is valid during the last 90ns of XMT2, and for approximately 15ns thereafter.
MISYSCLK	This signal is the memory interface's version ot the system clock. It is true during the last 25ns of every microinstruction.

[1]See Alto: A Personal Computer System - C. Thacker, E. McCreight

The timing of a memory cycle is shown in Figure 1. The memory address register is loaded at the end of a cycle in which a MAR-microinstruction is executed. The next four cycles, X, 0, 1, 2 comprise the memory cycle. Data is transferred during MT2, and will be stored if the microinstruction executed during MT2 is MD←, feteched if the microinstruction specifies ←MD.

Due to critical timing restrictions, I/O devices do not use the normal signals which indicate, during MT2, whether a store or a fetch is being done. Instead, the address determines the nature of the reference, and fetches are directed only to addresses capable of delivering data, stores are directed only to addresses capable of accepting data.

Whenever an I/O device capable of delivering data detects its address, it places data on the memory bus. If it needs to know that the reference is finished, (to clear a buffer full indicator, for example) it uses XMT2 for this purpose.

Devices which accept stores should be designed to clock data from MD' on the falling edge of XMT2, or, if latches are used, to load them with ADDRESS DETECTED·XMT2·MISYSCLK.

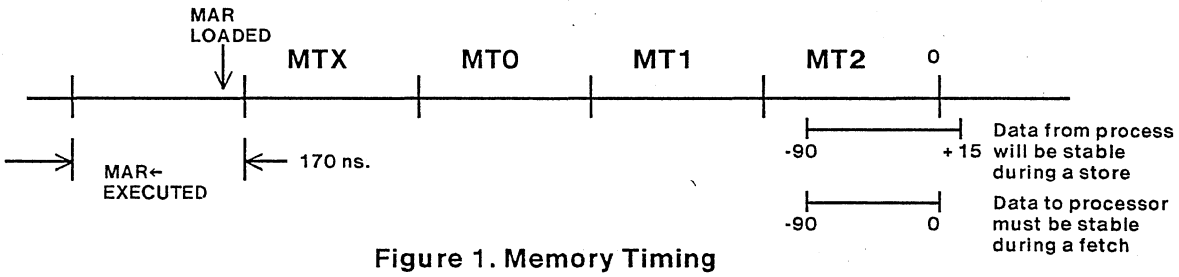

Figure 1. Memory Timing

Processor Bus Interfacing

Devices which interface the processor bus have at their disposal most of the facilities provided by the processor. Rather than attempting to discuss all possible signalling sequences, the application of each set of signals available at the card connector will be discussed.

WAKEn', nACT'

These signals are the wakeup request and task active lines for task n. When the I/O device requires service from its associated task, it drops WAKEn'. When the tasks becomes active, the processor drops nACT'. The wakeup request line should be synchronous with the system clock, as shown in Figure 2. The taks will retain control of the processor until it executes a TASK function and,

(1) there is a higher priority wakeup request, or
(2) the wakeup request is removed.

It is the responsibility of the taks microcode to cause the wakeup request to be removed when the service which triggered the request has been provided. All signals which gate registers to, or loged registers from., the bus should be gated with nACT'.

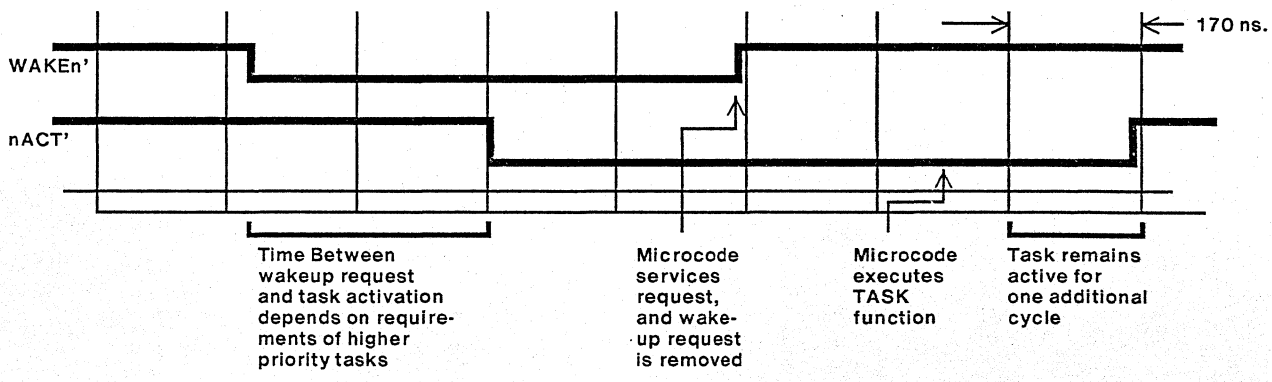

Figure 2: WAKEUP TIMING

Next(05)'-Next(09)'

The NEXT bus is a low true, open collector bus which specifies the address of the microinstruction which will be feteched from the control memory during the next cycle and executed during the cycle following that. Branches are done in the Alto by Oring onto this bus, and the low 5 bits of the NEXT bus are provided on the backplane for this purpose.

By convention, branches are usually controlled by F2's. A typical instruction sequence for a branch function which ORs a 1 into NEXT(09) if its condition is true is:

```
!1,2,No,Yes                ;assembler directive specifying that
                           ;YES should be in an odd location, NO
                           ;in an even location

BRANCHFUNCTION             ;During the execution of this
                           ;instruction, the NEXT bus contains
                           ;the address specified by the following
                           ;instruction (i.e. NO). If the branch
                           ;condition is true, the NEXT bus
                           ;contains NO+1

:NO                        ;colon is the assembler syntax for
                           ;specifying the address of the
                           ;subsequent instruction.

NO: mumble                 ;get here on failure.

YES: foo                   ;get here on success.
```

F1(0)-F1(3),F2(0)-F2(3)

These lines are the outputs of MIR. By convention, F1's and F2's from 0-7 are task independent, those from 10B to 17B are interpreted according to the active task. These lines are stable for the entire cycle, and change at approximately -15ns. Because these lines can change during the system clock, they should be buffered in a latch if they are used to gate other clocks (see figure 3).

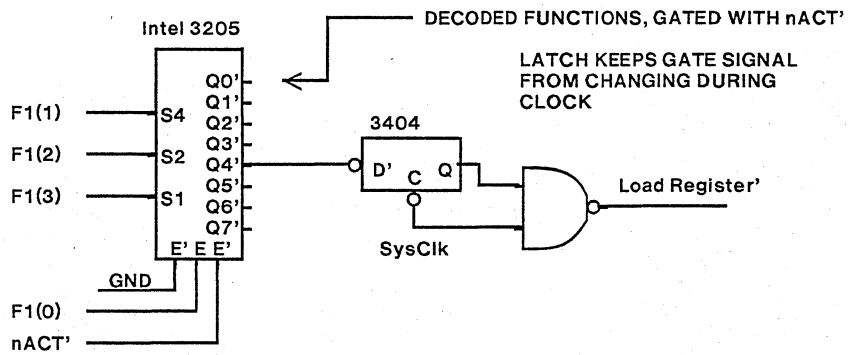

Figure 3. FUNCTION DECODE

BUS(00)-BUS(15)

The system data bus is high true, and drivers may be either open collector or tri-state. Data will be stable on

the bus no later than 80ns into a cycle, and will remain stable until the end of the cycle.

Start

This signal is generated when the emulator executes the SIO instruction. This signal is true for an entire cycle, and during the cycle, ACO is placed on the bus. The bits of the bus may be used for arbitrary control functions.

OKTORUN

This signal is provided for devices which require initialization. It remains false from the time power is turned on until +5v is at nominal level, and the bootstrap button has been pressed and released. When power is established, OKTORUN follows the bootstrap button.

SYSCLK and ARC

These are the two versions of the system clock. SYSCLK occurs at the end of every microinstruction, and is used to load registers, etc. Some memory reference sequences may stop the processor for from one to three cycles by disabling SYSCLK. ARC (always running clock) has the same phase as SYSCLK, but is never stopped.

These are the standard signals provided on the edge connectors for processor bus devices. These should be the only signals required to interface most I/O devices. For any further information, consult me.