Accessing the NS16550A UART in the PS/2 Model 50, 60, 70, and 80

National Semiconductor
Application Note 628
Martin S. Michael
July 1989

INTRODUCTION

This paper reviews fundamental concepts of the Micro Channel Architecture and their relation to the NS16550A UART. All 4 of the PS/2 personal computers reviewed, use the NS16550A for asynchronous serial communication.

The first part is an overview of the PS/2 system board and Micro Channel Architecture (MCA) in the Models 50, 60, 70, and 80 personal computers. The next part explains the basic configuration and system initialization procedures for the UART that occur after power-up. The last part describes the overall interrupt procedure and the advantages of using the on-chip FIFOs of the NS16550A. These explanations describe the CPU accesses to the UART via MCA. Timing diagrams in the appendix show these accesses to the UART.

OVERVIEW OF THE PS/2 MODEL 50, 60, 70, AND 80 SYSTEM ARCHITECTURE

The block diagram indicates a number of identical functions that all system boards have *(Figure 1).* Each system CPU has an 8 channel DMA Controller and an optional math coprocessor associated with it via the local bus. The DMA Controller emulates the dual 8237 DMA controllers found on the IBM AT. Additionally, this DMA Controller provides Extended and Virtual Mode operation. These modes allow it to interface with various DMA slave devices and the CPU to dynamically select the arbitration level for 2 of the DMA channels. A central arbitration point allows certain adapter cards and system peripherals to compete for DMA transfers. These adapter cards must have the appropriate arbitration and DMA logic.

Buffers condition the bus signals from the system CPU and send them directly to the Micro Channel Interface. These signals, after further buffering, reach the system memory and the system peripherals. The system ROM on the Models 50 and 60 also interfaces via these buffers to the 80286 CPU. In the Models 70 and 80 the 128 kbyte ROM interfaces via the local bus to the 80386 CPU.

The dynamic RAM is expandable on the system board or on adapter cards. DMA controller addressing capability limits the total DRAM available on any of these systems to 16 Mbytes. The maximum DRAM available on the various system boards is:

1. Model 50; Type 1 = 1 Mbyte, Type 2 = 2 Mbytes
2. Model 60; = 1 Mbyte
3. Model 70; Type 1 or Type 2 = 6 Mbytes
4. Model 80; Type 1 = 2 Mbytes, Type 2 = 4 Mbytes

Beyond the memory, coprocessor, and DMA there are a number of major peripheral functions resident on the system board. These are:

1. serial port (NS16550A)
2. video graphics controller
3. diskette controller
4. parallel port
5. keyboard and pointing device controller
6. CMOS clock and configuration RAM
7. dual interrupt controllers (16 channels)
8. timer (3 channels)

The configuration software for the serial port on the system board restricts the addressing of the NS16550A to COM1 and COM2 on the Models 50, 60, 70, and 80. Adapter card serial ports, however, may be assigned any 1 of the 8 base I/O addresses.

Adding adapter cards extends the PS/2 functionality beyond the system board. These cards plug into the Micro Channel Bus connectors and conform to the MCA protocols.

OVERVIEW OF THE MICRO CHANNEL ARCHITECTURE

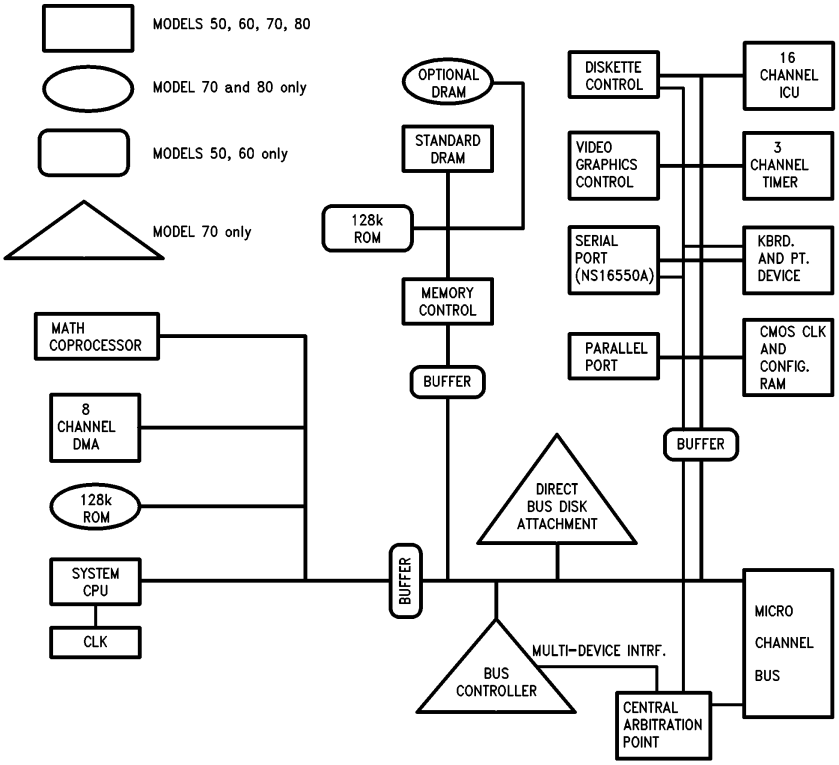
MCA functionality increases as the data bus width increases from 16 to 32 bits. Both bus widths support certain fundamental features regardless of the data bus width. One of these is a centralized arbitration controller that allows up to 16 devices to contend for the 8 available DMA channels. These devices compete based on an assigned priority level for the DMA resource. A ''fairness'' option allows lower priority devices to compete successfully for a DMA channel, even though, higher priority devices may require a transfer. If the fairness option is enabled, each device that has received DMA service must wait until all other devices requesting the DMA have been serviced before they are allowed to compete for the DMA resource, again. MCA fixes the priority levels of the DMA channels, except for channels 0 and 4, which the CPU can program to any priority level. The DMA channels are capable of both 8- and 16-bit transfers.

MCA also features level sensitive interrupts, provides for interrupt sharing and brings 11 of the 16 hardware interrupts out to the Micro Channel Bus for the adapters to use. The last section of this paper describes interrupt handling in more detail.

Previous PC architectures used jumpers and switches to configure the adapter cards. MCA uses programmable configuration registers on each adapter card, instead. This adds to system flexibility by allowing automatic card configuration via software.

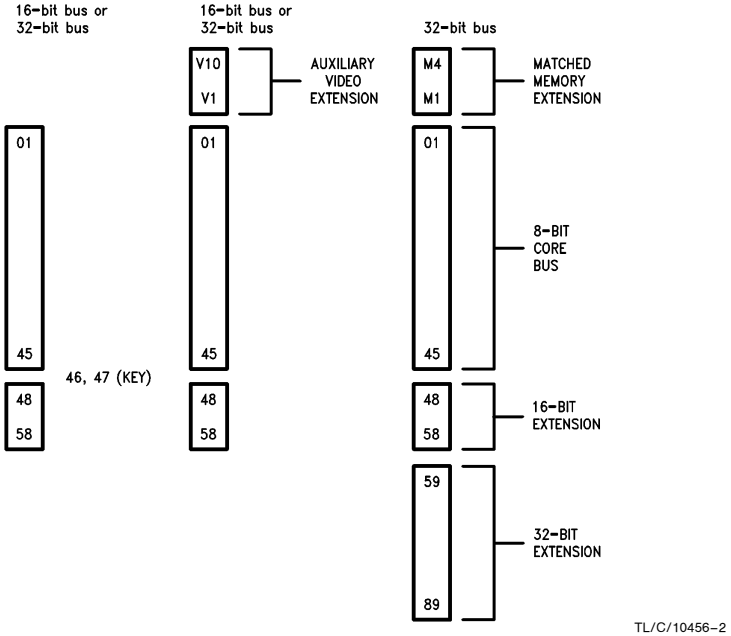
The Models 50 and 60 support 8- or 16-bit transfers over a 64 kbyte range of I/O addresses and over a 16 Mbyte range of memory addresses. The Models 70 and 80 have all of these capabilities and can execute 32-bit transfers over the 64 kbyte I/O address range or the 4 Gbyte memory address range.

RRD-B30M75/Printed in U. S. A.



TL/C/10456–1

FIGURE 1. PS/2 Block Diagram



TL/C/10456–2

FIGURE 2. Micro Channel Architecture

SUBDIVIDING THE MICRO CHANNEL ARCHITECTURE

MCA has an 8-bit core set of bus signals *(Figure 2)* and 4 types of extensions. If the system CPU is an 80286, a 16-bit extension and an auxiliary video extension are present. If the system CPU is an 80386, a 32-bit extension and a Matched Memory extension are present in addition to the 16-bit and auxiliary video extensions.

The **8-bit core** set of 90 signals are composed of 9 groups. Most of these are the typical bus signals associated with any CPU:

24 address	14 ground
8 data	8 DMA/arbitration
14 control	2 audio
6 interrupts	a 14.3 MHz clock signal.
9 power	

IBM reserved four signals in this set.

The 14 control signals can be grouped by function. A typical data transfer uses 7. The bus master uses 4 to sense card and channel status. Reset and channel configuration require 2, and a DRAM refresh cycle activates 1 signal.

The centralized arbitration controller uses 8 DMA/arbitration signals to support its features plus the facility for single or burst mode transfers. One signal notifies the DMA slave when the DMA channel it is using reaches its terminal count.

An audio summation input is available with a separate audio ground, so that all cards can drive the speaker.

Power and ground pin spacing keeps all bus signals within 0.1 inch of an AC ground potential, thus minimizing EMI.

This core set of signals provides the fundamental structure of MCA. Four extensions contain the remaining MCA Bus signals.

The **16-bit Extension** widens the data bus and adds more interrupts. It adds to the core 8 more data lines, 5 more interrupt lines, and a high byte enable line. This extension also provides a status signal driven by the adapter card to indicate its 16-bit wide data bus capability.

The **Auxiliary Video Extension** provides signals from the system board for use by video adapter cards. These are all of standard video sync signals (i.e., vsync, hsync, and blank) along with enable lines for the Video DAC clock and data lines. Through these enable lines the adapter controls whether it will source or receive the video DAC clock and data. Both of these extensions have the appropriate number of power and ground lines to maintain the reduced EMI specification.

Two additional bus extensions found only in the 32-bit MCA are the 32-bit extension and the Matched Memory Extension. The **32-bit Extension** widens the data bus by adding:

16 data signals
8 address signals
4 byte enable signals
3 32-bit data transfer status signals
15 power and ground signals to maintain reduced EMI

The **Matched Memory Extension** provides signal lines to the adapter, so that the CPU can access RAM on the adapter cards as fast as it accesses system RAM. During Matched Memory cycles to the adapter card, the accesses will take 3 CPU clock cycles instead of 4. Matched Memory cycles use three signal lines. Two of these allow the CPU to indicate when it will provide a Matched Memory Cycle and when valid data is on the bus. The adapter card uses the third signal to request a Matched Memory Cycle.

In summary, the MCA provides all signals necessary for CPU or DMA data transfer to additional memory and peripherals and for monitoring card or bus status. It also supports some miscellaneous functions (e.g., audio, more interrupts, a clock oscillator, etc.). The four extensions provide for data bus expansion to 32 bits, fast memory access and auxiliary video control.

OVERVIEW OF THE PROGRAMMABLE OPTION SELECT (POS)

The Models 50, 60, 70, and 80 use software to configure the system peripherals and adapter cards, rather than providing switches and jumpers. This software is called the Configuration Utilities. These utilities match the system peripherals and adapter cards to their appropriate initialization files. The initialization files are known as Adapter Description Files (ADFs). These files each have an I.D. number that associates each ADF with the matching adapter card I.D. number. The ADF contains specific information used by the operator during system configuration such as the adapter card name, the system resources the adapter can use, and help information. It indicates to the system the minimum resources required for the adapter card to run. It also contains the codes used to record the specific options chosen by the operator (e.g., one of the options for an asynchronous communication card is the assignment of the COM port number to each UART). The ADF will specify the interrupt level, the arbitration level, the I/O register addresses, and the memory range addresses of the adapter card. The Configuration Utilities use this information to configure the adapter card and provide selectable options to the operator. ADFs are ASCII files.

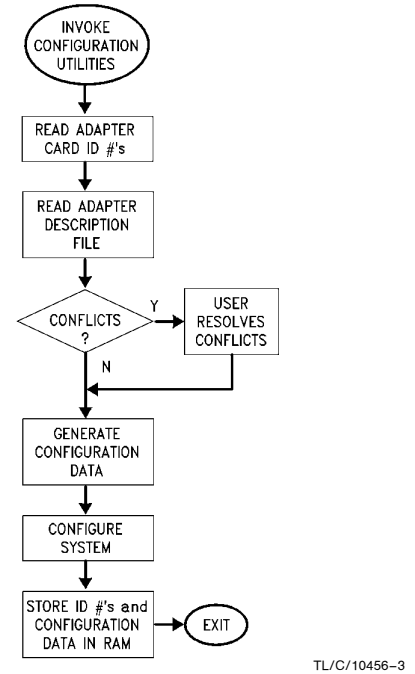
When invoked, the Configuration Utilities begin reading the adapter card I.D. numbers and then read the ADF I.D. numbers *(Figure 3)*. They disable any cards that don't have an ADF and indicate any conflict of resources (e.g., the same COM port address assigned to two different UARTs) to the operator. When there are no conflicts or when the system is automatically configured, the utilities generate the system configuration data. The 64-byte CMOS RAM (all models) and the 2 kbyte CMOS RAM extension (Model 60, 70, and 80) store the configuration data. Once stored the system is ready for normal power-up operation.

During normal power-up *(Figure 4)* the Power On Self-Test (POST) software tests the hardware and compares the adapter I.D. numbers to those in the configuration RAM. If the numbers match, it initializes each adapter card. If they don't match, it requests that the Configuration Utilities be run to resolve the conflicts.

UART ACCESSES—SOFTWARE

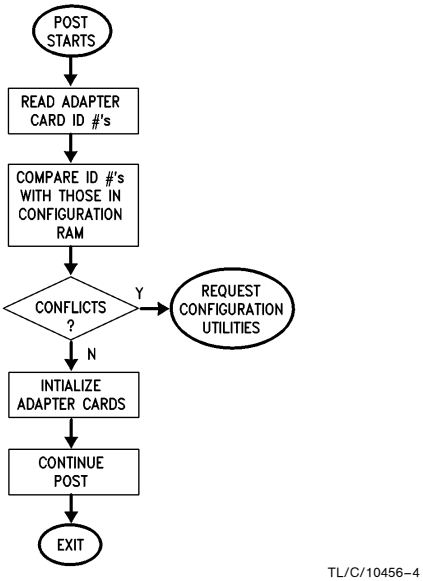
During normal operation UART accesses are done by the applications software via DOS routines, BIOS routines, or by direct access to the UARTs designated addresses. The addresses of each UART in the system are assigned during system configuration.

Using the Configuration Utilities the operator assigns 1 of 8 available base addresses to the asynchronous communications ports on the adapter cards. Table 1 lists all possible addresses for the asynchronous communications ports; they include the original ''COM1'' and ''COM2'' addresses of the IBM AT in order to maintain software compatibility.



TL/C/10456–3

FIGURE 3. Configuration Utilities



TL/C/10456–4

FIGURE 4. POST and Configuration Utilities

TABLE 1. Asynchronous Communication Port Addresses

Name	I/O Addr.	Interrupt	Comment
Serial 1	03f8–03ff	IRQ 4	COM1
Serial 2	02f8–02ff	IRQ 3	COM2
Serial 3	3220–3227	IRQ 3	
Serial 4	3228–322f	IRQ 3	
Serial 5	4220–4227	IRQ 3	
Serial 6	4228–422f	IRQ 3	
Serial 7	5220–5227	IRQ 3	
Serial 8	5228–522f	IRQ 3	

The system board description file restricts the addresses of the system board NS16550A to either COM1 or COM2. Adapter card ADFs determine which addresses are available for the UARTs on the adapter cards. Each adapter card contains a set of registers in the adapter I/O address space from 0100 to 0107 hex and also at 94 and 96 hex. Since these registers are at identical locations on all adapter cards, the system decodes a unique signal for each card when it needs to address these registers. This unique signal is called Card Setup [−CD SETUP (n)]. These ten registers in the adapter I/O address space:

1. Enable either the adapter card or the system board.
2. Store the adapter card I.D. number.
3. Record the selected card options.
4. Store card initialization data contained in the ADF.
5. Provide error status or a pointer to error status.

The NMI error handler uses this error status when the adapter card signals (via channel check, CHCK) a serious error. The error must be one that threatens the continued operation of the system (i.e., a parity error). One of the selectable options that the POS registers store is the address assigned to each UART on the adapter card.

The UART and these POS registers may be accessed at any time through the DOS Debug port I/O utility. This is done by enabling the setup of a particular adapter card through POS registers 0094, 0102 hex and then transferring the data to the assigned UART addresses. The Debug utility allows the adapter card configuration to be changed without running the Configuration Utilities. By not using the Configuration Utilities the user can easily cause configuration conflicts and should be cautious. IBM does not recommend this method of access, but is useful when testing new hardware.

UART ACCESSES—HARDWARE

Appendix A contains timing diagrams of accesses to the NS16550A addressed as COM2 on an IBM Dual Asynchronous Card. Signals from the Micro Channel Bus access this card. The timing of these signals should be the same for all PS/2 systems with MCA, however the measurements in Appendix A were done on only the Model 50 and Model 80. The read pulse width is approximately 390 ns. The write pulse width is approximately 220 ns and the chip select setup time is approximately 290 ns. The first two diagrams in Appendix A illustrate the basic read and write accesses to the UART. The middle two diagrams illustrate ''back-to-back'' write and read accesses to the Scratch Pad Register in the UART. The last two diagrams illustrate the initialization steps done during POST.

INTERRUPT HANDLING AND USE OF THE FIFOS

Interrupts on the PS/2 Models 50, 60, 70, and 80 are level-sensitive low active. This differs from the positive edge-sensitive interrupts on the IBM PCs, XTs, and ATs. There are several reasons for this change. One of the main reasons is interrupt sharing. It is apparent from the serial port addresses listed in Table 1, that 7 of the 8 serial ports will activate IRQ 3 for interrupt service. In an edge-sensitive system the Interrupt Control Unit (ICU) will not record any interrupt edges arriving after the first edge, but before the first interrupt is cleared. This makes interrupt sharing in an edge-sensitive system impossible, unless each device sharing the interrupt is sophisticated enough to only issue an interrupt when another device hasn't.

In a level-sensitive system multiple open-collector devices can hold the same interrupt line low until the CPU services each one. The only additional hardware required is an interrupt pending latch on each adapter card so that the CPU, can identify the card with an active interrupt. The CPU then executes the appropriate service routine for that card. This normally clears the device interrupt. The software then sends an End of Interrupt (EOI) to the ICU. If the same interrupt is still pending (interrupt sharing) the CPU checks the next card that could activate the interrupt signal for an active interrupt pending bit. If the bit is active the CPU services the interrupt as described above. It continues trying to clear this interrupt by checking and servicing the cards left in the chain until the interrupt clears or a higher priority event occurs.

Devices sharing the same interrupt level may have a lengthy wait before the CPU can service their interrupts. This can have a significant impact on the performance of a serial channel receiver. Since the receiver has no immediate control over the arrival of the incoming data without CPU intervention, it must be able to store the data for a period longer than the interrupt latency time of the CPU. The NS16550A has a provision for long interrupt latencies. Both the receiver and transmitter have 16-byte FIFOs. The CPU enables these FIFOs by writing an x1 hex to the third register (FCR) of the UART. All UARTs that have this FIFO capability will set two indicator bits in their Interrupt Identification Register (IIR).

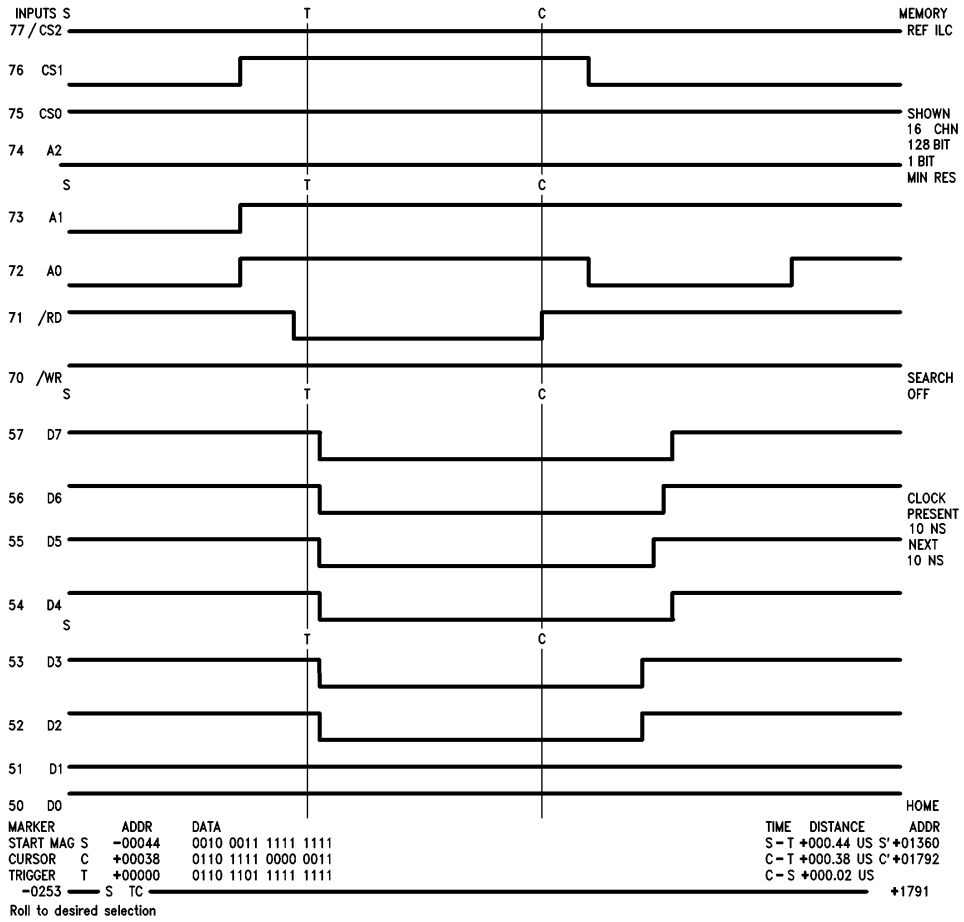
Therefore, the software must read a Cx hex from the third register (IIR) of the UART before relying on the FIFOs. The receiver FIFO buffers incoming data. As this receiver FIFO fills, it will activate an interrupt. The CPU programs the level of receiver FIFO ''fullness'' needed to trigger this interrupt. The CPU selects one of these interrupt trigger levels during the UART initialization. This programmable trigger level allows the receiver FIFO to accommodate varying system interrupt latency times. When the receiver FIFO fills to this trigger level the UART issues an interrupt.

Another advantage of having the FIFOs on both the transmitter and receiver is the reduction in the number of interrupts the CPU must process. The NS16550A with enabled FIFOs can issue $1/16$th the number of transmitter interrupts to the CPU as compared to one with disabled FIFOs. The reduction in receiver interrupts is proportional to the number of bytes stored in the FIFO before the programmed trigger level is reached. Handling fewer interrupts reduces the amount of overhead the CPU needs to execute. Using the transmitter FIFO allows the CPU to send the same amount of data while executing $1/16$th the overhead.

The PS/2 architecture is well structured. It has many of the data and control pathways one expects in a more advanced system and some unexpected ones. This structure ensures that the software accesses to the NS16550A are identical, whether the UART is on the system board or an MCA adapter card. Allow the programmer needs to know is the serial port address of the target UART and its register set. The timing of signals accessing the UART on the MCA Bus is identical regardless of the system CPU. The timing of these signals is more than adequate for reliable operation of the NS16550A. As more multitasking and multiuser applications are written for the serial channels, the benefits of on-chip FIFOs will be apparent.

APPENDIX A

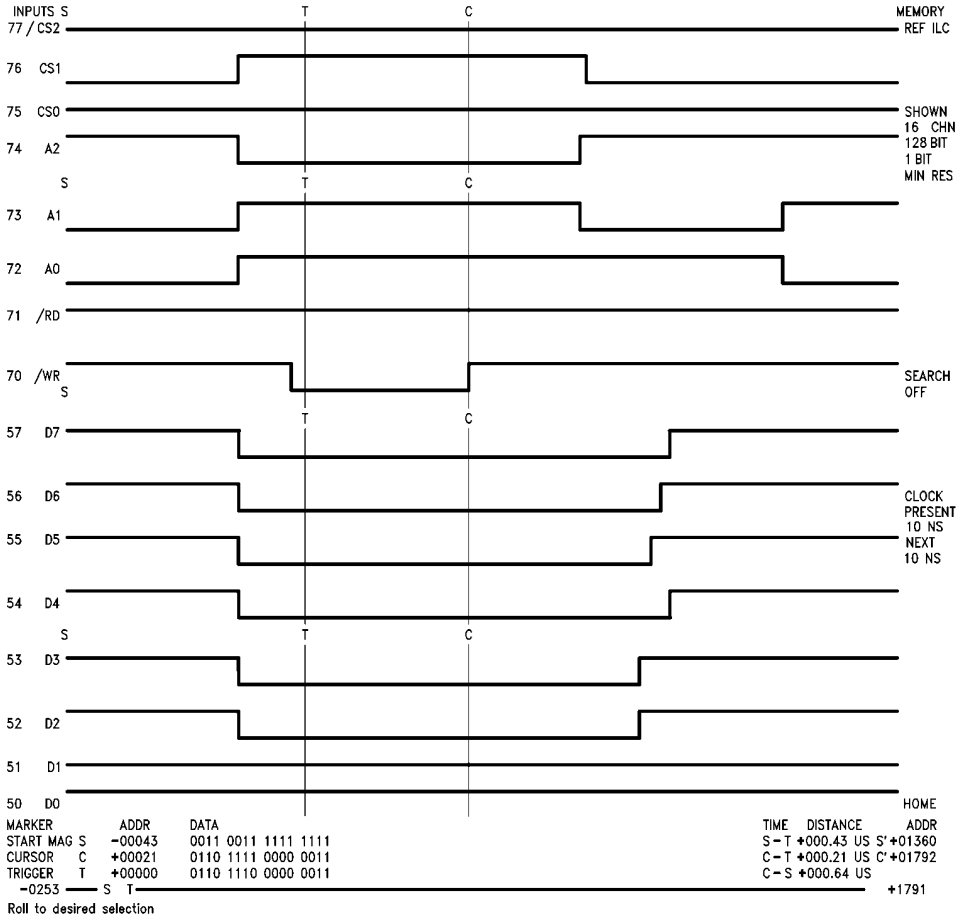
Dual ASYNC Card Read and Write



TL/C/10456–5

PS/2 Model 50 & Model 80 read from Dual ASYNC card (03 from LCR) read width 390 ns

Data file M50 IORD-Ref, Kontron PS/2 info Disk

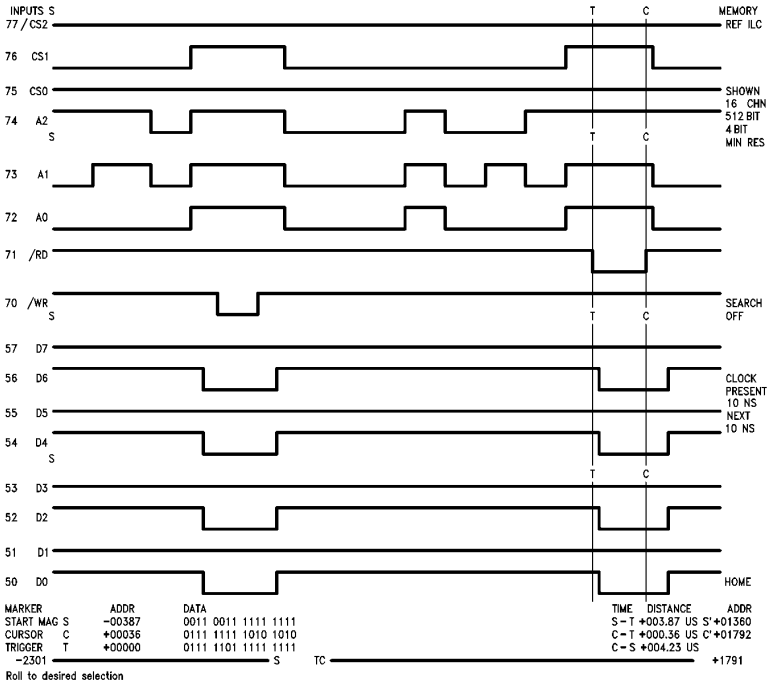


TL/C/10456–6

PS/2 Model 50 & 80 write to Dual ASYNC card (03 to LCR) write width 220 ns

Data file M50 IOWR-Ref, Kontron PS/2 info Disk

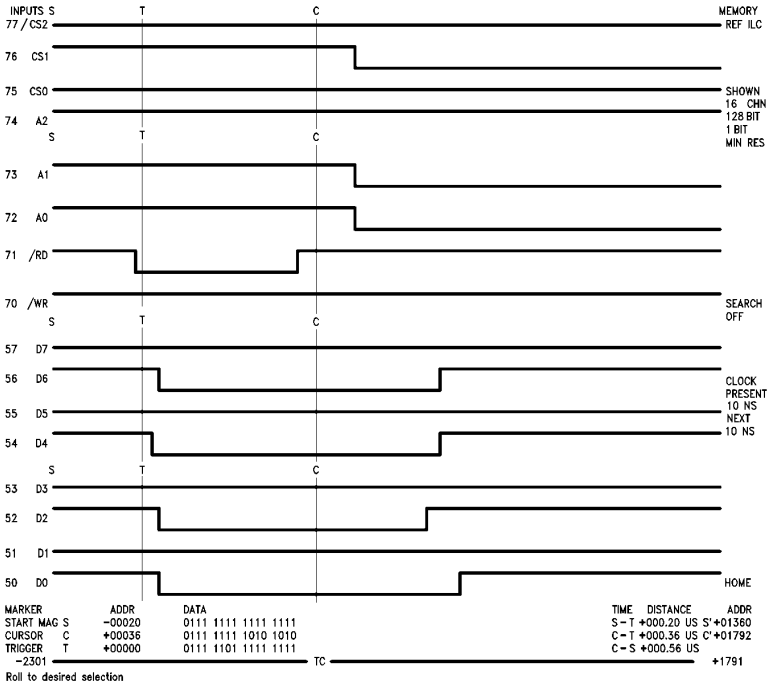
Back-to-Back Write to SCR and Read from SCR



TL/C/10456–7

Back to back write to SCR and read from SCR during power-up

Write width 220 ns
Read width 340 ns
time /wr to /rd model 50 279 ns, model 80 192 ns

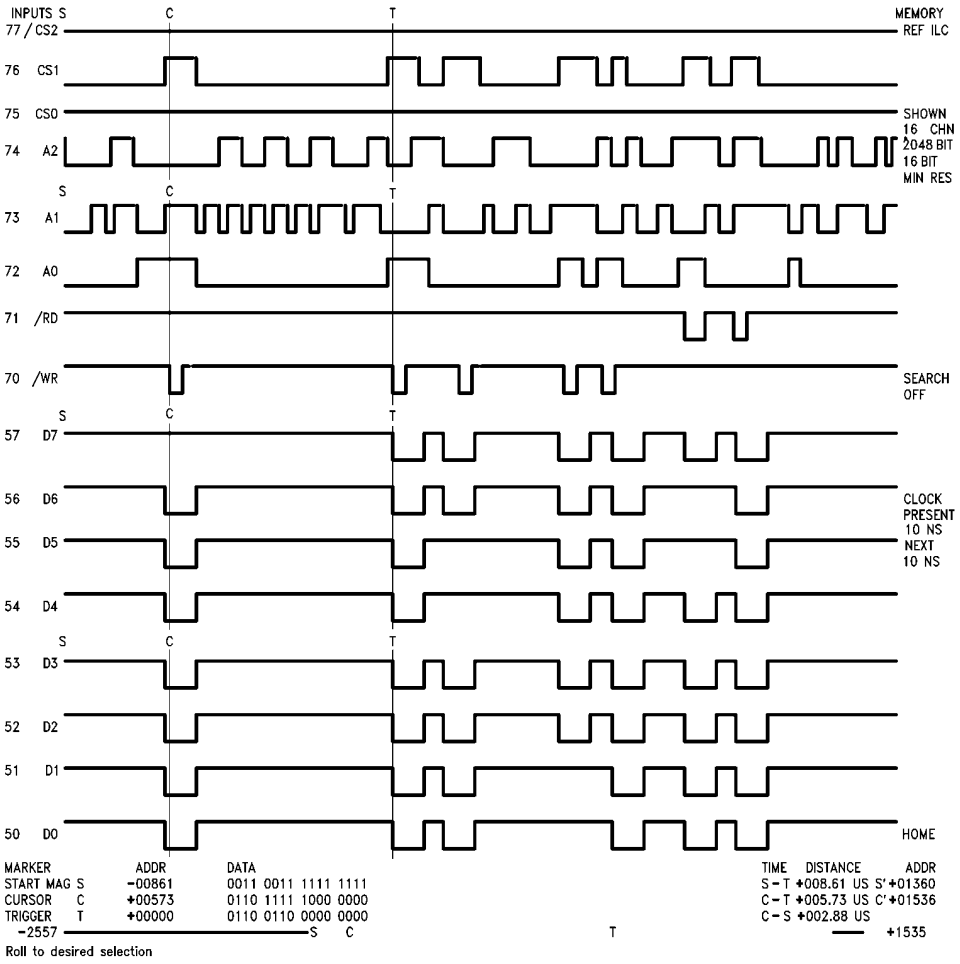


TL/C/10456–8

Detail of read cycle shown above

File M50 bk-bk ref, Kontron PS/2 info Disk

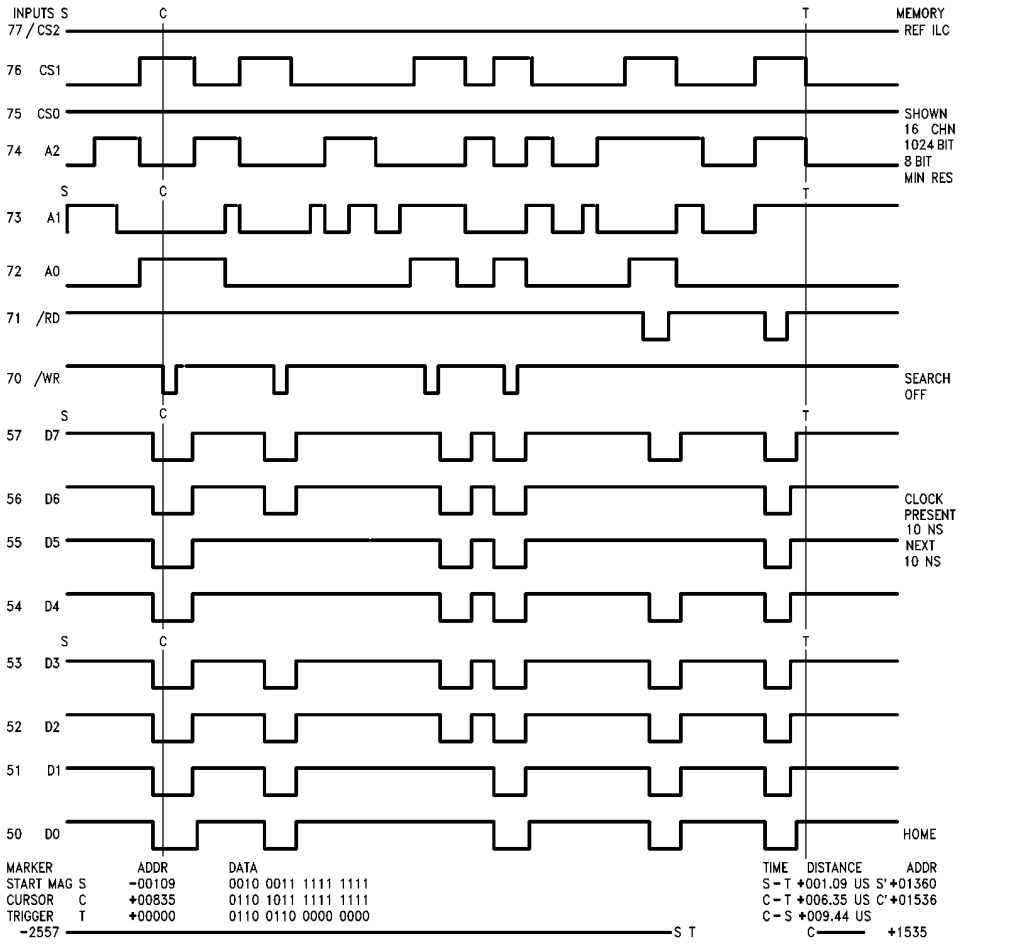
COM2 Initialization



TL/C/10456–9

PS/2 Model 50 and 80 Dual ASYNC card initialization for COM2. The cursor shows the second write to the UART. The first write is shown on the ''Back-TO-Back write to SCR Read from SCR'' Timing Diagram

Detail of Last Part of Initialization



TL/C/10456–10

Initialization Sequence (Including SCR)

WR	SCR	AAH	Check for 8250-B Part
RD	SCR	AAH	
WR	CCR	80H	Set Dlab
WR	DLH	00H	Set Baud to 2400
WR	DLL	30H	
WR	LCR	03H	8 Data, 1 Stop, no Parity
WR	IER	00H	Disable Intrs
RD	LSR	60H	Clear Status
RD	MSR	00H	Clear Status

Lit. # 100628

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

National Semiconductor Corporation
2900 Semiconductor Drive
P.O. Box 58090
Santa Clara, CA 95052-8090
Tel: 1(800) 272-9959
TWX: (910) 339-9240

National Semiconductor GmbH
Livry-Gargan-Str. 10
D-82256 Fürstenfeldbruck
Germany
Tel: (81-41) 35-0
Telex: 527649
Fax: (81-41) 35-1

National Semiconductor Japan Ltd.
Sumitomo Chemical
Engineering Center
Bldg. 7F
1-7-1, Nakase, Mihama-Ku
Chiba-City,
Ciba Prefecture 261
Tel: (043) 299-2300
Fax: (043) 299-2500

National Semiconductor Hong Kong Ltd.
13th Floor, Straight Block,
Ocean Centre, 5 Canton Rd.
Tsimshatsui, Kowloon
Hong Kong
Tel: (852) 2737-1600
Fax: (852) 2736-9960

National Semiconductores Do Brazil Ltda.
Rue Deputado Lacorda Franco
120-3A
Sao Paulo-SP
Brazil 05418-000
Tel: (55-11) 212-5066
Telex: 391-1131931 NSBR BR
Fax: (55-11) 212-1181

National Semiconductor (Australia) Pty, Ltd.
Building 16
Business Park Drive
Monash Business Park
Nottinghill, Melbourne
Victoria 3168 Australia
Tel: (3) 558-9999
Fax: (3) 558-9998