

Tandy 1000 TX

Technical Reference Manual

TANDY®

TANDY COMPUTER PRODUCTS

**Tandy 1000 TX
Page Insertion Guide**

Important Customer Note:

A gray stripe has been printed along the right edge of the title page of each of the sections to facilitate your finding the begining of the section. Also, a tabbed divider for each section has been provided for insertion at this point.

**Foldout Pages To Be Inserted At The End
of the Main Logic Section**

System Clock	
Timing Diagram	Sheets 1 of 3 thru 3 of 3
CPU Control	
Timing Diagram	Sheets 1 of 5 thru 5 of 5
Main Logic Schematic	
Rev.A	8000288
Rev.C	Sheets 1 of 12 thru 12 of 12

**Foldout Page To Be Inserted At The End
of the Disk Drive Section**

Exploded View	8790142
	Sheet 1 of 1
Schematic	8790142
	Sheet 1 of 1



TANDY COMPUTER PRODUCTS

**TANDY 1000 TX
TECHNICAL REFERENCE MANUAL
Cat. No. 25-1514**

TANDY COMPUTER PRODUCTS

**Tandy 1000 TX Technical Reference Manual
Copyright 1987, Tandy Corporation
All Rights Reserved.**

Reproduction or use of any portion of this manual without written permission from Tandy Corporation and/or its licensor, is prohibited. While reasonable efforts have been taken in the preparation of this manual to assure its accuracy, Tandy Corporation assumes no liability resulting from any errors in or omissions from this manual, or from the use of the information obtained herein.

**Tandy 1000 TX
Technical Reference Manual
Contents**

Sections

Main Logic Board
Devices
Power Supplies
Keyboard
Disk Drive
Options
Software

Important Customer Note:

A gray stripe has been printed along the right edge of the title page of each of the sections to facilitate your finding the begining of the section. Also, a tabbed divider for each section has been provided for insertion at this point.

TANDY COMPUTER PRODUCTS

1000 TX Main Logic Board

1000 TX Main Logic Board
Contents

Section	Page
Introduction	3
Specifications	4
Standard Features	4
Optional Features	5
Physical Specifications	6
Switch Settings and Jumper Pin Configurations	7
Theory of Operation	8
80286 Processor	8
80287 Processor	9
Clock Generation	10
Command and Control Signal Generation.....	12
Command Buffer	13
DRAM Control	13
Refresh Control	15
BIOS ROM Control	15
Reset Circuit	15
Wait State and Ready Logic	16
NMI Logic	16
80287 Control Logic	16
CPU Address Buffers	19
Data Buffers and Conversion Logic	20
I/O Decode	21
Floppy Disk Controller	21
Interrupt Controller	22
Video Controller	23
Timer Counter	24
Joystick Interface	24
Keyboard Interface	25
Sound Circuit	25
DMA Controller	26
RS232 Serial Port Interface	27
Parallel Printer Port Interface	27
Expansion Ports.....	28
System Expansion Bus.....	28
Expansion Bus Signal Descriptions	29
Memory Map of System	32
I/O Port Map of System	33
Timing Charts	59
System Clock Timing	59
CPU Control Timing	71
Schematic Diagrams	81

INTRODUCTION TO THE TANDY® 1000 TX

The Tandy® 1000 TX is modular in design to allow maximum flexibility in system configuration. The computer consists of a main unit, a detachable keyboard with coiled cable, and a monitor. The main unit is supplied with one internal 3½" 720K floppy disk drive. The standard types of monitors used with the Tandy 1000 TX are the monochrome composite and the color RGB monitor. Since these units are modular, you can place them on top of the main unit or at any convenient location.

The Tandy 1000 TX comes standard with 640K of RAM. An optional 128K upgrade can be added to the main logic board, for a total of 768K of RAM. The upper 128K is used exclusively as video RAM, leaving 640K for system use, which is the maximum RAM supported by the TX memory map.

The Tandy 1000 TX comes standard with 640K of system RAM. An optional 128K RAM can be added on the system board to expand the memory to a full 768K bytes, the maximum RAM allowed by the system memory map.

Other features include a parallel printer port, a serial port, two built-in joystick interfaces, a speaker for audio feedback, and a headphone jack with volume control.

The main unit is the heart of the Tandy 1000 TX. It houses the main logic assembly, system power supply, and floppy disk drive.

The main logic assembly is a large board mounted to the bottom of the main unit and interconnected to the keyboard, power supply, and disk drive by a series of cables.

The power supply is a 67W switching regulator type, designed to provide adequate power capacity for a fully configured system that has all the option slots in use.

The floppy disk drive uses 3½" double-sided, double-density diskettes to read, write, or store data. These are soft sector diskettes. The disk drive assembly comes installed in the main unit. The floppy diskette stores approximately 720K bytes (formatted) of data. All system programs, with the exception of the system startup sequence, are stored on diskette.

SPECIFICATIONS

Standard Features

- 80286 CPU running at 8MHz, switchable to 4.77 MHz
- Socket for 80287 numerical co-processor
- 640K bytes DRAM upgradeable to 768K bytes (16 bit data bus)
- 16K bytes BIOS ROM (16 bit data bus)
- Tandy 1000 TX video controller that supports:
 - 128K bytes DRAM (used as system and video memory)
 - alphanumeric mode
 - graphics modes including:
 - 160 X 200 16-color
 - 320 X 200 4-color
 - 320 X 200 16-color
 - 640 X 200 2-color
 - 640 X 200 4-color
- 8237-5 DMA controller that supports:
 - 4 DMA channels
 - 8 bit transfers
 - 4 MHz clock speed
- 8259A interrupt controller for 8 interrupts
- 8253 interval timer that supports:
 - system interrupt timing
 - sound timing
 - refresh timing
- Custom keyboard interface controller
- 90-key keyboard that includes 12 function keys
- Custom parallel printer port
- Serial port (RS-232-C)
- Audio interface circuit that supports:
 - internal 8-Ohm speaker
 - headphone jack with user accessible volume control
- Joystick interface for two joysticks
- Custom floppy disk controller circuit that supports:
 - 5 $\frac{1}{4}$ " 360K floppy disk drives
 - 3 $\frac{1}{2}$ " 720K floppy disk drives
- One 3 $\frac{1}{2}$ " 720K floppy disk drive
- Five 8-bit expansion slots
- Reset button and support logic
- 67-Watt power supply

Optional Features

- 80287 numerical math co-processor
- 128K DRAM upgrade (16-bit data bus memory)
- 5 $\frac{1}{4}$ " 360K floppy disk drive
- 3 $\frac{1}{2}$ " 720K floppy disk drive
- Hard disk controller
- Hard disk card (20 meg)
- Display adapter boards that support mono, EGA, or other special video modes
- 300, 1200, or 2400 baud modem boards

Physical Specifications

(Computer and Keyboard)

Processor: Intel® 80286

Dimensions: Computer - 16" x 13½" x 5"

Keyboard - 16½" x 8" x 1½"

Weight: Computer - 18 lbs. (with 2 floppy disk drives)
Keyboard - 3 lbs. 4 oz.

Power Requirements:

Range: 105 VAC to 135 VAC

Nominal: 120 VAC, 60 Hz, 3 Amp maximum

With 1 Floppy Disk Drive, 640K Memory:

AC Current: 350 - 400 mA with floppy doing R/W tests.

Disk Drive:

	+5 VDC	+12 VDC
R/W	560 mA (Min.)	340 mA (Max.)

Main Logic Board: 1700 mA 450 mA

Operating Environment:

Temperature: 55 to 85 degrees F (13 to 30 degrees C)

Humidity: 40% to 80% non-condensing

Non-Operating Environment:

Temperature: -40 to +160 degrees F (-40 to 71 degrees C)

Humidity: 20% to 90% non-condensing

Disk Drive Specifications

Power:

Supply Voltage	+5 VDC Input	+12 VDC Input
Ripple 0 to 50 kHz	100 mV	100 mV
Tolerance Including Ripple	+/-5%	+/-5%
Standby Current Nominal	190 mA	160 mA
Worst Case	220 mA	190 mA
Operating Current Nominal	260 mA	600 mA
Worst Case	300 mA	1000 mA

SWITCH SETTINGS AND JUMPER PIN CONFIGURATIONS

Jumper	Function	Interrupt	As Shipped
S2-1	Color/Monochrome Monitor on = color off = monochrome/composite		on
S2-2	Interrupt 5 Sync Source on = internal sync off = external sync	Int. 5	on
S2-3	FDC Interrupt on = on board FDC controller off = optional FDC controller	Int. 6	on
S2-4	Printer Interrupt on = enable printer interrupt off = disable printer interrupt	Int. 7	on
E1-E2	Serial (Com) Port Enable installed = port enabled not installed = port disabled	Int. 4	installed
E3-E4	COM Port Select installed = COM1 selected not installed = COM2 selected		installed
E9-E10	Memory Configuration installed = 512K system memory not installed = 640K system memory		installed

Table 1. Switch Settings and Jumper Configurations.

THEORY OF OPERATION

80286 Microprocessor

The 80286 (U13) is an advanced, high-performance, 16-bit microprocessor with special capabilities for multi-tasking and multi-user systems. Two modes of operation are available in the 80286, the Real Address mode, and the Protected Virtual Address mode. In the Real Address mode, the 80286 is compatible with existing 8086 and 8088 software and allows addressing of one megabyte of memory space. The Tandy 1000 TX does not support the Protected Virtual Address mode.

Optional 80287 Numerical Math Co-Processor

The optional 80287 (U15) performs high-speed arithmetic and logarithmic functions and trigonometric operations that increase the performance of an 80286 system. Performance increases are obtained by the 80287's ability to perform math calculations faster than the 80286, and also by the executing math instructions in parallel with the 80286.

Clock Generation (Night Blue)

All clocks required by the system are generated by the custom CPU Controller (U52). There are two independent clock circuits supplied by a Dual Oscillator Clock (Y1) from which all other clocks are derived.

The 16 MHz Clock is routed into the CPU Controller (U52), which generates the output signals PRCLK, DMACLK, and SCLK. The Clock Switch circuitry required to toggle the 80286 Microprocessor between 8 MHz and 4 MHz mode, as well as the logic to prevent any short cycling during a clock switch cycle, are implemented in the CPU Controller IC (U52). If the signal XD3 is asserted high during an I/O write to port 062 (hex), then the output signal PRCLK is 16 MHz, which operates the 80286 in 8 MHz mode. If the signal XD3 is asserted low during an I/O write to port 062 (hex), the output signal PRCLK is at 8 MHz, operating the 80286 in the 4 MHz mode. When Reset is generated, the signal RES* is asserted low and defaults the Tandy 1000 TX to the 8 MHz mode.

The CPU Controller Chip also controls wait states to insert the proper number of wait states required for a two clock mode of operation. When the PRCLK signal is 16 MHz (8 MHz Mode), then four wait states are inserted in all 8-bit Memory and I/O cycles. When the signal PRCLK is 8 MHz (4 MHz mode) then two wait states are inserted during all 8-bit Memory and I/O cycles. During all 16-bit memory cycles, only one wait state is inserted in both the 8MHz and 4MHz modes.

PRCLK is then routed through three damping resistors to produce the signals PRCLK for the 80286, PRCLKA for the 80287 math co-processor, and PRCLKB for the DRAM/DMA control logic.

DMACLK and SCLK are output signals for system use. The DMACLK output frequency is $\frac{1}{4}$ of the PRCLK signal, and the SCLK output frequency is $\frac{1}{2}$ of the PRCLK signal. Both are synchronized by Reset to the PRCLK output signal. After a Reset, DMACLK and SCLK are held low until the 80286 asserts status S1 = 0. SCLK and DMACLK make the first transition on the falling edge of PRCLK, following with a TS state that synchronizes them to PRCLK.

SCLK is buffered by IC U68 (74ALS244) and filtered by R51, C198, and FB7, and is then routed to the Expansion Bus for option board use. DMACLK is filtered by R37, FB3, and C171, and is then routed to the DMA Controller (U11).

Table 2 shows all the clocks generated from 16 MHz in both modes:

	8 MHz Mode	4 MHz Mode
PRCLK	16 MHz	8 MHz
SCLK	8 MHz	4 MHz
DMACLK	4 MHz	2 MHz

Table 2. Clocks Generated From 16MHz.

The CPU Controller generates three other clocks. They are derived from the 28.63636 MHz Clock Oscillator (Y1). These are OSC, 3.58 MHz, and 1.19 MHz. The 28MHZ input signal is divided by 2 to generate OSC, which is buffered by U68 and filtered by R49, C190, C190A, and FB6. OSC is then routed to the expansion bus slots for video clocks for any optional video boards. The 28MHz signal is also divided by 8, generating the 3.58 MHz clock which is routed to the Sound Generator (U7). 28MHz is also divided by 24 to become the 1.19 MHz clock, which is routed to the clock input of the Interval Timer 8254-2 (U12).

Command and Control Signal Generation

The command and control signals required for the Tandy 1000 TX operation are generated by the CPU Controller (U52). The command signals are decoded from the CPU status signals S0*-S1* and M/IO* during the Ts cycle. The decoded signals indicate the type of cycle that is to be executed (MEMR*, MEMW*, IOR*, IOW*, INTA*). The control signals (ALE, DT/R*, DSDEN0*, DSDEN1*, MEMCYC) control the latching of addresses, determine the direction and enabling of the data bus buffers, and start a memory cycle. Table 3 indicates the decoding of the CPU status signals.

M/IO*	S1*	S0*	Type of Bus Cycle
0	0	0	Interrupt Acknowledge
0	0	1	I/O Read
0	1	0	I/O Write
0	1	1	None: Idle
1	0	0	Halt or Shutdown
1	0	1	Memory Read
1	1	0	Memory Write
1	1	1	None: Idle

Table 3. CPU Status Signal Decoding.

A0 and BHE* are decoded to determine the data transfer width to and from the CPU. Table 4 shows the data transfer width depending on the state of A0 and BHE*.

BHE* A0	Width of Data Transfer
0 0	Word Transfer
0 1	Byte Transfer D8 ~ D15
1 0	Byte Transfer D0 ~ D7
1 1	Not Used

Table 4. Data Transfer Width Decode.

Command Buffer

Some of the command signals generated by the CPU Controller (U52) require buffering to the system. The ICs (U68) and (U22) implement this function. The ICs buffer the command signals to the system bus for the expansion bus slots, the peripheral devices, and also the Video Controller. The LMEGCS* and CPUHLDA signals control the buffers. When CPUHLDA is asserted active (high), it disables buffer (U22). (Outputs are tri-stated.) This indicates to the system that the DMA Controller has accessed the system bus. When CPUHLDA is negated (low), then the buffers are enabled, which allows the bus to be driven.

DRAM Control

The CPU address decode for the Dynamic Random Access Memory (DRAM) array is generated by the Custom DRAM/DMA Control IC (U53). These signals are latched by ALE internally to the DRAM/DMA Control IC and held for the complete cycle. The address decode signals are RAS0*, RAS1*, RAS2*, CAS1*, and CASH*. These are decoded by a certain address, depending on the type of DRAM used and the configuration of the MCO and MCL jumpers. Memory configurations supported by the Tandy 1000 TX are 640K bytes or 768K bytes (including 128K of video memory). Table 5 shows the different options available on the DRAM/DMA Control IC.

Memory Option	Jumper Config	MCO	MCL	System Memory	Total System Memory*	Control	Bank	Address Range
1	E9-E10	0	1	512K	640K	RAS0	512K	000000-07FFFF
2	None	1	1	640K	768K	RAS0 RAS1	512K 128K	000000-07FFFF 080000-09FFFF

• Note: Total system memory includes 128K of video memory.

Table 5. Memory Configurations.

MEMCYC triggers the control signals for the DRAM array. The MEMCYC signal (generated by the CPU Controller (U52) indicates that a DRAM BUS cycle is in progress. MEMCYC enables RAS0*, RAS1*, RAS2*, CASH*, and CASL*, depending on the address of the bus cycle. The selected RAS(x)* lines become active at the next falling edge of PRCLKB.

After $\frac{1}{2}$ PRCLKB cycle at the rising edge of the clock, MUX is generated and switches the DRAM address (MA0-MA8) from Row Address to Column Address. After another PRCLKB cycle at the next rising edge of the clock, CASL* and/or CASH* are asserted. Two CAS signals are generated internally to the DRAM/DMA Control IC to provide the ability to access word or byte cycles in the DRAM array. Table 6 shows the state of each control signal during each type of bus cycle.

Address Range	Bus Width	RAS0*	RAS1*	RAS2*	CASL*	CASH*
000000-07FFFFH	Even	Byte	0	1	1	0
000000-07FFFFH	Odd	Byte	0	1	1	0
000000-07FFFFH	Even	Word	0	1	1	0
080000-09FFFFFFH	Even	Byte	1	0	1	0
080000-09FFFFFFH	Odd	Byte	1	0	1	0
080000-09FFFFFFH	Even	Word	1	0	1	0

Table 6. Signal State Control Signals.

The signals WE0* and WE1* provide read and write control. Both are asserted at the same time by IC (U24) and are controlled by MEMW* (memory write). If WE0* and WE1* are asserted high, it is a read cycle; if they are asserted low, it is a write cycle.

Refresh Control

The OUT1 pin of the 8254-2 Internal Timer (U12) generates an active high pulse every 15 usec. The rising edge of the OUT1 signal clocks $\frac{1}{4}$ of a 74LS74 (U4), which generates the signal REFREQ. REFREQ is an input signal to the 8237 DMA controller (U11). This input to U11 is actually Data Transfer Request 0, (DREQ0), which requests the DMA to perform a DMA cycle. The DMA controller channel 0 has been programmed to perform a single transfer from memory to an I/O device, such as a floppy drive. This causes a memory read at a certain address to be performed. Each time the REFREQ signal is generated, the DMA controller increments the address and performs another memory read. This causes all memory rows to be read every 4ms to keep data in the DRAMs stable. Refer to the section on the DMA controller for more information on DMA cycles.

BIOS ROM Control

The DRAM/DMA IC (U53) provides the CPU address decode used for the ROM select. The signal generated is called ROMCS* (ROM Chip Select) and is used as the Chip Enable for the BIOS ROMs U38 and U39. This output is asserted whenever any of three addressed ranges is detected, CPHLDA is inactive, and ALE is asserted. The three address ranges are 0E0000-0FFFFFH, EE0000-EFFFFFH, and FE0000-FFFFFFH. The address lines S11-S15 are provided to the BIOS ROMs for lower address control. The standard size of EPROM used in the Tandy 1000 TX is 16K X 8 (128K bit). The 1000 TX accepts larger EPROMs of 32K X 8 (256K bit) if required.

The data is buffered onto the MD0-MD15 data bus, controlled by the 82C205 IC (U14).

Reset Circuit

The CPU Control IC (U52) controls the system reset required either to initialize the complete system after power-up or to reboot. Two reset output signals, RESET and RESCPU, are active high and generated when a power-up condition is detected or when the reset button on the front of the computer is pressed.

The RESET signal is used as a general system reset, while the RESCPU signal resets the 80286 Processor. The RES* signal is the input to (U52), which signifies a reset condition. The RES* signal is generated either from an RC network during power up or from the reset switch. During a power-up, RES* is held low for the time period generated by the RC time constant of R24 and C135. This is the time it takes C135 to change to an active high. Also, if the reset switch is pressed, it applies a ground to C135 and discharges it. This asserts RES* low until C135 charges again to a logic high.

During power-up, the RES* signal is generated twice to provide a proper reset to the CPU control IC (U52). A second RC time constant is generated by R47A and C171A to the input of a schmidt trigger inverter (U66). This holds the input pin of U4 ($\frac{1}{2}$ of a 74LS74) high for approximately 150ms. When RESCPU is negated after the first reset, the CPU issues the first command to the CPU control IC by driving SI* low. This generates the first rising edge of DMACLK, which latches into U4. The Q output of U4 is then routed to an open collector inverter, (U67), which discharges C135, again asserting RES* low and generating the second reset. CR1A provides a reset to U4 when C135 is discharged to at least .7v. After U4 is reset, RES* is released, and C135 is allowed to charge, negating RES* and finishing the second reset pulse. When the CPU issues the first command again, which starts DMACLK, U4 latches a low. This is because the D input of U4 has transitioned to a low by the end of the second reset.

The CPU Control IC (U52) also internally controls the RESCPU signal to meet the requirements of the 80286 during a detected shutdown condition.

Wait State and Ready Logic

Wait state control is implemented internally to the CPU Control IC. The function of the wait state control logic is to match the speed of the various devices in the Tandy 1000 TX to the speed of the 80286 CPU. Two circuits assert wait states within memory and I/O cycles. One method is controlled by the device being accessed, using the IOCHRDY signal input to the CPU Control IC. If a device requires additional wait states within the bus cycle, the device should negate IOCHRDY low until it can service the bus cycle. After the required number of wait states have been inserted, the device should assert IOCHRDY, causing the READY* output of the CPU Control IC to be asserted low, which tells the CPU to terminate the cycle.

The second method (internal to the CPU Control IC) is several default wait states during the various bus cycles. During a 16-bit memory cycle (which is determined by the assertion of AFL16*), one wait state is automatically inserted. The default of an 8-bit memory or I/O cycle is four wait states. This can be overridden by driving IOCHRDY low as mentioned above. As long as IOCHRDY is at a logic low level, wait states are inserted indefinitely.

Note: IOCHRDY should not be held low for longer than 15 usec because it will stop DRAM refresh cycles.

NMI Logic

In the Tandy 1000 TX, the Non-Maskable Interrupt (NMI) indicates an I/O error condition or a Numerical Math Co-Processor 80287 error condition signal (NMI*). Both error conditions are enabled by the NMIEN signal, which is generated from the Video Controller, Big Blue. The INT287* signal (from the CPU Control IC) becomes active when the ERROR* signal is asserted by the Numerical Math Co-Processor.

80287 Control Logic

Incorporated into the CPU Control IC is the logic required to interface the 80287 Math Co-processor to the 80286 CPU. This logic decodes the signals that select and reset the 80287 and also handles the Busy*/Error* signals from the 80287 to the CPU.

The input signal 287CS* is a user I/O address decoded signal used by the CPU Control IC to generate the control signals to the 80287 IC. The 287CS* signal is asserted during I/O address 0F0h - OFFh. Further decoding is provided by the CPU Control IC, which generates RES287, NPCS*, and BUSY287* signals. Table 7 defines the internal decode.

Hex Address	Description
0F0	Clear Math Co-processor Busy
0F1	Reset Math Co-processor Busy
0F8-0FF	Math Co-processor Chip Select

Table 7. 287CS* Decode.

Given the command to perform a task, the 80287 co-processor issues a BUSY* signal to the CPU Control IC. With the assertion of the BUSY287* output, this signal is passed to the CPU. Normally the BUSY* input is passed through to the BUSY287* output. Deassertion of BUSY* results in deassertion of BUSY287*. The BUSY287* output is latched, and the INT287* output pin is forced HIGH during this busy period if the ERROR* input becomes active (signaling a Numerical Processor error). Until cleared by an I/O write cycle to address 0F0h or 0Flh, both signals then remain active. Both the interrupt latch for INT287* and the busy latch for BUSY287* are cleared after a system reset.

The CPU Control IC's RES287 output pin handles resetting of the 80287 co-processor. This can be activated by a system reset or an I/O write to address 0F1h. This signal is active only for the period of time that the source signal is active, as it is not latched internally.

CPU Address Buffers

U20, U21 (74ALS573), and U22 (74ALS244) implement buffering of the address lines to the system. SA01-SA19 are buffered and latched for the expansion bus slots and I/O peripherals. ALE is used to latch SA01-SA19 and held for the complete bus cycle. SA01-SA19 are also used to address the BIOS ROMs and DRAM/DMA Control. A17-A23 are routed directly to DRAM/DMA control to generate memory address decoded signals. The multiplexed address lines MA0-MA7 are also generated and buffered to the DRAM memory by the DRAM/DMA Control IC (U53). To meet address requirements for the DRAMs, the MUX signal multiplexes SA1-SA16 internally to the DRAM/DMA Control IC.

During a DMA cycle, a 74LS245 (U3), is used to buffer S0-S7 directly from the DMA controller (U11). S8-S16 are buffered internally to the DRAM/DMA controller. S17-S19 are buffered by U22 (1/2 of a 74ALS244).

Data Buffers and Conversion Logic

The 82A205 IC (U14) provides the data buses, buffers, and drivers for D0-D15 to the system. Three data buses are generated, SD0-SD7 for the expansion bus slots, MD0-MD15 for memory access from ROM and DRAM, and D0-D15, which is routed to the 80286 CPU and 80287 Co-processor data bus. The direction and control of the data buffers are provided by the input signals to the 82A205 IC (DT/R*, DSDEN0*, DSDEN1*, SBHE*, and SA0).

DT/R* controls the direction of the data path during a read or write. The DSDEN0* and DSDEN1* signals control the word and byte data transfers, while SBHE* and SA0 determine the buffers to be enabled during a byte access.

Conversion logic is also implemented in the 82A205 IC, controlled by ENHLB* and DIRHLB. This conversion logic allows data to be transferred from the lower to upper or upper to lower data byte to meet the requirements of the CPU or receiving device.

A 74ALS245 buffers the Internal Peripheral Bus XDO-XD7 from the System Data Bus SD0-SD1. The enabling and direction are controlled by the signals XBUFDIR* and XBUFEN*, which are generated by a PLS173 IFL (U19).

I/O Decode

The Video Control IC (U36), a 74HCT138 (U49), a PLS173 IFL (U19), and a 74HCT138 IC (U34) accomplish the I/O Address decoding. These four ICs provide all the necessary chip select signals to the system. The AA0, BA1, and CA2 output signals of the Video Controller are encoded device select lines that are fed directly to the 74HCT138 IC, in which I/O address decoding is generated. The second 74HCT138 (U34) and the PLS173 (U19) decode the system address to generate the other I/O address decode select signals. Refer to the Tandy 1000 TX I/O Map for details.

Floppy Disk Controller

The onboard Floppy Disk Controller (FDC) interfaces the system to the Floppy Disk Drive (FDD). Up to two internal 720K (double density MFM format) FDDs can be accommodated.

The FDC circuit can be organized into the following subsections:

- uPD765A FDC Chip
- System Interface
- Clock Generation
- Precompensation
- Data Separator
- Disk Drive Interface

uPD765A Chip. The uPD765A FDC chip (U51) integrates most of the control logic necessary to:

- interface the Serial bit stream to or from the FDD to the parallel bus of the system
- implement the commands necessary to operate the FDD
- maintain information about the status of the FDD

During a read or write data operation to the FDD, the FDC chip generates a DMA request for a byte transfer to or from memory. The FDC chip continues to generate DMA requests until the preprogrammed amount of data is transferred as signified by generation of a Termination Count (TC) Signal. After the TC is reached, the FDC chip generates an interrupt to the system through INTFDC so that status and result data can be serviced.

Refer to the device data sheet for complete descriptions of the available commands and the command and status registers.

System Interface. Various ICs latch and buffer data to and from the system. A DORWR* is generated at pin 6 of (U62) on an I/O write to port 3F2 (hex). This signal latches the data byte that is bit defined as the Drive Select, XDS0* and XDS1*, Motor On, MTRON*, DMA and Interrupt Request (DMA/INTE), and a reset signal to the FDC controller U51 (FDCRST*).

Clock Generation. The FDC Support IC (U64) generates all clocks required by the Floppy Disk circuit. These clocks are derived from a 16 MHz input signal. FDCLK, required by the FDC Controller (U51), is derived by dividing the 16 MHz clock by 4. The resulting 4 MHz clock is also used as a delay counter for the DMA request signal DRQ as well as a reference clock for the write precompensation circuit. The 4 MHz clock also generates a 250 nanosecond pulse at a frequency of 500 KHz. The 500 KHz signal is used as a write clock for the FDC Controller.

Precompensation. The precompensation circuit is implemented internally to the FDC Support IC (U64). The write data bit can be shifted either early or late in the serial bit stream, depending on the requirements of the Floppy Disk Drive. This function is programmable and controlled by the FDC IC signals PS0 and PS1.

Data Separator. The FDC Support IC (U64) also contains the data separator circuit. The data separator recovers the clock and data signals from the serial bit stream of the Floppy Disk Drive. The FDC Support IC supports only MFM or Double-density mode.

Disk Drive Interface. All FDC outputs to the FDD are driven by high current 7414 open collector buffers or 7416 open collector inverters. All FDC inputs from the FDD are buffered by 74HCT14 SCHMIDT triggered inverters. The inputs are pulled up on board by 150 ohm terminating resistors. All outputs should be terminated on the last FDD by 1500 ohm resistors.

Interrupt Controller

The 8259A Interrupt Controller chip (U16) supplies the maskable interrupt input to the CPU. The 8259A has eight interrupt inputs controlled through software commands. It can mask (disable) and prioritize (arrange priority) to generate the interrupt input to the CPU. The eight interrupts are assigned as follows:

#0	Timer Channel 0	Software Timer
#1	Keyboard	Keyboard Code Received
#2	Interrupt on the Bus	Hard Disk Controller
#3	Interrupt on the Bus	Modem
#4	Interrupt on the Bus	RS-232
#5	Vertical Sync	Optional Bus Interrupt
#6	Floppy Disk Controller	Optional Bus Interrupt
#7	Printer	Optional Bus Interrupt

Interrupts 0 and 1 are connected to system board functions as indicated in the chart. Interrupts 2-4 are connected directly to the Expansion Bus, with the normal assigned functions listed in the chart. Interrupts 5-7 are connected through a switch, (S2), to the system board function listed, and are also connected to the Expansion Bus. To use interrupts 5, 6, or 7 on the Bus, the system board function must be disconnected (by setting the appropriate switch to off). Note that disconnecting the normal system board function might cause some application programs to fail or operate incorrectly.

Video Controller

The next major block of the Tandy 1000 TX is the video interface circuitry. This custom part contains all the logic necessary to generate an IBM-compatible color video display. The video interface logic consists of the 84-pin custom video circuit (U36), four 64K X 4 RAMS (U18, U33, U35, and U48), a 74LS244 buffer (U71), and associated logic for generating composite and RGBI video.

The Tandy 1000 TX video interface circuitry controls 128K of memory. This RAM is shared by the CPU and the video. Normally, the video requires only 16K or 32K for the video screen, and the remainder of the 128K is available for system memory use.

The Tandy 1000 TX video interface custom circuit is composed of a 6845 equivalent design, dynamic RAM address generation/timing, and video attribute controller logic.

Normal function of the video interface custom circuit is as follows. After the 6845 is programmed with a correct set of operating values, a 4:1 multiplexer generates the address inputs to the dynamic RAMs. This MUX switches between video (6845) address and CPU address as well as between row and column address. Also, the video interface chip provides the RAM timing signals and generates a wait signal, VIDWAIT*, to the CPU for proper synchronization with the video RAM access cycles.

The outputs from the RAM chips are connected only to the video interface custom circuit, so all CPU read/write operations are buffered by this part. During a normal display cycle, video data from the RAM chips is first latched in the Video Attribute latch and the Video Character latch. The video interface requires a memory organization of 64K X 16 and latches 16 bits of memory during each access to RAM. From the output of the two latches, the data is supplied to the character ROM for the alpha modes or to the shift registers for the graphics modes. A final 2:1 MUX switches between foreground or background in the alpha modes.

From the 2:1 MUX, the RGBI data is combined with the PC color select data and latched in the Pre-Palette latch. This latch synchronizes the RGBI data before it is used to address the Palette. The palette mask MUX switches between incoming RGBI data and the palette address register. During a CPU write to the palette, this address register selects one of the 16 palette locations. Also, the Palette mask MUX allows any of the input RGBI bits to be set to zero.

The palette allows the 16 colors to be remapped in any desired organization. Normally, the palette is set for a 1:1 mapping (red = red, blue = blue, and so on) for PC compatibility. However, instantly changing the on-screen colors is a powerful tool for animation or graphics programs.

After the Palette, the RGBI data is resynchronized in the Post Palette register. The final logic before the RGBI data is buffered off the chip is the Border MUX. This MUX allows the Border to be replaced with any color selected by the border color latch. This latch is normally disabled in PC modes, but it is used in all PC jr modes.

Timer

The final Tandy 1000 TX function other than I/O is the 8254-2 timer chip (U12). This part is composed of three independent programmable counters. The clock for all three counters is 1.1925 MHz. Counters 0 and 1 are permanently enabled. Counter 2 is controlled by port Hex 0062, bit 0. Counter 0 is connected to system interrupt 0 and is used for software timing functions. Counter 1 is used for refresh function timing. Counter 2 is connected to the sound circuit and also to port Hex 0061, bit 5.

Joystick Interface

The joystick interface converts positional information from hand-held joysticks (1 or 2) into CPU data. Each joystick provides one or two push-buttons and X,Y position for a total of four bits each. You can use two joysticks.

The joystick handle is connected to two potentiometers mounted perpendicular to each other; one for X position, one for Y position. Through the cable, the main logic board applies +5 VDC to one side and ground to the other of the pots. The pot wiper is the position signal: a voltage between 0 and +5 VDC. This signal is applied to one input of a comparator U9. The other comparator input is the reference signal (a ramp between 0.0 to +5.0 volts).

When the position signal is equal to or less than the reference signal, the comparator output goes true. This comparator output is the X or Y position data bit. The ramp is reset to 0.0 VDC whenever an I/O Write is made at Port 200/201 Hex. The IOW* and JOYCFS* signals turn on Q2, which drains C114 to 0.0 volts. When Q2 is turned off, Q1, R15, R13, R14, and C11 create a constant-current source that linearly charges C114 to +5.0 VDC in 1.12 milliseconds. The joystick information is "read" by the CPU at Port 200/201 Hex through U10. See the Joystick Block Diagram.

Keyboard Interface

The next I/O function of the Tandy 1000 TX is the Keyboard interface custom circuit. The heart of this custom part is several read-write registers that are used to control the keyboard interface logic. For the interface to the keyboard connector, a 164-type shift register is used to load the serial data and allow the CPU to read it as 8 parallel bits.

Sound Circuit

The sound circuit is one of the five I/O functions of the Tandy 1000 TX. The circuit provides sound output for the internal speaker as well as for an external sound circuit.

The main source of sound in the Tandy 1000 TX is the 76496 complex sound generator (U7). This device has three tone generators and one white noise generator. Each tone generator can be programmed for frequency and attenuation. Also, this device has an audio input pin connected to the gated output of timer channel 2. This audio input signal is mixed with the sound generator signal and supplied to the audio output pin.

From the output of the 76496, the sound signal is connected to a dual analog multiplexer (U8). The multiplexer is switched by port 61, bit 4, and turns off the audio signal to the speaker, headphone jacks, and external audio output. The output of the multiplexer (U8) is routed to audio amplifiers U76 for the external audio output and U2 for the internal speaker and headphone jacks. The volume of the internal speaker can be adjusted by a user-accessible volume control (R6). When the headphone jack is used, the internal speaker is disabled.

DMA Controller

The major components of the Direct Memory Access (DMA) circuit consists of an 8237A-5 DMA controller (U11), the DRAM/DMA Control (U53), and a bi-directional address buffer 74ALS245 (U3).

Initialization--A DMA Operation. When a DMA operation is requested by software or by a peripheral through a DREQ line, the 8237A-5 DMA controller initiates a Bus Hold Request to the 80286 CPU through the CPU Controller IC. The CPU Controller arbitrates the CPU Hold Request from the DMA controller to the CPU.

When the CPU acknowledges the Hold request, the CPU control, address, and data lines are tri-stated. The CPU Controller controls the direction and enables the memory or peripheral address and data buses that correspond to the requested DMA operation.

During the DMA operation, the 8237A-5 acts as the bus master and, along with the CPU Controller IC, generates all bus control signals and address and data signals. The DMA transfers continue for the number of counts and to the destination address that was previously programmed into the DMA registers. See the device data sheet and the IO map for complete descriptions of the registers, their locations, and their functions.

DMA Bus Cycles. During the data bus cycle, the 8237A-5 first outputs the upper address (A8-A15) on its data outputs (XD0-XD7), to be latched in the buffer internally to the DRAM/DMA Control by the address strobe signal (AS) from the 8237A-5. Next, the lower address (A0-A7) is put directly on the S address bus by the 8237A-5.

The DMA request acknowledge signals, DACK2 and DACK3, are used along with RFRSH* and ACK* to enable the page register to be output as the upper address (A16 and A17 through A23), which are buffered by U22 to the system expansion slots.

A DMA bus cycle can be extended by the RDY input of the DMA controller. The DMA memory read DMAMR* is routed to the CPU Controller IC for extending the DMA bus cycle by inserting on DMA clock period as a wait state. The CPU Controller inserts the wait state by controlling the DMARDY input of the DMA controller.

I/O devices can extend the DMA bus cycle by controlling the IOCHRDY signal of the expansion bus. Setup times must be observed for IOCHRDY to be recognized.

RS232 Serial Port Interface

The RS232 Port is a single-channel, asynchronous communications port. The heart of the serial port is the WD8250-A Asynchronous Communications Element ACE (U70) that functions as a serial data input/output interface. It performs serial-to-parallel conversion on data characters received from a peripheral device or modem and parallel-to-serial conversion on data characters received from the CPU.

Status information reported includes the type and condition of the ACE's transfer operations as well as any error conditions detected during serial data operations. The WD8250-A includes a programmable Baud Rate Generator that allows operation from 50 to 9600 Baud. The WD8250-A is supplied with a clock of 1.8432 MHz from oscillator (Y2). The WD8250-A can be tailored to the user's requirements by being able to remove start bits, stop bits, and parity bits. It supports 5, 6, 7, or 8 data bit characters with 1, 1½, or 2 stop bits. Diagnostic capabilities provide loopback functions of transmit/receive and input/output signals.

The ACE is programmed by selecting the I/O address 3F8-3FE hex for primary and 2F8-2FE hex for secondary and writing data out to the port. Address bits A0, A1, and A2 are used to define the modes of operation by selecting the different registers to be programmed or read.

One interrupt is provided to the system from IRQ4 for primary operation and IRQ3 for secondary operation. This interrupt is active high. Bit 3 of the modem control register must be set high in order to send interrupts to the system. When this bit is high, any interrupts allowed by the interrupt enable register cause an interrupt.

Parallel Printer Port Interface

The final I/O interface of the Tandy 1000 TX is the Printer Interface. This part supplies all the signals required to interface to a typical parallel printer. These signals are 8 data out lines, plus various handshake control signals. Also, the printer interface generates an interrupt to the CPU if enabled.

Expansion Ports

System Expansion Bus

This section identifies the I/O interface requirements for the 8-bit, PC-compatible option cards. Each of the five slots has a 62-pin connector socket.

The following connector pin assignment is used on the PC option slots; this connector socket has 62 pins.

Pin	Signal Name	I/O	Pin	Signal Name	I/O
A1	NMI*	I	B1	GND	GROUND
A2	SD7	I/O	B2	BRESET	O
A3	SD6	I/O	B3	+5V	POWER
A4	SD5	I/O	B4	IRQ2	I
A5	SD4	I/O	B5	-5V	POWER
A6	SD3	I/O	B6	FDCDRQ	I
A7	SD2	I/O	B7	-12V	POWER
A8	SD1	I/O	B8	N/C	
A9	SD0	I/O	B9	+12V	POWER
A10	IOCHRDY	I	B10	GND	GROUND
A11	AEN	O	B11	SMEMW*	O
A12	SA19	O	B12	SMEMR*	O
A13	SA18	O	B13	IOW*	O
A14	SA17	O	B14	IOR*	O
A15	SA16	O	B15	DACK3*	O
A16	SA15	O	B16	DRQ3	I
A17	SA14	O	B17	DACK1*	O
A18	SA13	O	B18	DRQ1	I
A19	SA12	O	B19	RFRSH*	O
A20	SA11	O	B20	SCLK	O
A21	SA10	O	B21	IRQ7	I
A22	SA9	O	B22	IRQ6	I
A23	SA8	O	B23	IRQ5	I
A24	SA7	O	B24	IRQ4	I
A25	SA6	O	B25	IRQ3	I
A26	SA5	O	B26	FDCDACK*	O
A27	SA4	O	B27	T/C	O
A28	SA3	O	B28	BALE	O
A29	SA2	O	B29	+5V	POWER
A30	SA1	O	B30	OSC	O
A31	SA0	O	B31	GND	GROUND

Expansion Bus Signal Description

The following signal descriptions for the System I/O Bus are for PC bus-compatible option cards. Note that all signal lines are TTL compatible levels and that I/O adapters should be designed with a maximum of two low power Shottky (LS) loads per line.

SCLK (B20). SCLK is the System clock and has a period of 125ns in 8MHz mode, or 250ns in 4MHz mode. It has a 50% duty cycle and is used only for synchronization with the CPU. It is not intended for uses requiring a fixed frequency.

SA0 through SA19 (A12-A31). These lines are 20 address bits used to address memory and I/O devices within the Tandy 1000 TX. They are gated on the system bus when the BALE signal is high and are latched on the falling edge of the BALE signal. Generation of these signals is accomplished by the CPU or a DMA controller. SA0-SA19 are active high.

BALE (B28). BALE is a Buffered Address Latch Enable generated by the CPU Control IC. It is used to latch valid addresses from the CPU, and can be used by an I/O board to indicate a valid CPU address, in conjunction with AEN. BALE is pulled to a high state during DMA cycles, which include Refresh cycles. BALE is active high.

AEN (A11). AEN is an Address Enable signal used to remove the CPU and other devices from the bus to allow DMA transfers to take place. During AEN active, the DMA controller has control of the address bus, the data bus, the READ command lines, and the WRITE command lines. AEN is active high.

SD0 through SD7 (A2-A9). These signals are the data bus bits 0 through 7 from the CPU to memory and I/O devices on the bus. SD0 is the least significant bit (lsb), and SD7 is the most significant bit (msb).

BRESET (B2). BRESET is used to reset or initialize the expansion logic during power-up time, line voltage outage, or when the Reset switch on the front panel is pressed. BRESET is active high.

NMI* (A1). This signal indicates an uncorrectable system error when active. The NMI* signal provides the system board with parity information about memory or devices on the bus. NMI* is active low.

IOCHRDY (A10). This signal is used to lengthen I/O or memory cycles when driven low by the active device. (This signal should not be held low more than 15 microseconds.) Any slow device using this line should drive it low immediately upon detecting its valid address and a READ or WRITE command. See the timing diagram for setup times. IOCHRDY is active high (Ready condition).

IRQ2 through IRQ7 (B4, B21-B25). These signals are used to tell the CPU that an I/O device needs attention. The Interrupt Requests are prioritized with IRQ2 having the highest priority and IRQ7 the lowest. An Interrupt Request is generated when any IRQ signal is driven high and held high until the CPU acknowledges the interrupt.

IOR* (B14). IOR* is a read signal that instructs an I/O device to drive its data onto the data bus (SD0-SD7). This line can be driven by the CPU Control IC or by the DMA controller. IOR* is active low.

IOW* (B13). IOW* is a write signal that instructs an I/O device to read, or latch, the data from the data bus (SD0-SD7). This line can be driven by the CPU Control IC or by the DMA controller. IOW* is active low.

SMEMR* (B12). SMEMR* is a read signal that instructs a memory device to drive its data onto the data bus (SD0-SD7). This line can be driven by the CPU Control IC or by the DMA controller through the CPU Control IC. SMEMR* is active only when the memory address is within the first 1 megabyte range (000000-0FFFFFH). SMEMR* is active low.

SMEMW* (B11). SMEMW* is a write signal that instructs a memory device to read, or latch, the data from the data bus (SD0-SD7). This line can be driven by the CPU Control IC or by the DMA controller through the CPU Control IC. SMEMW* is active only when the memory address is within the first 1 megabyte range (000000-0FFFFFH). SMEMW* is active low.

DRQ1, FDCDRQ, and DRQ3 (B18, B6, B16). These lines are asynchronous DMA requests by peripheral devices to gain DMA service. They are prioritized with DRQ1 having the highest priority, FDCDRQ next, and DRQ3 lowest. A DMA request is generated by driving a DRQ line active high and holding it until the corresponding DACK (DMA acknowledge) signal goes active. DRQ1, FDCDRQ, and DRQ3 perform only 8-bit transfers. All DRQ lines are active high.

DACK1*, FDCACK*, and DACK3* (B17, B26, B15). These lines are DMA acknowledge signals used to acknowledge DMA requests DRQ1, FDCDRQ, and DRQ3. All DACK signals are active low.

RFRSH* (B19). This signal is used to indicate a refresh cycle that can be used by a memory board to refresh Dynamic memory. RFRSH* is active low and generated every 15 usec.

T/C (B27). T/C is a signal that provides a pulse when the terminal count for any DMA channel is reached. T/C is active high.

OSC (B30). OSC is an oscillator signal that is a high-speed clock with a 70 nanosecond period (14.31818 megahertz). It has a 50% duty cycle.

Memory Map

<u>Address</u>	<u>Name</u>	<u>Allocated Function</u>
000000-07FFFF	512K System RAM	System Memory
080000-09FFFF	128K System/Video RAM or 128K Expansion Memory	System Memory and Video Display Memory or System Memory
0A0000-0BFFFF	128K Video RAM	Reserved For Graphics Display Memory
0E0000-0FFFFF, EE0000-EFFFFF, or FE0000-FFFFFF	16K BIOS ROM Memory	Reserved For BIOS ROM Memory

I/O Port Map of System

I/O Port Map Summary

<u>Block</u>	<u>Usage</u>	<u>Function</u>
0000-001F	0000-001F	DMA Function
0020-003F	0020-0027	Interrupt Controller
0040-005F	0040-0047	Timer
0060-007F	0060-006F	PIO Function
0080-009F	0080-009F	DMA Page Register
00A0-00BF	00A0	NMI Mask Register
00C0-00DF	00C0-00C7	Sound Generator
00E0-00FF	00E0-00FF	Numerical Co-processor
0100-01FF		Reserved
0200-020F	0200-0207	Joystick Interface
0210-02F7		Reserved
02F8-02FF	02F8-02FF	Serial Port Secondary (COM2)
0300-031F		Reserved
0320-032F		Hard Disk Controller (optional)
0330-036F		Reserved
0370-0377	0370-0377	Floppy Disk Controller 2 (optional)
0378-037F	0378-037F	Printer
0380-03CF		Reserved
03D0-03DF	03D0-03DF	System Video
03E0-03EF		Reserved
03F0-03F7	03F0-03F7	Floppy Disk Controller 1
03F8-03FF	03F8-03FF	Serial Port Primary (COM1)
0400-FFFF		Not Usable

<u>Address</u>	<u>Description</u>
0000	DMA Controller IOW* = 0: Channel 0 Base and Current Address Internal Flip/Flop = 0: Write A0-A7 Internal Flip/Flop = 1: Write A8-A15 IOR* = 0: Channel 0 Current Address Internal Flip/Flop = 0: Read A0-A7 Internal Flip/Flop = 1: Read A8-A15
0001	DMA Controller IOW* = 0: Channel 0 Base and Current Word Count Internal Flip/Flop = 0: Write W0-W7 Internal Flip/Flop = 1: Write AW-W15 IOR* = 0: Channel 0 Current Word Count Internal Flip/Flop = 0: Read W0-W7 Internal Flip/Flop = 1: Read W8-W15
0002	DMA Controller IOW* = 0: Channel 1 Base and Current Address Internal Flip/Flop = 0: Write A0-A7 Internal Flip/Flop = 1: Write A8-A15 IOR* = 0: Channel 1 Current Address Internal Flip/Flop = 0: Read A0-A7 Internal Flip/Flop = 1: Read A8-A15
0003	DMA Controller IOW* = 0: Channel 1 Base and Current Word Count Internal Flip/Flop = 0: Write W0-W7 Internal Flip/Flop = 1: Write AW-W15 IOR* = 0: Channel 1 Current Word Count Internal Flip/Flop = 0: Read W0-W7 Internal Flip/Flop = 1: Read W8-W15
0004	DMA Controller IOW* = 0: Channel 2 Base and Current Address Internal Flip/Flop = 0: Write A0-A7 Internal Flip/Flop = 1: Write A8-A15 IOR* = 0: Channel 2 Current Address Internal Flip/Flop = 0: Read A0-A7 Internal Flip/Flop = 1: Read A8-A15
0005	DMA Controller IOW* = 0: Channel 2 Base and Current Word Count Internal Flip/Flop = 0: Write W0-W7 Internal Flip/Flop = 1: Write AW-W15 IOR* = 0: Channel 2 Current Word Count Internal Flip/Flop = 0: Read W0-W7 Internal Flip/Flop = 1: Read W8-W15
0006	DMA Controller IOW* = 0: Channel 3 Base and Current Address Internal Flip/Flop = 0: Write A0-A7 Internal Flip/Flop = 1: Write A8-A15 IOR* = 0: Channel 3 Current Address Internal Flip/Flop = 0: Read A0-A7 Internal Flip/Flop = 1: Read A8-A15

<u>Address</u>	<u>Description</u>
0007	DMA Controller IOW* = 0: Channel 3 Base and Current Word Count Internal Flip/Flop = 0: Write W0-W7 Internal Flip/Flop = 1: Write AW-W15 IOR* = 0: Channel 3 Current Word Count Internal Flip/Flop = 0: Read W0-W7 Internal Flip/Flop = 1: Read W8-W15
0008	DMA Controller IOW* = 0, Write Command Register
Bit	Description
0	0 = Memory to Memory Disable 1 = Memory to Memory Enable
1	0 = Channel 0 Address Hold Disable 1 = Channel 0 Address Hold Enable X If Bit 0 = 0
2	0 = Controller Enable 1 = Controller Disable
3	0 = Normal Timing 1 = Compressed Timing X If Bit 0 = 1
4	0 = Fixed Priority 1 = Rotating Priority
5	0 = Late Write Selection 1 = Extended Write Selection X If Bit 3 = 1
6	0 = DREQ Sense Active High 1 = DREQ Sense Active Low
7	0 = DACK Sense Active Low 1 = DACK Sense Active High
	IOR* = 0, Read Status Register
Bit	Description
0	1 = Channel 0 Has Reached TC
1	1 = Channel 1 Has Reached TC
2	1 = Channel 2 Has Reached TC
3	1 = Channel 3 Has Reached TC
4	1 = Channel 0 Request
5	1 = Channel 1 Request
6	1 = Channel 2 Request
7	1 = Channel 3 Request

<u>Address</u>	<u>Description</u>	
0009	DMA Controller	IOW* = 0, Write Request Register
Bit	Description	
0-1	Bit1 Bit0	
	0 0	Select Channel 0
	0 1	Select Channel 1
	1 0	Select Channel 2
	1 1	Select Channel 3
2	0	Reset Request Bit
	1	Set Request Bit
3-7		Don't Care
	IOR* = 0, Illegal	
000A	DMA Controller	IOW* = 0, Write Single Mask Register
Bit	Description	
0-1	Bit1 Bit0	
	0 0	Select Channel 0 Mask Bit
	0 1	Select Channel 1 Mask Bit
	1 0	Select Channel 2 Mask Bit
	1 1	Select Channel 3 Mask Bit
2	0	Clear Mask Bit (Enable Channel)
	1	Set Mask Bit (Disable Channel)
3-7		Don't Care
	IOR* = 0, Illegal	
000B	DMA Controller	IOW* = 0, Write Mode Register
Bit	Description	
0-1	Bit1 Bit0	
	0 0	Channel 0 Select
	0 1	Channel 1 Select
	1 0	Channel 2 Select
	1 1	Channel 3 Select
2-3	Bit3 Bit2	
	0 0	Verify Transfer
	0 1	Write Transfer To Memory
	1 0	Read Transfer To Memory
	1 1	Illegal
	X	If Bits 6 and 7 = 11
4	0	Autoinitialization Enable
	1	Autoinitialization Disable
5	0	Address Increment Select
	1	Address Decrement Select
6-7	Bit7 Bit6	
	0 0	Demand Mode Select
	0 1	Single Mode Select
	1 0	Block Mode Select
	1 1	Cascade Mode Select
	IOR* = 0, Illegal	

<u>Address</u>	<u>Description</u>
000C	DMA Controller IOW* = 0, Clear Byte Pointer Flip/Flop IOR* = 0, Illegal
000D	DMA Controller IOW* = 0, Master Clear IOR* = 0, Read Temporary Register
000E	DMA Controller IOW* = 0, Clear Mask Register IOR* = 0, Illegal
000F	DMA Controller IOW* = 0, Write All Mask Register Bits Bit Description 0 0 = Clear Channel 0 Mask Bit (Enable) 1 = Set Channel 0 Mask Bit (Disable) 1 0 = Clear Channel 1 Mask Bit (Enable) 1 = Set Channel 1 Mask Bit (Disable) 2 0 = Clear Channel 2 Mask Bit (Enable) 1 = Set Channel 2 Mask Bit (Disable) 3 0 = Clear Channel 3 Mask Bit (Enable) 1 = Set Channel 3 Mask Bit (Disable) 4-7 Don't Care IOR* = 0, Illegal
0010 - 001F	Same as 0000-000F

<u>Address</u>	<u>Description</u>	
0020	8259A Interrupt Controller	
Note: Initialization Words are set up by the operating system and are generally not to be changed. Writing an initialization word might cancel pending interrupts.		
Bit4 = 1 Initialization Command Word 1		
Bit0 = 0	ICW4 Needed	
= 1	ICW4 Not Needed	
Bit1 = 0	Cascade Mode	
= 1	Single Mode	
Bit2	Not Used	
Bit3 = 0	Edge Triggered Mode	
= 1	Level Triggered Mode	
Bit5-7	Not Used	
Bit4 = 0 &	Operation Control Word 2	
Bit3 = 0	Bit0-2: Determine The Interrupt Level Acted On When The SL Bit Is Active	
	Interrupt Level = 0 1 2 3 4 5 6 7	
	Bit0 (L0): 0 1 0 1 0 1 0 1	
	Bit1 (L1): 0 0 1 1 0 0 1 1	
	Bit2 (L2): 0 0 0 0 1 1 1 1	
Bit5-7: Control Rotate And End Of Interrupt Modes		
B7 B6 B5		
0 0 1	Non-Specific EOI Command	End Of Interrupt
0 1 1	Specific EOI Command	End Of Interrupt
1 0 1	Rotate On Non-Specific EOI	Auto Rotation
1 0 0	Rotate In Automatic EOI Mode (Set)	Auto Rotation
0 0 0	Rotate In Automatic EOI Mode (Clear)	Auto Rotation
1 1 1	*Rotate On Specific EOI Command	Specific Rotation
1 1 0	*Set Priority Command	Specific Rotation
0 1 0	No Operation	
(*L0 - L2 Are Used)		

<u>Address</u>	<u>Description</u>			
Bit4 = 0 &	Operation Control Word 3			
Bit3 = 1	Bit 0-1:			
Bit1	Bit0 - Read Register Command			
0	0	No Action		
0	1	No Action		
1	0	Read IR Register On Next IOR* Pulse		
1	1	Read IS Register On Next IOR* Pulse		
Bit2 = 0:	No Poll Command			
= 1:	Poll Command			
Bit5-6				
Bit5	Bit6	- Special Mask Mode		
0	0	No Action		
0	1	No Action		
1	0	Reset Special Mask		
1	1	Set Special Mask		
Bit7 = 0				
0021	8259A Interrupt Controller			
Initialization Control Word 2				
Bit0-7: Not Used				
Bit3-7: T3-T7 Of Interrupt Vector Address				
(8086/8088/80286 Mode)				
Initialization Control Word 3 (Master Device)				
Bit0-7: = 1	Indicated IR Input Has A Slave			
= 0	Indicated IR Input Does Not Have A Slave			
Initialization Control Word 3 (Slave Device)				
Bit0-2 = ID0-2	Bit0	Bit1	Bit2	- Slave ID #
	0	0	0	0
	0	0	1	1
	0	1	0	2
	0	1	1	3
	1	0	0	4
	1	0	1	5
	1	1	0	6
	1	1	1	7
Bit3-7 = 0	(Not Used)			

<u>Address</u>	<u>Description</u>
	Initialization Control Word 4
Bit0:	Type Of Processor =0 MCS-80/85 Mode =1 8086/8088/80286 Mode
Bit1:	Type Of End Of Interrupt =0 Normal EOI =1 Auto EOI
Bit2-3:	Buffering Mode
Bit3	Bit2 0 X Non-Buffered Mode 1 0 Buffered Mode/Slave 1 1 Buffered Mode/Master
Bit4:	Nesting Mode =0 Not Special Fully Nested Mode =1 Special Fully Nested Mode
Bit5-7:	=0 (Not Used)
	Operation Control Word 1 (IOR*/IOW*)
Bit0-7:	Interrupt Mask For IRQ0-IRQ7 =0 Mask Reset (Enable) =1 Mask Set (Disable)

Note: Peripherals requesting an interrupt service must generate a low to high edge and then remain at a logic high level until service is acknowledged. Failure to do so results in a Default Service for IRQ7.

0022-0027 Same as 0020-0021

0028-003F Not Used

0040/0044	8254-2 Timer
IOW* = 0:	Load Counter No. 0
IOR* = 0:	Read Counter No. 0
0041/0045	8254-2 Timer
IOW* = 0:	Load Counter No. 1
IOR* = 0:	Read Counter No. 1
0042/0046	8254-2 Timer
IOW* = 0:	Load Counter No. 2
IOR* = 0:	Read Counter No. 2

<u>Address</u>	<u>Description</u>		
0043/0047	8254-2 Timer		
IOW* = 0:	Write Mode Word		
Control Word Format			
Bit 0: BCD			
=0: BCD Counter (4 Decades)			
=1: Binary Counter 16 Bits			
Bit1-3: Mode Selection			
Bit3 Bit2 Bit1			
0 0 0	Mode 0		
0 0 1	Mode 1		
X 1 0	Mode 2		
X 1 1	Mode 3		
1 0 0	Mode 4		
1 0 1	Mode 5		
Bit4-5: Read/Load			
Bit5 Bit4			
0 0 Counter Latching Operation			
0 1 Read/Load LSB Only			
1 0 Read/Load MSB Only			
1 1 Read/Load LSB First, Then MSB			
Bit6-7: Select Counter			
Bit7 Bit6			
0 0 Select Counter 0			
0 1 Select Counter 1			
1 0 Select Counter 2			
1 1 Illegal			
IOR* = 0: No-Operation 3-State			
0048-005F	Not Used		
0060	Port A / Keyboard Interface Control Ports (Read Only)		
Bit	Description		
0	Keyboard Bit 0-LSB		
1	Keyboard Bit 1		
2	Keyboard Bit 2		
3	Keyboard Bit 3		
4	Keyboard Bit 4		
5	Keyboard Bit 5		
6	Keyboard Bit 6		
7	Keyboard Bit 7-MSB		

<u>Address</u>	<u>Description</u>
0061	Port B Read Or Write Description Bit 0 = 8253 Gate 2 Enable 1 = Speaker Data Out Enable 2 Not Used 3 Not Used 4 1 = Disable Internal Speaker (Sound Control2) 5 Not Used 6 0 = HOLOCK (If IBM PC Keyboard Mode) 7 1 = Keyboard Clear
0062	Port C Read/Write: Bits 0-3; Read Only Bits: 4-7 Description Bit 0 Read/Write - Not Used 1 Read/Write - Not Used 2 Read/Write - Not Used 3 (Output) CPU Clock Rate 0 = 4.77 MHz (PC Compatible Rate) 1 = 8.00 MHz (Default By Boot ROM) 4 Video RAM Size 0 = 128K Video 1 = 256K Video 5 8253 Out #2 6 Monochrome Mode 0 = Color Monitor 1 = 350 Line Monitor, Mono 7 Reserved
0063-0066	Reserved
0067	Port H Reserved
0068	Port I A20 Gate And CPU Reset Bit0: Not Used Bit1: 0 = GA20 Always 0 1 = A20 Equals To GA20 (Reset Clears Bit1 To "0") Bit2: CPU Reset Write 0 = Reset 1 = No Effect Read 0 = Power On Reset 1 = CPU Reset Bit3: Not Used Bit4: Not Used Bit5: Not Used Bit6: Not Used Bit7: Not Used

<u>Address</u>	<u>Description</u>
0069	Port J Read And Write 8 Bit Data Storage For Shutdown Status Byte Bit0: Data Bit 0 Bit1: Data Bit 1 Bit2: Data Bit 2 Bit3: Data Bit 3 Bit4: Data Bit 4 Bit5: Data Bit 5 Bit6: Data Bit 6 Bit7: Data Bit 7
006A-007F	Reserved
0080	DMA Page Register (Reserved for Diagnostics) Write Only
0081	DMA Channel 2 Page Register -Write Only
Address Description	
Bit0	Address A16
Bit1	Address A17
Bit2	Address A18
Bit3	Address A19
Bit4	Address A20
Bit5	Address A21
Bit6	Address A22
Bit7	Address A23
0082	DMA Channel 3 Page Register -Write Only
Address Description	
Bit0	Address A16
Bit1	Address A17
Bit2	Address A18
Bit3	Address A19
Bit4	Address A20
Bit5	Address A21
Bit6	Address A22
Bit7	Address A23
0083	DMA Channel 0-1 Page Register -Write Only
Address Description	
Bit0	Address A16
Bit1	Address A17
Bit2	Address A18
Bit3	Address A19
Bit4	Address A20
Bit5	Address A21
Bit6	Address A22
Bit7	Address A23

<u>Address</u>	<u>Description</u>						
0084-008F	Same as 0080-0083						
00A0	NMI Mask Register, Write Only						
Bit	Description						
0	External Video						
0	0 = Normal Operation						
1	1 = All Video Addresses And Ports Are Disabled						
1	MEMCONFIG 1 - A17 128K SW						
2	MEMCONFIG 2 - A18 256K SW						
3	MEMCONFIG 3 - A19 512K SW						
4	"0" Enable 128K Of Video RAM (Always "0")						
5	Not Used						
6	Not Used						
7	1 = Enable NMI						
	0 = Disabled						
00A1-00A7	Reserved						
Bit	Bit	Bit	Bit	Memory Start	Memory Length	Memory Range	
4	3	2	1				
256K	Enable A19	A18	A17				
0	0	0	0	0 0000	128K	0 0000-1 FFFF	
0	0	0	1	2 0000	128K	2 0000-3 FFFF	
0	0	1	0	4 0000	128K	4 0000-5 FFFF	
0	0	1	1	6 0000	128K	6 0000-7 FFFF	
0	1	0	0	8 0000	128K	8 0000-9 FFFF	
0	1	1	1	B 0000	128K	B 0000-B FFFF (4 Page)	
1	0	0	1	0 0000	256K	0 0000-3 FFFF	
1	0	1	0	2 0000	256K	2 0000-5 FFFF	
1	0	1	1	4 0000	256K	4 0000-7 FFFF	
1	1	0	0	6 0000	256K	6 0000-9 FFFF	
1	1	1	1	B 8000	256K	B 0000-B FFFF (8 Page)	

NOTE: To turn off on-board Video, be sure Port A0H, Data Bit 0 is a "1" And Video Array Register 3.

00A8-00AF Not Used

<u>Address</u>	<u>Description</u>							
00C0-00C7	Sound SN76496							
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
= 1	0	0	0	F6	F7	F8	F9	Update Tone Frequency 1
= 0	X	F0	F1	F2	F3	F4	F5	Additional Frequency Data
= 1	0	0	1	A0	A1	A2	A3	Update Tone Attenuation 1
= 1	0	1	0	F6	F7	F8	F9	Update Tone Frequency 2
= 0	X	F0	F1	F2	F3	F4	F5	Additional Frequency Data
= 1	0	1	1	A0	A1	A2	A3	Update Tone Attenuation 2
= 1	1	0	0	F6	F7	F8	F9	Update Tone Frequency 3
= 0	X	F0	F1	F2	F3	F4	F5	Additional Frequency Data
= 1	1	0	1	A0	A1	A2	A3	Update Tone Attenuation 3
= 1	1	1	0	X	FB	NFO	NFI	Update Noise Control
= 1	1	1	1	A0	A1	A2	A3	Update Noise Attenuation
00C8-00CF	Reserved							
00E0-00EF	Reserved							
00F0	Clear Numerical Co-Processor Busy							
00F1	Reset Numerical Co-Processor To Real Mode							
00F2	Same As 00F0							
00F3	Same As 00F1							
00F4	Same As 00F0							
00F5	Same As 00F1							
00F6	Same As 00F0							
00F7	Same As 00F1							
00F8-00FF	Math Co-Processor Chip Select							
0100-01FF	Reserved							
0200-0207	Joystick							
	Clear (Resets Integrator To 0)							
0201	Read R = Right Joystick, L = Left Joystick							
Bit	Description							
0	R - X Horizontal Position							
1	R - Y Vertical Position							
2	L - X Horizontal Position							
3	L - Y Vertical Position							
4	R Button #1 (Logic 0 = Button Pressed)							
5	R Button #2 (Logic 0 = Button Pressed)							
6	L Button #1 (Logic 0 = Button Pressed)							
	L Button #2 (Logic 0 = Button Pressed)							
0208-020F	Not Used							
0210-02F7	Reserved							

<u>Address</u>	<u>Description</u>
2F8-2FF	Serial Port Secondary (COM2)
02F8	Write Transmitter Holding Register (Character to send)
<u>Bit</u>	<u>Description</u>
0	Bit 0 - LSB (First Bit Sent Serially)
1	Bit 1
2	Bit 2
3	Bit 3
4	Bit 4
5	Bit 5
6	Bit 6
7	Bit 7 - MSB
02F8	Read Receiver Buffer Register (Character Received)
<u>Bit</u>	<u>Description</u>
0	Bit 0 - LSB (First Bit Received Serially)
1	Bit 1
2	Bit 2
3	Bit 3
4	Bit 4
5	Bit 5
6	Bit 6
7	Bit 7 - MSB
02F8	Divisor Latch LSB (Divisor Latch Access Bit DLAB = "1")
<u>Bit</u>	<u>Description</u>
0	Bit 0
1	Bit 1
2	Bit 2
3	Bit 3
4	Bit 4
5	Bit 5
6	Bit 6
7	Bit 7

<u>Address</u>	<u>Description</u>
02F9	Divisor Latch MSB (Divisor Latch Access Bit DLAB = "1")
<u>Bit</u>	<u>Description</u>
0	Bit 0
1	Bit 1
2	Bit 2
3	Bit 3
4	Bit 4
5	Bit 5
6	Bit 6
7	Bit 7
02F9	Interrupt Enable Register
<u>Bit</u>	<u>Description</u>
0	"1" = Enables the Received Data Available Interrupt
1	"1" = Enables the Transmitter Holding Register Int
2	"1" = Enables Receive Line Status Interrupt
3	"1" = Enables the Modem Status Interrupt
4-7	Always Logical "0"
02FA	Interrupt Identification Register
<u>Bit</u>	<u>Description</u>
0	"0" = Interrupt Pending
1-2	Bit 2 Bit 1 "0" "0" Fourth Level Priority "0" "1" Third Level Priority "1" "0" Second Level Priority "1" "1" Highest Level Priority
3-7	Always Logical "0"

<u>Address</u>	<u>Description</u>
02FB	Line Control Register
<u>Bit</u>	<u>Description</u>
0-1	Bit 1 Bit 0
	"0" "0" Five Bit Word Length
	"0" "1" Six Bit Word Length
	"1" "0" Seven Bit Word Length
	"1" "1" Eight Bit Word Length
2	"0" = One Stop Bit
	"1" = 1½ Stop Bits When Five Bit Length Selected
	Two Stop Bits With Six, Seven, or Eight Bit
3	"1" = Parity Enable
4	"0" = Odd Parity Select
	"1" = Even Parity Select
5	Stick Parity Bit
6	"1" = Set Break Enable
7	"1" = Divisor Latch Access Bit Enable
02FC	
<u>Bit</u>	<u>Description</u>
0	"1" = Data Terminal Ready Set (DTR)
	"0" = Data Terminal Ready Reset (DTR)
1	Request To Send (RTS)
2	Out 1
3	Out 2
4	Loop
5-7	Always Logical "0"
02FD	Line Status Register
<u>Bit</u>	<u>Description</u>
0	Data Ready (DR)
1	Overrun Error (OR)
2	"1" = Detect Parity Error (PE)
3	"1" = Detect Framing Error (FE)
4	"1" = Break Interrupt (BI)
5	Transmitter Holding Register
	"1" = Character Transferred From Holding To Shift
	Register
	"0" = Loading Transmitter Holding Register
6	Transmitter Shift Register Empty
	"1" = Shift Register Idle
	"0" = Data Transfer From Holding Register
7	Always Logical "0"

<u>Address</u>	<u>Description</u>
02FE	Modem Status Register
<u>Bit</u>	<u>Description</u>
0	Delta Clear To Send (DCTS)
1	Delta Data Set Ready (DDSR)
2	Trailing Edge Ring Indicator "1" = On "0" = Off
3	Delta Received Line Signal Detect (If Bit 0, 1, 2, or 3 is set to a "1" modem status interrupt is generated)
4	"0" = Clear To Send (CTS)
5	"0" = Data Set Ready (DSR)
6	"0" = Ring Indicator (RI)
7	"0" = Received Line Signal Detect (RLSD)
02FF	Reserved
0300-036F	Reserved
0370-0377	Floppy Disk Controller 2 (optional)
0378	Printer - Data Latch
<u>Bit</u>	<u>Description</u>
0	Bit 0 - LSB
1	Bit 1
2	Bit 2
3	Bit 3
4	Bit 4
5	Bit 5
6	Bit 6
7	Bit 7 - MSB
0379	Printer - Read Status
<u>Bit</u>	<u>Description</u>
0	Not Used
1	Not Used
2	Not Used
3	"0" = Error
4	"1" = Printer Select
5	"0" = Out of Paper
6	"0" = Acknowledge
7	"0" = Busy

<u>Address</u>	<u>Description</u>
037A (037E)	Printer - Control Latch
<u>Bit</u>	<u>Description</u>
0	"0" = Strobe
1	"0" = Auto FD XT
2	"0" = Initialize
3	"0" = Select Printer
4	"1" = Enable Interrupt
5	"0" = Enable Output Data
6	Not Used
7	Not Used
037B	Not Used
037C	Printer - Data Latch
037D	Printer - Read Status
037F-03CF	Not Used
03D0-03D3	Not Used

<u>Address</u>	<u>Description</u>
03D4	6845 Address Register
03D5	6845 Data Register
03D6	Not Used
03D7	Not Used
03D8	Mode Select Register
Bit0	High Resolution Clock = 0 Selects 40 By 25 Alphanumeric Mode = 1 Selects 80 By 25 Alphanumeric Mode
Bit1	Graphics Select = 0 Selects Alphanumeric Mode = 1 Selects 320 By 200 Graphics Mode
Bit2	Black And White = 0 Selects Color Mode = 1 Selects Black And White Mode
Bit3	Video Enable = 0 Disables Video Signal = 1 Enables Video Signal
Bit4	640 Dot Graphics = 0 Disables 640 By 200 B&W Graphics Mode = 1 Enables 640 By 200 B&W Graphics Mode
Bit5	Blink Enable = 0 Disables Blinking = 1 Enables Blinking
03D9	Color Select Register
Bit0	Background Blue
Bit1	Background Green
Bit2	Background Red
Bit3	Background Intensity
Bit4	Foreground Intensity
Bit5	Color Select

<u>Address</u>	<u>Description</u>	
03DA-03DE	Write Video Array Address And Read Status '03DA) Write Video Array Data (03DE)	
Read (03DA)		
00 Bit0	Display Inactive	Not Used
00 Bit1	Light Pen Set	Not Used
00 Bit2	Light Switch Status	Not Used
00 Bit3	Vertical Retrace	Not Used
00 Bit4	Not Used	Not Used
01 Bit0		Palette Mask 0
01 Bit1		Palette Mask 1
01 Bit2		Palette Mask 2
01 Bit3		Palette Mask 3
02 Bit0		Border Blue
02 Bit1		Border Green
02 Bit2		Border Red
02 Bit3		Border Intensity
02 Bit5		Reserved = 0
03 Bit0		Mono Enable = 1
03 Bit1		Reserved = 0
03 Bit2		Border Enable
03 Bit3		4-Color High Resolution
03 Bit4		16 Color Mode
03 Bit5		Extra Video Mode
03DB	Clear Light Pen Latch	
03DC	Preset Light Pen Latch	
03DD	Extended RAM Page Register - CPU Relative	
Bit	Description	
0	Extended Addressing Modes	
1	Not Used	
2	Not Used	
3	CRT Video Page Address "17"	
4	CRT Video Page Address "18"	
5	CPU Page Address "17"	
6	CPU Page Address "18"	
7	Select 64K Or 256K RAM	

<u>Address</u>	<u>Description</u>
03DF	CRT Processor Page Register - Video Memory Relative
Bit0 A14	CRT Page 0
Bit1 A15	CRT Page 1
Bit2 A16	CRT Page 2
Bit3 A14	Processor Page 0
Bit4 A15	Processor Page 1
Bit5 A16	Processor Page 2
Bit6	Video Address Mode 0
Bit7	Video Address Mode 1
Video Descriptions	D0 D7 D6 D5
8p 1 - 16K	0 0 0 0
8p 2 - 8K	0 0 0 1
4p 2 - 16K	0 1 0 0
4p 4 - 8K	0 1 1 1
4p 1 - 32K	1 0 0 0
2p 2 - 32K	1 0 1 1
03E0-03EF	Reserved
03F0	Not Used
03F1	Drive Select Switch
Bit0	Not Used
Bit1	"1" DS0 = DS0 "0" DS0 = DS1
Bit2	Not Used
Bit3	Mux FDCTC (Write Only) "0" FDCTC Out "1" Input
Bit4	Not Used
Bit5	Not Used
Bit6	Not Used
Bit7	Not Used
03F2	DOR Register (Write Only)
Bit0-1	Drive Select
Bit1	Bit0
0	0 Drive Select A*
0	1 Drive Select B*
Bit2	0 = FDC Reset
Bit3	1 = Enable DMA Request/Interrupt
Bit4	1 = Drive A Motor On
Bit5	1 = Drive B Motor On
Bit6	1 = FDC Terminal Count
Bit7	Not Used

<u>Address</u>	<u>Description</u>
03F3	Not Used
03F4	FDC - Status (Read Only) See FDC Specification
03F5	FDC - Data (R/W) See FDC Specification
03F6-03F7	Reserved
03F8-03FF	Serial Port Primary (COM1)
03F8	Write Transmitter Holding Register (Character to send)
<u>Bit</u>	<u>Description</u>
0	Bit 0 - LSB (First Bit Sent Serially)
1	Bit 1
2	Bit 2
3	Bit 3
4	Bit 4
5	Bit 5
6	Bit 6
7	Bit 7 - MSB
03F8	Read Receiver Buffer Register (Character Received)
<u>Bit</u>	<u>Description</u>
0	Bit 0 - LSB (First Bit Received Serially)
1	Bit 1
2	Bit 2
3	Bit 3
4	Bit 4
5	Bit 5
6	Bit 6
7	Bit 7 - MSB
03F8	Divisor Latch LSB (Divisor Latch Access Bit DLAB = "1")
<u>Bit</u>	<u>Description</u>
0	Bit 0
1	Bit 1
2	Bit 2
3	Bit 3
4	Bit 4
5	Bit 5
6	Bit 6
7	Bit 7

<u>Address</u>	<u>Description</u>
03F9	Divisor Latch MSB (Divisor Latch Access Bit DLAB = "1")
<u>Bit</u>	<u>Description</u>
0	Bit 0
1	Bit 1
2	Bit 2
3	Bit 3
4	Bit 4
5	Bit 5
6	Bit 6
7	Bit 7
03F9	Interrupt Enable Register
<u>Bit</u>	<u>Description</u>
0	"1" = Enables the Received Data Available Interrupt
1	"1" = Enables the Transmitter Holding Register Int
2	"1" = Enables Receive Line Status Interrupt
3	"1" = Enables the Modem Status Interrupt
4-7	Always Logical "0"
03FA	Interrupt Identification Register
<u>Bit</u>	<u>Description</u>
0	"0" = Interrupt Pending
1-2	Bit 2 Bit 1
	"0" "0" Fourth Level Priority
	"0" "1" Third Level Priority
	"1" "0" Second Level Priority
	"1" "1" Highest Level Priority
3-7	Always Logical "0"

<u>Address</u>	<u>Description</u>	
03FB	Line Control Register	
<u>Bit</u>	<u>Description</u>	
0-1	Bit 1 Bit 0	
	"0" "0"	Five Bit Word Length
	"0" "1"	Six Bit Word Length
	"1" "0"	Seven Bit Word Length
	"1" "1"	Eight Bit Word Length
2	"0" = One Stop Bit	
	"1" = 1½ Stop Bits When Five Bit Length Selected	
	Two Stop Bits With Six, Seven, or Eight Bit	
3	"1" = Parity Enable	
4	"0" = Odd Parity Select	
	"1" = Even Parity Select	
5	Stick Parity Bit	
6	"1" = Set Break Enable	
7	"1" = Divisor Latch Access Bit Enable	
03FC		
<u>Bit</u>	<u>Description</u>	
0	"1" = Data Terminal Ready Set (DTR)	
	"0" = Data Terminal Ready Reset (DTR)	
1	Request To Send (RTS)	
2	Out 1	
3	Out 2	
4	Loop	
5-7	Always Logical "0"	
03FD	Line Status Register	
<u>Bit</u>	<u>Description</u>	
0	Data Ready (DR)	
1	Overrun Error (OR)	
2	"1" = Detect Parity Error (PE)	
3	"1" = Detect Framing Error (FE)	
4	"1" = Break Interrupt (BI)	
5	Transmitter Holding Register	
	"1" = Character Transferred From Holding To Shift Register	
	"0" = Loading Transmitter Holding Register	
6	Transmitter Shift Register Empty	
	"1" = Shift Register Idle	
	"0" = Data Transfer From Holding Register	
7	Always Logical "0"	

<u>Address</u>	<u>Description</u>
03FE	Modem Status Register
<u>Bit</u>	<u>Description</u>
0	Delta Clear To Send (DCTS)
1	Delta Data Set Ready (DDSR)
2	Trailing Edge Ring Indicator "1" = On "0" = Off
3	Delta Received Line Signal Detect (If Bit 0, 1, 2, or 3 is set to a "1" modem status interrupt is generated)
4	"0" = Clear To Send (CTS)
5	"0" = Data Set Ready (DSR)
6	"0" = Ring Indicator (RI)
7	"0" = Received Line Signal Detect (RLSD)
03FF	Reserved

TIMING SPECIFICATIONS FOR SYSTEM CLOCK TIMING (SH 1 OF 3)

!SYM!PARAMETER	! MIN	TYP	MAX	UNIT	TEST	CON.
!T1 !16MHZ clock period	!	62.5!	!	ns!	!	!
!T2 !16MHZ clock high time	!	25!	31.2!	37.5!	ns!	!
!T2a!16MHZ clock low time	!	25!	31.2!	37.5!	ns!	!
!T3 !8MHZ clock period	!	125.0!	!	ns!	!	!
!T4 !8MHZ clock high time	!	62.5!	!	ns!	!	!
!T4a!8MHZ clock low time	!	62.5!	!	ns!	!	!
!T5 !PRCLK,PRCLKA,PRCLKB period	!	62.5!	!	ns!	!	!
! fast mode	!	!	!	!	!	!
!T6 !PRCLK,PRCLKA,PRCLKB period	!	1125.0!	!	ns!	!	!
! slow mode	!	!	!	!	!	!
!T7 !SCLK clock period fast mode	!	1125.0!	!	ns!	!	!
!T8 !SCLK clock period slow mode	!	1250.0!	!	ns!	!	!
!T9 !DMACLK clock period fast mode	!	1250.0!	!	ns!	!	!
!T10 !DMACLK clock period slow mode	!	1500.0!	!	ns!	!	!
!T11!28MHZ clock period	!	34.9!	!	ns!	!	!
!T12!OSC clock period	!	69.8!	!	ns!	!	!
!T13!3.58MHZ clock period	!	1279.4!	!	ns!	!	!
!T14!1.19MHZ clock period	!	1838.1!	!	ns!	!	!

TIMING SPECIFICATIONS FOR ROM, DRAM AND SYSTEM TIMING (SH 2 OF 3)

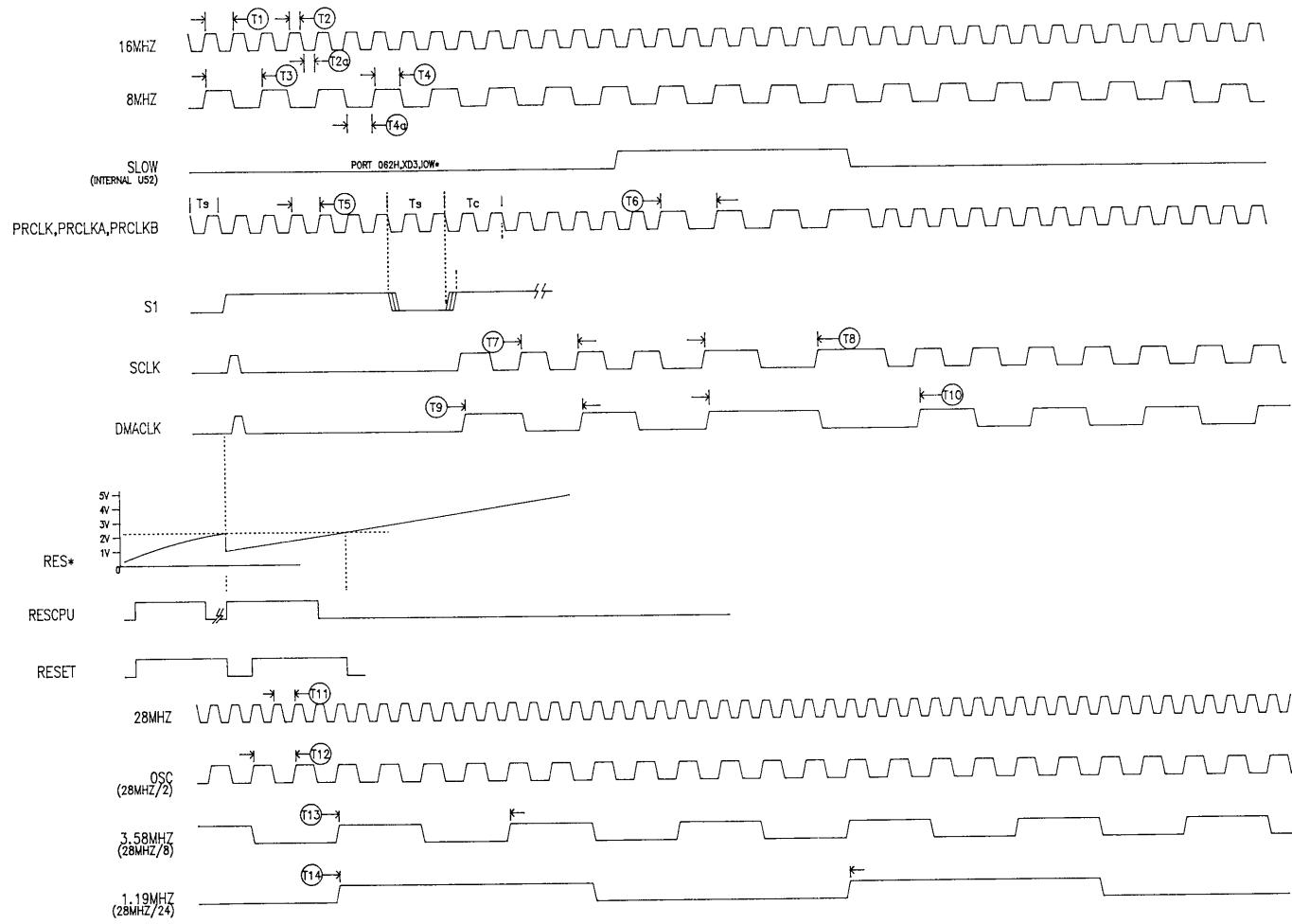
!SYM!PARAMETER	!MIN	!MAX	!UNIT	TEST CON.
!T1 !System Clock period	!62.5!	125	!	!
!T2 !System Clock High time	!31.2!	62.5!	ns	!
!T3 !Status/PEACK* valid delay	! 1 !	33	ns	!
!T4 !Address valid delay	! 1 !	60	ns	!
!T5 !ALE active delay from Status	! 5 !	25	ns	!
!T6 !ALE inactive delay	! 5 !	25	ns	!
!T7 !MEMR*,MEMW* active delay	! 5 !	45	ns	!
!T8 !MEMR*,MEMW* inactive delay	! 5 !	45	ns	!
!T9 !ROMCS* active delay from A0-23	! 0 !	40	ns	!
!T10 !ROMCS* inactive delay	! 0 !	40	ns	!
!T11 !MEMCYC active delay from PRCLK	! 5 !	25	ns	!
!T12 !MEMCYC inactive delay from PRCLK	! 5 !	35	ns	!
!T13 !RASX* active delay from PRCLK	! 5 !	45	ns	at 100pf
!T14 !RASX* Inactive delay from PRCLK	! 5 !	45	ns	at 100pf
!T15 !CASX* active delay from PRCLK	! 5 !	40	ns	!
!T16 !CASX* inactive delay from	! 5 !	30	ns	!
! !MEMCYC	! 1 !	1	!	!
!T17 !Memory Address delay from ALE	! 8 !	50	ns	!
!T18 !RASX* Address hold time	!20 !	--	ns	!
!T19 !CASX* Address setup time	! 5 !	--	ns	!
!T20 !RASX* hold time	!80 !	--	ns	!
!T21 !ROM access delay from MEMR*	!--	75	ns	!
!T22 !DRAM access time	! 1 !	75	ns	!
!T23 !DBUS setup time (CPU)	!10 !	--	ns	!
!T24 !DBUS delay from MDBUS valid	! 8 !	30	ns	!

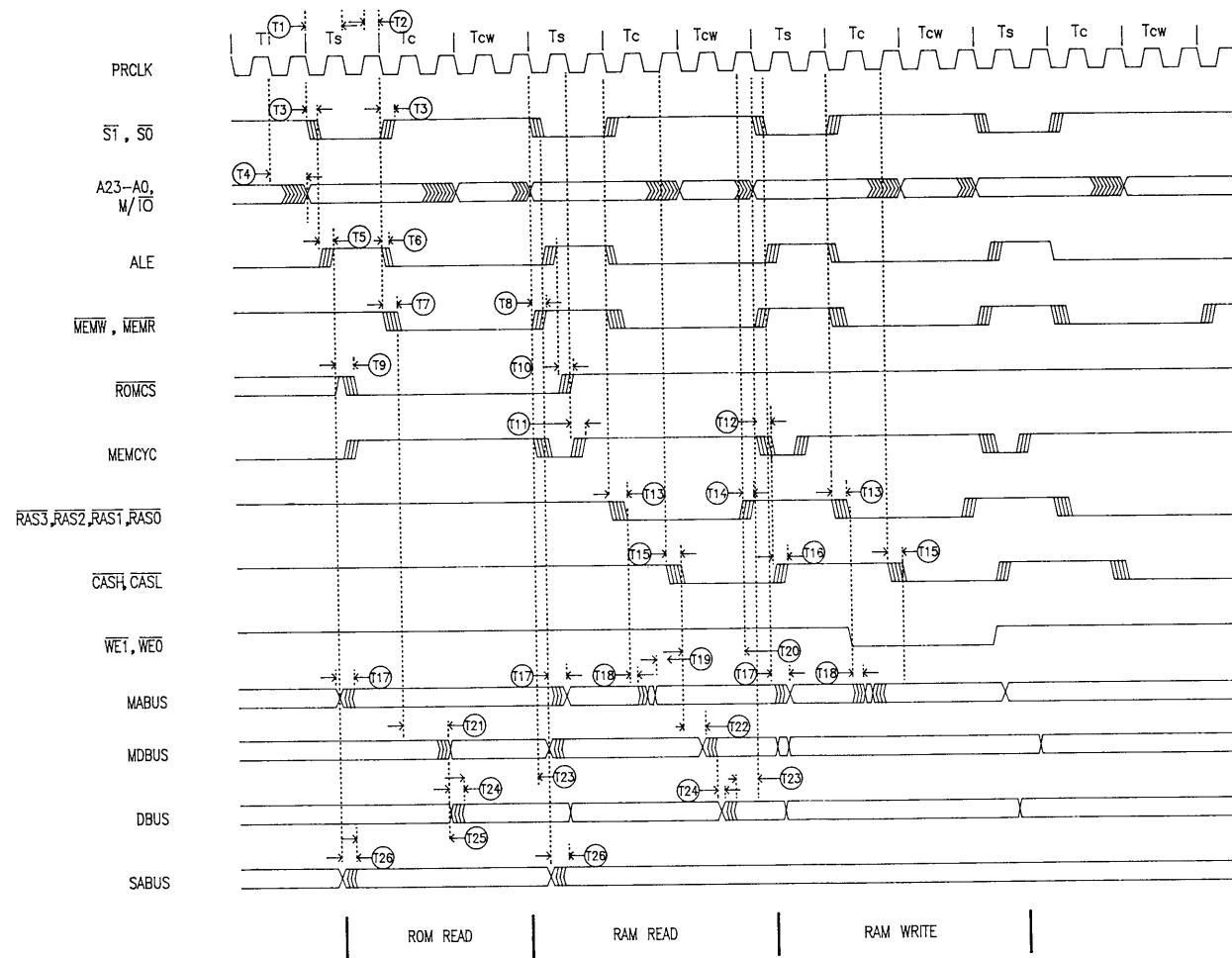
!T25!ROM access time from ! valid SABUS	! 0 ! 200 !ns !
!-----!	!-----!
!T26!SABUS delay from ALE	! 0 ! 23 !ns !
=====	=====

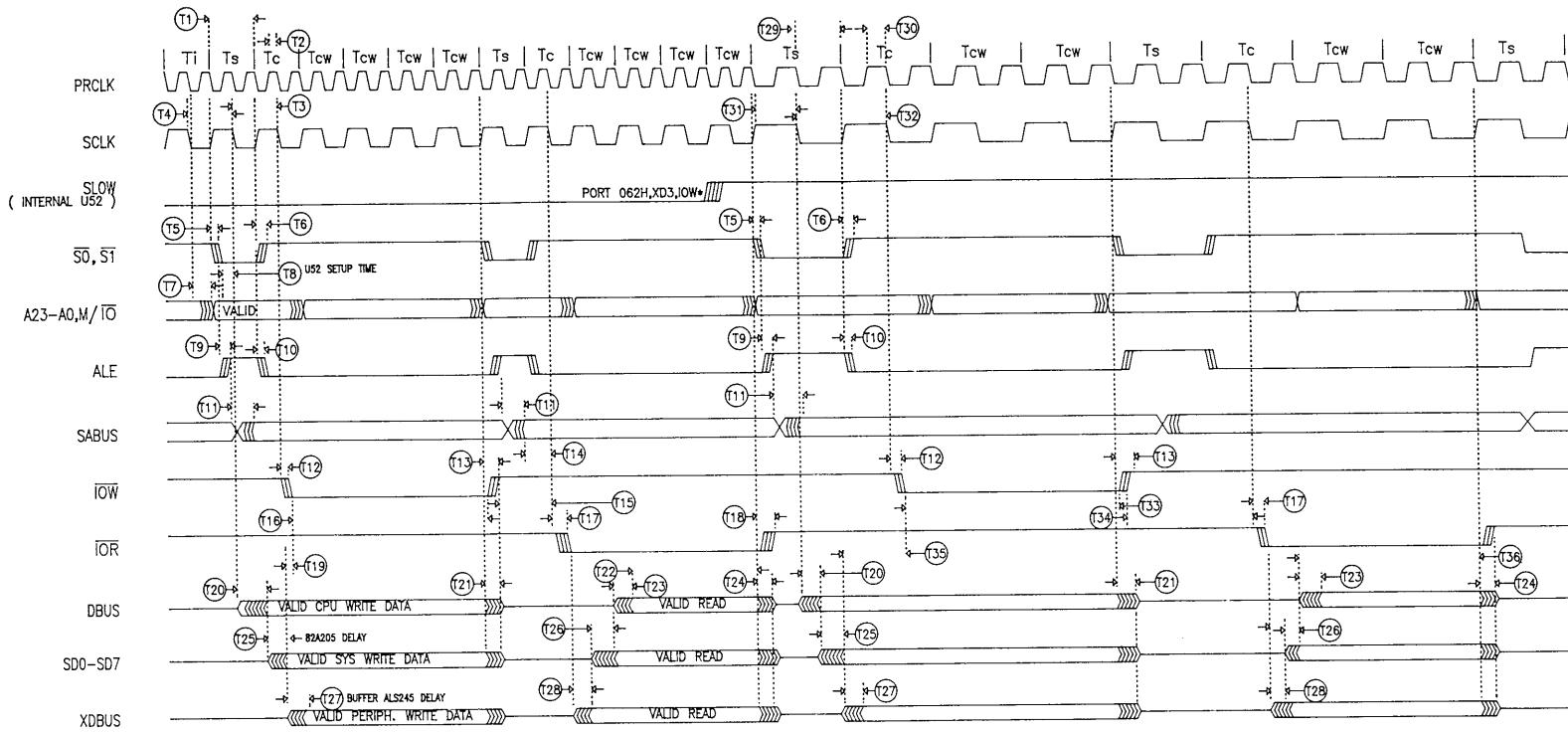
TIMING SPECIFICATIONS FOR IO R/W TIMING (SH 3 OF 3)

!SYM!PARAMETER	!MIN!	TYP	! MAX	!UNIT!	TEST COND.
!T1 !PRCLK clock period	! 1	62.5!	! ns	!	!
!T2 !PRCLK high period	! 25!	31.2!	! 37	! ns	!
!T3 !SCLK clock period	! 1	125	! 1	! ns	!
!T4 !SCLK high period	! 1	162.5!	! ns	!	!
!T5 !Status/PEACK active delay	! 1!	! 33	! ns	!	!
!T6 !Status/PEACK inactive delay	! 1!	! 33	! ns	!	!
!T7 !CPU address valid delay	! 1!	! 60	! ns	!	!
!T8 !U52 setup time	! 24!	! ---	! ns	!	!
!T9 !ALE active delay	! 5!	! 25	! ns	!	!
!T10 !ALE inactive delay	! 5!	! 25	! ns	!	!
!T11 !SABUS delay time from ALE	! 0!	! 23	! ns	!	!
!T12 !IOW* active delay	! 5!	! 45	! ns	!	!
!T13 !IOW* inactive delay	! 5!	! 45	! ns	!	!
!T14 !SABUS setup time	! 100!	! ---	! ns	!	!
!T15 !IOW*/IOR* precharge	! 140!	! ---	! ns	!	!
!T16 !IOW* pulse width	! 562!	! ---	! ns	!	!
!T17 !IOR* active delay	! 5!	! 45	! ns	!	!
!T18 !IOR* inactive delay	! 5!	! 45	! ns	!	!
!T19 !SDBUS setup time	! 50!	! ---	! ns	!	!
!T20 !DBUS delay time	! 0!	! 50	! ns	!	!
!T21 !DBUS hold time from IOW*	! 12!	! ---	! ns	!	!
!T22 !SDBUS setup time to IOR*	! 42!	! ---	! ns	!	!
!T23 !DBUS delay time	! 8!	! 32	! ns	!	!

!SYM!PARAMETER	!MIN!	!TYP	! MAX	!UNIT!	TEST COND.
!T24!DBUS float from IOR*	!	!	!	50	!ns !
!T25!SDBUS write delay from DBUS!	!	!	!	25	!ns !
!T26!SDBUS read delay from XDBUS!	!	!	!	10	!ns !
!T27!XDBUS write delay	!	!	!	10	!ns !
!T28!XDBUS read delay from IOR*	!	!	!	465.5	!ns !
!T29!PRCLK clock period	!	!	125	!	!ns !
!T30!PRCLK high period	!	!	62.5	!	!ns !
!T31!SCLK high period	!	!	125	!	!ns !
!T32!SCLK clock period	!	!	250	!	!ns !
!T33!IOW* pulse width	!	1625	!	!	!ns !
!T34!IOW*/IOR* overlap time	!	1330	!	!	!ns !
!T35!SDBUS setup time	!	50	!	!	!ns !
!T36!IOR* pulse width	!	1625	!	!	!ns !

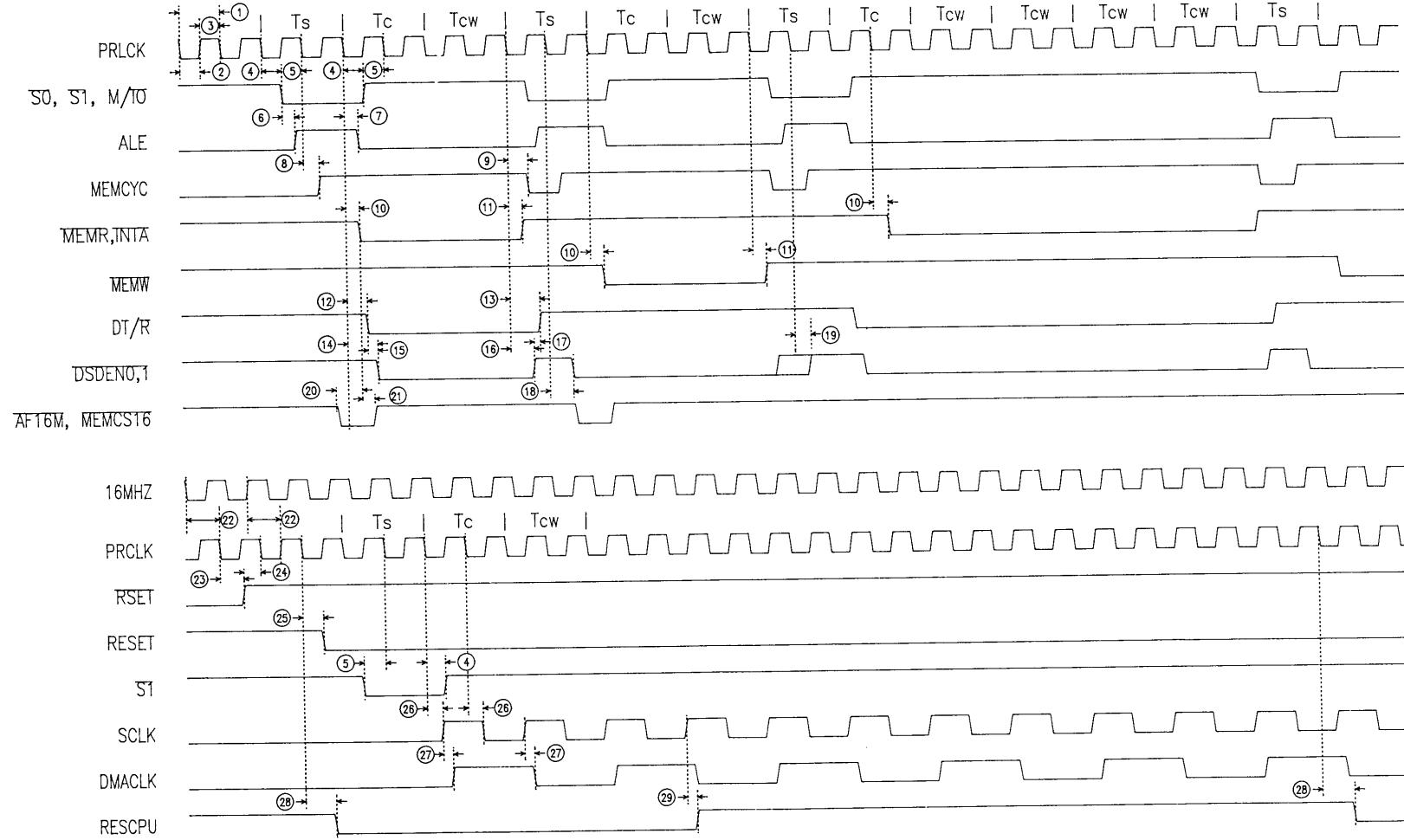


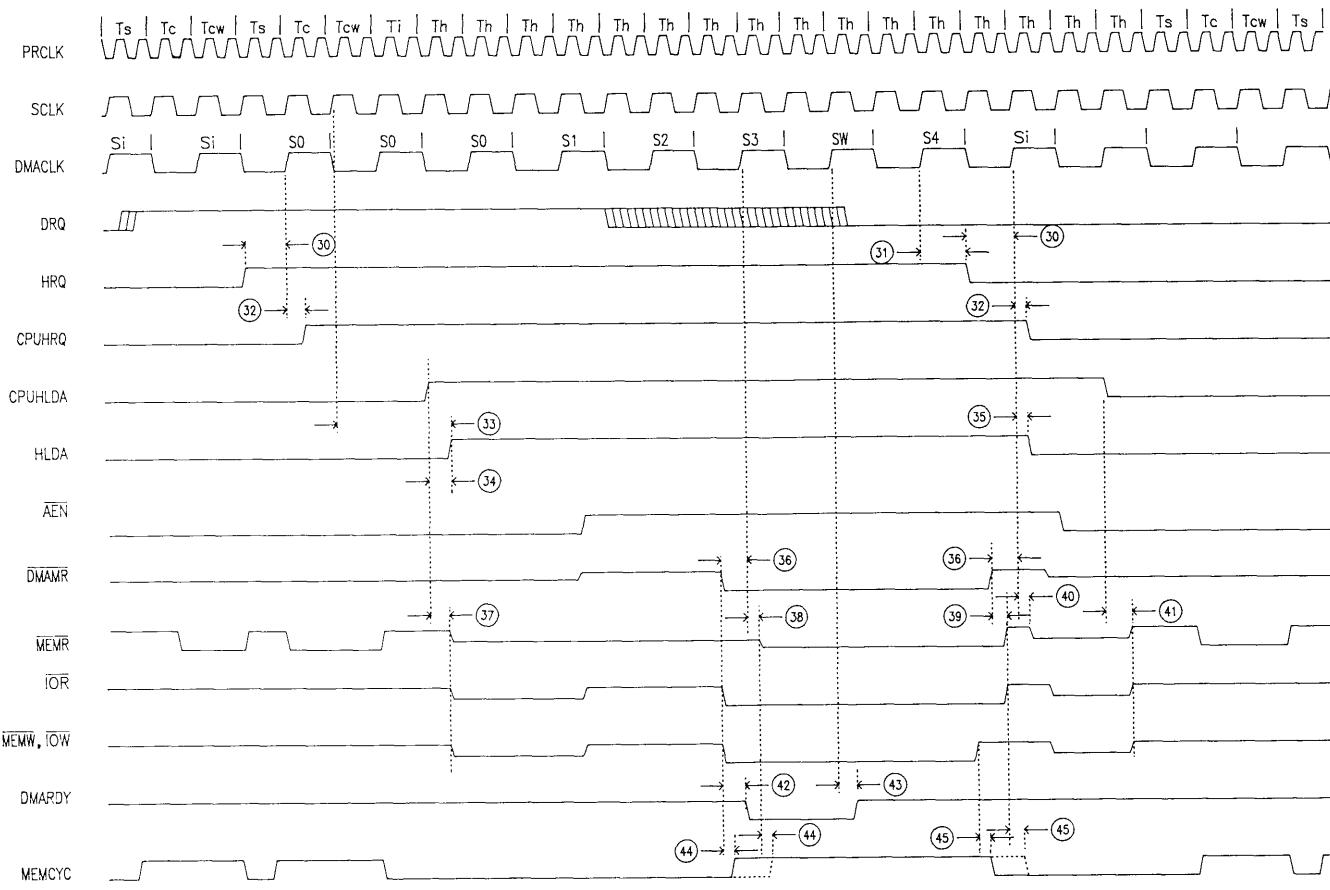


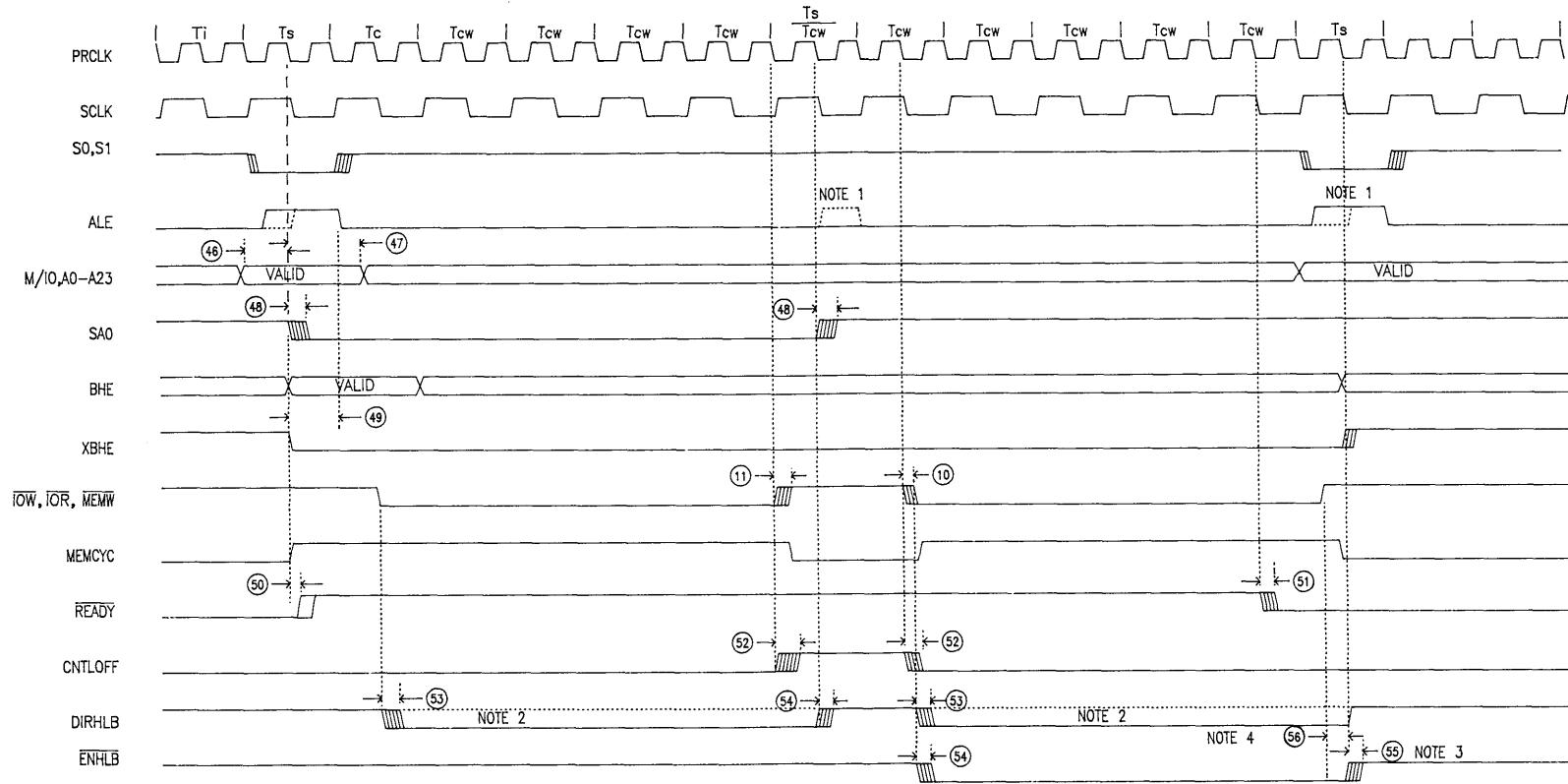


8 MHZ AND FOUR WAIT STATES

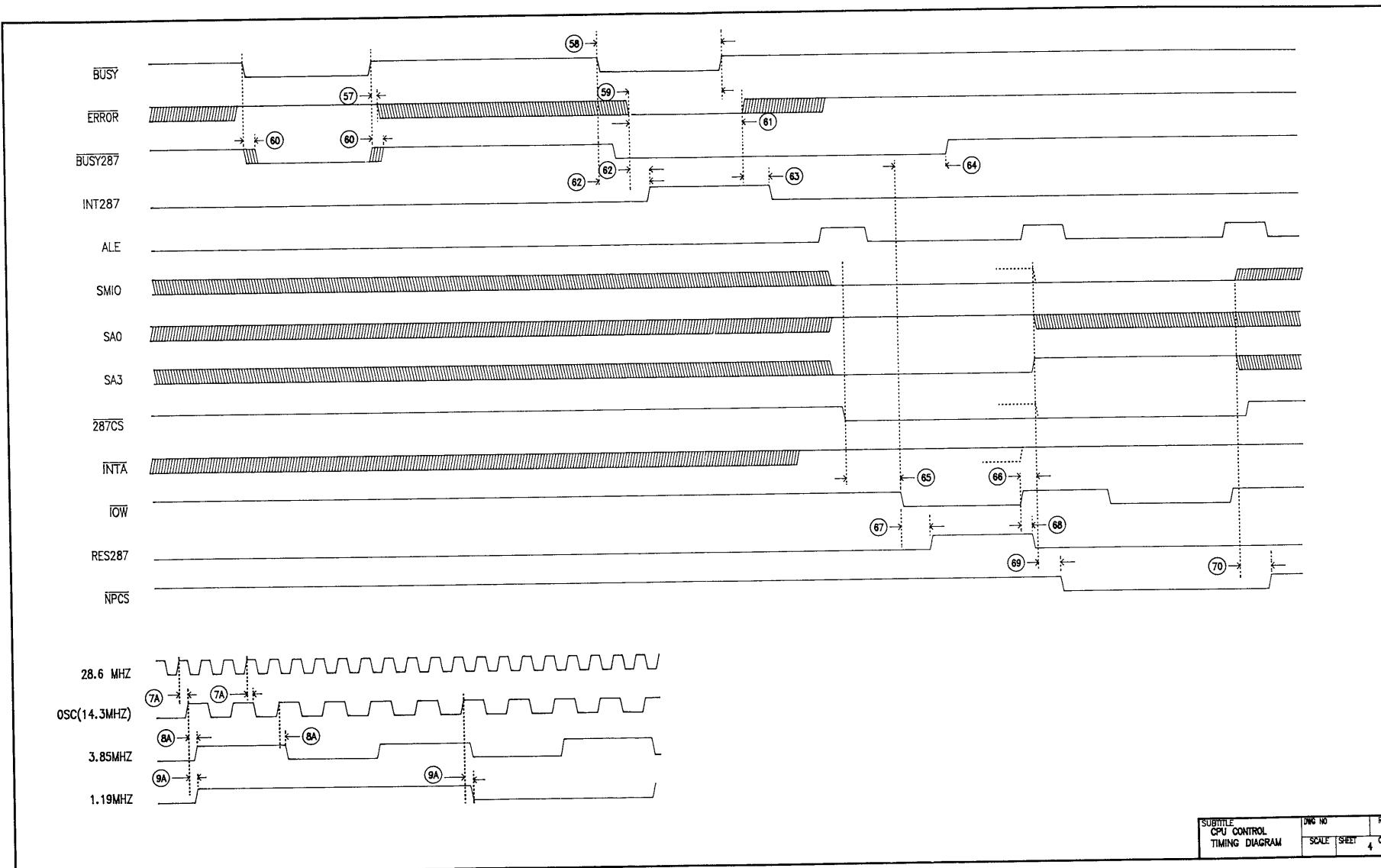
4 MHZ AND TWO WAIT STATES

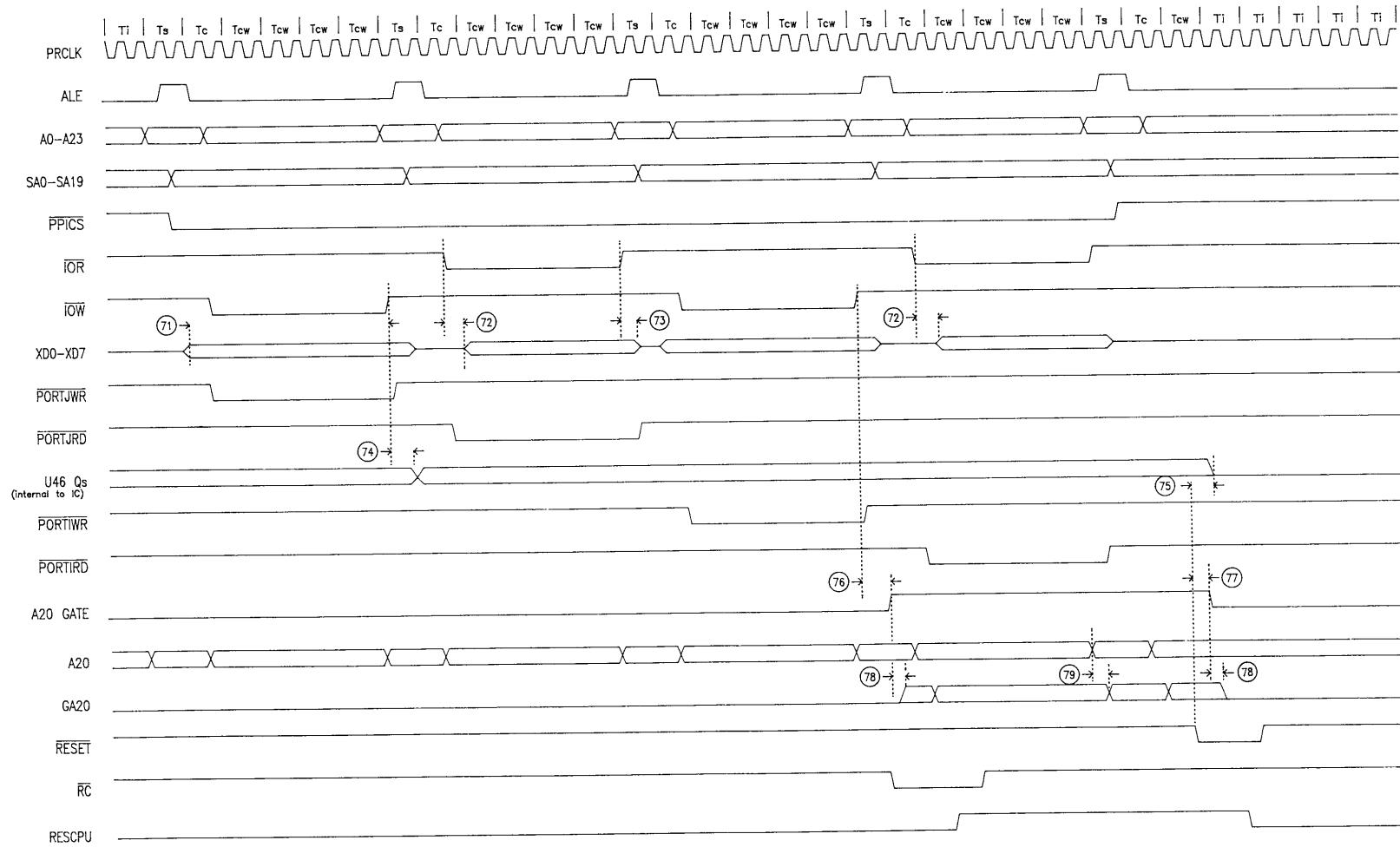




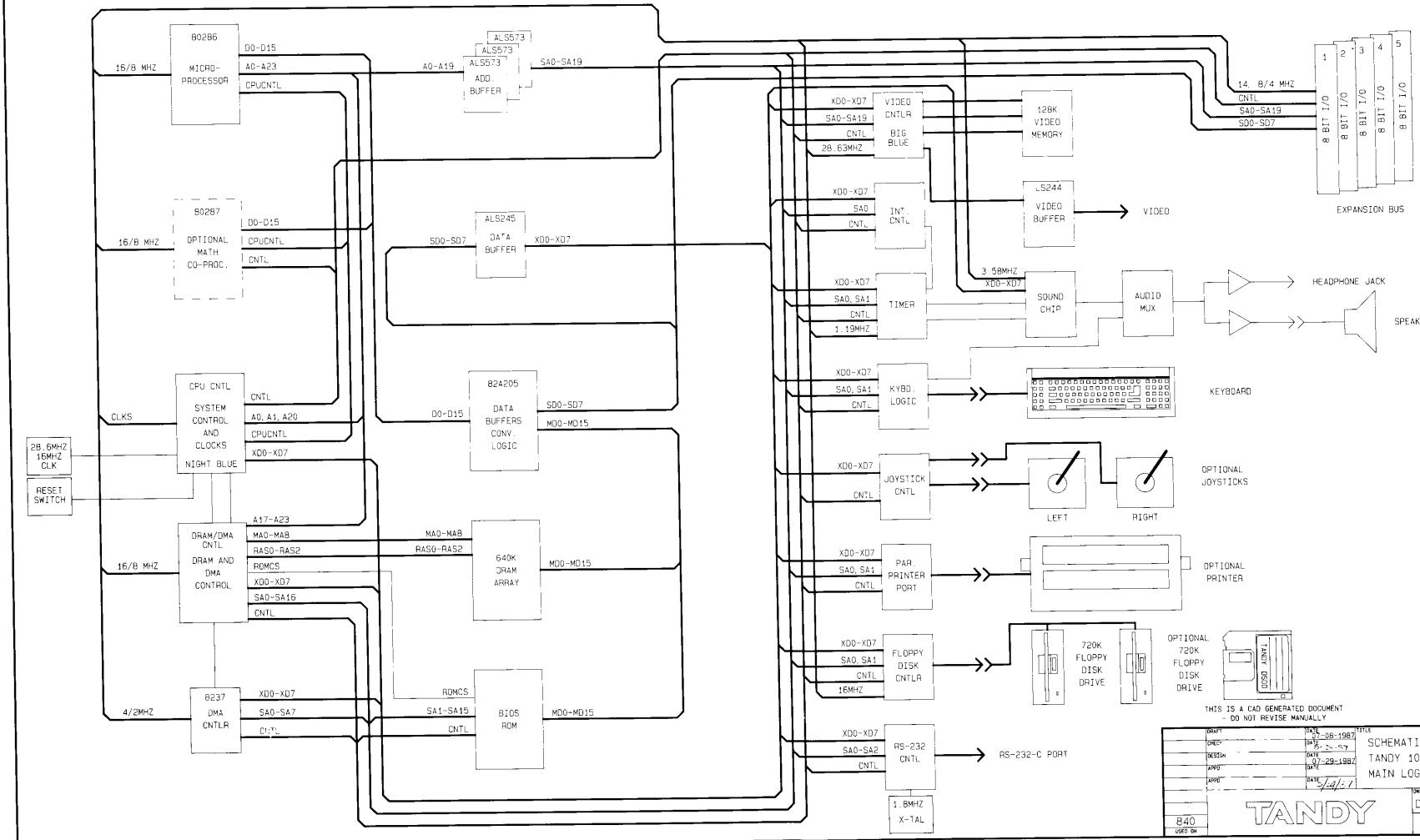


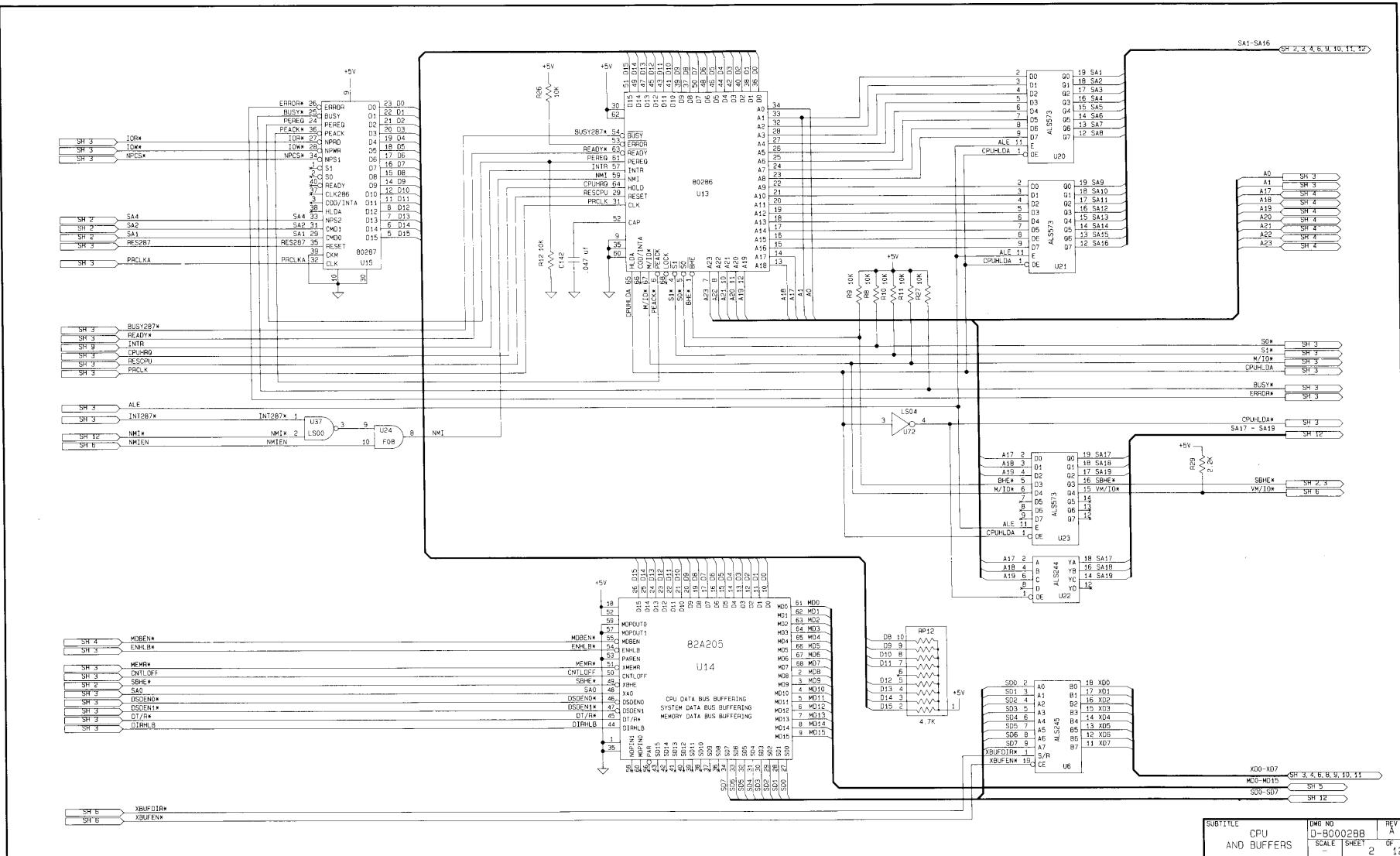
NOTES
 1. SALE and CONVALE internal signals shown for reference only.
 2. DIRHLB in solid line for I/O write cyc. dotted line for I/O read cycle.
 3. Timing shown during I/O write cycle.
 4. Timing shown during I/O read cycle.





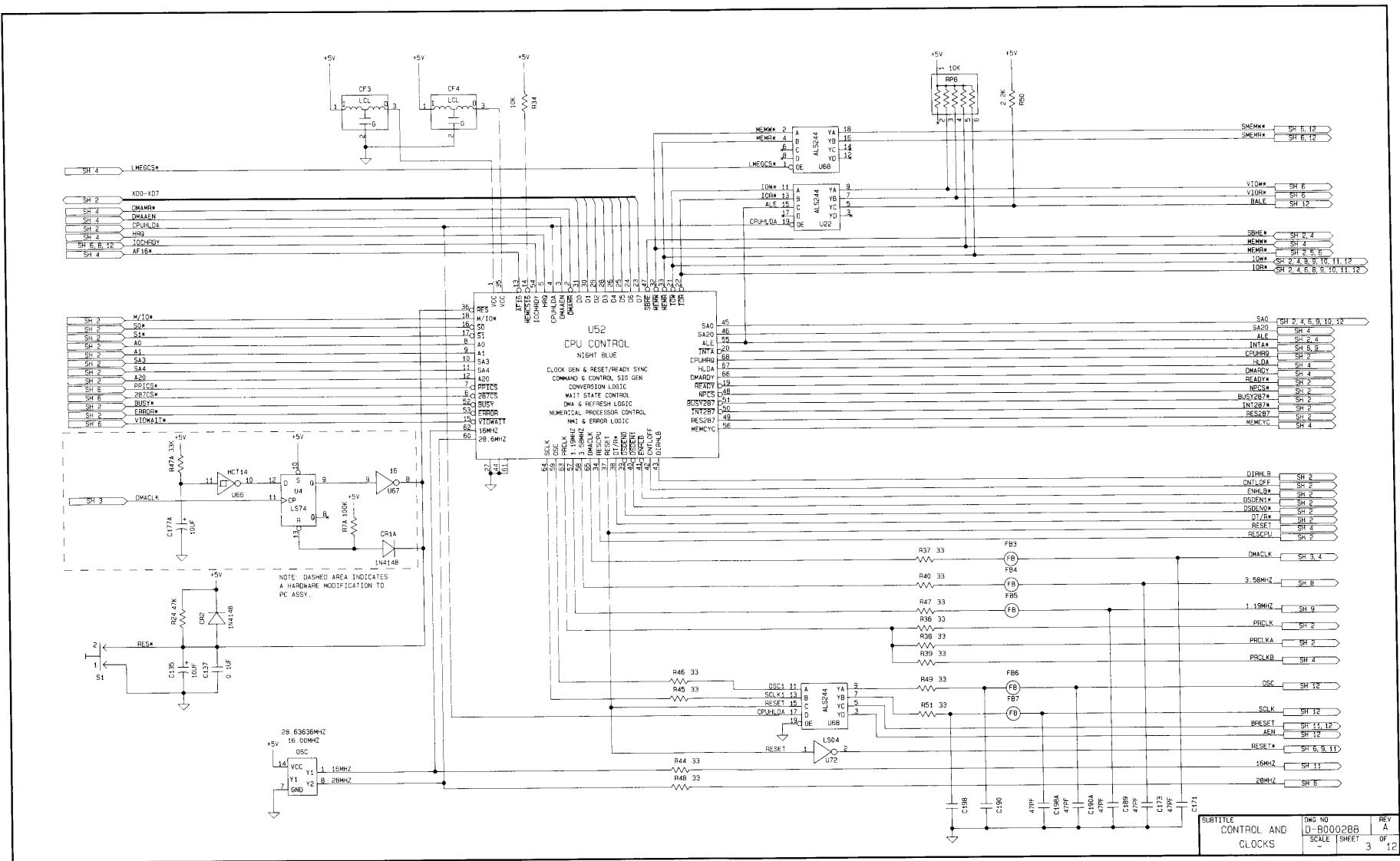
LETTER	REVISIION DESCRIPTION	DATE	APPROVED
A	CHANGE PER TCO# 13244, 13250, 13363-13365	7/8/87	

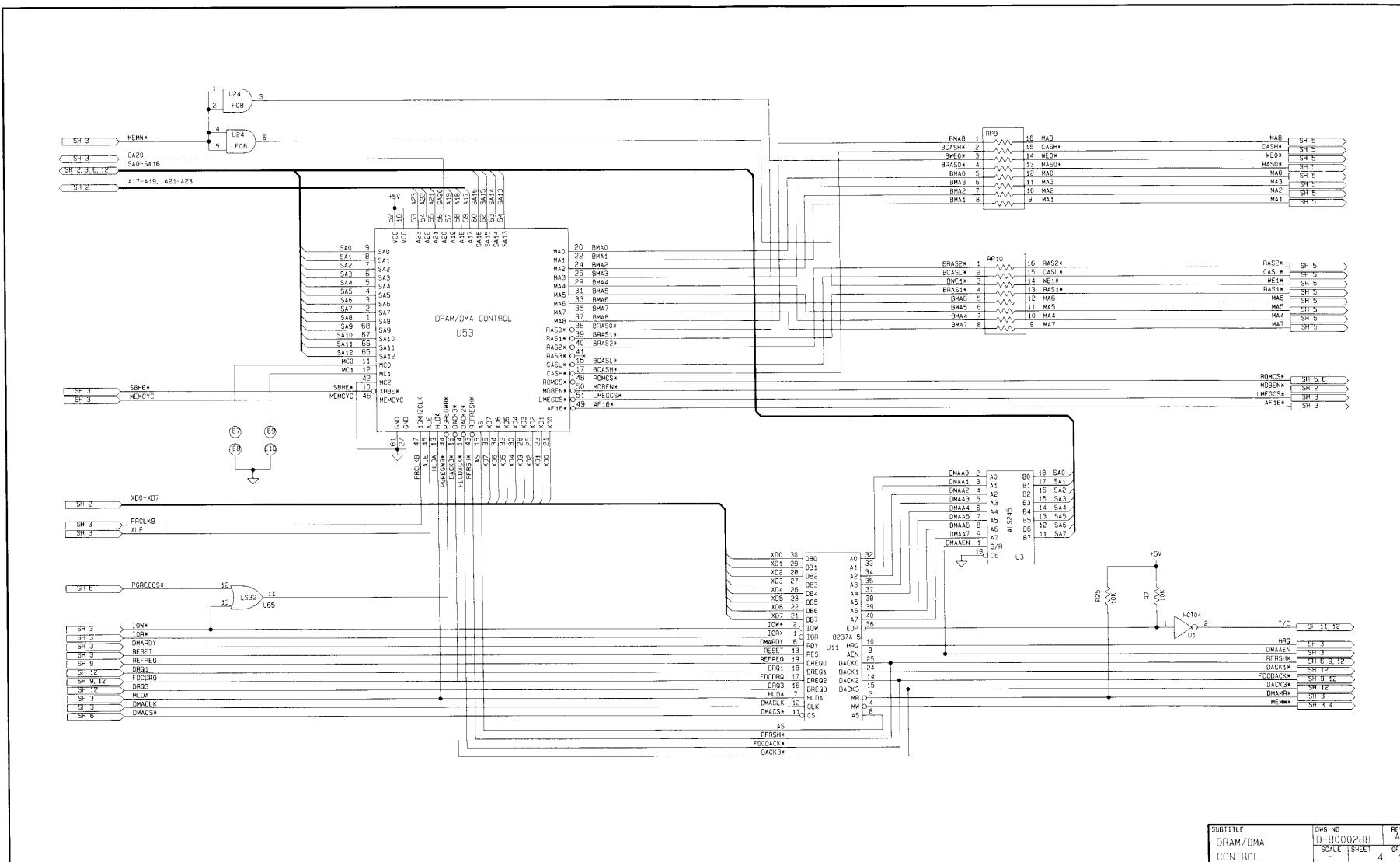


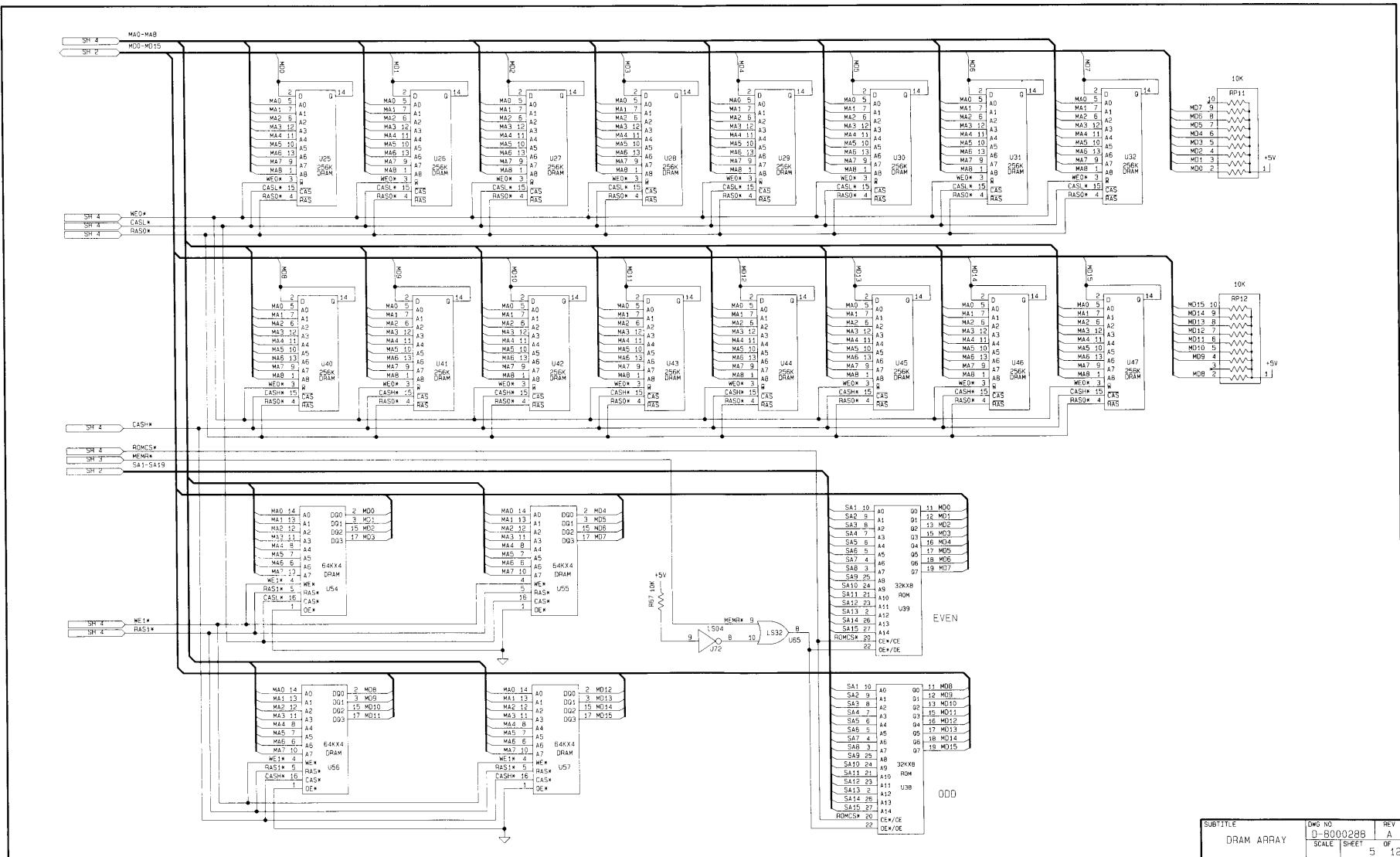


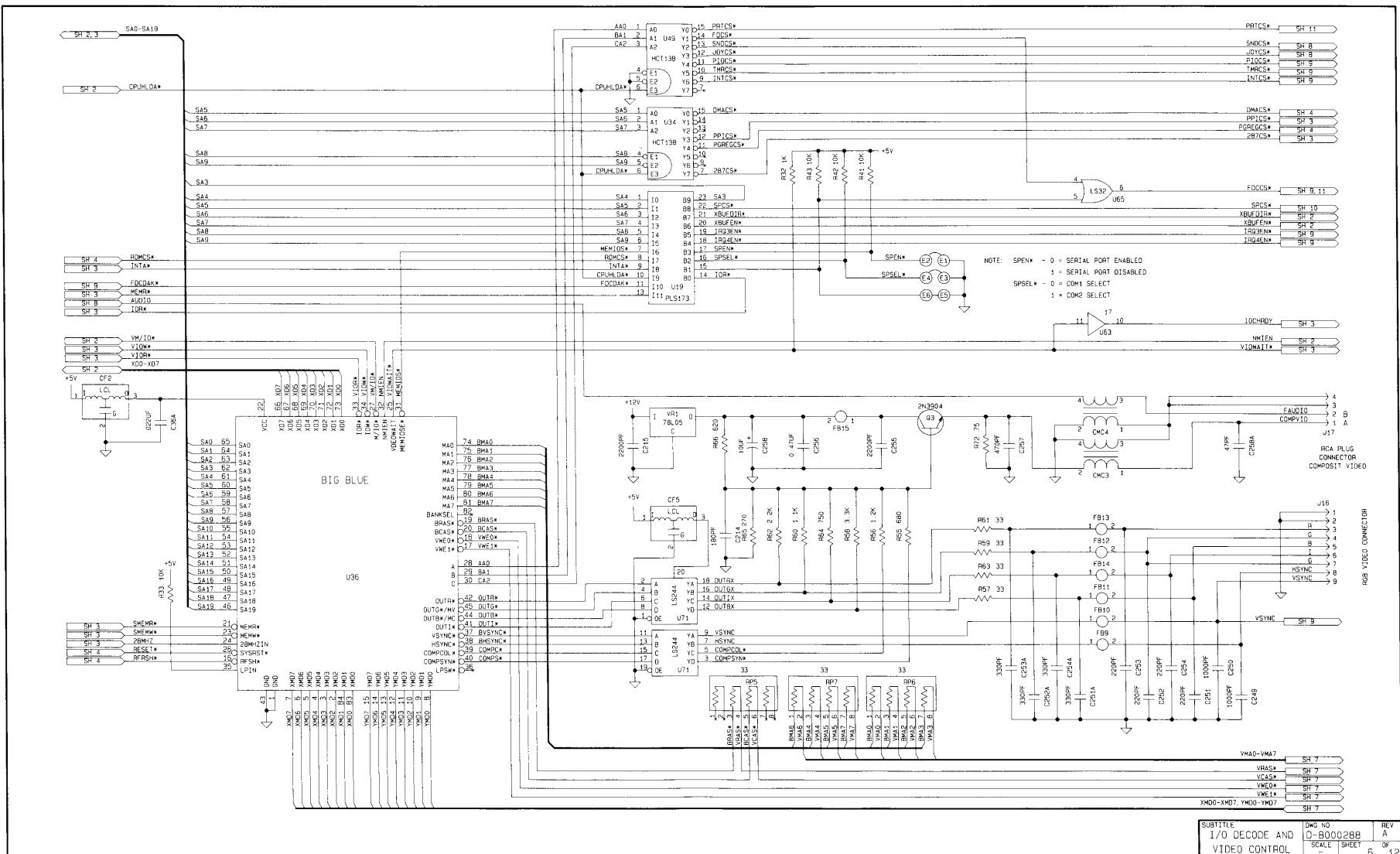
SUBTITLE: CPU AND BUFFERS
DWG NO: D-8000266
SCALE SHEET: 2 of 12
REV: A

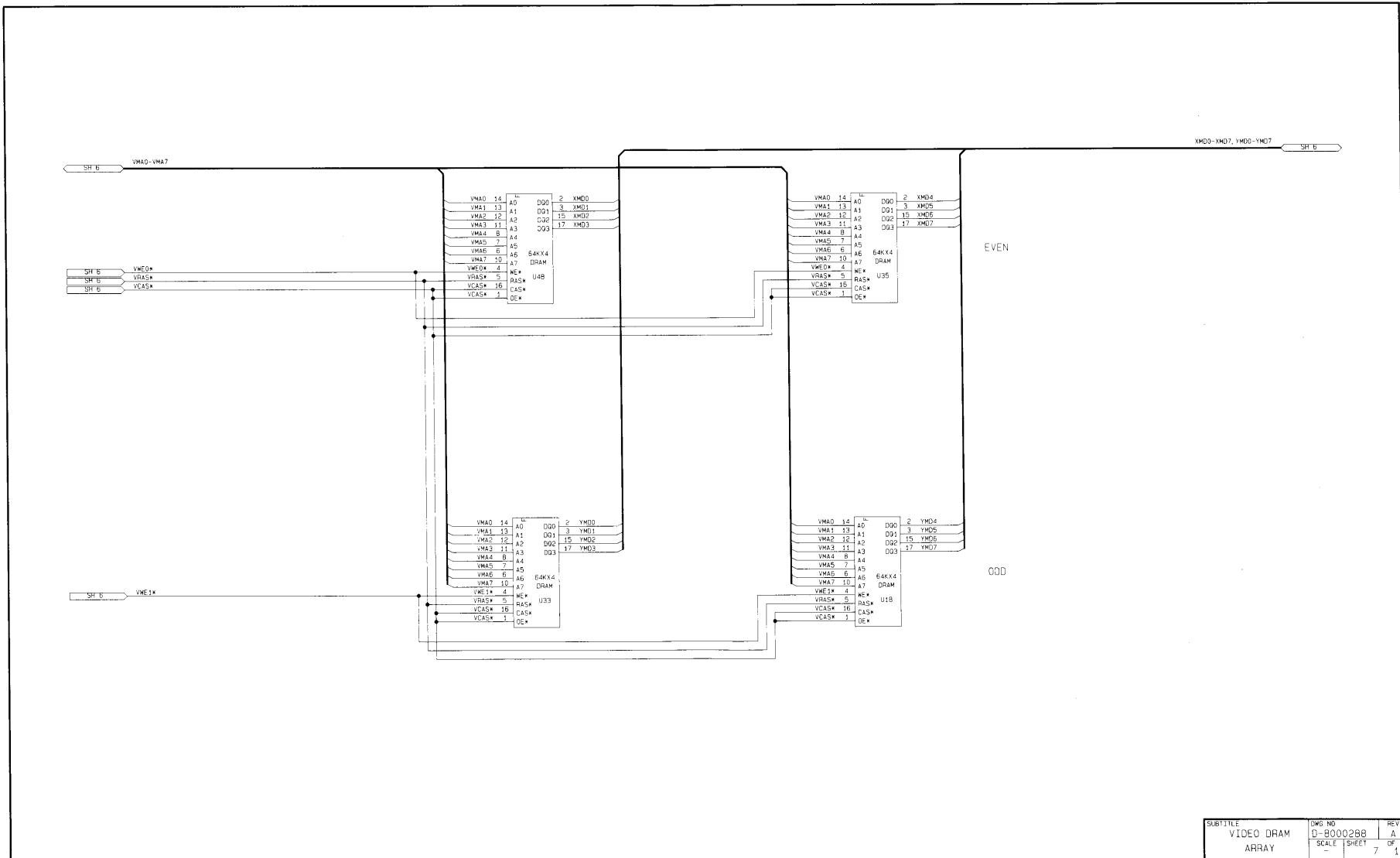
X00-X07 SH 3, 4, 6, 9, 10, 11
M00-M15 SH 5
S00-S07 SH 12

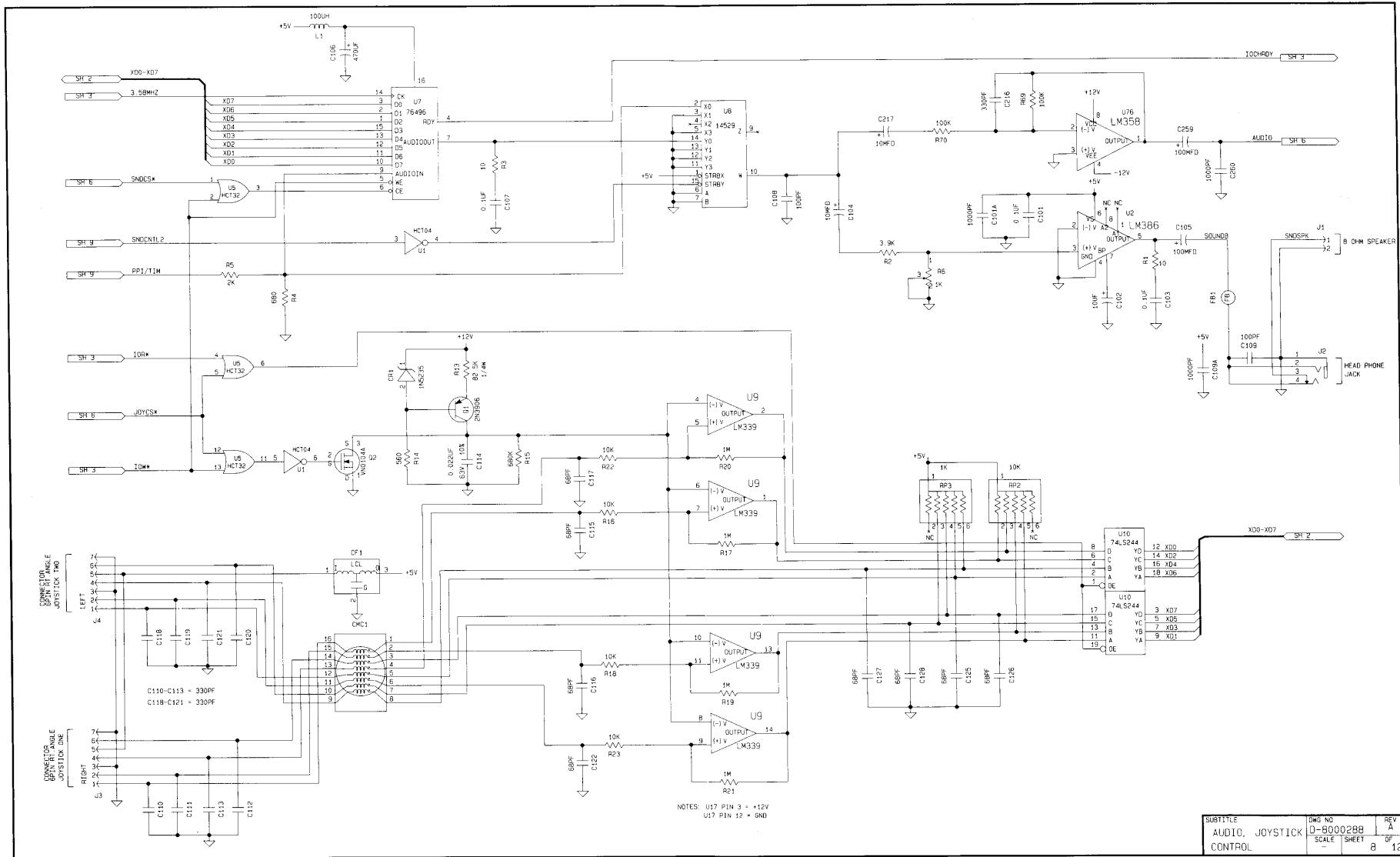


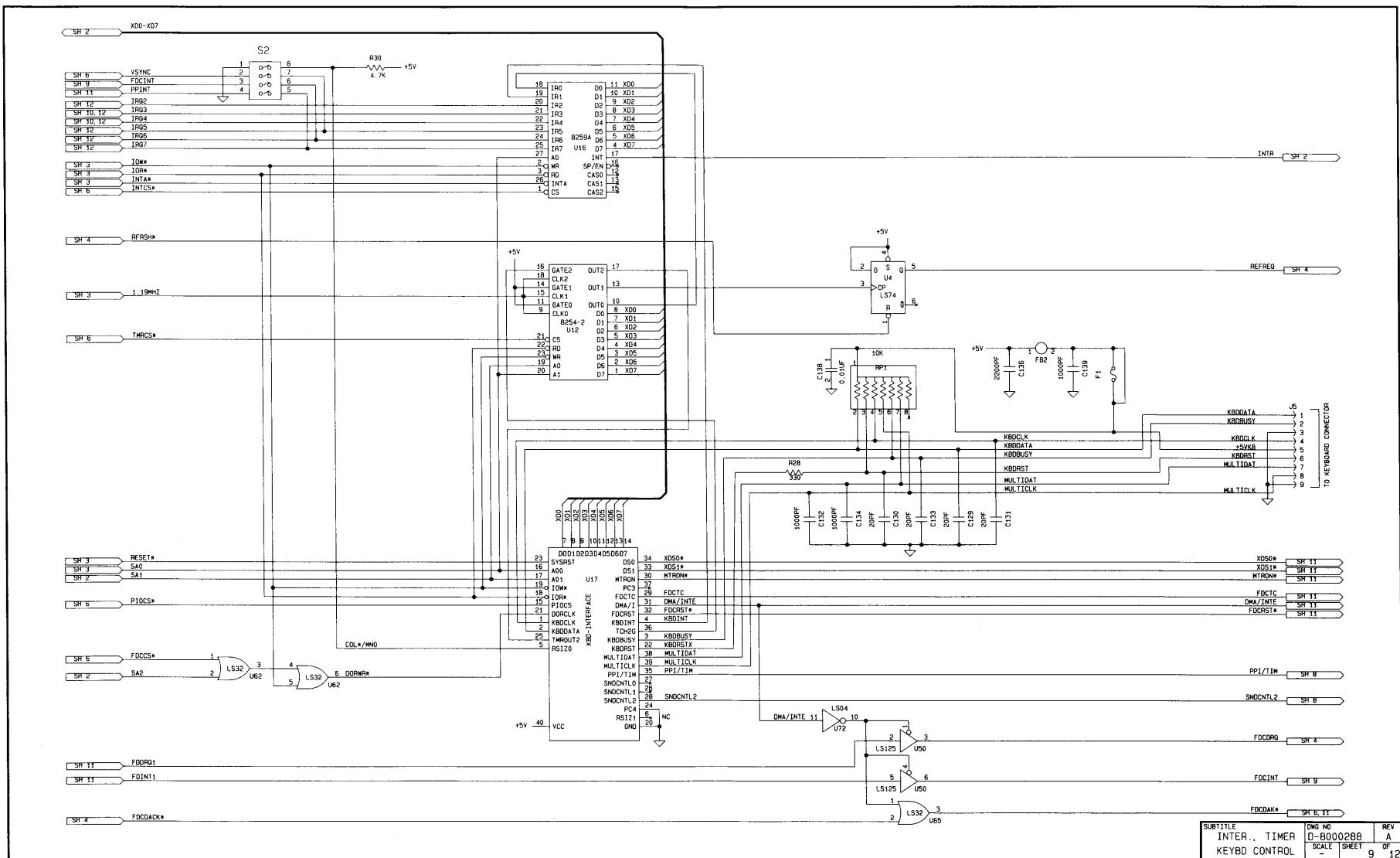


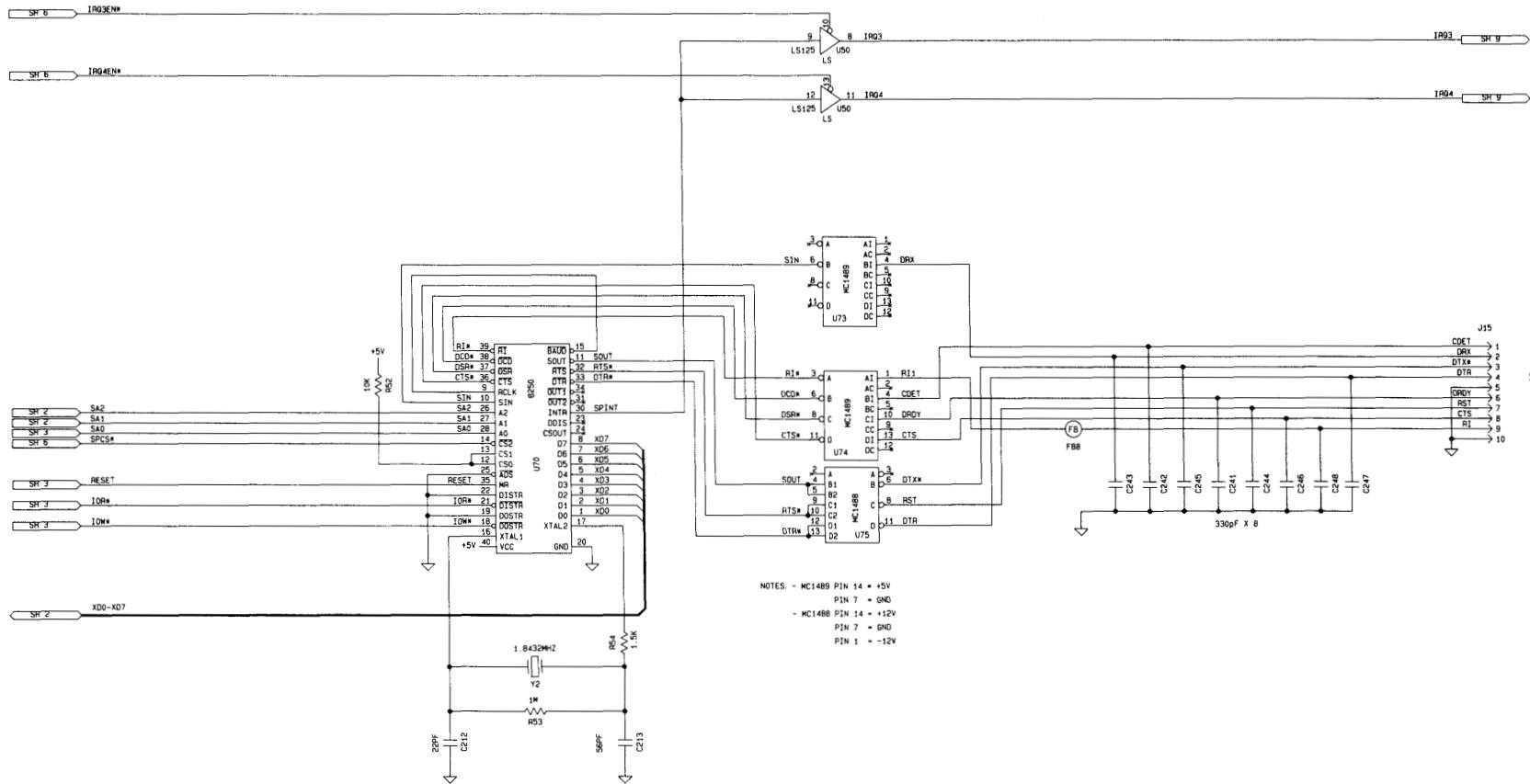


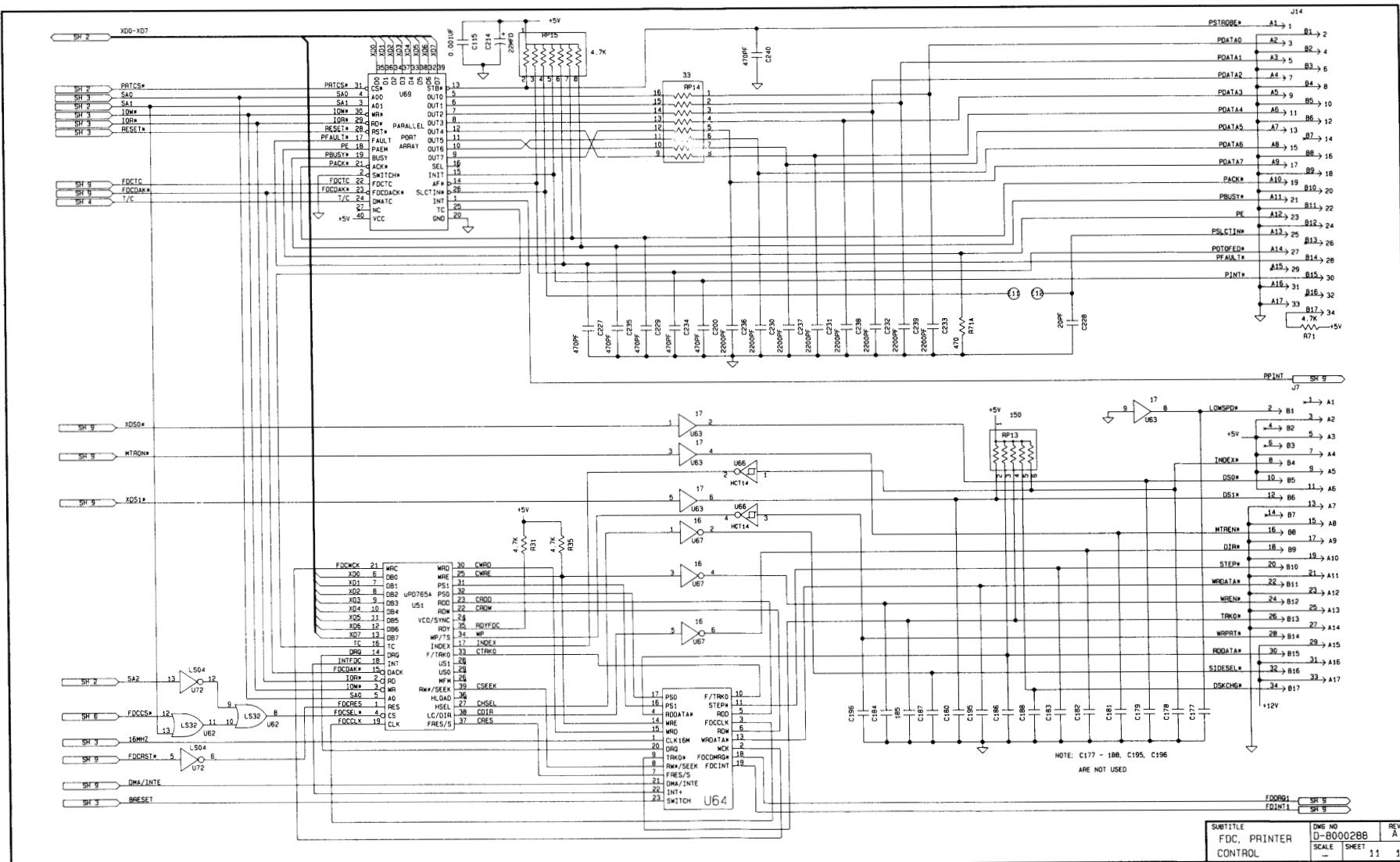


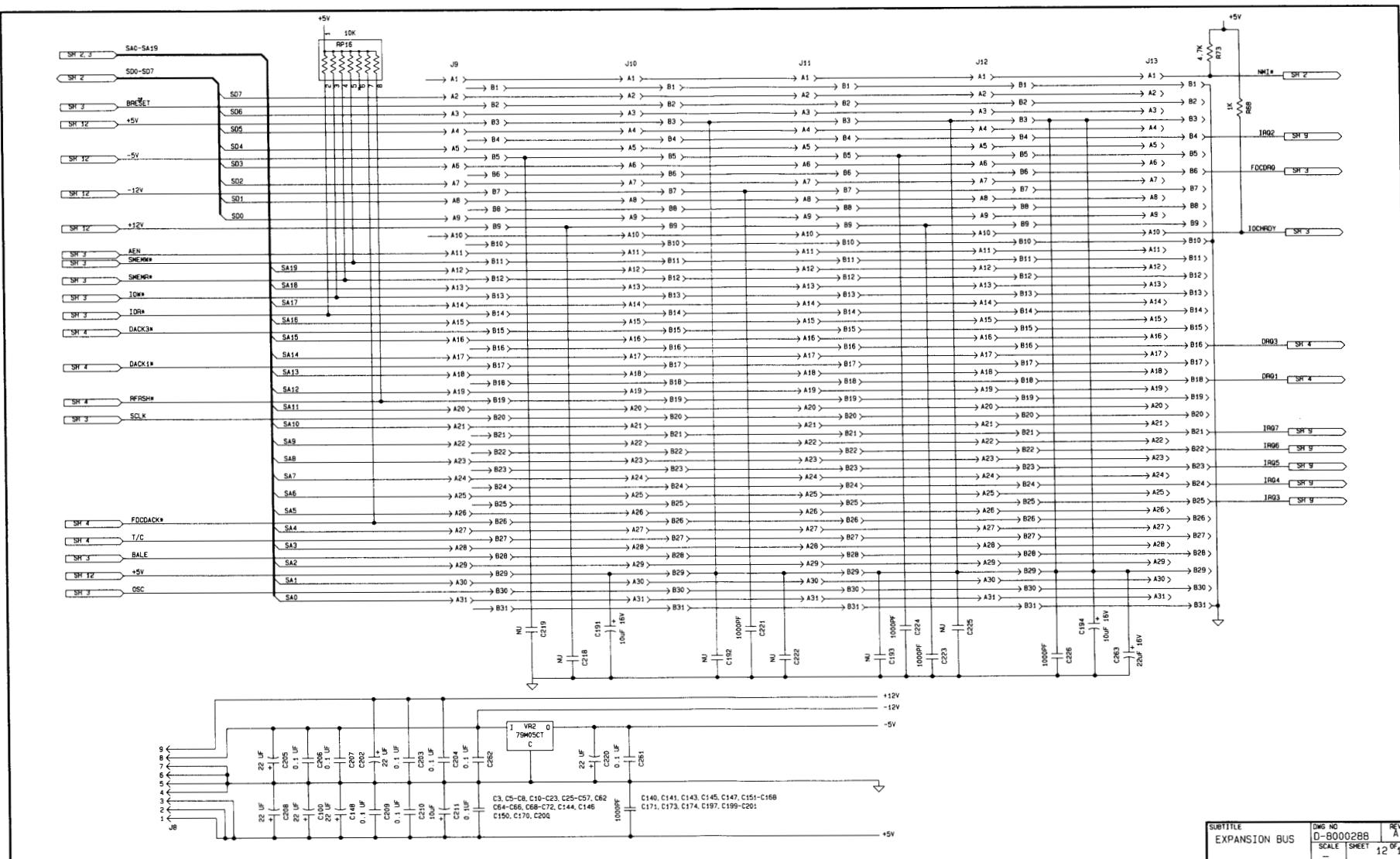




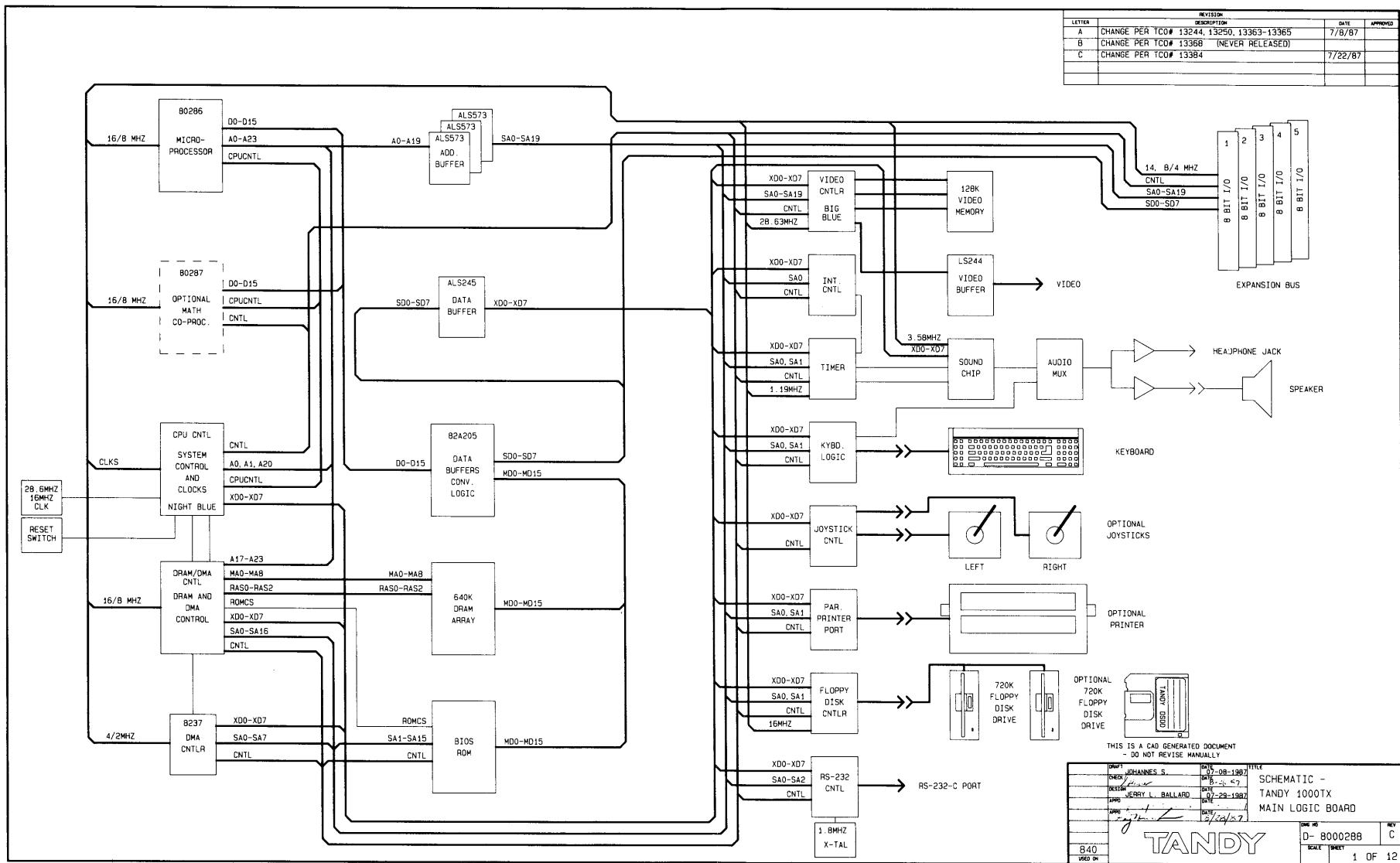


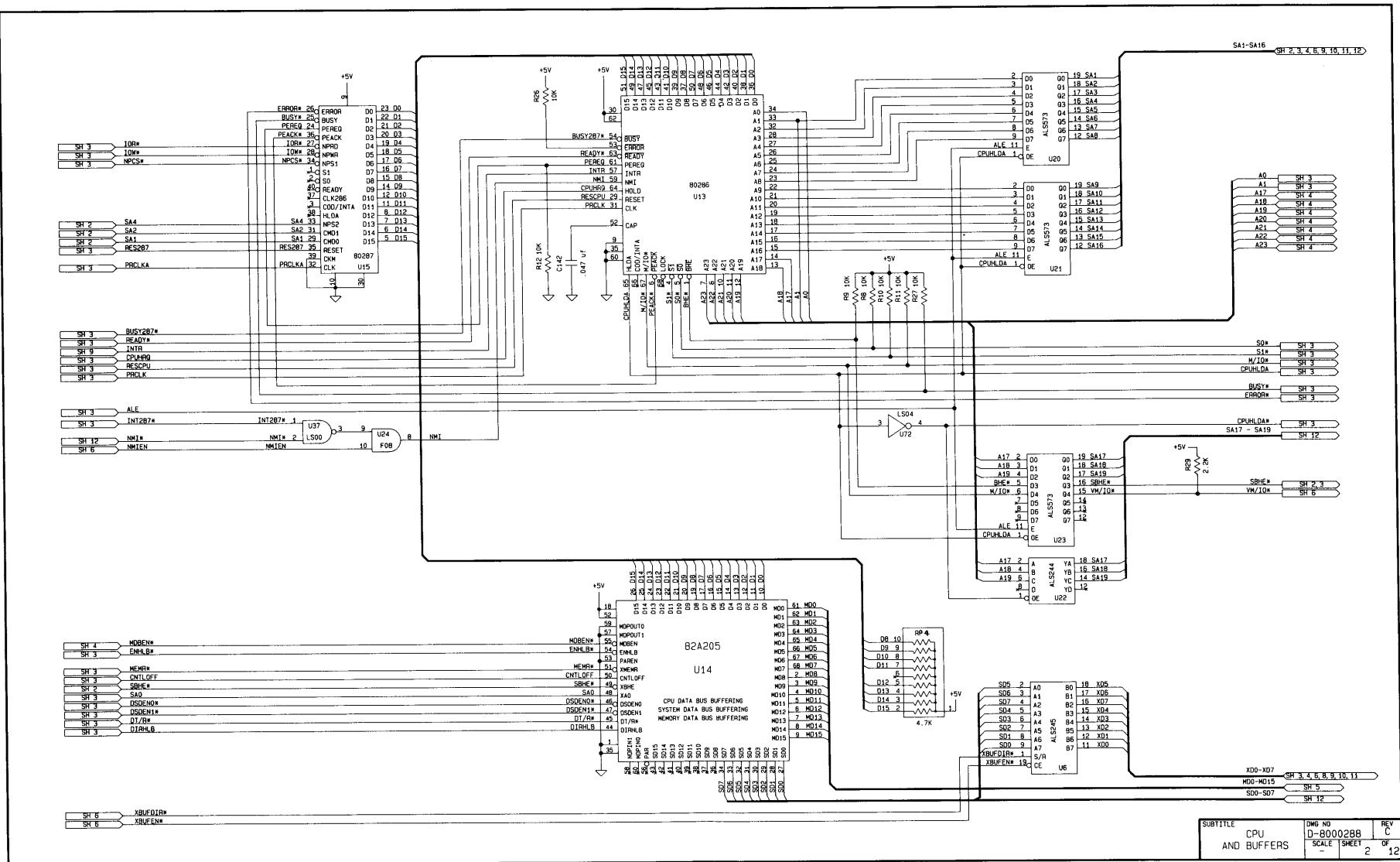


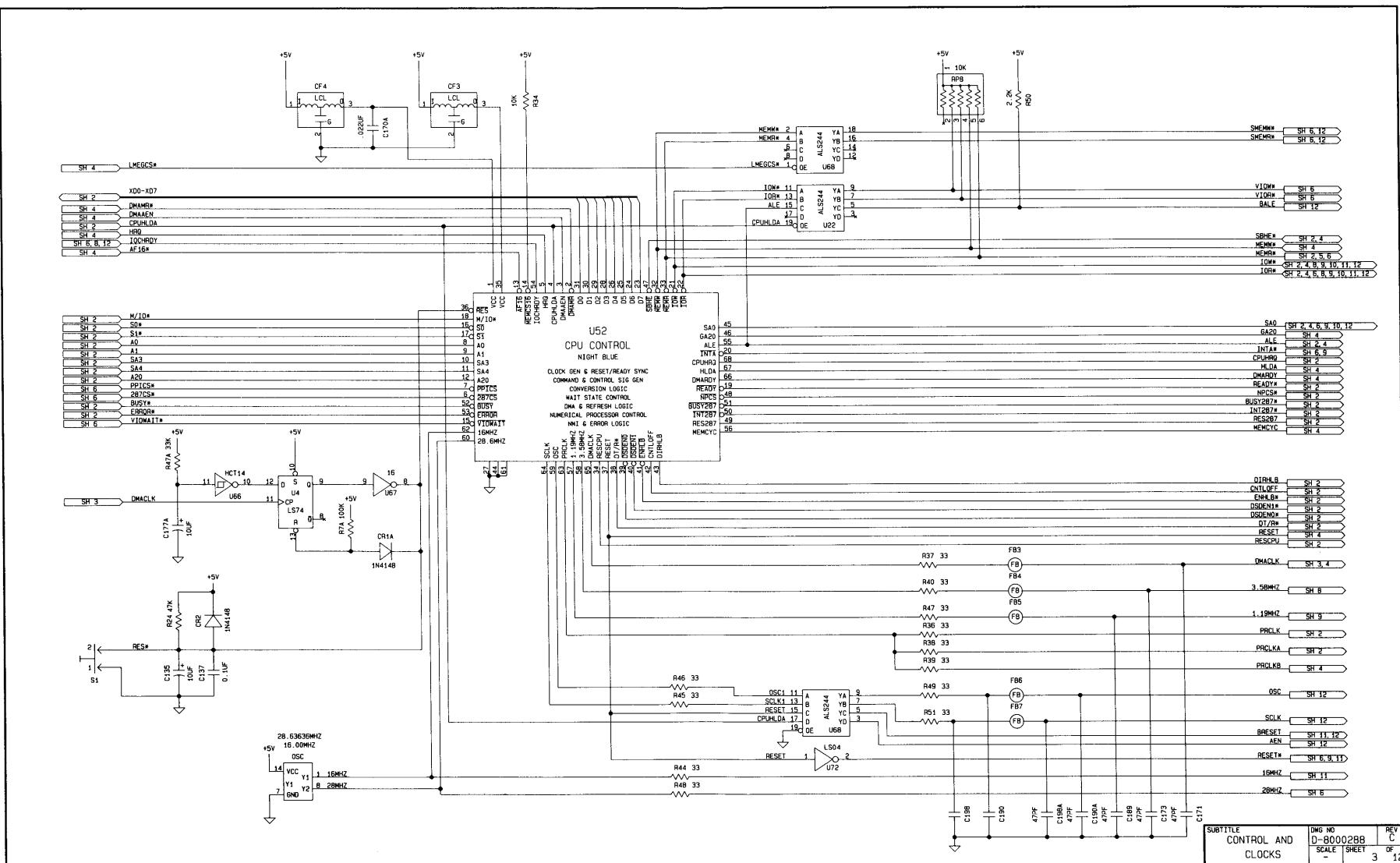


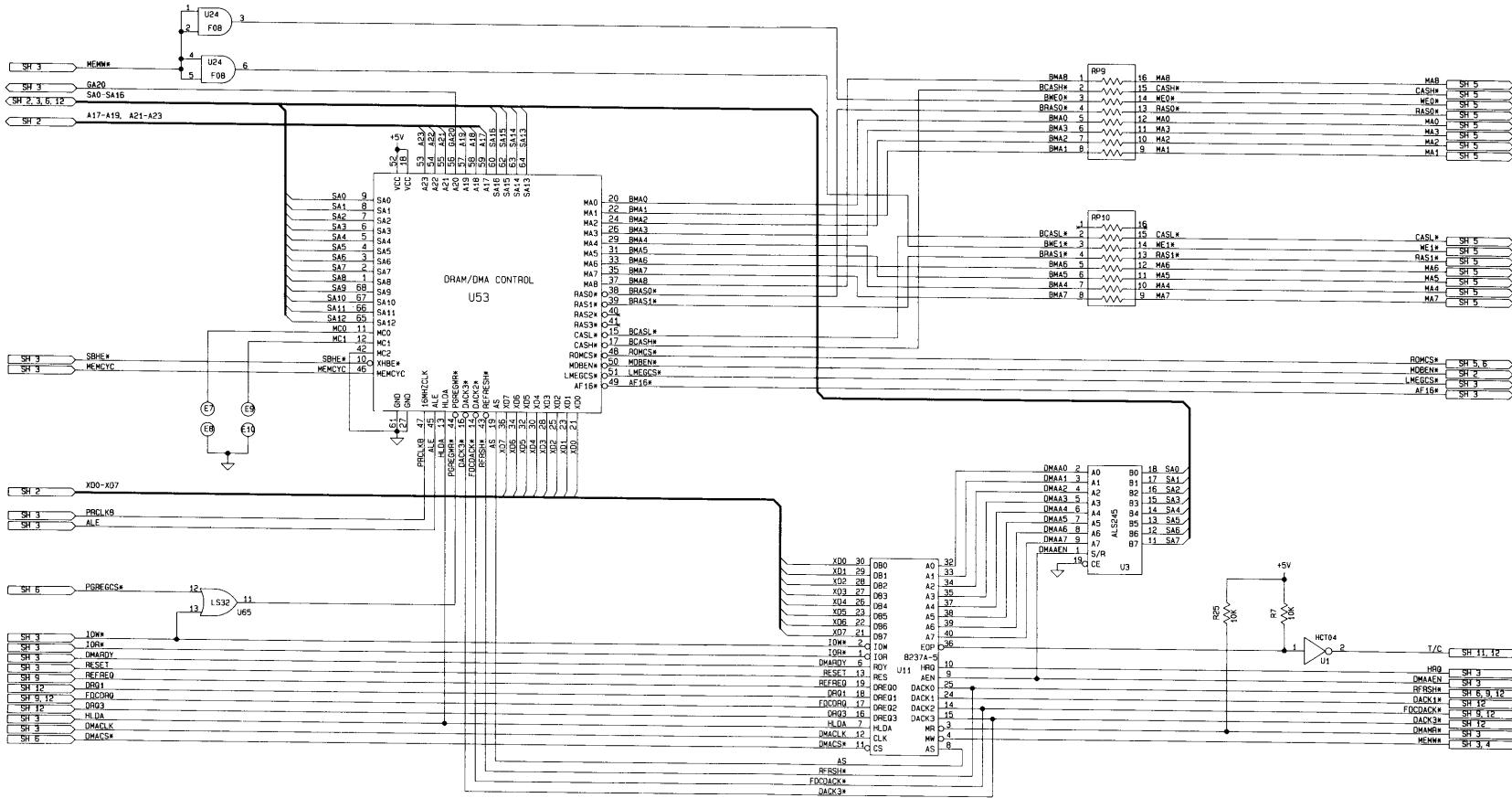


REVISION		DATE	APPROVED
LETTER	DESCRIPTION		
A	CHANGE PER TCO# 13244, 13250, 13363-13365	7/8/87	
B	CHANGE PER TCO# 13368 (NEVER RELEASED)		
C	CHANGE PER TCO# 13384	7/22/87	

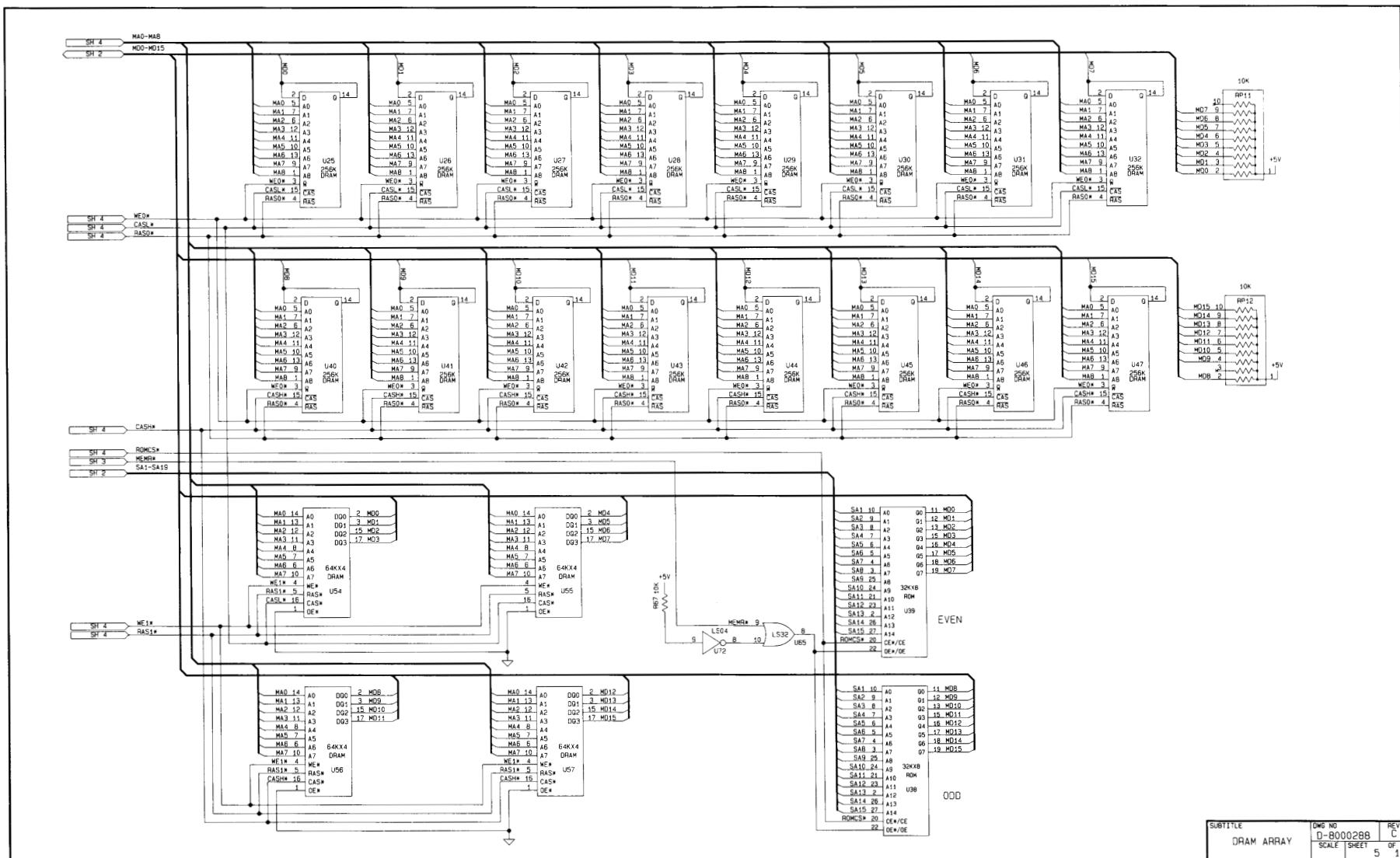


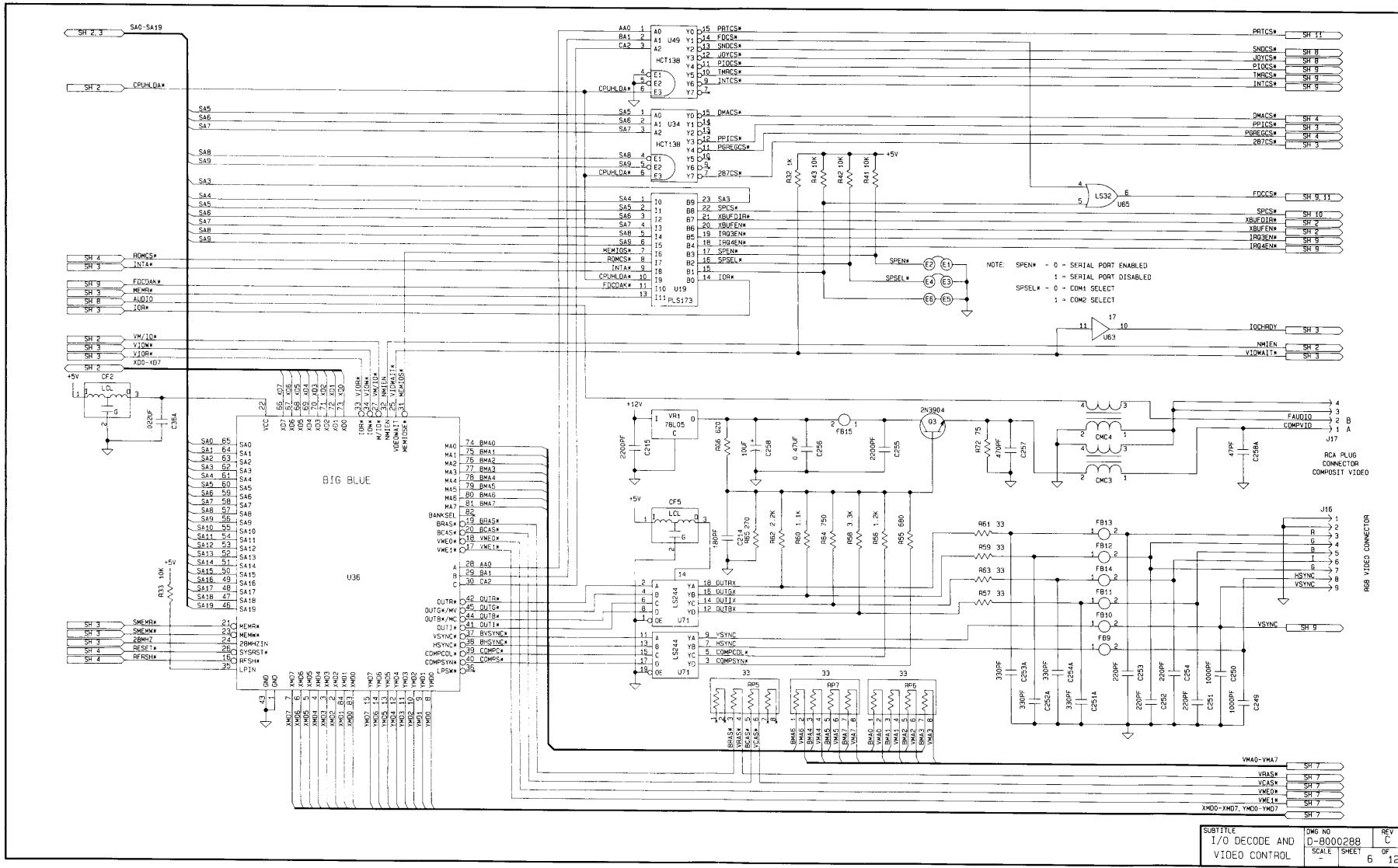


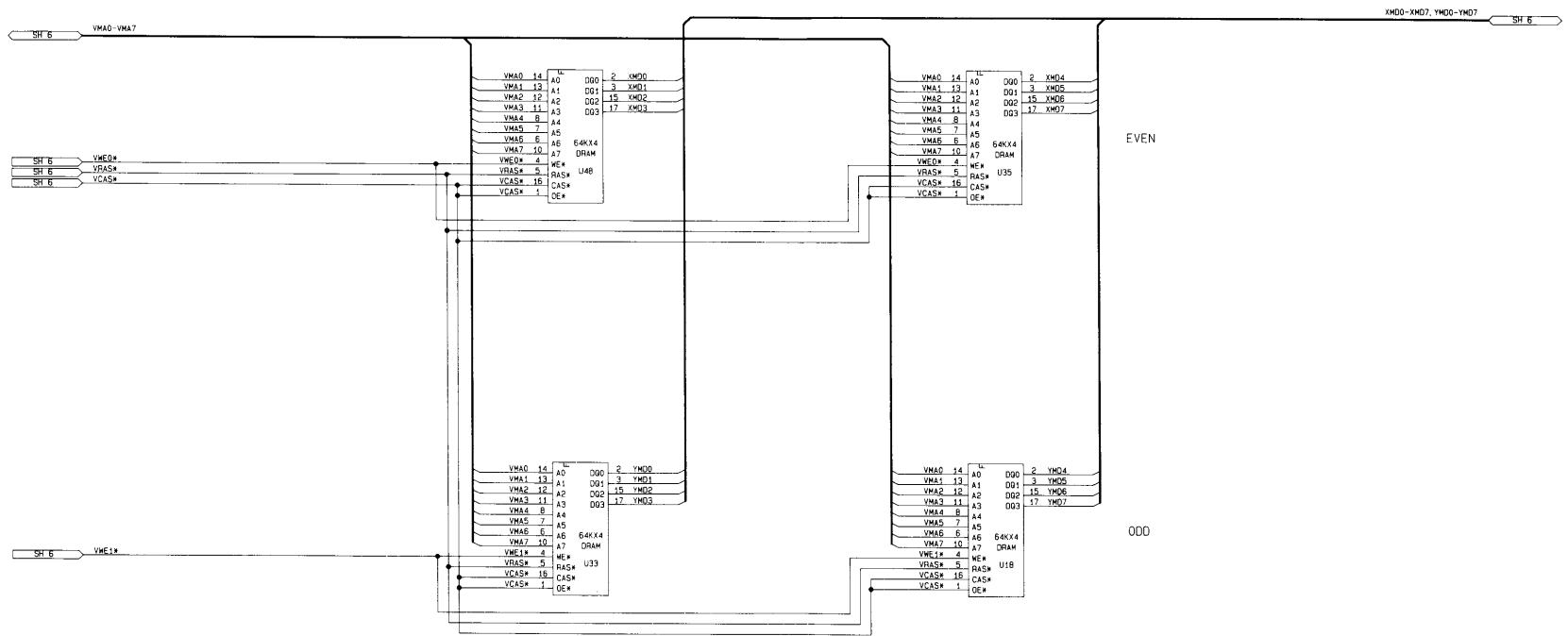


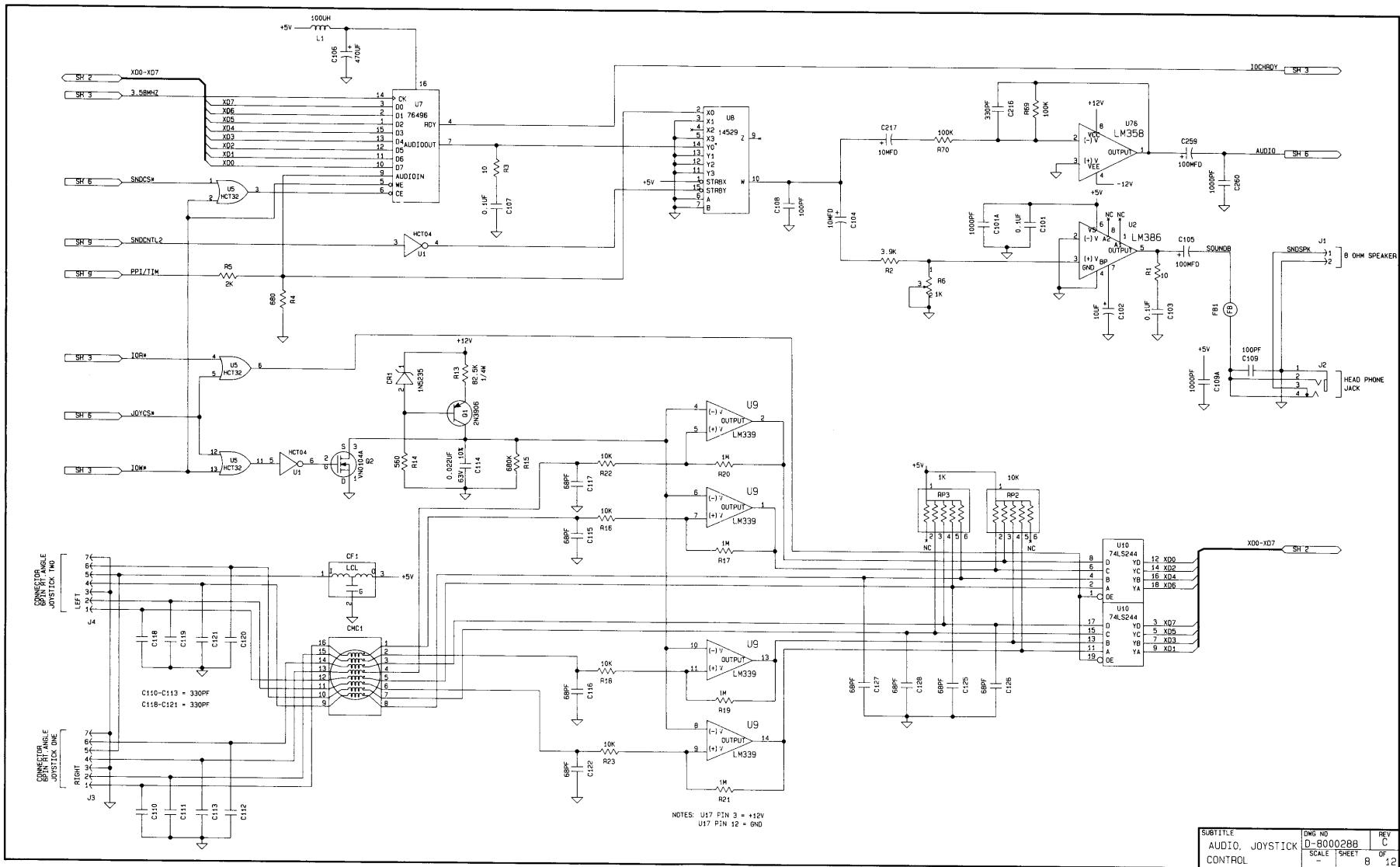


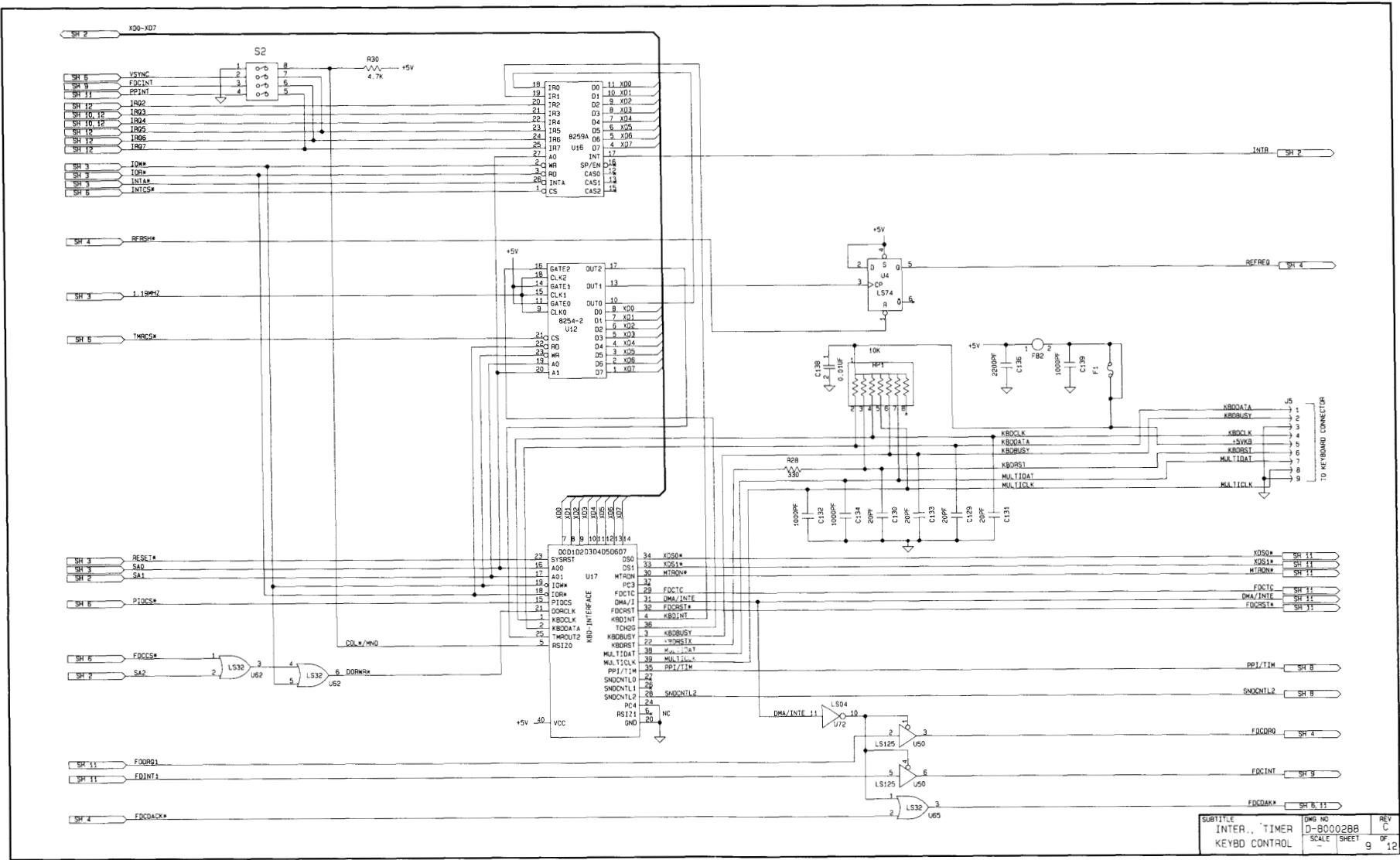
SUBTITLE DRAM/DMA CONTROL	DMG NO D-8000288	REV C
	SCALE -	SHEET 4 OF 12

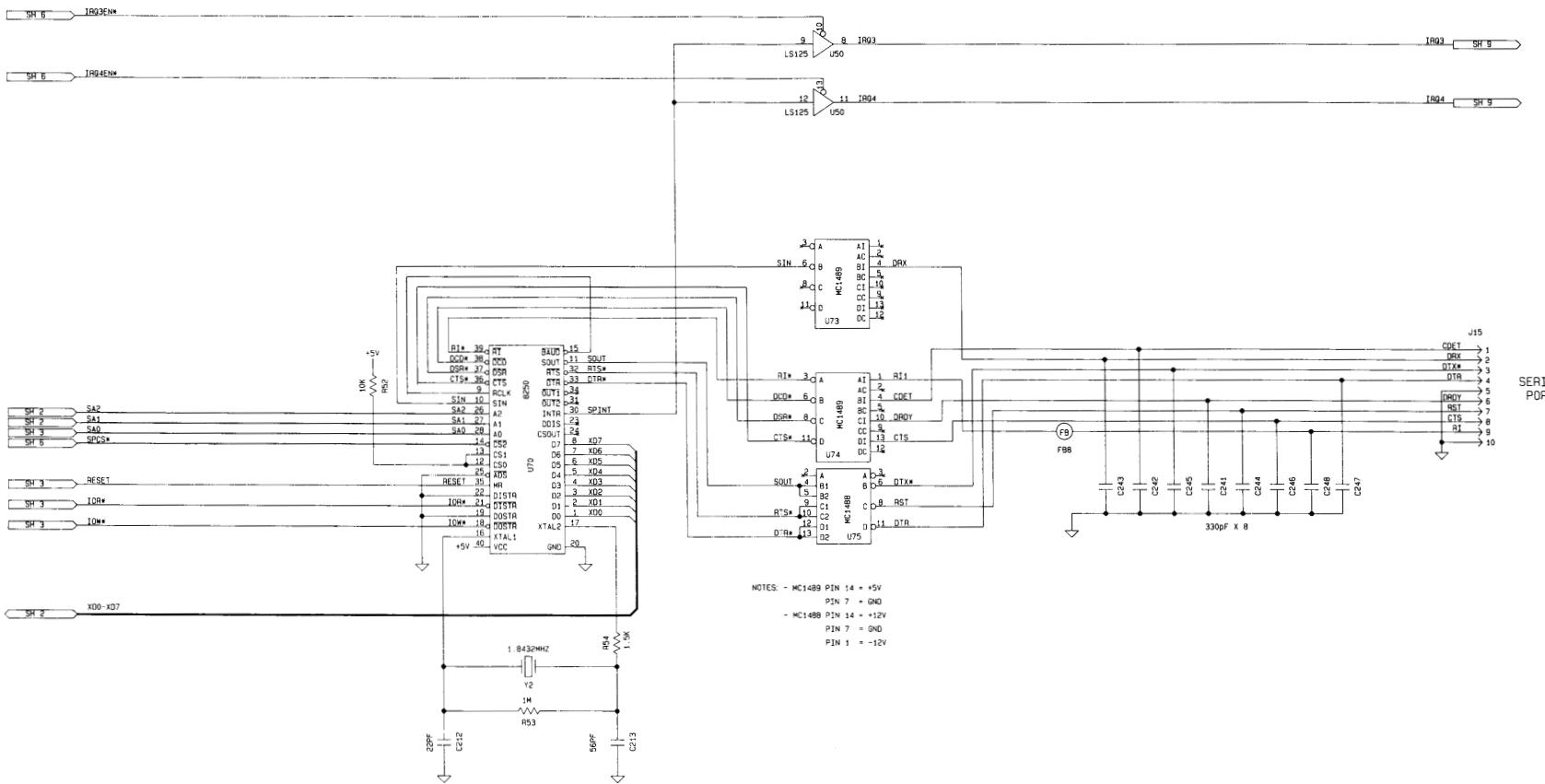




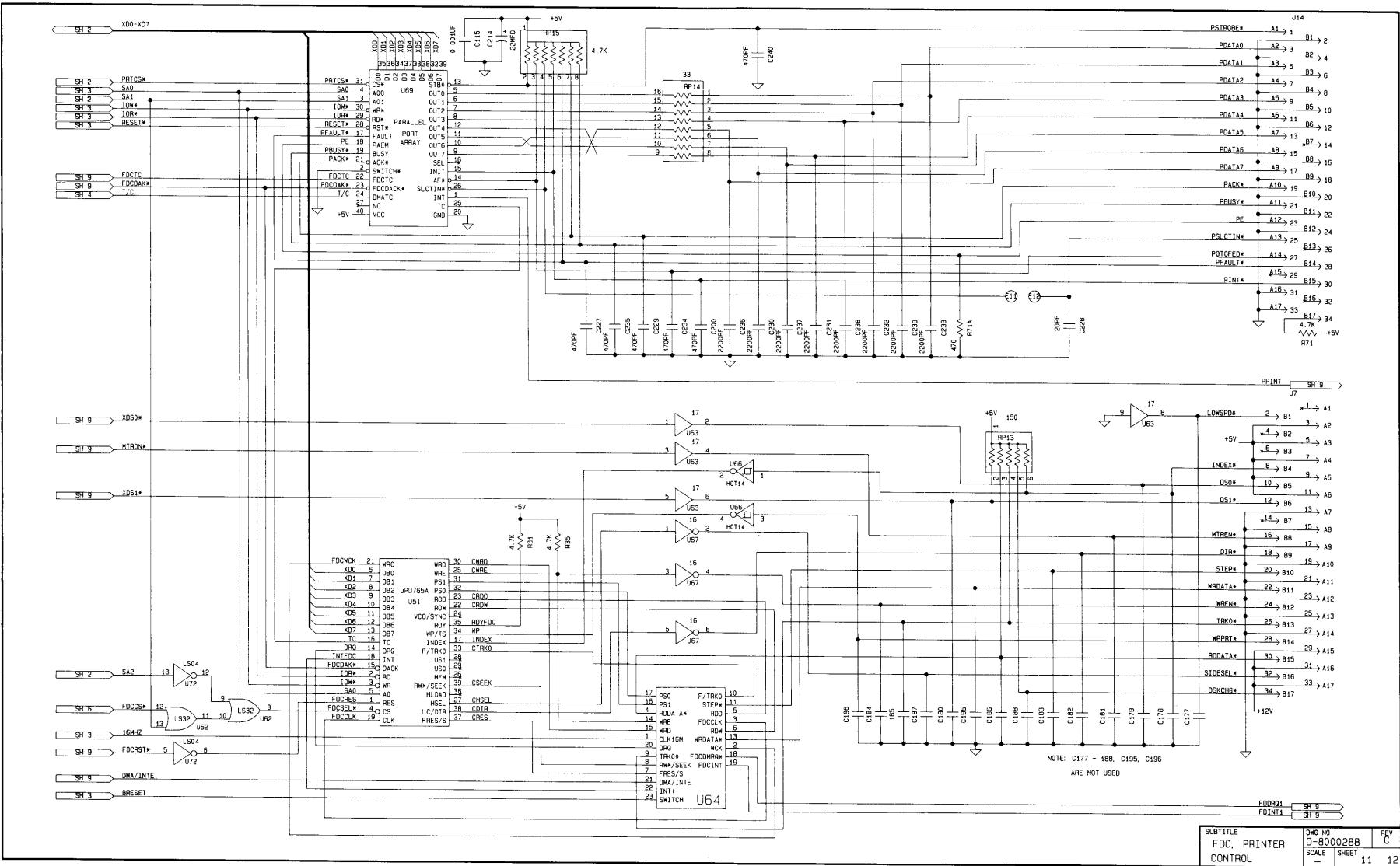


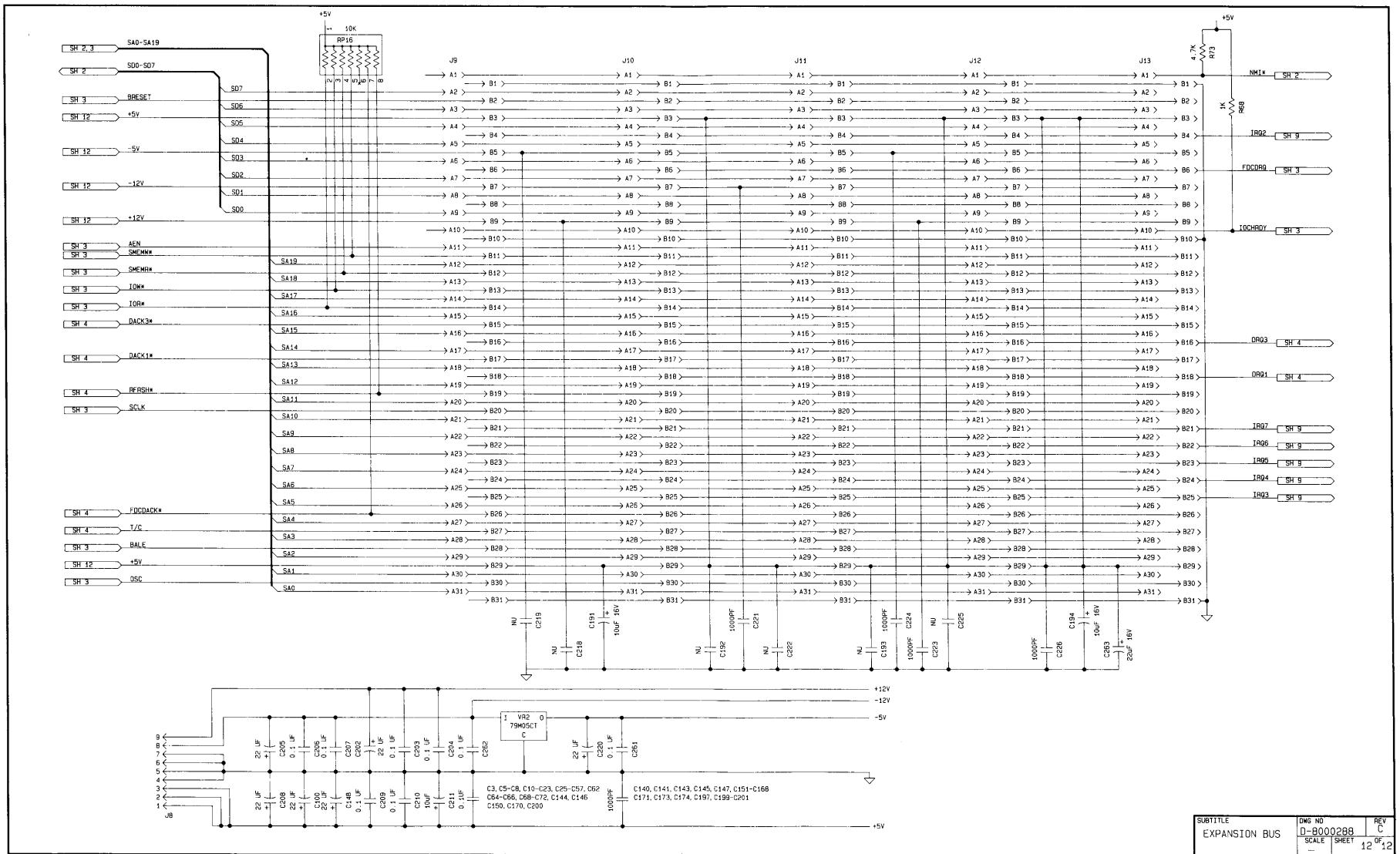






SUBTITLE	DWG NO	REV
SERIAL PORT	D- 80002BB	C
SCALE	HEET	OF
-	10	12





TANDY COMPUTER PRODUCTS

1000 TX Devices

TANDY COMPUTER PRODUCTS

1000 TX Devices
Contents

Device	Manufacturer
82A205	Chips and Technology
8237A-5	intel
8254-2	intel
8259A	intel
8272A	intel
80286-8	intel
80287-6	intel
8250AN	National Semiconductor
8496(Sound)	NCR
MC14529	Motorola
MC1488	Motorola
MC1489	Motorola
U19 (I/O Address Decode) DRAM/DMA Cntl.	Tandy
FDC Chip	Tandy
Printer Interface	Tandy
Keyboard I/F Chip	Universal
CPU Cntl. (Nt. Blue)	VTI

Note: All Intel Chip Specifications are reprinted by permission
of Intel Corporation.

Note: All Motorola Chip Specifications are:

Copyright of Motorola, Inc.
and are Used by Permission

CHIPS

**82A205
PC/AT COMPATIBLE CHIP**

- Fully IBM™ PC AT Compatible
 - Flexible architecture allows usage in any iAPX 286 design
 - 10 or 8 MHz with One Wait State or 6 MHz with Zero Wait State Capability
 - Complete System Board Memory Decode
 - Configurable RAM Selects
 - 16 Bit to 8 Bit Conversion Logic
 - Variable Wait State Selection
 - 24 mA sink and -3.3 mA source current for System Bus outputs
 - Single 5 Volt Supply

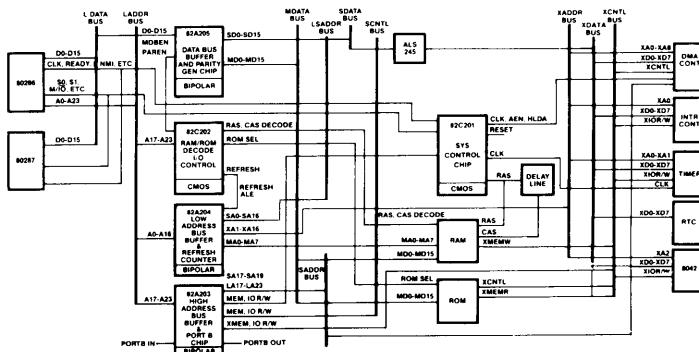
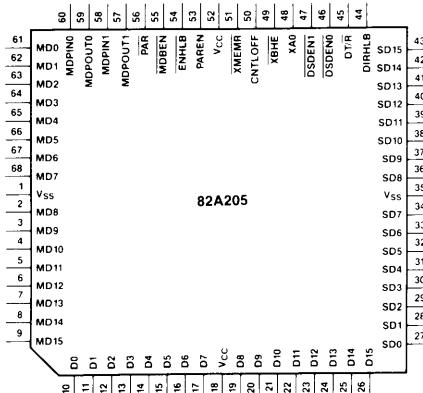


Figure 1. PC/AT Block Diagram



82A205 Pin Description

Pin No.	Pin Type	Symbol	Description
1	—	V _{SS}	Ground.
61-68 2-9	I/O I/O	MD0-7 MD8-15	Memory data bus for the on board memory.
10-17 19-26	I/O I/O	D0-D7 D8-D15	CPU data bus signals from/to the CPU.
18	—	V _{CC}	5 Volt Power Supply.
27-34 36-43	I/O I/O	SD0-SD7 SD8-15	System Data bus for the expansion bus. Its direction is determined by DT/R signal from the 82C201.
35	—	V _{SS}	Ground.
44	I	DIRHLB	DIRHLB is generated by the 82C201 and controls the direction of low to high byte conversion during data transfers to and from 8 bit peripherals.
45	I	DT/R	Data Transmit/Receive is generated by the 82C201. It determines direction of data to and from the memory. A HIGH on the pin indicates a write cycle and a LOW indicates a read cycle.
46, 47	I	DSDEN0 DSDEN1	Data Strobe Enable 0 and 1 are generated by the 82C201. These signals enable the data transceivers connected to the LOW and HIGH data bytes.
48	I	XA0	XA0 is address signal 0 for the peripheral bus. It is generated by the 82A203. It is used to condition the bus transceiver for the memory data bus.
49	I	XBHE	XBHE is the Bus High Enable signal generated by the 82A203. It is used to condition the bus transceiver for the memory data bus, and is active during a high byte transfer.
50	I	CNTLOFF	Control Off is generated by the 82C201 and is used to enable low byte data bus latch during byte access.
51	I	XMEMR	Memory Read is generated by the 82A203 and is used to enable the parity generation logic on the device, and to set the direction on the output transceiver for the memory data bus.
52	—	V _{CC}	5 V Power Supply.
53	I	PAREN	Parity Enable allows the parity check to be done on parity on the parity bit read from the memory
54	I	ENHLB	ENHLB enables the high to low byte conversion in conjunction with DIRHLB signal. It is generated by the 82C201.

82A205 Pin Description

(Continued)

Pin No.	Pin Type	Symbol	Description
55	I	MDBEN	Memory Data Bus Enable is generated by the 82C202. It enables the data bus transceivers connected to the memory devices.
56	O	PAR	Parity signal, when active, signifies a parity error on a memory read cycle.
59,57	I	MDPOUT0, MDPOUT1	Memory Data Parity Out 0 and 1 are the parity bits read from the memory banks 0 and 1. They are used to compute the parity during a ready cycle.
60, 58	O	MDPIN0, MDPIN1	Memory Data Parity In 0 and 1 are the parity bits written to the memory banks 0 and 1 during a memory write cycle.

CHIPS

Functional Description 82A205

Figure 9 illustrates a TTL equivalent of logic implemented by the 82A205. The chip provides the data bus buffers and drivers for D0-D15. The three data buses controlled are the CPU bus (D0-D15), the System bus (SD0-SD15), and the Memory Data bus (MD0-MD15). The direction and control for these drivers are provided by the DT/R, DSDEN0, DSDEN1, XBHE, and XA0 inputs, as shown in the figure. The low byte to high byte conversion logic is

also implemented on the chip. The conversion logic is controlled by the ENHLB and DIRHLB inputs. The chip also integrates the parity generation and check logic. The parity is computed on the memory data bus signals and output as MDPIN0 and MDPIN1. During a read cycle, the parity check is computed on the data read from the memory and the parity bits MDPOUT0 and MDPOUT1. On a parity error, the PAR output is activated.

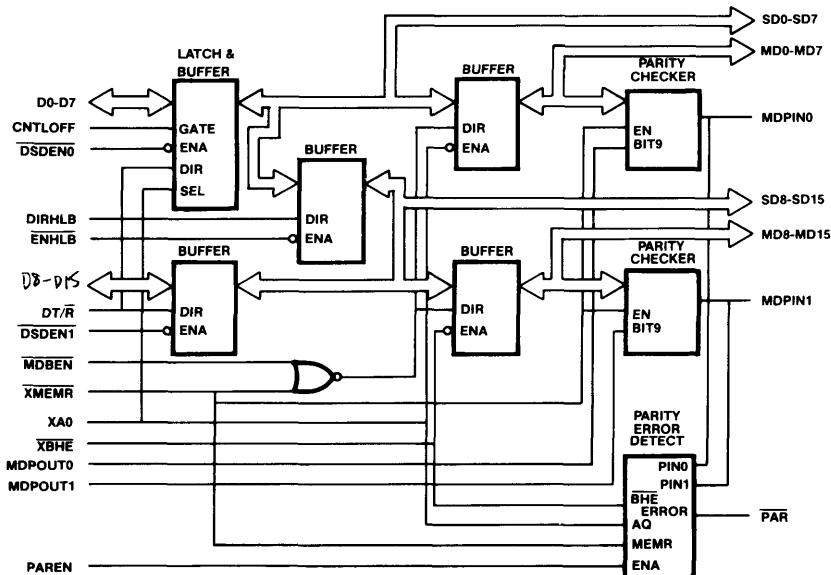


Figure 9: 82A205 Functional Block Diagram

CHIPS

82A205 Absolute Maximum Ratings

Parameter	Symbol	Min.	Max.	Units
Supply Voltage	V_{CC}	—	7.0	V
Input Voltage	V_I	-0.5	5.5	V
Output Voltage	V_O	-0.5	5.5	V
Operating Temperature	T_{op}	-25	85	C
Storage Temperature	T_{stg}	-40	125	C

NOTE: Permanent device damage may occur if Absolute Maximum Ratings are exceeded. Functional operation should be restricted to the conditions described under Operating Conditions.

82A205 Operating Conditions

Parameter	Symbol	Min.	Max.	Units
Supply Voltage	V_{CC}	4.75	5.25	V
Ambient Temperature	T_A	0	70	C

82A205 DC Characteristics

Parameter	Symbol	Min.	Max.	Units
Input Low Voltage	V_{IL}		0.8	V
Input High Voltage	V_{IH}	2.0		V
Output Low Voltage $I_{OL} = 10\text{mA}$ (Note 1)	V_{OL1}		0.5	V
Output Low Voltage $I_{OL} = 24\text{mA}$ (Note 2)	V_{OL2}		0.5	V
Output High Voltage $I_{OH} = -3.3\text{mA}$ (Note 3)	V_{OH}	2.4		V
Input Low Current $V_I = 0.5\text{V}$, $V_{CC} = 5.25\text{V}$	I_{IL}		-200	μA
Input High Current $V_I = 2.4\text{V}$, $V_{CC} = 5.25\text{V}$	I_{IH}		20	μA
Input High Current $V_I = 5.5\text{V}$, $V_{CC} = 5.25\text{V}$	I_I		200	μA
Output Short Circuit Current $V_O = 0\text{V}$	I_{OS}	-15	-100	mA
Input Clamp Voltage $I_I = -18\text{mA}$, $V_{CC} = 4.75\text{V}$	V_{IC}		-1.5	V
Power Supply Current	I_{CC}	180	300	mA
Output Hi-Z Leak Current 3-State Output Pins	I_{OZ1}	-100	100	μA
Output Hi-Z Leak Current Bidirectional Pins	I_{OZ2}	-300	120	μA

NOTES 1. All non-system and bus outputs only.

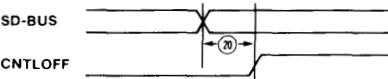
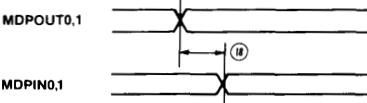
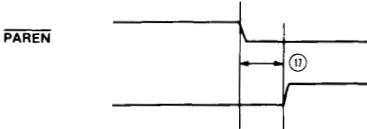
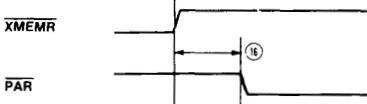
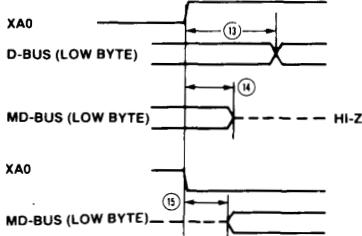
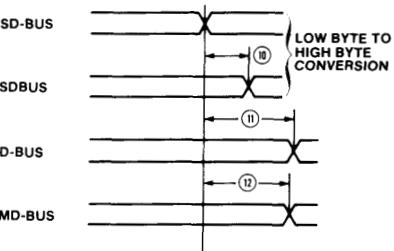
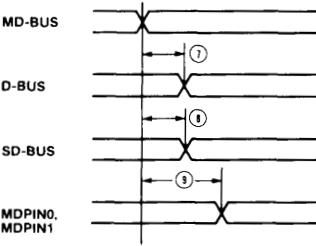
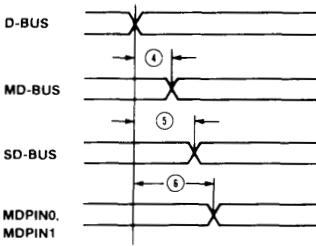
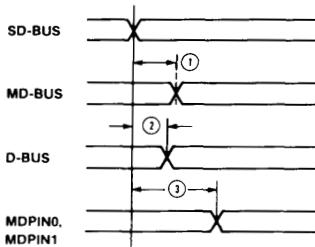
2. All system bus outputs, SD0-15, are specified at $I_{OL} = 24\text{mA}$ @ $V_{OL} = 0.5\text{V}$.

3. All outputs and bidirectional pins.

82A205 AC Characteristics(TA + 0° C to 70° C, V_{CC} + 5V ± 5%)

Sym	Description	Min.	Max.	Units
t1	System Data Bus to Memory Bus Delay	8	32	ns
t2	System Data Bus to CPU Data Bus Delay	8	32	ns
t3	System Data Bus to Parity Bits MDPIN0, 1 Output	12	42	ns
t4	CPU Data Bus to Memory Data Bus Delay	8	30	ns
t5	CPU Data Bus to System Data Bus Delay	5	25	ns
t6	CPU Data Bus to Parity Bits MDPIN0, MDPIN1 Output	12	42	ns
t7	Memory Data Bus to CPU Data Bus Delay	8	30	ns
t8	Memory Data Bus to System Data Bus Delay	6	27	ns
t9	Memory Data Bus to Parity Bits MDPIN0, 1 Output	10	38	ns
t10	System Data Bus Low Byte to High Byte Conversion	8	32	ns
t11	System Bus to CPU Data Bus Hi-Lo Byte Conversion	10	33	ns
t12	System Bus to Mem Data Bus Hi-Lo Byte Conversion	10	35	ns
t13	XA0 to CPU Data Bus Low Byte Delay	8	30	ns
t14	XA0 to Memory Data Bus HI-Z	6	26	ns
t15	XA0 to Memory Data Bus Address Valid	8	30	ns
t16	XMEMR Going High to Parity Delay	7	28	ns
t17	PAREN to Parity Delay	4	20	ns
t18	Parity Input Bits to Parity Output Bits Delay	12	40	ns
t19	ENHLB to SD8-SD15 delay	—	30	ns
t20	SD BUS to CNTLOFF set-up time	10		ns

CHIPS



Load Circuit Measurement Conditions

Parameter	Output Type	Symbol	C_L (pF)	R_1 (Ω)	R_L (Ω)	SW_1	SW_2
Propagation Delay Time	Totem pole 3-state	t_{PLH} t_{PHL}	50	--	1.0K	OFF	ON
Time	Bidirectional						
Propagation Delay Time	Open drain or Open Collector	t_{PLH} t_{PHL}	50	0.5K	--	ON	OFF
Disable Time	3-state Bidirectional	t_{PZL} t_{PHZ}	5	0.5K	1.0K	ON	ON
Enable Time	3-state Bidirectional	t_{PZH} t_{PLZ}	50	0.5K	1.0K	ON	ON

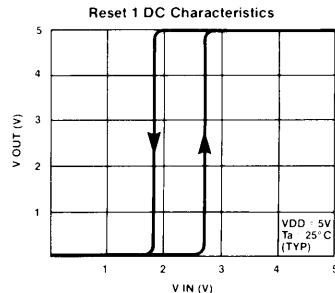
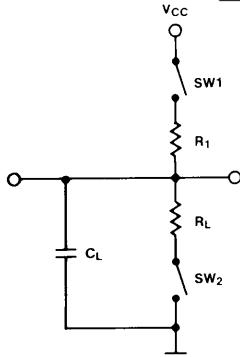
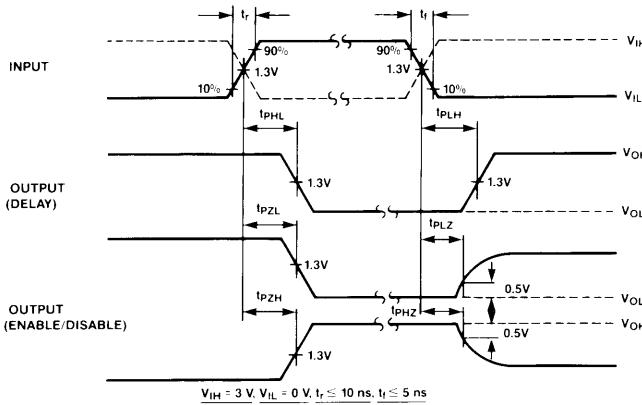
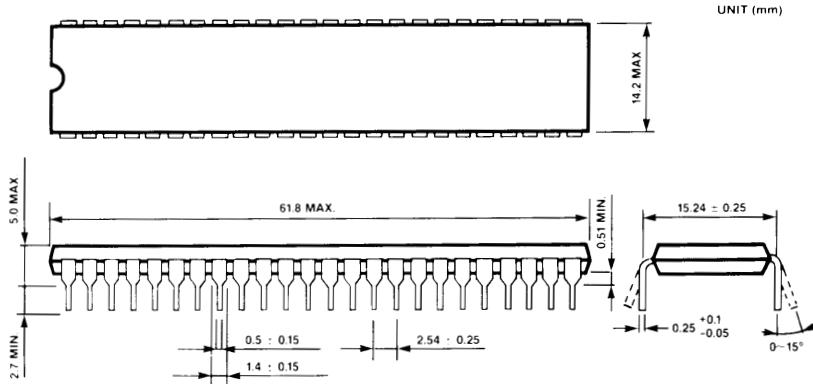


Figure 10. Load Circuit and AC Characteristics Measurement Waveform

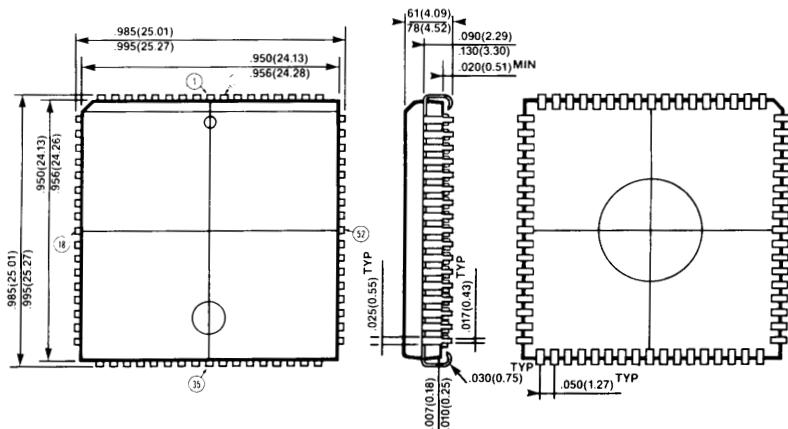
CHIPS

48-PIN PLASTIC DUAL-IN-LINE PACKAGE



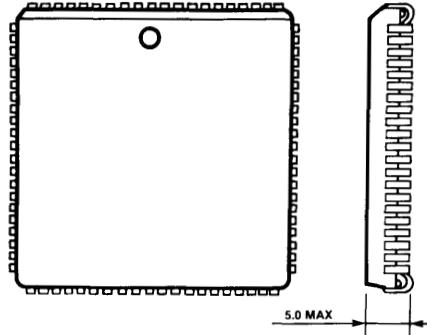
68-LEAD PLASTIC CHIP CARRIER

DIMENSIONS IN INCHES (MILLIMETERS) S = 3.6/1

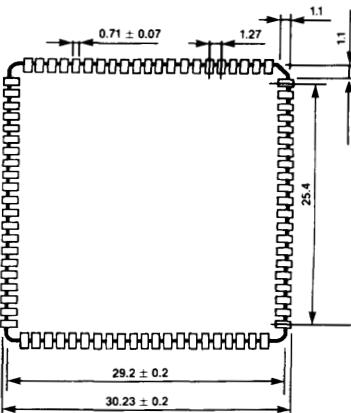


CHIPS

84-PIN PLASTIC LEADED CHIP CARRIER



UNIT (mm)



Ordering Information

Order Number	Package Type Note 1	Remarks
P82C201, P82C201-10	PLCC-84	C (Note 2)
P82C202	PDIP-48	C
P82A203	PLCC-68	C
P82A204	PLCC-68	C
P82A205	PLCC-68	C
CS8220	—	Standard CHIPSet (Note 3)
CS8220-10	—	10MHz CHIPSet (Note 4)

NOTES

1. PLCC = Plastic Leaded Chip Carrier 84 Pins
PDIP = Plastic Dual-In-Line Package 48 Pins
2. C = Commercial Range, 0 to 70°C, $V_{DD} = 4.75$ to 5.25 V
3. CS8220 consists of P82C201, P82C202, P82A203, P82A204, P82A205
4. CS8220-10 consists of P82C201-10, P82C202, P82A203, P82A204, P82A205



8237A/8237A-4/8237A-5 HIGH PERFORMANCE PROGRAMMABLE DMA CONTROLLER

- Enable/Disable Control of Individual DMA Requests
- Four Independent DMA Channels
- Independent Autoinitialization of All Channels
- Memory-to-Memory Transfers
- Memory Block Initialization
- Address Increment or Decrement
- High Performance: Transfers up to 1.6M Bytes/Second with 5 MHz 8237A-5
- Directly Expandable to Any Number of Channels
- End of Process Input for Terminating Transfers
- Software DMA Requests
- Independent Polarity Control for DREQ and DACK Signals
- Available In EXPRESS — Standard Temperature Range
- Available In 40-Lead Cerdip and Plastic Packages

(See Packaging Spec, Order #231369)

The 8237A Multimode Direct Memory Access (DMA) Controller is a peripheral interface circuit for microprocessor systems. It is designed to improve system performance by allowing external devices to directly transfer information from the system memory. Memory-to-memory transfer capability is also provided. The 8237A offers a wide variety of programmable control features to enhance data throughput and system optimization and to allow dynamic reconfiguration under program control.

The 8237A is designed to be used in conjunction with an external 8-bit address latch. It contains four independent channels and may be expanded to any number of channels by cascading additional controller chips. The three basic transfer modes allow programmability of the types of DMA service by the user. Each channel can be individually programmed to Autoinitialize to its original condition following an End of Process (EOP). Each channel has a full 64K address and word count capability.

The 8273A-4 and 8237A-5 are 4 MHz and 5 MHz versions of the standard 3 MHz 8237A respectively.

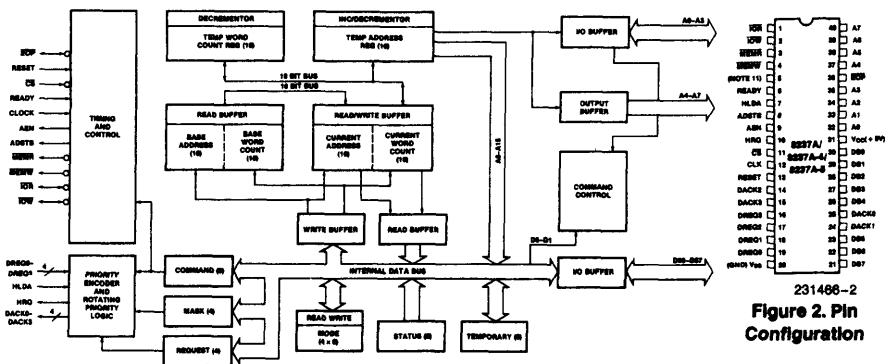


Figure 1. Block Diagram

231466-1

Figure 2. Pin Configuration

November 1986

Order Number: 231466-002

Table 1. Pin Description

Symbol	Type	Name and Function
V _{CC}		POWER: + 5V supply.
V _{SS}		GROUND: Ground.
CLK	I	CLOCK INPUT: Clock Input controls the internal operations of the 8237A and its rate of data transfers. The input may be driven at up to 3 MHz for the standard 8237A and up to 5 MHz for the 8237A-5.
CS	I	CHIP SELECT: Chip Select is an active low input used to select the 8237A as an I/O device during the Idle cycle. This allows CPU communication on the data bus.
RESET	I	RESET: Reset is an active high input which clears the Command, Status, Request and Temporary registers. It also clears the first/last flip/flop and sets the Mask register. Following a Reset the device is in the Idle cycle.
READY	I	READY: Ready is an input used to extend the memory read and write pulses from the 8237A to accommodate slow memories or I/O peripheral devices. Ready must not make transitions during its specified setup/hold time.
HLDA	I	HOLD ACKNOWLEDGE: The active high Hold Acknowledge from the CPU indicates that it has relinquished control of the system busses.
DREQ0-DREQ3	I	DMA REQUEST: The DMA Request lines are individual asynchronous channel request inputs used by peripheral circuits to obtain DMA service. In fixed Priority, DREQ0 has the highest priority and DREQ3 has the lowest priority. A request is generated by activating the DREQ line of a channel. DACK will acknowledge the recognition of DREQ signal. Polarity of DREQ is programmable. Reset initializes these lines to active high. DREQ must be maintained until the corresponding DACK goes active.
DB0-DB7	I/O	DATA BUS: The Data Bus lines are bidirectional three-state signals connected to the system data bus. The outputs are enabled in the Program condition during the I/O Read to output the contents of an Address register, a Status register, the Temporary register or a Word Count register to the CPU. The outputs are disabled and the inputs are read during an I/O Write cycle when the CPU is programming the 8237A control registers. During DMA cycles the most significant 8 bits of the address are output onto the data bus to be strobed into an external latch by ADSTB. In memory-to-memory operations, data from the memory comes into the 8237A on the data bus during the read-from-memory transfer. In the write-to-memory transfer, the data bus outputs place the data into the new memory location.
IOR	I/O	I/O READ: I/O Read is a bidirectional active low three-state line. In the Idle cycle, it is an input control signal used by the CPU to read the control registers. In the Active cycle, it is an output control signal used by the 8237A to access data from a peripheral during a DMA Write transfer.
IOW	I/O	I/O WRITE: I/O Write is a bidirectional active low three-state line. In the Idle cycle, it is an input control signal used by the CPU to load information into the 8237A. In the Active cycle, it is an output control signal used by the 8237A to load data to the peripheral during a DMA Read transfer.

Table 1. Pin Description (Continued)

Symbol	Type	Name and Function
EOP	I/O	END OF PROCESS: End of Process is an active low bidirectional signal. Information concerning the completion of DMA services is available at the bidirectional EOP pin. The 8237A allows an external signal to terminate an active DMA service. This is accomplished by pulling the EOP input low with an external EOP signal. The 8237A also generates a pulse when the terminal count (TC) for any channel is reached. This generates an \overline{EOP} signal which is output through the EOP line. The reception of EOP, either internal or external, will cause the 8237A to terminate the service, reset the request, and, if Autoinitialize is enabled, to write the base registers to the current registers of that channel. The mask bit and TC bit in the status word will be set for the currently active channel by EOP unless the channel is programmed for Autoinitialize. In that case, the mask bit remains unchanged. During memory-to-memory transfers, EOP will be output when the TC for channel 1 occurs. EOP should be tied high with a pull-up resistor if it is not used to prevent erroneous end of process inputs.
A0-A3	I/O	ADDRESS: The four least significant address lines are bidirectional three-state signals. In the Idle cycle they are inputs and are used by the CPU to address the register to be loaded or read. In the Active cycle they are outputs and provide the lower 4 bits of the output address.
A4-A7	O	ADDRESS: The four most significant address lines are three-state outputs and provide 4 bits of address. These lines are enabled only during the DMA service.
HRQ	O	HOLD REQUEST: This is the Hold Request to the CPU and is used to request control of the system bus. If the corresponding mask bit is clear, the presence of any valid DREQ causes 8237A to issue the HRQ.
DACK0-DACK3	O	DMA ACKNOWLEDGE: DMA Acknowledge is used to notify the individual peripherals when one has been granted a DMA cycle. The sense of these lines is programmable. Reset initializes them to active low.
AEN	O	ADDRESS ENABLE: Address Enable enables the 8-bit latch containing the upper 8 address bits onto the system address bus. AEN can also be used to disable other system bus drivers during DMA transfers. AEN is active HIGH.
ADSTB	O	ADDRESS STROBE: The active high, Address Strobe is used to strobe the upper address byte into an external latch.
MEMR	O	MEMORY READ: The Memory Read signal is an active low three-state output used to access data from the selected memory location during a DMA Read or a memory-to-memory transfer.
MEMW	O	MEMORY WRITE: The Memory Write is an active low three-state output used to write data to the selected memory location during a DMA Write or a memory-to-memory transfer.

FUNCTIONAL DESCRIPTION

The 8237A block diagram includes the major logic blocks and all of the internal registers. The data interconnection paths are also shown. Not shown are the various control signals between the blocks. The 8237A contains 344 bits of internal memory in the form of registers. Figure 3 lists these registers by name and shows the size of each. A detailed description of the registers and their functions can be found under Register Description.

Name	Size	Number
Base Address Registers	16 bits	4
Base Word Count Registers	16 bits	4
Current Address Registers	16 bits	4
Current Word Count Registers	16 bits	4
Temporary Address Register	16 bits	1
Temporary Word Count Register	16 bits	1
Status Register	8 bits	1
Command Register	8 bits	1
Temporary Register	8 bits	1
Mode Registers	6 bits	4
Mask Register	4 bits	1
Request Register	4 bits	1

Figure 3. 8237A Internal Registers

The 8237A contains three basic blocks of control logic. The Timing Control block generates internal timing and external control signals for the 8237A. The Program Command Control block decodes the various commands given to the 8237A by the microprocessor prior to servicing a DMA Request. It also decodes the Mode Control word used to select the type of DMA during the servicing. The Priority Encoder block resolves priority contention between DMA channels requesting service simultaneously.

The Timing Control block derives internal timing from the clock input. In 8237A systems, this input will usually be the ϕ_2 TTL clock from an 8224 or CLK from an 8085AH or 8284A. 33% duty cycle clock generators, however, may not meet the clock high time requirement of the 8237A-5 of the same frequency. For example, 82C84A-5 CLK output violates the clock high time requirement of 8237A-5. In this case 82C84A CLK can simply be inverted to meet 8237A-5 clock high and low time requirements. For 8085AH-2 systems above 3.9 MHz, the 8085 CLK(OUT) does not satisfy 8237A-5 clock LOW and HIGH time requirements. In this case, an external clock should be used to drive the 8237A-5.

DMA OPERATION

The 8237A is designed to operate in two major cycles. These are called Idle and Active cycles. Each device cycle is made up of a number of states. The 8237A can assume seven separate states, each composed of one full clock period. State I (SI) is the Inactive state. It is entered when the 8237A has no

valid DMA requests pending. While in SI, the DMA controller is inactive but may be in the Program Condition, being programmed by the processor. State S0 (S0) is the first state of a DMA service. The 8237A has requested a hold but the processor has not yet returned an acknowledge. The 8237A may still be programmed until it receives HLDA from the CPU. An acknowledge from the CPU will signal that DMA transfers may begin. S1, S2, S3 and S4 are the working states of the DMA service. If more time is needed to complete a transfer than is available with normal timing, wait states (SW) can be inserted between S2 or S3 and S4 by the use of the Ready line on the 8237A. Note that the data is transferred directly from the I/O device to memory (or vice versa) with IOR and MEMW (or MEMR and IOW) being active at the same time. The data is not read into or driven out of the 8237A in I/O-to-memory or memory-to-I/O DMA transfers.

Memory-to-memory transfers require a read-from and a write-to-memory to complete each transfer. The states, which resemble the normal working states, use two digit numbers for identification. Eight states are required for a single transfer. The first four states (S11, S12, S13, S14) are used for the read-from-memory half and the last four states (S21, S22, S23, S24) for the write-to-memory half of the transfer.

IDLE CYCLE

When no channel is requesting service, the 8237A will enter the Idle cycle and perform "SI" states. In this cycle the 8237A will sample the DREQ lines every clock cycle to determine if any channel is requesting a DMA service. The device will also sample CS, looking for an attempt by the microprocessor to write or read the internal registers of the 8237A. When CS is low and HLDA is low, the 8237A enters the Program Condition. The CPU can now establish, change or inspect the internal definition of the part by reading from or writing to the internal registers. Address lines A0-A3 are inputs to the device and select which registers will be read or written. The IOR and IOW lines are used to select and time reads or writes. Due to the number and size of the internal registers, an internal flip-flop is used to generate an additional bit of address. This bit is used to determine the upper or lower byte of the 16-bit Address and Word Count registers. The flip-flop is reset by Master Clear or Reset. A separate software command can also reset this flip-flop.

Special software commands can be executed by the 8237A in the Program Condition. These commands are decoded as sets of addresses with the CS and IOW. The commands do not make use of the data bus. Instructions include Clear First/Last Flip-Flop and Master Clear.

ACTIVE CYCLE

When the 8237A is in the Idle cycle and a non-masked channel requests a DMA service, the device will output an HRQ to the microprocessor and enter the Active cycle. It is in this cycle that the DMA service will take place, in one of four modes:

Single Transfer Mode—In Single Transfer mode the device is programmed to make one transfer only. The word count will be decremented and the address decremented or incremented following each transfer. When the word count "rolls over" from zero to FFFFH, a Terminal Count (TC) will cause an Auto-initialize if the channel has been programmed to do so.

DREQ must be held active until DACK becomes active in order to be recognized. If DREQ is held active throughout the single transfer, HRQ will go inactive and release the bus to the system. It will again go active and, upon receipt of a new HLDA, another single transfer will be performed. In 8080A, 8085AH, 8088, or 8086 system, this will ensure one full machine cycle execution between DMA transfers. Details of timing between the 8237A and other bus control protocols will depend upon the characteristics of the microprocessor involved.

Block Transfer Mode—In Block Transfer mode the device is activated by DREQ to continue making transfers during the service until a TC, caused by word count going to FFFFH, or an external End of

Process (EOP) is encountered. DREQ need only be held active until DACK becomes active. Again, an Autoinitialization will occur at the end of the service if the channel has been programmed for it.

Demand Transfer Mode—In Demand Transfer mode the device is programmed to continue making transfers until a TC or external EOP is encountered or until DREQ goes inactive. Thus transfers may continue until the I/O device has exhausted its data capacity. After the I/O device has had a chance to catch up, the DMA service is re-established by means of a DREQ. During the time between services when the microprocessor is allowed to operate, the intermediate values of address and word count are stored in the 8237A Current Address and Current Word Count registers. Only an EOP can cause an Autoinitialize at the end of the service. EOP is generated either by TC or by an external signal. DREQ has to be low before S4 to prevent another transfer.

Cascade Mode—This mode is used to cascade more than one 8237A together for simple system expansion. The HRQ and HLDA signals from the additional 8237A are connected to the DREQ and DACK signals of a channel of the initial 8237A. This allows the DMA requests of the additional device to propagate through the priority network circuitry of the preceding device. The priority chain is preserved and the new device must wait for its turn to acknowledge requests. Since the cascade channel of the initial 8237A is used only for prioritizing the additional device, it does not output any address or control

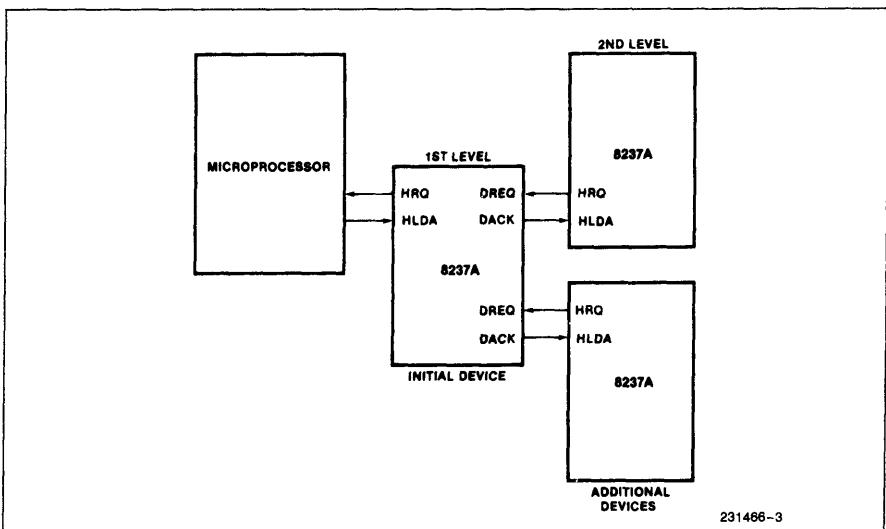


Figure 4. Cascaded 8237As

signals of its own. These could conflict with the outputs of the active channel in the added device. The 8237A will respond to DREQ and DACK but all other outputs except HRQ will be disabled. The ready input is ignored.

Figure 4 shows two additional devices cascaded into an initial device using two of the previous channels. This forms a two level DMA system. More 8237As could be added at the second level by using the remaining channels of the first level. Additional devices can also be added by cascading into the channels of the second level device, forming a third level.

TRANSFER TYPES

Each of the three active transfer modes can perform three different types of transfers. These are Read, Write and Verify. Write transfers move data from an I/O device to the memory by activating MEMW and IOR. Read transfers move data from memory to an I/O device by activating MEMR and IOW. Verify transfers are pseudo transfers. The 8237A operates as in Read or Write transfers generating addresses, and responding to EOP, etc. However, the memory and I/O control lines all remain inactive. The ready input is ignored in verify mode.

Memory-to-Memory—To perform block moves of data from one memory address space to another with a minimum of program effort and time, the 8237A includes a memory-to-memory transfer feature. Programming a bit in the Command register selects channels 0 and 1 to operate as memory-to-memory transfer channels. The transfer is initiated by setting the software DREQ for channel 0. The 8237A requests a DMA service in the normal manner. After HLDA is true, the device, using four state transfers in Block Transfer mode, reads data from the memory. The channel 0 Current Address register is the source for the address used and is decremented or incremented in the normal manner. The data byte read from the memory is stored in the 8237A internal Temporary register. Channel 1 then performs a four-state transfer of the data from the Temporary register to memory using the address in its Current Address register and incrementing or decrementing it in the normal manner. The channel 1 current Word Count is decremented. When the word count of channel 1 goes to FFFFH, a TC is generated causing an EOP output terminating the service.

Channel 0 may be programmed to retain the same address for all transfers. This allows a single word to be written to a block of memory.

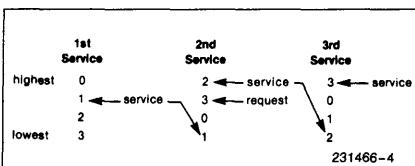
The 8237A will respond to external EOP signals during memory-to-memory transfers. Data comparators in block search schemes may use this input to terminate the service when a match is found. The timing of memory-to-memory transfers is found in Figure 12. Memory-to-memory operations can be detected as an active AEN with no DACK outputs.

Autoinitialize—By programming a bit in the Mode register, a channel may be set up as an Autoinitialize channel. During Autoinitialize initialization, the original values of the Current Address and Current Word Count registers are automatically restored from the Base Address and Base Word count registers of that channel following EOP. The base registers are loaded simultaneously with the current registers by the microprocessor and remain unchanged throughout the DMA service. The mask bit is not altered when the channel is in Autoinitialize. Following Autoinitialize the channel is ready to perform another DMA service, without CPU intervention, as soon as a valid DREQ is detected. In order to Autoinitialize both channels in a memory-to-memory transfer, both word counts should be programmed identically. If interrupted externally, EOP pulses should be applied in both bus cycles.

Priority—The 8237A has two types of priority encoding available as software selectable options. The first is Fixed Priority which fixes the channels in priority order based upon the descending value of their number. The channel with the lowest priority is 3 followed by 2, 1 and the highest priority channel, 0. After the recognition of any one channel for service, the other channels are prevented from interfering with that service until it is completed.

After completion of a service, HRQ will go inactive and the 8237A will wait for HLDA to go low before activating HRQ to service another channel.

The second scheme is Rotating Priority. The last channel to get service becomes the lowest priority channel with the others rotating accordingly.



With Rotating Priority in a single chip DMA system, any device requesting service is guaranteed to be recognized after no more than three higher priority services have occurred. This prevents any one channel from monopolizing the system.

Compressed Timing—In order to achieve even greater throughput where system characteristics permit, the 8237A can compress the transfer time to two clock cycles. From Figure 11 it can be seen that state S3 is used to extend the access time of the read pulse. By removing state S3, the read pulse width is made equal to the write pulse width and a transfer consists only of state S2 to change the address and state S4 to perform the read/write. S1 states will still occur when A8-A15 need updating (see Address Generation). Timing for compressed transfers is found in Figure 14.

Address Generation—In order to reduce pin count, the 8237A multiplexes the eight higher order address bits on the data lines. State S1 is used to output the higher order address bits to an external latch from which they may be placed on the address bus. The falling edge of Address Strobe (ADSTB) is used to load these bits from the data lines to the latch. Address Enable (AEN) is used to enable the bits onto the address bus through a three-state enable. The lower order address bits are output by the 8237A directly. Lines A0-A7 should be connected to the address bus. Figure 11 shows the time relationships between CLK, AEN, ADSTB, DB0-DB7 and A0-A7.

During Block and Demand Transfer mode services, which include multiple transfers, the addresses generated will be sequential. For many transfers the data held in the external address latch will remain the same. This data need only change when a carry or borrow from A7 to A8 takes place in the normal sequence of addresses. To save time and speed transfers, the 8237A executes S1 states only when updating of A8-A15 in the latch is necessary. This means for long services, S1 states and Address Strobes may occur only once every 256 transfers, a savings of 255 clock cycles for each 256 transfers.

REGISTER DESCRIPTION

Current Address Register—Each channel has a 16-bit Current Address register. This register holds the value of the address used during DMA transfers. The address is automatically incremented or decremented after each transfer and the intermediate values of the address are stored in the Current Address register during the transfer. This register is written or read by the microprocessor in successive 8-bit bytes. It may also be reinitialized by an Autoinitialize back to its original value. Autoinitialize takes place only after an EOP.

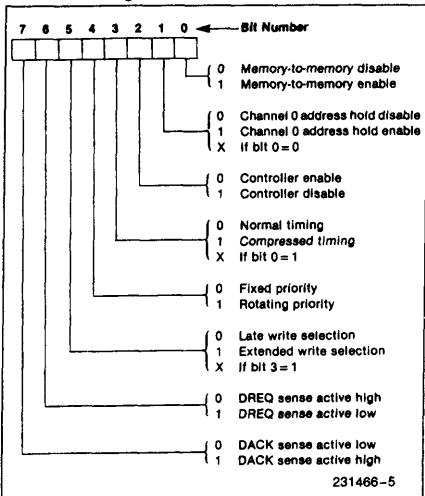
Current Word Register—Each channel has a 16-bit Current Word Count register. This register determines the number of transfers to be performed. The actual number of transfers will be one more than the number programmed in the Current Word Count register (i.e., programming a count of 100 will result in 101 transfers). The word count is decremented after each transfer. The intermediate value of the word count is stored in the register during the transfer. When the value in the register goes from zero to FFFFH, a TC will be generated. This register is loaded or read in successive 8-bit bytes by the microprocessor in the Program Condition. Following the end of a DMA service it may also be reinitialized by an Autoinitialize back to its original value. Autoinitialize can occur only when an EOP occurs. If it is not Autoinitialized, this register will have a count of FFFFH after TC.

Base Address and Base Word Count Registers—Each channel has a pair of Base Address and Base Word Count registers. These 16-bit registers store the original value of their associated current registers. During Autoinitialize these values are used to restore the current registers to their original values. The base registers are written simultaneously with their corresponding current register in 8-bit bytes in the Program Condition by the microprocessor. These registers cannot be read by the microprocessor.

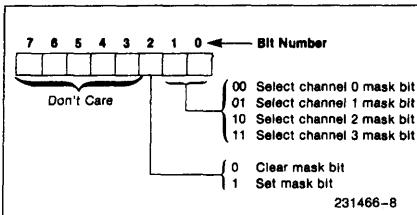
Command Register—This 8-bit register controls the operation of the 8237A. It is programmed by the microprocessor in the Program Condition and is cleared by Reset or a Master Clear instruction. The following table lists the function of the command bits. See Figure 6 for address coding.

Mode Register—Each channel has a 6-bit Mode register associated with it. When the register is being written to by the microprocessor in the Program Condition, bits 0 and 1 determine which channel Mode register is to be written.

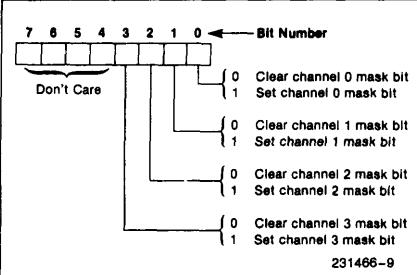
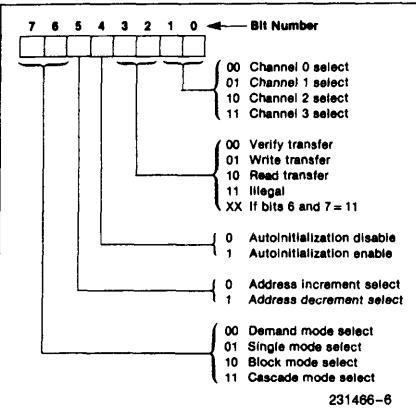
Request Register—The 8237A can respond to requests for DMA service which are initiated by software as well as by a DREQ. Each channel has a request bit associated with it in the 4-bit Request register. These are non-maskable and subject to prioritization by the Priority Encoder network. Each register bit is set or reset separately under software control or is cleared upon generation of a TC or external EOP. The entire register is cleared by a Reset. To set or reset a bit, the software loads the proper form of the data word. See Figure 5 for register address coding. In order to make a software request, the channel must be in Block Mode.

Command Register

Mask Register—Each channel has associated with it a mask bit which can be set to disable the incoming DREQ. Each mask bit is set when its associated channel produces an EOP if the channel is not programmed for Autoinitiate. Each bit of the 4-bit Mask register may also be set or cleared separately under software control. The entire register is also set by a Reset. This disables all DMA requests until a clear Mask register instruction allows them to occur. The instruction to separately set or clear the mask bits is similar in form to that used with the Request register. See Figure 5 for instruction addressing.

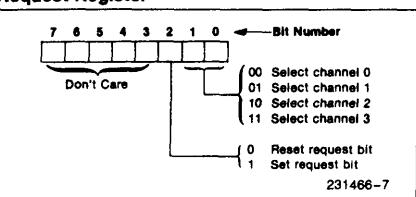


All four bits of the Mask register may also be written with a single command.

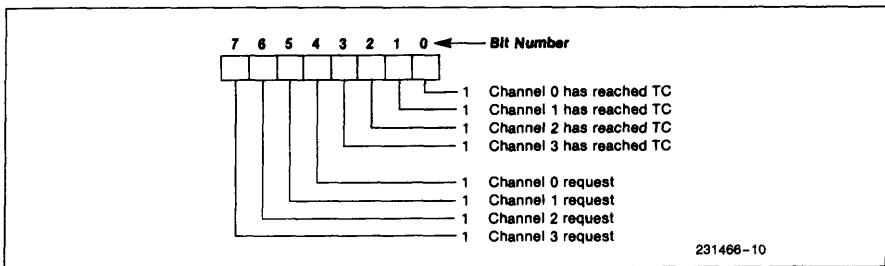
Mode Register

Register	Operation	Signals							
		CS	iOR	iOW	A3	A2	A1	A0	
Command	Write	0	1	0	1	0	0	0	
Mode	Write	0	1	0	1	0	1	1	
Request	Write	0	1	0	1	0	0	1	
Mask	Set/Reset	0	1	0	1	0	1	0	
Mask	Write	0	1	0	1	1	1	1	
Temporary	Read	0	0	1	1	1	0	1	
Status	Read	0	0	1	1	0	0	0	

Figure 5. Definition of Register Codes

Request Register

Status Register—The Status register is available to be read out of the 8237A by the microprocessor. It contains information about the status of the devices at this point. This information includes which channels have reached a terminal count and which chan-



231466-10

nels have pending DMA requests. Bits 0–3 are set every time a TC is reached by that channel or an external EOP is applied. These bits are cleared upon Reset and on each Status Read. Bits 4–7 are set whenever their corresponding channel is requesting service.

Temporary Register—The Temporary register is used to hold data during memory-to-memory transfers. Following the completion of the transfers, the last word moved can be read by the microprocessor in the Program Condition. The Temporary register always contains the last byte transferred in the previous memory-to-memory operation, unless cleared by a Reset.

Software Commands—These are additional special software commands which can be executed in the Program Condition. They do not depend on any specific bit pattern on the data bus. The three software commands are:

Clear First/Last Flip-Flop: This command must be executed prior to writing or reading new address or word count information to the 8237A. This initializes the flip-flop to a known state so that subsequent accesses to register contents by the microprocessor will address upper and lower bytes in the correct sequence.

Master Clear: This software instruction has the same effect as the hardware Reset. The Command, Status, Request, Temporary, and Internal First/Last Flip-Flop registers are cleared and the Mask register is set. The 8237A will enter the Idle cycle.

Clear Mask Register: This command clears the mask bits of all four channels, enabling them to accept DMA requests.

Figure 6 lists the address codes for the software commands.

Signals						Operation
A3	A2	A1	A0	IOR	IOW	
1	0	0	0	0	1	Read Status Register
1	0	0	0	1	0	Write Command Register
1	0	0	1	0	1	Illegal
1	0	0	1	1	0	Write Request Register
1	0	1	0	0	1	Illegal
1	0	1	0	1	0	Write Single Mask Register Bit
1	0	1	1	0	1	Illegal
1	0	1	1	1	0	Write Mode Register
1	1	0	0	0	1	Illegal
1	1	0	0	1	0	Clear Byte Pointer Flip/Flop
1	1	0	1	0	1	Read Temporary Register
1	1	0	1	1	0	Master Clear
1	1	1	0	0	1	Illegal
1	1	1	0	1	0	Clear Mask Register
1	1	1	1	0	1	Illegal
1	1	1	1	1	0	Write All Mask Register Bits

Figure 6. Software Command Codes

Channel	Register	Operation	Signals						Internal Flip-Flop	Data Bus DB0-DB7	
			CS	IOR	IOW	A3	A2	A1	A0		
0	Base and Current Address	Write	0	1	0	0	0	0	0	0	A0-A7 A8-A15
		Read	0	1	0	0	0	0	0	1	A0-A7 A8-A15
	Base and Current Word Count	Write	0	0	1	0	0	0	0	0	W0-W7 W8-W15
		Read	0	1	0	0	0	0	1	1	W0-W7 W8-W15
1	Base and Current Address	Write	0	1	0	0	0	1	0	0	A0-A7 A8-A15
		Read	0	1	0	0	0	1	0	1	A0-A7 A8-A15
	Base and Current Word Count	Write	0	0	1	0	0	1	0	0	W0-W7 W8-W15
		Read	0	1	0	0	0	1	1	1	W0-W7 W8-W15
2	Base and Current Address	Write	0	1	0	0	1	0	0	0	A0-A7 A8-A15
		Read	0	1	0	0	1	0	0	1	A0-A7 A8-A15
	Base and Current Word Count	Write	0	0	1	0	1	0	0	0	W0-W7 W8-W15
		Read	0	1	0	0	1	0	1	1	W0-W7 W8-W15
3	Base and Current Address	Write	0	1	0	0	1	1	0	0	A0-A7 A8-A15
		Read	0	1	0	0	1	1	0	1	A0-A7 A8-A15
	Base and Current Word Count	Write	0	0	1	0	1	1	0	0	W0-W7 W8-W15
		Read	0	1	0	0	1	1	1	1	W0-W7 W8-W15

Figure 7. Word Count and Address Register Command Codes

PROGRAMMING

The 8237A will accept programming from the host processor any time that HLDA is inactive; this is true even if HRQ is active. The responsibility of the host is to assure that programming and HLDA are mutually exclusive. Note that a problem can occur if a DMA request occurs, on an unmasked channel while the 8237A is being programmed. For instance, the CPU may be starting to reprogram the two byte Address register of channel 1 when channel 1 receives a DMA request. If the 8237A is enabled (bit 2 in the command register is 0) and channel 1 is unmasked, a DMA service will occur after only one byte of the Address register has been reprogrammed. This can be avoided by disabling the controller (setting bit 2 in the command register) or masking the channel before programming any other registers. Once the programming is complete, the controller can be enabled/unmasked.

After power-up it is suggested that all internal locations, especially the Mode registers, be loaded with some valid value. This should be done even if some

channels are unused. An invalid mode may force all control signals to go active at the same time.

APPLICATION INFORMATION(1)

Figure 8 shows a convenient method for configuring a DMA system with the 8237A controller and an 8080A/8085AH microprocessor system. The multimode DMA controller issues a HRQ to the processor whenever there is at least one valid DMA request from a peripheral device. When the processor replies with a HLDA signal, the 8237A takes control of the address bus, the data bus and the control bus. The address for the first transfer operation comes out in two bytes—the least significant 8 bits on the eight address outputs and the most significant 8 bits on the data bus. The contents of the data bus are then latched into an 8-bit latch to complete the full 16 bits of the address bus. The 8282 is a high speed, 8-bit, three-state latch in a 20-pin package. After the initial transfer takes place, the latch is updated only after a carry or borrow is generated in the least significant address byte. Four DMA channels are provided when one 8237A is used.

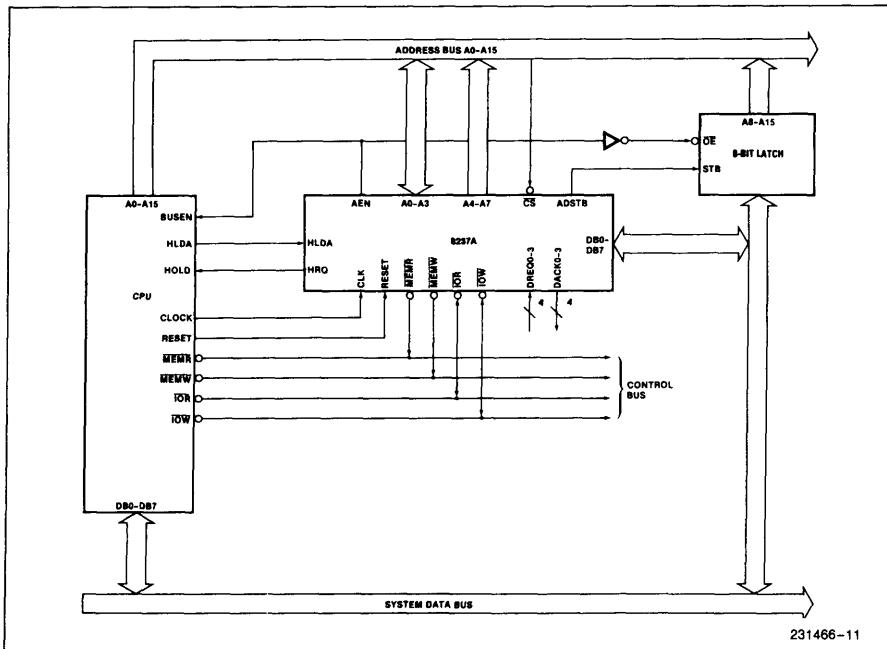


Figure 8. 8237A System Interface

NOTE:

1. See Application Note AP-67 for 8086 design information.

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature under Bias 0°C to 70°C
 Case Temperature 0°C to +75°C
 Storage Temperature -65°C to +150°C
 Voltage on Any Pin with
 Respect to Ground -0.5V to +7V
 Power Dissipation 1.5 Watt

*Notice: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

D.C. CHARACTERISTICS

$T_A = 0^\circ\text{C}$ to 70°C , $T_{\text{CASE}} = 0^\circ\text{C}$ to 75°C , $V_{\text{CC}} = +5.0\text{V} \pm 5\%$, $\text{GND} = 0\text{V}$

Symbol	Parameter	Min	Typ(1)	Max	Unit	Test Conditions
V_{OH}	Output High Voltage	2.4			V	$I_{\text{OH}} = -200\ \mu\text{A}$
		3.3			V	$I_{\text{OH}} = -100\ \mu\text{A}$ (HRQ Only)
V_{OL}	Output LOW Voltage			0.40	V	$I_{\text{OL}} = 3.2\ \text{mA}$
V_{IH}	Input HIGH Voltage	2.0		$V_{\text{CC}} + 0.5$	V	(Note 8)
V_{IL}	Input LOW Voltage	-0.5		0.8	V	
I_{LI}	Input Load Current			± 10	μA	$0\text{V} \leq V_{\text{IN}} \leq V_{\text{CC}}$
I_{LO}	Output Leakage Current			± 10	μA	$0.45\text{V} \leq V_{\text{OUT}} \leq V_{\text{CC}}$
I_{CC}	V_{CC} Supply Current	110	130	150	mA	$T_A = +25^\circ\text{C}$
			130	150	mA	$T_A = 0^\circ\text{C}$
C_O	Output Capacitance		4	8	pF	$f_c = 1.0\ \text{MHz}$, Inputs = 0V
C_I	Input Capacitance		8	15	pF	
C_{IO}	I/O Capacitance		10	18	pF	

A.C. CHARACTERISTICS—DMA (MASTER) MODET_A = 0°C to 70°C, T_{CASE} = 0°C to 75°C, V_{CC} = +5V ± 5%, GND = 0V

Symbol	Parameter	8237A		8237A-4		8237A-5		Unit
		Min	Max	Min	Max	Min	Max	
Tael	AEN HIGH from CLK LOW (S1) Delay Time	300		225		200		ns
taet	AEN LOW from CLK HIGH (S1) Delay Time	200		150		130		ns
Tafab	ADR Active to Float Delay from CLK HIGH	150		120		90		ns
Tafc	READ or WRITE Float from CLK HIGH	150		120		120		ns
Tafdb	DB Active to Float Delay from CLK HIGH	250		190		170		ns
TAHR	ADR from READ HIGH Hold Time	TCY-100		TCY-100		TCY-100		ns
TAHS	DB from ADSTB LOW Hold Time	40		40		30		ns
TAHW	ADR from WRITE HIGH Hold Time	TCY-50		TCY-50		TCY-50		ns
TAK	DACK Valid from CLK LOW Delay Time (Note 7)	250		220		170		ns
	EOP HIGH from CLK HIGH Delay Time (Note 10)	250		190		170		ns
	EOP LOW from CLK HIGH Delay Time	250		190		170		ns
Tasm	ADR Stable from CLK HIGH	250		190		170		ns
Tass	DB to ADSTB LOW Setup Time	100		100		100		ns
Tch	Clock High Time (Transitions ≤ 10 ns)	120		100		80		ns
Tcl	Clock Low Time (Transitions ≤ 10 ns)	150		110		68		ns
Tcy	CLK Cycle Time	320		250		200		ns
tdcl	CLK HIGH to READ or WRITE LOW Delay (Note 4)	270		200		190		ns
tdctr	READ HIGH from CLK HIGH (S4) Delay Time (Note 4)	270		210		190		ns
tdctw	WRITE HIGH from CLK HIGH (S4) Delay Time (Note 4)	200		150		130		ns
tdq1	HRQ Valid from CLK HIGH Delay Time (Note 5)	160		120		120		ns
tdq2		250		190		120		ns
teps	EOP LOW from CLK LOW Setup Time	60		45		40		ns
tepw	EOP Pulse Width	300		225		220		ns
tfaab	ADR Float to Active Delay from CLK HIGH	250		190		170		ns
tfac	READ or WRITE Active from CLK HIGH	200		150		150		ns
tfadb	DB Float to Active Delay from CLK HIGH	300		225		200		ns
ths	HLDA Valid to CLK HIGH Setup Time	100		75		75		ns
tidh	Input Data from MEMR HIGH Hold Time	0		0		0		ns
tids	Input Data to MEMR HIGH Setup Time	250		190		170		ns
todh	Output Data from MEMW HIGH Hold Time	20		20		10		ns
todv	Output Data Valid to MEMW HIGH	200		125		125		ns
tqs	DREQ to CLK LOW (S1, S4) Setup Time (Note 7)	0		0		0		ns
trh	CLK to READY LOW Hold Time	20		20		20		ns
trs	READY to CLK LOW Setup Time	100		60		60		ns
tstl	ADSTB HIGH from CLK HIGH Delay Time	200		150		130		ns
tstt	ADSTB LOW from CLK HIGH Delay Time	140		110		90		ns

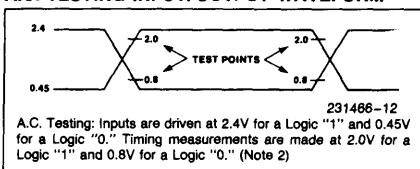
A.C. CHARACTERISTICS—PERIPHERAL (SLAVE) MODE

TA = 0°C to 70°C, TCASE = 0°C to 75°C, VCC = +5V ± 5%, GND = 0V

Symbol	Parameter	8237A		8237A-4		8237A-5		Unit
		Min	Max	Min	Max	Min	Max	
TAR	ADR Valid or CS LOW to READ LOW	50		50		50		ns
TAW	ADR Valid to WRITE HIGH Setup Time	200		150		130		ns
TCW	CS LOW to WRITE HIGH Setup Time	200		150		130		ns
TDW	Data Valid to WRITE HIGH Setup Time	200		150		130		ns
TRA	ADR or CS Hold from READ HIGH	0		0		0		ns
TRDE	Data Access from READ LOW (Note 12)		200		200		140	ns
TRDF	DB Float Delay from READ HIGH	20	100	20	100	0	70	ns
TRSTD	Power Supply HIGH to RESET LOW Setup Time	500		500		500		ns
TRSTS	RESET to First IOWR		2TCY		2TCY		2TCY	ns
TRSTW	RESET Pulse Width	300		300		300		ns
TRW	READ Width	300		250		200		ns
TWA	ADR from WRITE HIGH Hold Time	20		20		20		ns
TWC	CS HIGH from WRITE HIGH Hold Time	20		20		20		ns
TWD	Data from WRITE HIGH Hold Time	30		30		30		ns
TWWS	Write Width	200		200		160		ns
TWR	End of Write to End of Read in DMA Transfer	0		0		0		ns

NOTES:

1. Typical values are for TA = 25°C, nominal supply voltage and nominal processing parameters.
2. Input timing parameters assume transition times of 20 ns or less. Waveform measurement points for both input and output signals are 2.0V for HIGH and 0.8V for LOW, unless otherwise noted.
3. Output loading is 1 TTL gate plus 150 pF capacitance, unless otherwise noted.
4. The net IOW or MEMW Pulse width for normal write will be TCY-100 ns and for extended write will be 2TCY-100 ns. The net IOR or MEMR pulse width for normal read will be 2TCY-50 ns and for compressed read will be TCY-50 ns.
5. TDQ1 is specified for two different output HIGH levels. TDQ1 is measured at 2.0V. TDQ2 is measured at 3.3V. The value for TDQ2 assumes an external 3.3 kΩ pull-up resistor connected from HRQ to VCC.
6. DREQ should be held active until DACK is returned.
7. DREQ and DACK signals may be active high or active low. Timing diagrams assume the active high mode.
8. Successive read and/or write operations by the external processor to program or examine the controller must be timed to allow at least 600 ns for the 8237A, at least 500 ns for the 8237A-4 and at least 400 ns for the 8237A-5, as recovery time between active read or write pulses. The same recovery time is needed between an active read or write pulse followed by a DMA transfer.
9. EOP is an open collector output. This parameter assumes the presence of a 2.2K pullup to VCC.
10. Pin 5 is an input that should always be at a logic high level. An internal pull-up resistor will establish a logic high when the pin is left floating. It is recommended however, that pin 5 be tied to VCC.
11. Output Loading on the Data Bus is 1 TTL Gate plus 100 pF capacitance.

A.C. TESTING INPUT/OUTPUT WAVEFORM

WAVEFORMS

SLAVE MODE WRITE TIMING

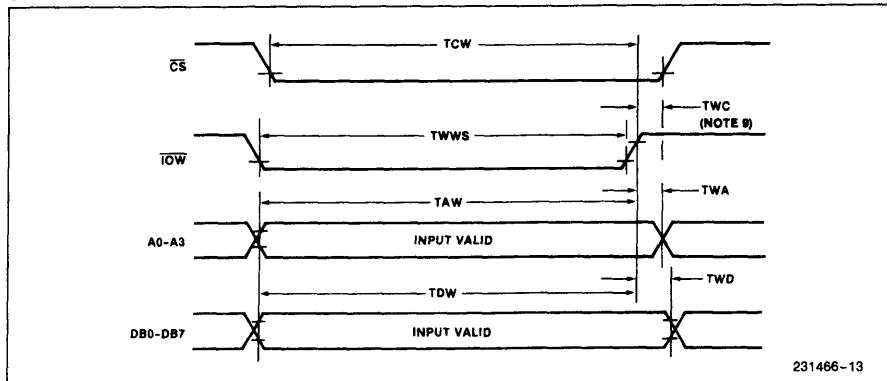


Figure 9. Slave Mode Write

SLAVE MODE READ TIMING

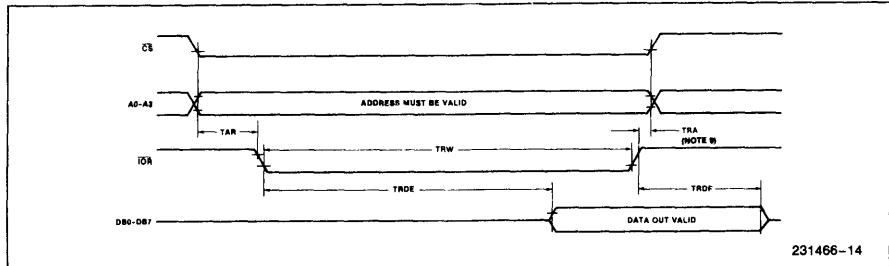
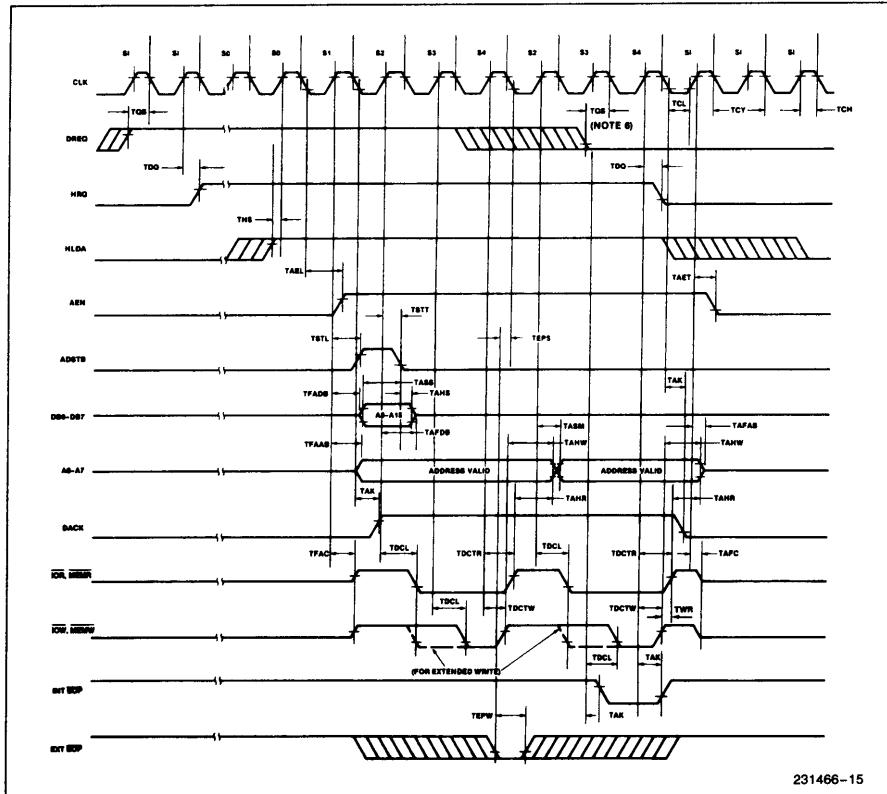
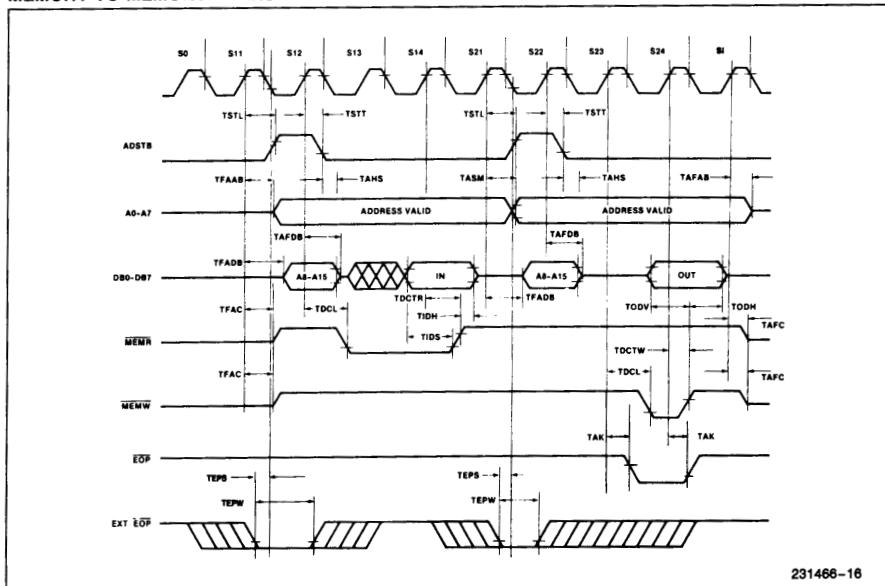
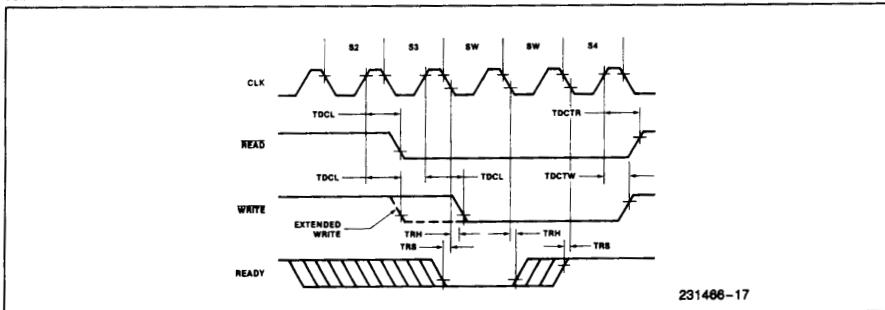
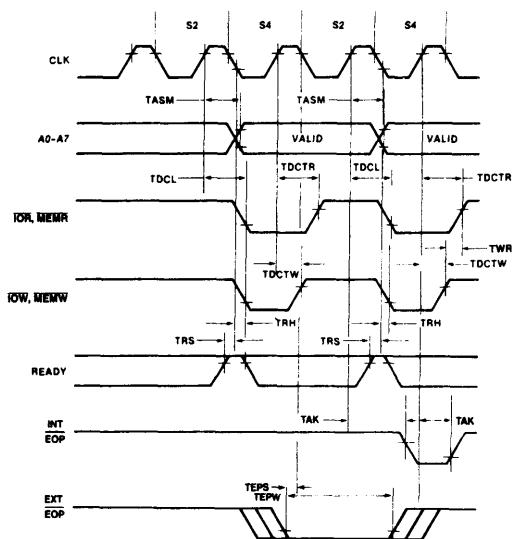


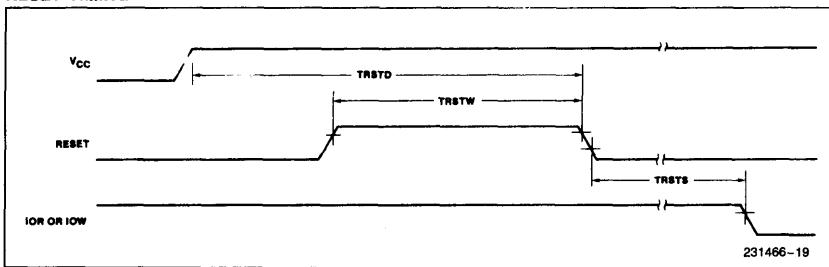
Figure 10. Slave Mode Read

WAVEFORMS (Continued)**DMA TRANSFER TIMING****Figure 11. DMA Transfer**

WAVEFORMS (Continued)**MEMORY-TO-MEMORY TRANSFER TIMING****Figure 12. Memory-to-Memory Transfer****READY TIMING****Figure 13. Ready**

WAVEFORMS (Continued)
COMPRESSED TRANSFER TIMING


231466-18

Figure 14. Compressed Transfer
RESET TIMING


231466-19

Figure 15. Reset

8254 PROGRAMMABLE INTERVAL TIMER

- Compatible with All Intel and Most Other Microprocessors
- Handles Inputs from DC to 10 MHz
 - 5 MHz 8254-5
 - 8 MHz 8254
 - 10 MHz 8254-2
- Status Read-Back Command
- Six Programmable Counter Modes
- Three Independent 16-Bit Counters
- Binary or BCD Counting
- Single +5V Supply
- Available in EXPRESS
 - Standard Temperature Range

The Intel® 8254 is a counter/timer device designed to solve the common timing control problems in microcomputer system design. It provides three independent 16-bit counters, each capable of handling clock inputs up to 10 MHz. All modes are software programmable. The 8254 is a superset of the 8253.

The 8254 uses HMOS technology and comes in a 24-pin plastic or CERDIP package.

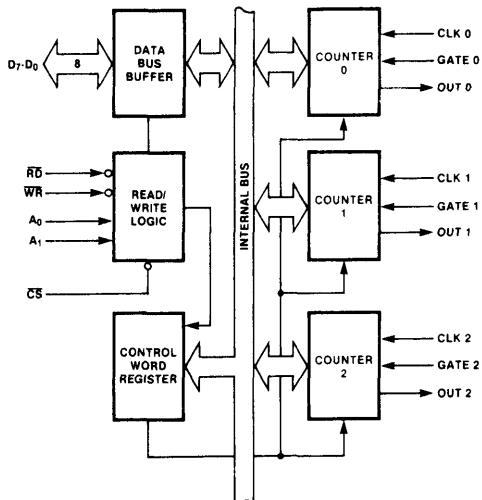


Figure 1. 8254 Block Diagram

231164-1

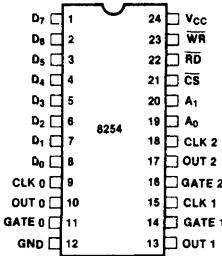


Figure 2. Pin Configuration

231164-2

Table 1. Pin Description

Symbol	Pin No.	Type	Name and Function															
D ₇ -D ₀	1-8	I/O	DATA: Bi-directional three state data bus lines, connected to system data bus.															
CLK 0	9	I	CLOCK 0: Clock input of Counter 0.															
OUT 0	10	O	OUTPUT 0: Output of Counter 0.															
GATE 0	11	I	GATE 0: Gate input of Counter 0.															
GND	12		GROUND: Power supply connection.															
V _{CC}	24		POWER: +5V power supply connection.															
WR	23	I	WRITE CONTROL: This input is low during CPU write operations.															
RD	22	I	READ CONTROL: This input is low during CPU read operations.															
CS	21	I	CHIP SELECT: A low on this input enables the 8254 to respond to RD and WR signals. RD and WR are ignored otherwise.															
A ₁ , A ₀	20-19	I	ADDRESS: Used to select one of the three Counters or the Control Word Register for read or write operations. Normally connected to the system address bus. <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>A₁</th> <th>A₀</th> <th>Selects</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>Counter 0</td> </tr> <tr> <td>0</td> <td>1</td> <td>Counter 1</td> </tr> <tr> <td>1</td> <td>0</td> <td>Counter 2</td> </tr> <tr> <td>1</td> <td>1</td> <td>Control Word Register</td> </tr> </tbody> </table>	A ₁	A ₀	Selects	0	0	Counter 0	0	1	Counter 1	1	0	Counter 2	1	1	Control Word Register
A ₁	A ₀	Selects																
0	0	Counter 0																
0	1	Counter 1																
1	0	Counter 2																
1	1	Control Word Register																
CLK 2	18	I	CLOCK 2: Clock input of Counter 2.															
OUT 2	17	O	OUT 2: Output of Counter 2.															
GATE 2	16	I	GATE 2: Gate input of Counter 2.															
CLK 1	15	I	CLOCK 1: Clock input of Counter 1.															
GATE 1	14	I	GATE 1: Gate input of Counter 1.															
OUT 1	13	O	OUT 1: Output of Counter 1.															

FUNCTIONAL DESCRIPTION

General

The 8254 is a programmable interval timer/counter designed for use with Intel microcomputer systems. It is a general purpose, multi-timing element that can be treated as an array of I/O ports in the system software.

The 8254 solves one of the most common problems in any microcomputer system, the generation of accurate time delays under software control. Instead of setting up timing loops in software, the programmer configures the 8254 to match his requirements and programs one of the counters for the desired delay. After the desired delay, the 8254 will interrupt the CPU. Software overhead is minimal and variable length delays can easily be accommodated.

Some of the other counter/timer functions common to microcomputers which can be implemented with the 8254 are:

- Real time clock
- Event-counter
- Digital one-shot
- Programmable rate generator
- Square wave generator
- Binary rate multiplier
- Complex waveform generator
- Complex motor controller

Block Diagram

DATA BUS BUFFER

This 3-state, bi-directional, 8-bit buffer is used to interface the 8254 to the system bus (see Figure 3).

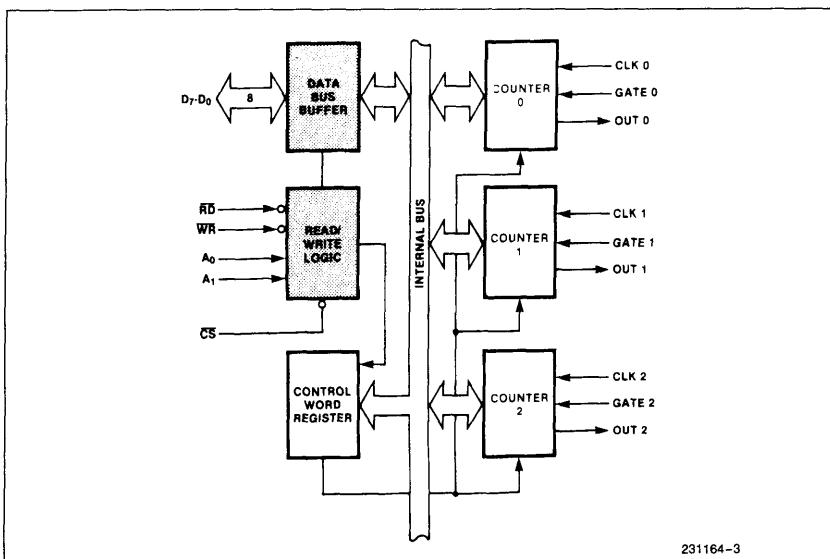


Figure 3. Block Diagram Showing Data Bus Buffer and Read/Write Logic Functions

READ/WRITE LOGIC

The Read/Write Logic accepts inputs from the system bus and generates control signals for the other functional blocks of the 8254. A₁ and A₀ select one of the three counters or the Control Word Register to be read from/written into. A "low" on the RD input tells the 8254 that the CPU is reading one of the counters. A "low" on the WR input tells the 8254 that the CPU is writing either a Control Word or an initial count. Both RD and WR are qualified by CS; RD and WR are ignored unless the 8254 has been selected by holding CS low.

CONTROL WORD REGISTER

The Control Word Register (see Figure 4) is selected by the Read/Write Logic when A₁, A₀ = 11. If the CPU then does a write operation to the 8254, the data is stored in the Control Word Register and is interpreted as a Control Word used to define the operation of the Counters.

The Control Word Register can only be written to; status information is available with the Read-Back Command.

COUNTER 0, COUNTER 1, COUNTER 2

These three functional blocks are identical in operation, so only a single Counter will be described. The internal block diagram of a single counter is shown in Figure 5.

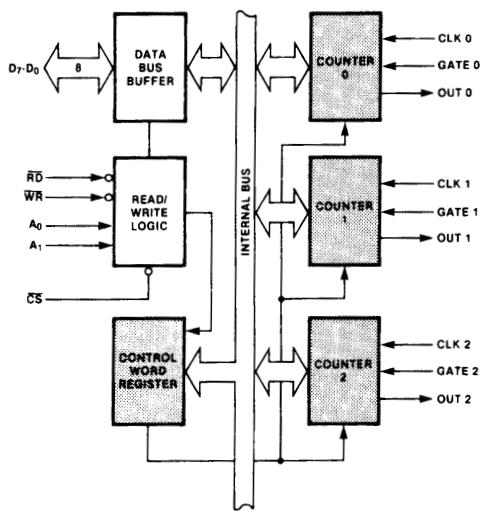
The Counters are fully independent. Each Counter may operate in a different Mode.

The Control Word Register is shown in the figure; it is not part of the Counter itself, but its contents determine how the Counter operates.

The status register, shown in Figure 5, when latched, contains the current contents of the Control Word Register and status of the output and null count flag. (See detailed explanation of the Read-Back command.)

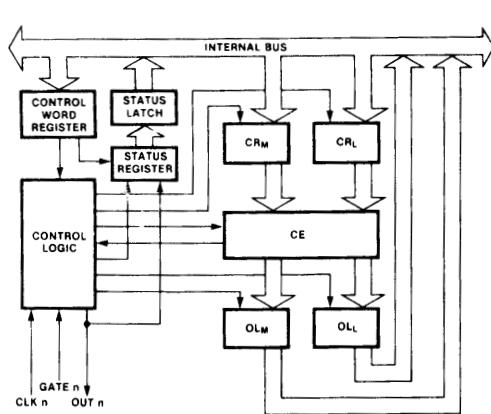
The actual counter is labelled CE (for "Counting Element"). It is a 16-bit presetable synchronous down counter.

OL_M and OL_L are two 8-bit latches. OL stands for "Output Latch"; the subscripts M and L stand for "Most significant byte" and "Least significant byte".



231164-4

Figure 4. Block Diagram Showing Control Word Register and Counter Functions



231164-5

Figure 5. Internal Block Diagram of a Counter

respectively. Both are normally referred to as one unit and called just OL. These latches normally "follow" the CE, but if a suitable Counter Latch Command is sent to the 8254, the latches "latch" the present count until read by the CPU and then return to "following" the CE. One latch at a time is enabled by the counter's Control Logic to drive the internal bus. This is how the 16-bit Counter communicates over the 8-bit internal bus. Note that the CE itself cannot be read; whenever you read the count, it is the OL that is being read.

Similarly, there are two 8-bit registers called CR_M and CR_L (for "Count Register"). Both are normally referred to as one unit and called just CR. When a new count is written to the Counter, the count is stored in the CR and later transferred to the CE. The Control Logic allows one register at a time to be loaded from the internal bus. Both bytes are transferred to the CE simultaneously. CR_M and CR_L are cleared when the Counter is programmed. In this way, if the Counter has been programmed for one byte counts (either most significant byte only or least significant byte only) the other byte will be zero. Note that the CE cannot be written into; whenever a count is written, it is written into the CR.

The Control Logic is also shown in the diagram. CLK n, GATE n, and OUT n are all connected to the outside world through the Control Logic.

8254 SYSTEM INTERFACE

The 8254 is a component of the Intel Microcomputer Systems and interfaces in the same manner as all

other peripherals of the family. It is treated by the system's software as an array of peripheral I/O ports; three are counters and the fourth is a control register for MODE programming.

Basically, the select inputs A₀, A₁ connect to the A₀, A₁ address bus signals of the CPU. The CS can be derived directly from the address bus using a linear select method. Or it can be connected to the output of a decoder, such as an Intel 8205 for larger systems.

OPERATIONAL DESCRIPTION

General

After power-up, the state of the 8254 is undefined. The Mode, count value, and output of all Counters are undefined.

How each Counter operates is determined when it is programmed. Each Counter must be programmed before it can be used. Unused counters need not be programmed.

Programming the 8254

Counters are programmed by writing a Control Word and then an initial count.

The Control Words are written into the Control Word Register, which is selected when A₁, A₀ = 11. The Control Word itself specifies which Counter is being programmed.

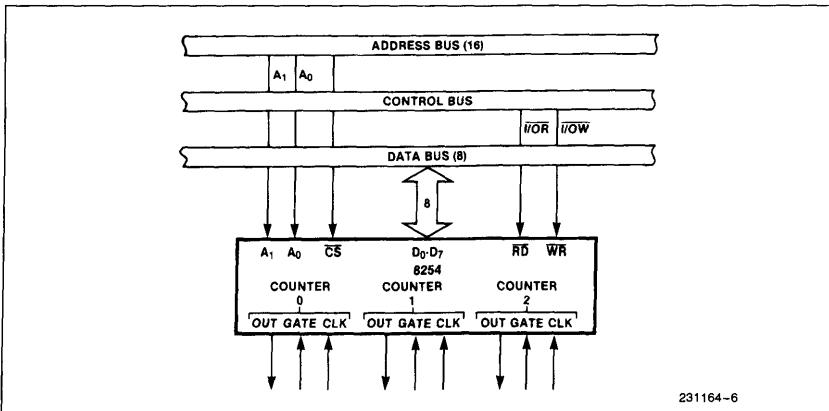


Figure 6. 8254 System Interface

Control Word Format $A_1, A_0 = 11 \quad \overline{CS} = 0 \quad \overline{RD} = 1 \quad \overline{WR} = 0$

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SC1	SC0	RW1	RW0	M2	M1	M0	BCD

SC—Select Counter		M—Mode					
SC1	SC0	M2	M1	M0			
0	0	0	0	0	Mode 0		
0	1	0	1	1	Mode 1		
1	0	1	0	0	Mode 2		
1	1	1	1	1	Mode 3		

RW—Read/Write		BCD					
RW1	RW0	0	Binary Counter 16-bits				
0	0	0	Binary Counter 16-bits				
0	1	1	Binary Coded Decimal (BCD) Counter (4 Decades)				
1	0	0	Binary Coded Decimal (BCD) Counter (4 Decades)				
1	1	1	Binary Coded Decimal (BCD) Counter (4 Decades)				

NOTE:
Don't care bits (X) should be 0 to insure compatibility with future Intel products.

Figure 7. Control Word Format

By contrast, initial counts are written into the Counters, not the Control Word Register. The A_1, A_0 inputs are used to select the Counter to be written into. The format of the initial count is determined by the Control Word used.

Write Operations

The programming procedure for the 8254 is very flexible. Only two conventions need to be remembered:

- 1) For each Counter, the Control Word must be written before the initial count is written.
- 2) The initial count must follow the count format specified in the Control Word (least significant byte only, most significant byte only, or least significant byte and then most significant byte).

Since the Control Word Register and the three Counters have separate addresses (selected by the A_1, A_0 inputs), and each Control Word specifies the Counter it applies to (SC0, SC1 bits), no special instruction sequence is required. Any programming sequence that follows the conventions in Figure 7 is acceptable.

A new initial count may be written to a Counter at any time without affecting the Counter's programmed Mode in any way. Counting will be affected as described in the Mode definitions. The new count must follow the programmed count format.

If a Counter is programmed to read/write two-byte counts, the following precaution applies: A program must not transfer control between writing the first and second byte to another routine which also writes into that same Counter. Otherwise, the Counter will be loaded with an incorrect count.

	A ₁	A ₀		A ₁	A ₀
Control Word—Counter 0	1	1	Control Word—Counter 2	1	1
LSB of count—Counter 0	0	0	Control Word—Counter 1	1	1
MSB of count—Counter 0	0	0	Control Word—Counter 0	1	1
Control Word—Counter 1	1	1	LSB of count—Counter 2	1	0
LSB of count—Counter 1	0	1	MSB of count—Counter 2	1	0
MSB of count—Counter 1	0	1	LSB of count—Counter 1	0	1
Control Word—Counter 2	1	1	MSB of count—Counter 1	0	1
LSB of count—Counter 2	1	0	LSB of count—Counter 0	0	0
MSB of count—Counter 2	1	0	MSB of count—Counter 0	0	0
	A ₁	A ₀		A ₁	A ₀
Control Word—Counter 0	1	1	Control Word—Counter 1	1	1
Control Word—Counter 1	1	1	Control Word—Counter 0	1	1
Control Word—Counter 2	1	1	LSB of count—Counter 1	0	1
LSB of count—Counter 2	1	0	Control Word—Counter 2	1	1
LSB of count—Counter 1	0	1	LSB of count—Counter 0	0	0
LSB of count—Counter 0	0	0	MSB of count—Counter 1	0	1
MSB of count—Counter 0	0	0	LSB of count—Counter 2	1	0
MSB of count—Counter 1	0	1	MSB of count—Counter 0	0	0
MSB of count—Counter 2	1	0	MSB of count—Counter 2	1	0

NOTE:
In all four examples, all Counters are programmed to read/write two-byte counts. These are only four of many possible programming sequences.

Figure 8. A Few Possible Programming Sequences

Read Operations

It is often desirable to read the value of a Counter without disturbing the count in progress. This is easily done in the 8254.

There are three possible methods for reading the counters: a simple read operation, the Counter Latch Command, and the Read-Back Command. Each is explained below. The first method is to perform a simple read operation. To read the Counter, which is selected with the A₁, A₀ inputs, the CLK input of the selected Counter must be inhibited by using either the GATE input or external logic. Otherwise, the count may be in the process of changing when it is read, giving an undefined result.

COUNTER LATCH COMMAND

The second method uses the "Counter Latch Command". Like a Control Word, this command is written to the Control Word Register, which is selected when A₁, A₀ = 11. Also like a Control Word, the SC0, SC1 bits select one of the three Counters, but two other bits, D5 and D4, distinguish this command from a Control Word.

A₁, A₀ = 11; CS = 0; RD = 1; WR = 0

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
SC1	SC0	0	0	X	X	X	X

SC1,SC0—specify counter to be latched

SC1 SC0 Counter

0	0	0
0	1	1
1	0	2
1	1	Read-Back Command

D5,D4=00 designates Counter Latch Command

X—don't care

NOTE:

Don't care bits (X) should be 0 to insure compatibility with future Intel products.

Figure 9. Counter Latching Command Format

The selected Counter's output latch (OL) latches the count at the time the Counter Latch Command is received. This count is held in the latch until it is read by the CPU (or until the Counter is reprogrammed). The count is then unlatched automatically and the OL returns to "following" the counting element (CE). This allows reading the contents of the Counters "on the fly" without affecting counting in progress. Multiple Counter Latch Commands may be used to latch more than one Counter. Each latched Counter's OL holds its count until it is read. Counter Latch Commands do not affect the programmed Mode of the Counter in any way.

If a Counter is latched and then, some time later, latched again before the count is read, the second Counter Latch Command is ignored. The count read will be the count at the time the first Counter Latch Command was issued.

With either method, the count must be read according to the programmed format; specifically, if the Counter is programmed for two byte counts, two bytes must be read. The two bytes do not have to be read one right after the other; read or write or programming operations of other Counters may be inserted between them.

Another feature of the 8254 is that reads and writes of the same Counter may be interleaved; for example, if the Counter is programmed for two byte counts, the following sequence is valid.

- 1) Read least significant byte.
- 2) Write new least significant byte.
- 3) Read most significant byte.
- 4) Write new most significant byte.

If a Counter is programmed to read/write two-byte counts, the following precaution applies: A program must not transfer control between reading the first and second byte to another routine which also reads from that same Counter. Otherwise, an incorrect count will be read.

READ-BACK COMMAND

The third method uses the Read-Back Command. This command allows the user to check the count value, programmed Mode, and current states of the OUT pin and Null Count flag of the selected counter(s).

The command is written into the Control Word Register and has the format shown in Figure 10. The command applies to the counters selected by setting their corresponding bits D3, D2, D1 = 1.

A0, A1 = 11		CS = 0	RD = 1	WR = 0			
D₇	D₆	D₅	D₄	D₃	D₂	D₁	D₀
1	1	COUNT	STATUS	CNT 2	CNT 1	CNT 0	0

D₅: 0 = Latch count of selected counter(s)
D₄: 0 = Latch status of selected counters(s)
D₃: 1 = Select Counter 2
D₂: 1 = Select Counter 1
D₁: 1 = Select Counter 0
D₀: Reserved for future expansion; Must be 0

Figure 10. Read-Back Command Format

The read-back command may be used to latch multiple counter output latches (OL) by setting the COUNT bit D5 = 0 and selecting the desired counter(s). This single command is functionally equivalent to several counter latch commands, one for each counter latched. Each counter's latched count is held until it is read (or the counter is reprogrammed). The counter is automatically unlatched when read, but other counters remain latched until they are read. If multiple count read-back commands are issued to the same counter without reading the count, all but the first are ignored; i.e., the count which will be read is the count at the time the first read-back command was issued.

The read-back command may also be used to latch status information of selected counter(s) by setting STATUS bit D4 = 0. Status must be latched to be read; status of a counter is accessed by a read from that counter.

The counter status format is shown in Figure 11. Bits D5 through D0 contain the counter's programmed Mode exactly as written in the last Mode Control Word. OUTPUT bit D7 contains the current state of the OUT pin. This allows the user to monitor the counter's output via software, possibly eliminating some hardware from a system.

D₇	D₆	D₅	D₄	D₃	D₂	D₁	D₀
Output	Null Count	RW1	RW0	M2	M1	M0	BCD

D₇ 1 = OUT Pin is 1
 0 = OUT Pin is 0
D₆ 1 = Null Count
 0 = Count available for reading
D₅-D₀ Counter programmed mode (see Figure 7)

Figure 11. Status Byte

NULL COUNT bit D6 indicates when the last count written to the counter register (CR) has been loaded into the counting element (CE). The exact time this happens depends on the Mode of the counter and is described in the Mode Definitions, but until the count is loaded into the counting element (CE), it can't be read from the counter. If the count is latched or read before this time, the count value will not reflect the new count just written. The operation of Null Count is shown in Figure 12.

This Action	Causes
A. Write to the control word register;(1) Null Count = 1	
B. Write to the count register (CR);(2)	Null Count = 1
C. New Count is loaded into	Null Count = 0
CE (CR → CE);	
NOTE:	
1. Only the counter specified by the control word will have its Null Count set to 1. Null count bits of other counters are unaffected.	
2. If the counter is programmed for two-byte counts (least significant byte then most significant byte) Null Count goes to 1 when the second byte is written.	

Figure 12. Null Count Operation

If multiple status latch operations of the counter(s) are performed without reading the status, all but the first are ignored; i.e., the status that will be read is the status of the counter at the time the first status read-back command was issued.

Both count and status of the selected counter(s) may be latched simultaneously by setting both

COUNT and **STATUS** bits D5,D4 = 0. This is functionally the same as issuing two separate read-back commands at once, and the above discussions apply here also. Specifically, if multiple count and/or status read-back commands are issued to the same counter(s) without any intervening reads, all but the first are ignored. This is illustrated in Figure 13.

If both count and status of a counter are latched, the first read operation of that counter will return latched status, regardless of which was latched first. The next one or two reads (depending on whether the counter is programmed for one or two type counts) return latched count. Subsequent reads return unlatched count.

CS	RD	WR	A1	A0	
0	1	0	0	0	Write into Counter 0
0	1	0	0	1	Write into Counter 1
0	1	0	1	0	Write into Counter 2
0	1	0	1	1	Write Control Word
0	0	1	0	0	Read from Counter 0
0	0	1	0	1	Read from Counter 1
0	0	1	1	0	Read from Counter 2
0	0	1	1	1	No-Operation (3-State)
1	X	X	X	X	No-Operation (3-State)
0	1	1	X	X	No-Operation (3-State)

Figure 14. Read/Write Operations Summary

Command								Description	Result
D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀		
1	1	0	0	0	0	1	0	Read back count and status of Counter 0	Count and status latched for Counter 0
1	1	1	0	0	1	0	0	Read back status of Counter 1	Status latched for Counter 1
1	1	1	0	1	1	0	0	Read back status of Counters 2, 1	Status latched for Counter 2, but not Counter 1
1	1	0	1	1	0	0	0	Read back count of Counter 2	Count latched for Counter 2
1	1	0	0	0	1	0	0	Read back count and status of Counter 1	Count latched for Counter 1, but not status
1	1	1	0	0	0	1	0	Read back status of Counter 1	Command ignored, status already latched for Counter 1

Figure 13. Read-Back Command Example

Mode Definitions

The following are defined for use in describing the operation of the 8254.

CLK Pulse: a rising edge, then a falling edge, in that order, of a Counter's CLK input.

Trigger: a rising edge of a Counter's GATE input.

Counter loading: the transfer of a count from the CR to the CE (refer to the "Functional Description")

MODE 0: INTERRUPT ON TERMINAL COUNT

Mode 0 is typically used for event counting. After the Control Word is written, OUT is initially low, and will remain low until the Counter reaches zero. OUT then goes high and remains high until a new count or a new Mode 0 Control Word is written into the Counter.

GATE = 1 enables counting; GATE = 0 disables counting. GATE has no effect on OUT.

After the Control Word and initial count are written to a Counter, the initial count will be loaded on the next CLK pulse. This CLK pulse does not decrement the count, so for an initial count of N, OUT does not go high until $N + 1$ CLK pulses after the initial count is written.

If a new count is written to the Counter, it will be loaded on the next CLK pulse and counting will continue from the new count. If a two-byte count is written, the following happens:

- 1) Writing the first byte disables counting. OUT is set low immediately (no clock pulse required)
- 2) Writing the second byte allows the new count to be loaded on the next CLK pulse.

This allows the counting sequence to be synchronized by software. Again, OUT does not go high until $N + 1$ CLK pulses after the new count of N is written.

If an initial count is written while GATE = 0, it will still be loaded on the next CLK pulse. When GATE goes high, OUT will go high N CLK pulses later; no CLK pulse is needed to load the Counter as this has already been done.

MODE 1: HARDWARE RETRIGGERABLE ONE-SHOT

OUT will be initially high. OUT will go low on the CLK pulse following a trigger to begin the one-shot pulse, and will remain low until the Counter reaches zero.

OUT will then go high and remain high until the CLK pulse after the next trigger.

After writing the Control Word and initial count, the Counter is armed. A trigger results in loading the Counter and setting OUT low on the next CLK pulse, thus starting the one-shot pulse. An initial count of N will result in a one-shot pulse N CLK cycles in duration. The one-shot is retriggerable, hence OUT will remain low for N CLK pulses after any trigger. The one-shot pulse can be repeated without rewriting the same count into the counter. GATE has no effect on OUT.

If a new count is written to the Counter during a one-shot pulse, the current one-shot is not affected unless the counter is retriggered. In that case, the Counter is loaded with the new count and the one-shot pulse continues until the new count expires.

MODE 2: RATE GENERATOR

This Mode functions like a divide-by-N counter. It is typically used to generate a Real Time Clock interrupt. OUT will initially be high. When the initial count has decremented to 1, OUT goes low for one CLK pulse. OUT then goes high again, the Counter reloads the initial count and the process is repeated. Mode 2 is periodic; the same sequence is repeated indefinitely. For an initial count of N , the sequence repeats every N CLK cycles.

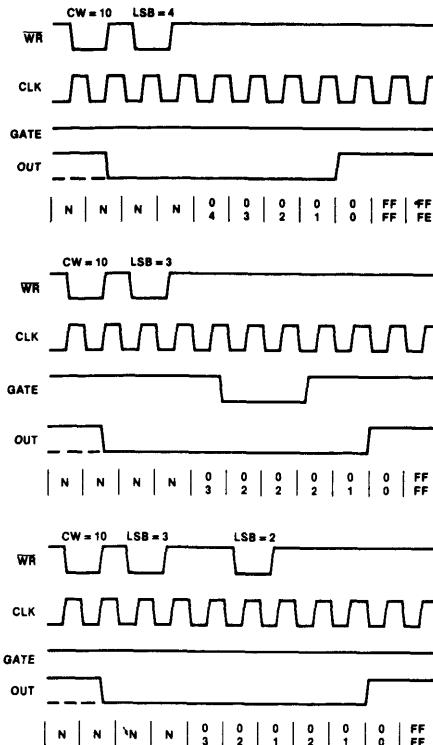
GATE = 1 enables counting; GATE = 0 disables counting. If GATE goes low during an output pulse, OUT is set high immediately. A trigger reloads the Counter with the initial count on the next CLK pulse; OUT goes low N CLK pulses after the trigger. Thus the GATE input can be used to synchronize the Counter.

After writing a Control Word and initial count, the Counter will be loaded on the next CLK pulse. OUT goes low N CLK Pulses after the initial count is written. This allows the Counter to be synchronized by software also.

Writing a new count while counting does not affect the current counting sequence. If a trigger is received after writing a new count but before the end of the current period, the Counter will be loaded with the new count on the next CLK pulse and counting will continue from the new count. Otherwise, the new count will be loaded at the end of the current counting cycle. In mode 2, a COUNT of 1 is illegal.

MODE 3: SQUARE WAVE MODE

Mode 3 is typically used for Baud rate generation. Mode 3 is similar to Mode 2 except for the duty cycle of OUT. OUT will initially be high. When half the



231164-7

NOTE:

- The following conventions apply to all mode timing diagrams:
1. Counters are programmed for binary (not BCD) counting and for reading/writing least significant byte (LSB) only.
 2. The counter is always selected (\bar{CS} always low).
 3. CW stands for "Control Word"; CW = 10 means a control word of 10 HEX is written to the counter.
 4. LSB stands for "Least Significant Byte" of count.
 5. Numbers below diagrams are count values. The lower number is the least significant byte. The upper number is the most significant byte. Since the counter is programmed to read/write LSB only, the most significant byte cannot be read.
- N stands for an undefined count.
Vertical lines show transitions between count values.

Figure 15. Mode 0

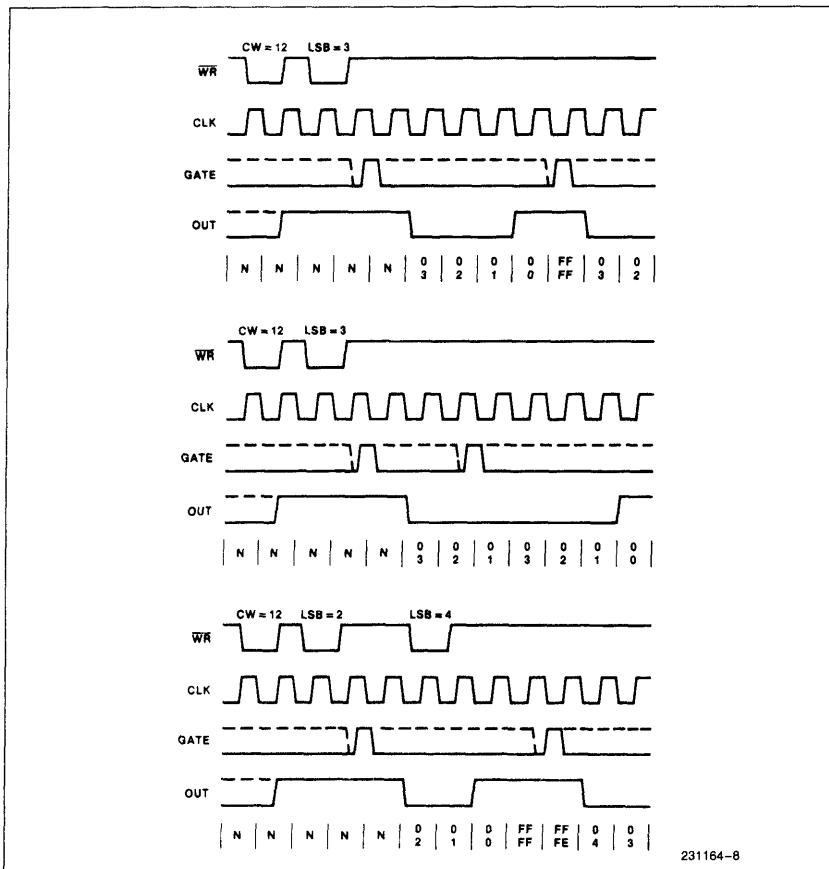


Figure 16. Mode 1

initial count has expired, OUT goes low for the remainder of the count. Mode 3 is periodic; the sequence above is repeated indefinitely. An initial count of N results in a square wave with a period of N CLK cycles.

GATE = 1 enables counting; GATE = 0 disables counting. If GATE goes low while OUT is low, OUT is set high immediately; no CLK pulse is required. A trigger reloads the Counter with the initial count on the next CLK pulse. Thus the GATE input can be used to synchronize the Counter.

After writing a Control Word and initial count, the Counter will be loaded on the next CLK pulse. This allows the Counter to be synchronized by software also.

Writing a new count while counting does not affect the current counting sequence. If a trigger is received after writing a new count but before the end of the current half-cycle of the square wave, the Counter will be loaded with the new count on the next CLK pulse and counting will continue from the

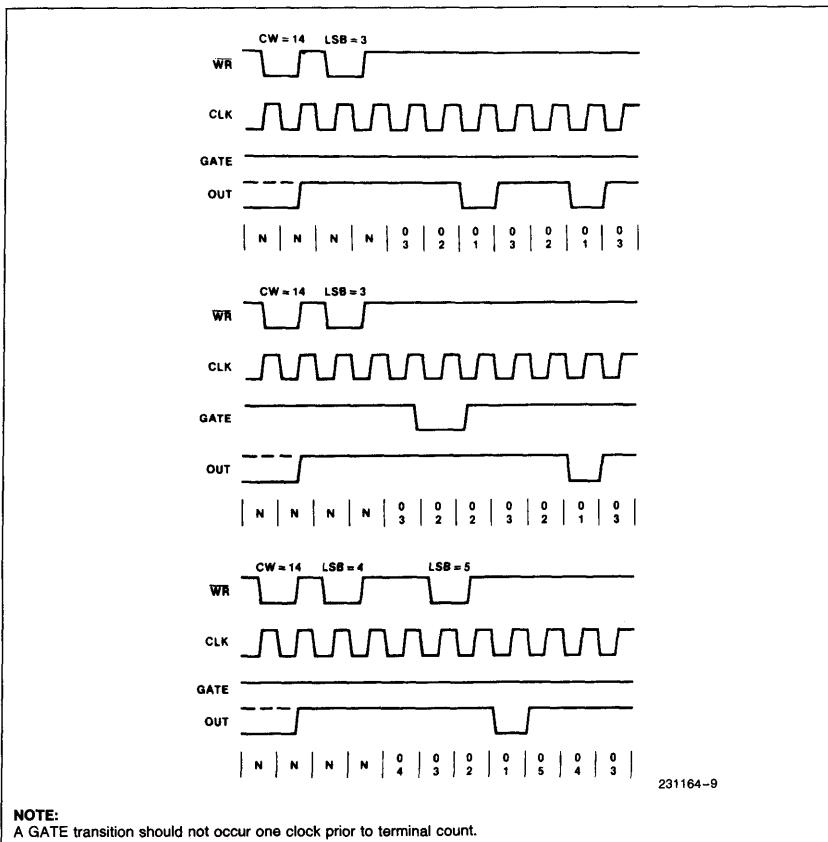


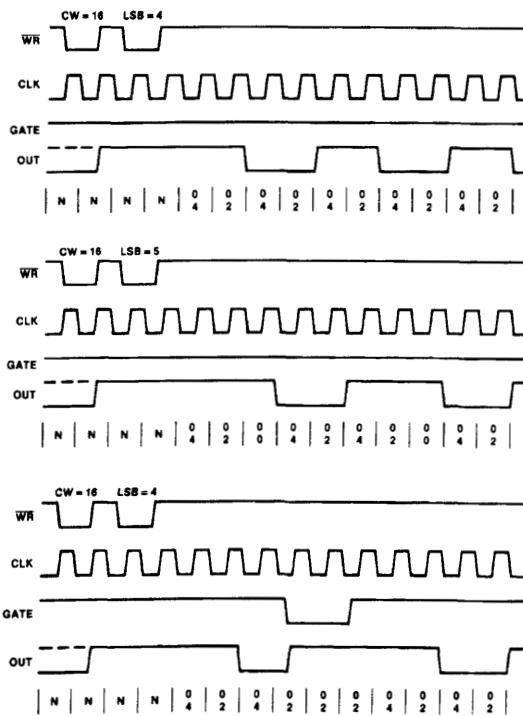
Figure 17. Mode 2

new count. Otherwise, the new count will be loaded at the end of the current half-cycle.

Mode 3 is implemented as follows:

Even counts: OUT is initially high. The initial count is loaded on one CLK pulse and then is decremented by two on succeeding CLK pulses. When the count expires OUT changes value and the Counter is reloaded with the initial count. The above process is repeated indefinitely.

Odd counts: OUT is initially high. The initial count minus one (an even number) is loaded on one CLK pulse and then is decremented by two on succeeding CLK pulses. One CLK pulse after the count expires, OUT goes low and the Counter is reloaded with the initial count minus one. Succeeding CLK pulses decrement the count by two. When the count expires, OUT goes high again and the Counter is reloaded with the initial count minus one. The above process is repeated indefinitely. So for odd counts, OUT will be high for $(N + 1)/2$ counts and low for $(N - 1)/2$ counts.



231164-10

NOTE:

A GATE transition should not occur one clock prior to terminal count.

Figure 18. Mode 3

MODE 4: SOFTWARE TRIGGERED STROBE

OUT will be initially high. When the initial count expires, OUT will go low for one CLK pulse and then go high again. The counting sequence is "triggered" by writing the initial count.

GATE = 1 enables counting; GATE = 0 disables counting. GATE has no effect on OUT.

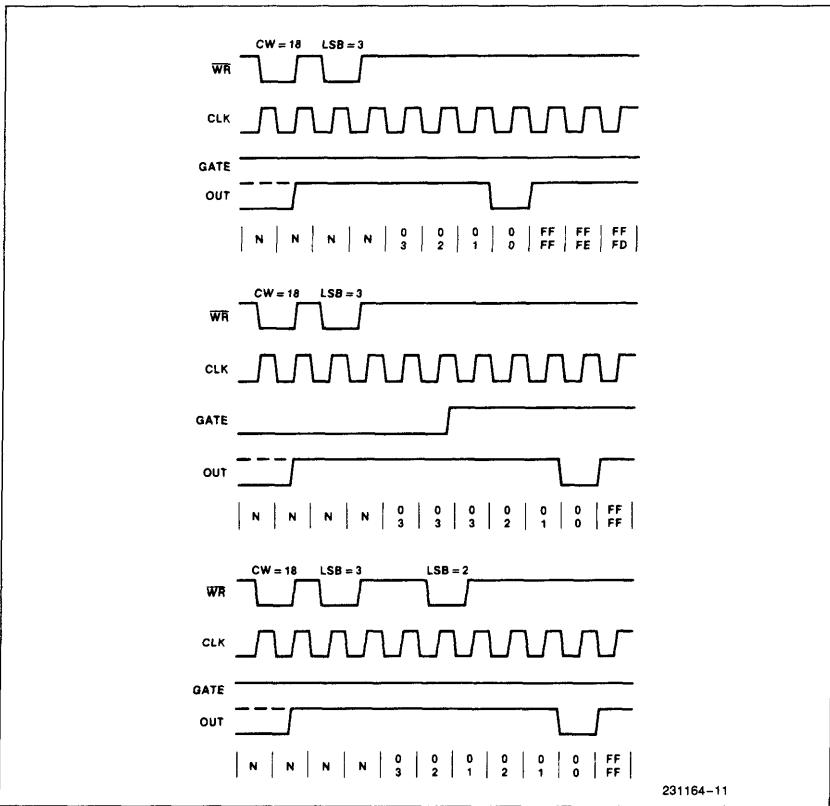
After writing a Control Word and initial count, the Counter will be loaded on the next CLK pulse. This CLK pulse does not decrement the count, so for an

initial count of N, OUT does not strobe low until N + 1 CLK pulses after the initial count is written.

If a new count is written during counting, it will be loaded on the next CLK pulse and counting will continue from the new count. If a two-byte count is written, the following happens:

- 1) Writing the first byte has no effect on counting.
- 2) Writing the second byte allows the new count to be loaded on the next CLK pulse.

This allows the sequence to be "retriggered" by software. OUT strobos low N + 1 CLK pulses after the new count of N is written.



231164-11

Figure 19. Mode 4

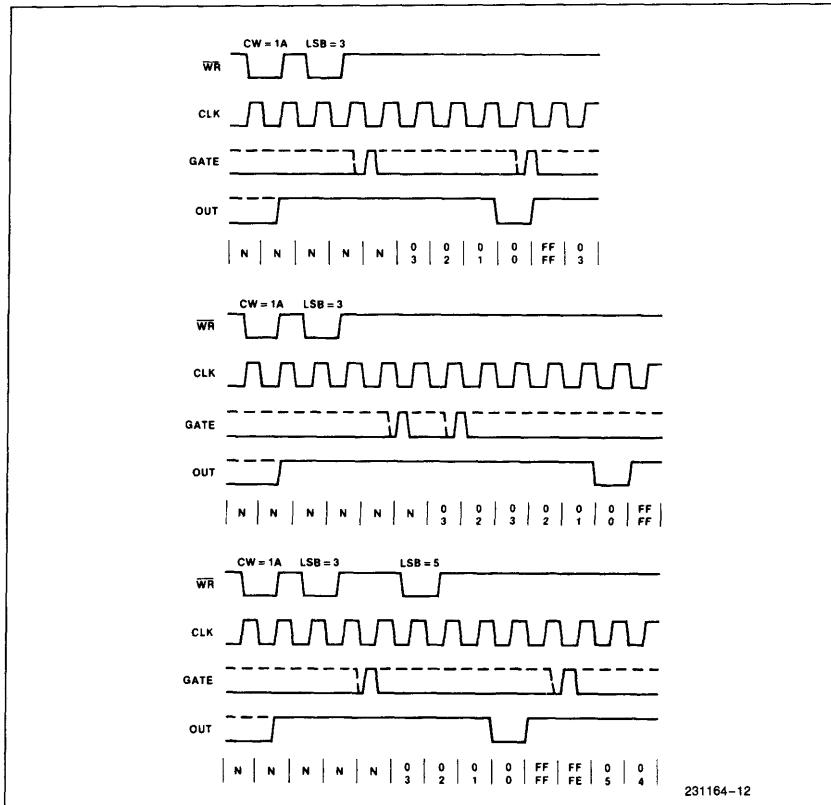
**MODE 5: HARDWARE TRIGGERED STROBE
(RETRIGGERABLE)**

OUT will initially be high. Counting is triggered by a rising edge of GATE. When the initial count has expired, OUT will go low for one CLK pulse and then go high again.

After writing the Control Word and initial count, the counter will not be loaded until the CLK pulse after a trigger. This CLK pulse does not decrement the count, so for an initial count of N, OUT does not strobe low until N + 1 CLK pulses after a trigger.

A trigger results in the Counter being loaded with the initial count on the next CLK pulse. The counting sequence is retriggerable. OUT will not strobe low for N + 1 CLK pulses after any trigger. GATE has no effect on OUT.

If a new count is written during counting, the current counting sequence will not be affected. If a trigger occurs after the new count is written but before the current count expires, the Counter will be loaded with the new count on the next CLK pulse and counting will continue from there.

**Figure 20. Mode 5**

Signal Status Modes	Low Or Going Low	Rising	High
0	Disables Counting	---	Enables Counting
1	---	1) Initiates Counting 2) Resets Output after Next Clock	---
2	1) Disables Counting 2) Sets Output Immediately High	Initiates Counting	Enables Counting
3	1) Disables Counting 2) Sets Output Immediately High	Initiates Counting	Enables Counting
4	Disables Counting	---	Enables Counting
5	---	Initiates Counting	---

Figure 21. Gate Pin Operations Summary

Mode	Min Count	Max Count
0	1	0
1	1	0
2	2	0
3	2	0
4	1	0
5	1	0

NOTE:
0 is equivalent to 2^{16} for binary counting and 10^4 for BCD counting.

Figure 22. Minimum and Maximum Initial Counts

Operation Common to All Modes

PROGRAMMING

When a Control Word is written to a Counter, all Control Logic is immediately reset and OUT goes to a known initial state; no CLK pulses are required for this.

GATE

The GATE input is always sampled on the rising edge of CLK. In Modes 0, 2, 3, and 4 the GATE input is level sensitive, and the logic level is sampled on the rising edge of CLK. In Modes 1, 2, 3, and 5 the GATE input is rising-edge sensitive. In these Modes, a rising edge of GATE (trigger) sets an edge-sensitive flip-flop in the Counter. This flip-flop is then sampled on the next rising edge of CLK; the flip-flop is reset immediately after it is sampled. In this way, a trigger will be detected no matter when it occurs—a high logic level does not have to be maintained until the next rising edge of CLK. Note that in Modes 2 and 3, the GATE input is both edge- and level-sensitive. In Modes 2 and 3, if a CLK source other than the system clock is used, GATE should be pulsed immediately following WR of a new count value.

COUNTER

New counts are loaded and Counters are decremented on the falling edge of CLK.

The largest possible initial count is 0; this is equivalent to 2^{16} for binary counting and 10^4 for BCD counting.

The Counter does not stop when it reaches zero. In Modes 0, 1, 4, and 5 the Counter "wraps around" to the highest count, either FFFF hex for binary counting or 9999 for BCD counting, and continues counting. Modes 2 and 3 are periodic; the Counter reloads itself with the initial count and continues counting from there.

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias	0°C to 70°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-0.5V to +7V
Power Dissipation	1W

*Notice: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

NOTICE: Specifications contained within the following tables are subject to change.

D.C. CHARACTERISTICS $T_A = 0^\circ\text{C}$ to 70°C , $V_{CC} = 5\text{V} \pm 10\%$

Symbol	Parameter	Min	Max	Units	Test Conditions
V_{IL}	Input Low Voltage	-0.5	0.8	V	
V_{IH}	Input High Voltage	2.0	$V_{CC} + 0.5\text{V}$	V	
V_{OL}	Output Low Voltage		0.45	V	$I_{OL} = 2.0\text{ mA}$
V_{OH}	Output High Voltage	2.4		V	$I_{OH} = -400\text{ }\mu\text{A}$
I_{IL}	Input Load Current		± 10	μA	$V_{IN} = V_{CC}$ to 0V
I_{OFL}	Output Float Leakage		± 10	μA	$V_{OUT} = V_{CC}$ to 0.45V
I_{CC}	V_{CC} Supply Current		170	mA	

CAPACITANCE $T_A = 25^\circ\text{C}$, $V_{CC} = \text{GND} = 0\text{V}$

Symbol	Parameter	Min	Max	Units	Test Conditions
C_{IN}	Input Capacitance		10	pF	$f_c = 1\text{ MHz}$
$C_{I/O}$	I/O Capacitance		20	pF	Unmeasured pins returned to V_{SS}

A.C. CHARACTERISTICS $T_A = 0^\circ\text{C}$ to 70°C , $V_{CC} = 5\text{V} \pm 10\%$, $\text{GND} = 0\text{V}$ **Bus Parameters(1)****READ CYCLE**

Symbol	Parameter	8254-5		8254		8254-2		Unit
		Min	Max	Min	Max	Min	Max	
t_{AR}	Address Stable Before $\overline{RD} \downarrow$	45		45		30		ns
t_{SR}	\overline{CS} Stable Before $\overline{RD} \downarrow$	0		0		0		ns
t_{RA}	Address Hold Time After $\overline{RD} \uparrow$	0		0		0		ns
t_{RR}	\overline{RD} Pulse Width	150		150		95		ns
t_{RD}	Data Delay from $\overline{RD} \downarrow$		120		120		85	ns
t_{AD}	Data Delay from Address		220		220		185	ns
t_{DF}	$\overline{RD} \uparrow$ to Data Floating	5	90	5	90	5	65	ns
t_{RV}	Command Recovery Time	200		200		165		ns

NOTE:

- AC timings measured at $V_{OH} = 2.0\text{V}$, $V_{OL} = 0.8\text{V}$.

A.C. CHARACTERISTICS $T_A = 0^\circ\text{C}$ to 70°C , $V_{CC} = 5V \pm 10\%$, GND = 0V (Continued)**WRITE CYCLE**

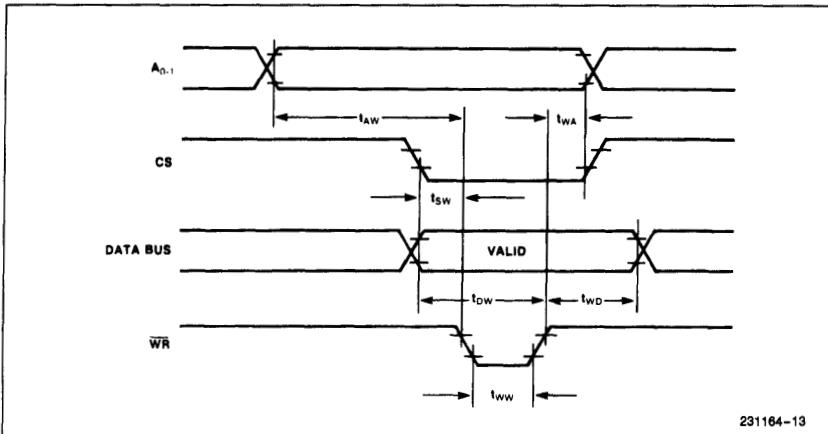
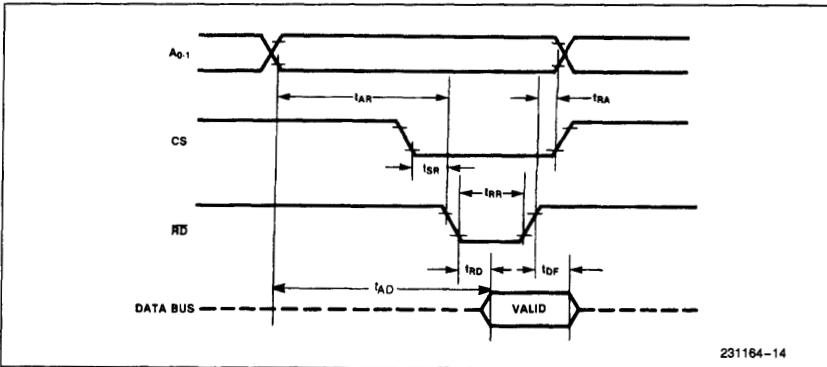
Symbol	Parameter	8254-5		8254		8254-2		Unit
		Min	Max	Min	Max	Min	Max	
t_{AW}	Address Stable Before $\overline{WR} \downarrow$	0		0		0		ns
t_{SW}	CS Stable Before $\overline{WR} \downarrow$	0		0		0		ns
t_{WA}	Address Hold Time After $\overline{WR} \downarrow$	0		0		0		ns
t_{WW}	\overline{WR} Pulse Width	150		150		95		ns
t_{DW}	Data Setup Time Before $\overline{WR} \uparrow$	120		120		95		ns
t_{WD}	Data Hold Time After $\overline{WR} \uparrow$	0		0		0		ns
t_{RV}	Command Recovery Time	200		200		165		ns

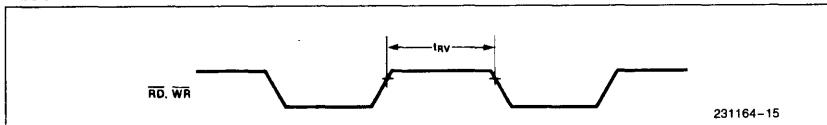
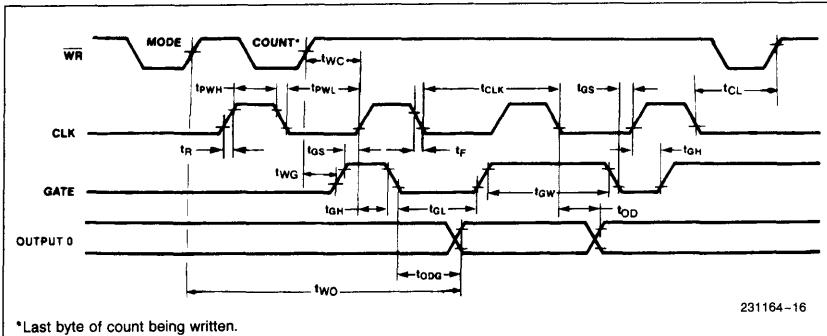
CLOCK AND GATE

Symbol	Parameter	8254-5		8254		8254-2		Unit
		Min	Max	Min	Max	Min	Max	
t_{CLK}	Clock Period	200	DC	125	DC	100	DC	ns
t_{PWH}	High Pulse Width	60(3)		60(3)		30(3)		ns
t_{PWL}	Low Pulse Width	60(3)		60(3)		50(3)		ns
t_R	Clock Rise Time		25		25		25	ns
t_F	Clock Fall Time		25		25		25	ns
t_{GW}	Gate Width High	50		50		50		ns
t_{GL}	Gate Width Low	50		50		50		ns
t_{GS}	Gate Setup Time to CLK \uparrow	50		50		40		ns
t_{GH}	Gate Setup Time After CLK \uparrow	50(2)		50(2)		50(2)		ns
t_{OD}	Output Delay from CLK \downarrow		150		150		100	ns
t_{ODG}	Output Delay from Gate \downarrow		120		120		100	ns
t_{WC}	CLK Delay for Loading \downarrow	0	55	0	55	0	55	ns
t_{WG}	Gate Delay for Sampling	-5	50	-5	50	-5	40	ns
t_{WO}	OUT Delay from Mode Write		260		260		240	ns
t_{CL}	CLK Set Up for Count Latch	-40	45	-40	45	-40	40	ns

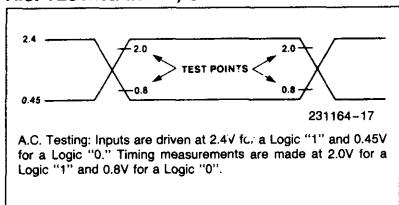
NOTES:

2. In Modes 1 and 5 triggers are sampled on each rising clock edge. A second trigger within 120 ns (70 ns for the 8254-2) of the rising clock edge may not be detected.
3. Low-going glitches that violate t_{PWH} , t_{PWL} may cause errors requiring counter reprogramming.

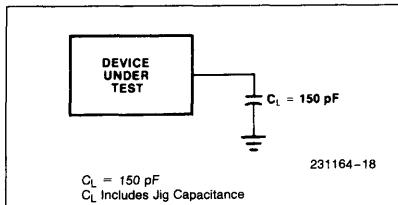
WAVEFORMS**WRITE****READ**

WAVEFORMS (Continued)**RECOVERY****CLOCK AND GATE**

*Last byte of count being written.

A.C. TESTING INPUT, OUTPUT WAVEFORM

A.C. Testing: Inputs are driven at 2.4V f.c. a Logic "1" and 0.45V for a Logic "0." Timing measurements are made at 2.0V for a Logic "1" and 0.8V for a Logic "0".

A.C. TESTING LOAD CIRCUIT

$C_L = 150 \text{ pF}$
 C_L Includes Jig Capacitance



8259A PROGRAMMABLE INTERRUPT CONTROLLER 8259A/8259A-2/8259A-8

- 8086, 8088 Compatible
- MCS-80®, MCS-85® Compatible
- Eight-Level Priority Controller
- Expandable to 64 Levels
- Programmable Interrupt Modes
- Individual Request Mask Capability
- Single + 5V Supply (No Clocks)
- 28-Pin Dual-In-Line Package
- Available in EXPRESS
 - Standard Temperature Range
 - Extended Temperature Range

The Intel 8259A Programmable Interrupt Controller handles up to eight vectored priority interrupts for the CPU. It is cascadable for up to 64 vectored priority interrupts without additional circuitry. It is packaged in a 28-pin DIP, uses NMOS technology and requires a single +5V supply. Circuitry is static, requiring no clock input.

The 8259A is designed to minimize the software and real time overhead in handling multi-level priority interrupts. It has several modes, permitting optimization for a variety of system requirements.

The 8259A is fully upward compatible with the Intel 8259. Software originally written for the 8259 will operate the 8259A in all 8259 equivalent modes (MCS-80/85, Non-Buffered, Edge Triggered).

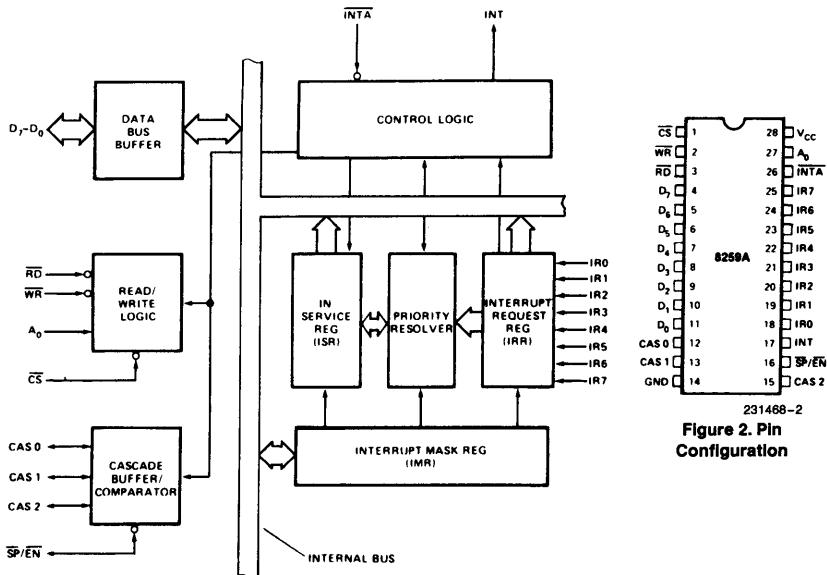


Figure 1. Block Diagram
Figure 2. Pin Configuration

Figure 1. Block Diagram

231468-1

Table 1. Pin Description

Symbol	Pin No.	Type	Name and Function
V _{CC}	28	I	SUPPLY: +5V Supply.
GND	14	I	GROUND
CS	1	I	CHIP SELECT: A low on this pin enables RD and WR communication between the CPU and the 8259A. INTA functions are independent of CS.
WR	2	I	WRITE: A low on this pin when CS is low enables the 8259A to accept command words from the CPU.
RD	3	I	READ: A low on this pin when CS is low enables the 8259A to release status onto the data bus for the CPU.
D ₇ -D ₀	4-11	I/O	BIDIRECTIONAL DATA BUS: Control, status and interrupt-vector information is transferred via this bus.
CAS ₀ -CAS ₂	12, 13, 15	I/O	CASCADE LINES: The CAS lines form a private 8259A bus to control a multiple 8259A structure. These pins are outputs for a master 8259A and inputs for a slave 8259A.
SP/EN	16	I/O	SLAVE PROGRAM/ENABLE BUFFER: This is a dual function pin. When in the Buffered Mode it can be used as an output to control buffer transceivers (EN). When not in the buffered mode it is used as an input to designate a master (SP = 1) or slave (SP = 0).
INT	17	O	INTERRUPT: This pin goes high whenever a valid interrupt request is asserted. It is used to interrupt the CPU, thus it is connected to the CPU's interrupt pin.
IR ₀ -IR ₇	18-25	I	INTERRUPT REQUESTS: Asynchronous inputs. An interrupt request is executed by raising an IR input (low to high), and holding it high until it is acknowledged (Edge Triggered Mode), or just by a high level on an IR input (Level Triggered Mode).
INTA	26	I	INTERRUPT ACKNOWLEDGE: This pin is used to enable 8259A interrupt-vector data onto the data bus by a sequence of interrupt acknowledge pulses issued by the CPU.
A ₀	27	I	AO ADDRESS LINE: This pin acts in conjunction with the CS, WR, and RD pins. It is used by the 8259A to decipher various Command Words the CPU writes and status the CPU wishes to read. It is typically connected to the CPU A0 address line (A1 for 8086, 8088).

FUNCTIONAL DESCRIPTION

Interrupts in Microcomputer Systems

Microcomputer system design requires that I.O. devices such as keyboards, displays, sensors and other components receive servicing in an efficient manner so that large amounts of the total system tasks can be assumed by the microcomputer with little or no effect on throughput.

The most common method of servicing such devices is the *Polling* approach. This is where the processor must test each device in sequence and in effect "ask" each one if it needs servicing. It is easy to see that a large portion of the main program is looping through this continuous polling cycle and that such a method would have a serious detrimental effect on system throughput, thus limiting the tasks that could be assumed by the microcomputer and reducing the cost effectiveness of using such devices.

A more desirable method would be one that would allow the microprocessor to be executing its main program and only stop to service peripheral devices when it is told to do so by the device itself. In effect, the method would provide an external asynchronous input that would inform the processor that it should complete whatever instruction that is currently being executed and fetch a new routine that will service the requesting device. Once this servicing is complete, however, the processor would resume exactly where it left off.

This method is called *Interrupt*. It is easy to see that system throughput would drastically increase, and thus more tasks could be assumed by the microcomputer to further enhance its cost effectiveness.

The Programmable Interrupt Controller (PIC) functions as an overall manager in an interrupt-driven system environment. It accepts requests from the peripheral equipment, determines which of the incoming requests is of the highest importance (priority), ascertains whether the incoming request has a higher priority value than the level currently being serviced, and issues an interrupt to the CPU based on this determination.

Each peripheral device or structure usually has a special program or "routine" that is associated with its specific functional or operational requirements; this is referred to as a "service routine". The PIC, after issuing an interrupt to the CPU, must somehow input information into the CPU that can "point" the Program Counter to the service routine associated with the requesting device. This "pointer" is an address in a vectoring table and will often be referred to, in this document, as vectoring data.

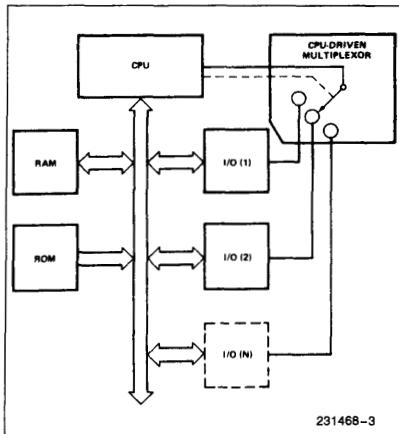


Figure 3a. Polled Method

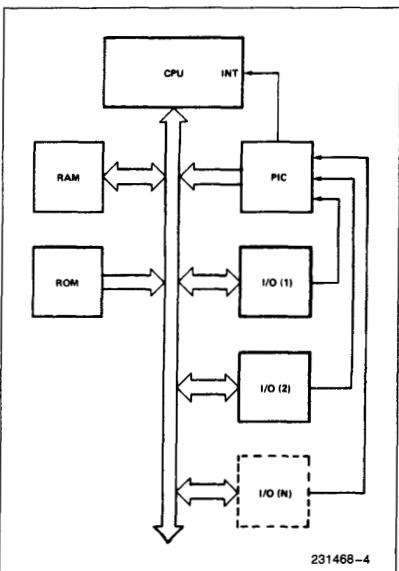


Figure 3b. Interrupt Method

The 8259A is a device specifically designed for use in real time, interrupt driven microcomputer systems. It manages eight levels or requests and has built-in features for expandability to other 8259A's (up to 64 levels). It is programmed by the system's software as an I/O peripheral. A selection of priority modes is available to the programmer so that the manner in which the requests are processed by the 8259A can be configured to match his system requirements. The priority modes can be changed or reconfigured dynamically at any time during the main program. This means that the complete interrupt structure can be defined as required, based on the total system environment.

INTERRUPT REQUEST REGISTER (IRR) AND IN-SERVICE REGISTER (ISR)

The interrupts at the IR input lines are handled by two registers in cascade, the Interrupt Request Register (IRR) and the In-Service Register (ISR). The IRR is used to store all the interrupt levels which are requesting service; and the ISR is used to store all the interrupt levels which are being serviced.

PRIORITY RESOLVER

This logic block determines the priorities of the bits set in the IRR. The highest priority is selected and strobed into the corresponding bit of the ISR during INTA pulse.

INTERRUPT MASK REGISTER (IMR)

The IMR stores the bits which mask the interrupt lines to be masked. The IMR operates on the IRR. Masking of a higher priority input will not affect the interrupt request lines of lower quality.

INT (INTERRUPT)

This output goes directly to the CPU interrupt input. The V_{OH} level on this line is designed to be fully compatible with the 8080A, 8085A and 8086 input levels.

INTA (INTERRUPT ACKNOWLEDGE)

INTA pulses will cause the 8259A to release vectoring information onto the data bus. The format of this data depends on the system mode (μ PM) of the 8259A.

DATA BUS BUFFER

This 3-state, bidirectional 8-bit buffer is used to interface the 8259A to the system Data Bus. Control words and status information are transferred through the Data Bus Buffer.

READ/WRITE CONTROL LOGIC

The function of this block is to accept OUTput commands from the CPU. It contains the Initialization Command Word (ICW) registers and Operation Command Word (OCW) registers which store the various control formats for device operation. This function block also allows the status of the 8259A to be transferred onto the Data Bus.

CS (CHIP SELECT)

A LOW on this input enables the 8259A. No reading or writing of the chip will occur unless the device is selected.

WR (WRITE)

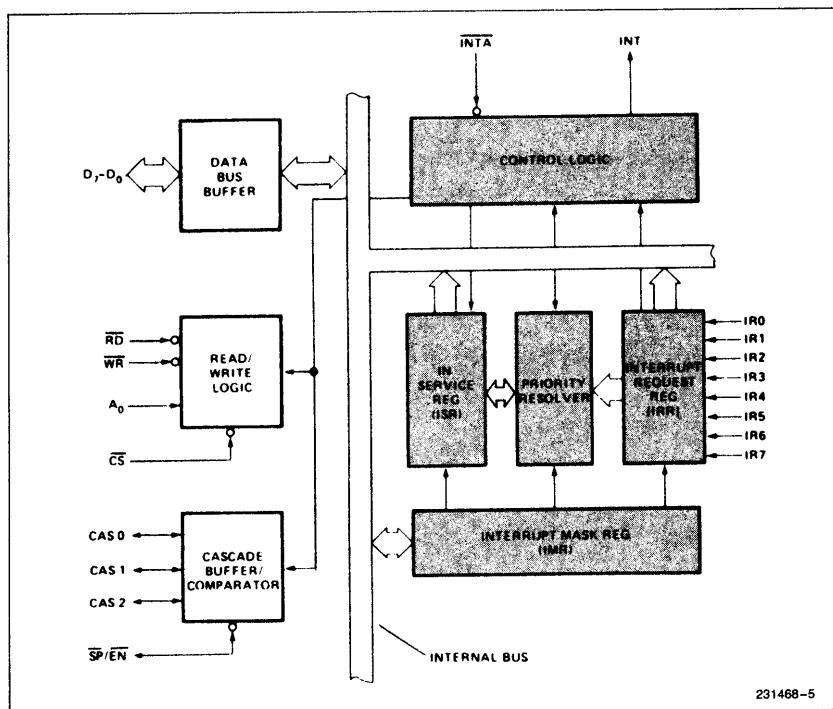
A LOW on this input enables the CPU to write control words (ICWs and OCWs) to the 8259A.

RD (READ)

A LOW on this input enables the 8259A to send the status of the Interrupt Request Register (IRR), In Service Register (ISR), the Interrupt Mask Register (IMR), or the interrupt level onto the Data Bus.

A₀

This input signal is used in conjunction with WR and RD signals to write commands into the various command registers, as well as reading the various status registers of the chip. This line can be tied directly to one of the address lines.



231468-5

Figure 4a. 8259A Block Diagram

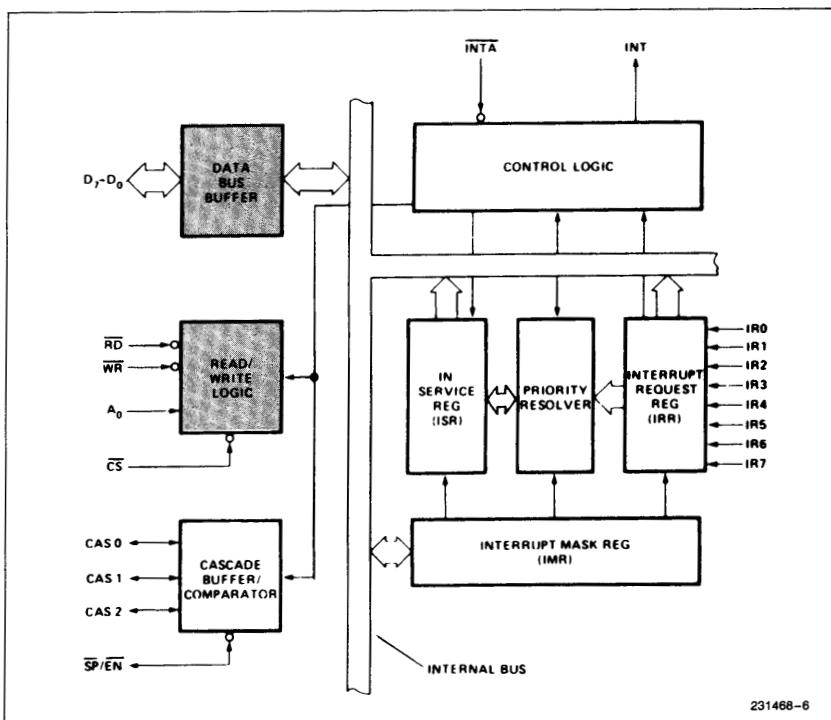


Figure 4b. 8259A Block Diagram

231468-6

THE CASCADE BUFFER/COMPARATOR

This function block stores and compares the IDs of all 8259A's used in the system. The associated three I/O pins (CAS0-2) are outputs when the 8259A is used as a master and are inputs when the 8259A is used as a slave. As a master, the 8259A sends the ID of the interrupting slave device onto the CAS0-2 lines. The slave thus selected will send its preprogrammed subroutine address onto the Data Bus during the next one or two consecutive INTA pulses. (See section "Cascading the 8259A".)

INTERRUPT SEQUENCE

The powerful features of the 8259A in a microcomputer system are its programmability and the interrupt routine addressing capability. The latter allows direct or indirect jumping to the specific interrupt routine requested without any polling of the interrupting devices. The normal sequence of events during an interrupt depends on the type of CPU being used.

The events occur as follows in an MCS-80/85 system:

1. One or more of the INTERRUPT REQUEST lines (IR7-0) are raised high, setting the corresponding IRR bit(s).
2. The 8259A evaluates these requests, and sends an INT to the CPU, if appropriate.
3. The CPU acknowledges the INT and responds with an INTA pulse.
4. Upon receiving an INTA from the CPU group, the highest priority ISR bit is set, and the corresponding IRR bit is reset. The 8259A will also release a CALL instruction code (11001101) onto the 8-bit Data Bus through its D7-0 pins.

5. This CALL instruction will initiate two more INTA pulses to be sent to the 8259A from the CPU group.
6. These two INTA pulses allow the 8259A to release its preprogrammed subroutine address onto the Data Bus. The lower 8-bit address is released at the first INTA pulse and the higher 8-bit address is released at the second INTA pulse.
7. This completes the 3-byte CALL instruction released by the 8259A. In the AEOI mode the ISR bit is reset at the end of the third INTA pulse. Otherwise, the ISR bit remains set until an appropriate EOI command is issued at the end of the interrupt sequence.

The events occurring in an 8086 system are the same until step 4.

4. Upon receiving an INTA from the CPU group, the highest priority ISR bit is set and the corresponding IRR bit is reset. The 8259A does not drive the Data Bus during this cycle.
5. The 8086 will initiate a second INTA pulse. During this pulse, the 8259A releases an 8-bit pointer onto the Data Bus where it is read by the CPU.
6. This completes the interrupt cycle. In the AEOI mode the ISR bit is reset at the end of the second INTA pulse. Otherwise, the ISR bit remains set until an appropriate EOI command is issued at the end of the interrupt subroutine.

If no interrupt request is present at step 4 of either sequence (i.e., the request was too short in duration) the 8259A will issue an interrupt level 7. Both the vectoring bytes and the CAS lines will look like an interrupt level 7 was requested.

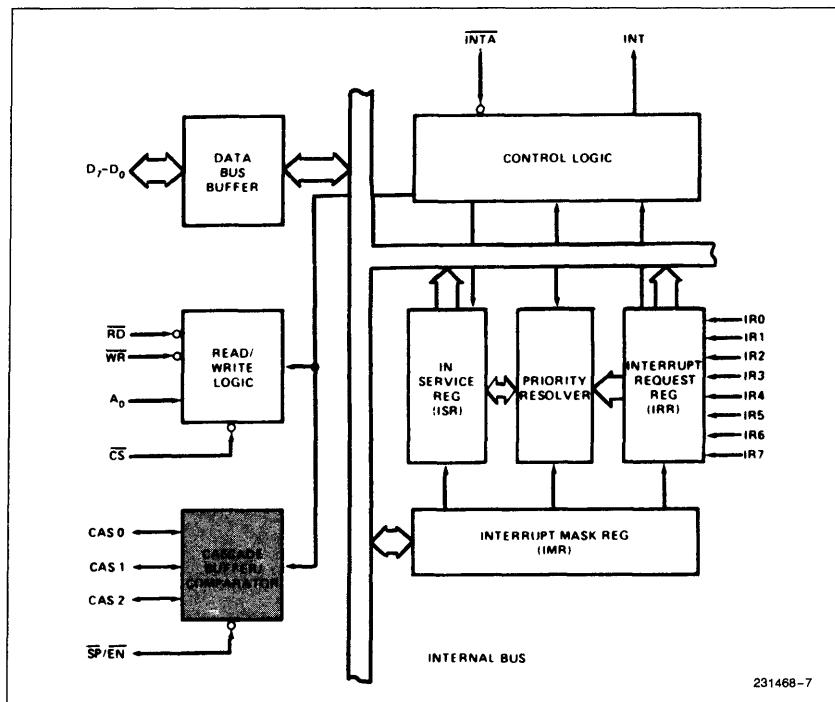


Figure 4c. 8259A Block Diagram

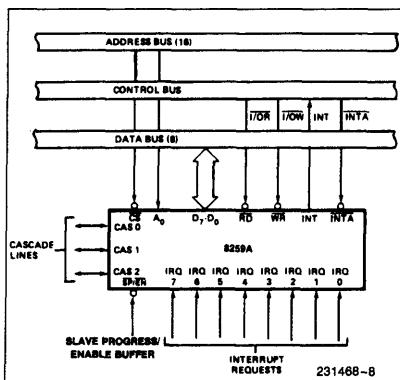


Figure 5. 8259A Interface to Standard System Bus

INTERRUPT SEQUENCE OUTPUTS**MCS-80®, MCS-85®**

This sequence is timed by three **INTA** pulses. During the first **INTA** pulse the CALL opcode is enabled onto the data bus.

Content of First Interrupt Vector Byte

D7	D6	D5	D4	D3	D2	D1	D0
CALL CODE							

1 1 0 0 1 1 0 1

During the second **INTA** pulse the lower address of the appropriate service routine is enabled onto the data bus. When Interval = 4 bits A₅-A₇ are programmed, while A₀-A₄ are automatically inserted by the 8259A. When Interval = 8 only A₆ and A₇ are programmed, while A₀-A₅ are automatically inserted.

Content of Second Interrupt Vector Byte

IR	Interval = 4							
	D7	D6	D5	D4	D3	D2	D1	D0
7	A7	A6	A5	1	1	1	0	0
6	A7	A6	A5	1	1	0	0	0
5	A7	A6	A5	1	0	1	0	0
4	A7	A6	A5	1	0	0	0	0
3	A7	A6	A5	0	1	1	0	0
2	A7	A6	A5	0	1	0	0	0
1	A7	A6	A5	0	0	1	0	0
0	A7	A6	A5	0	0	0	0	0

IR	Interval = 8							
	D7	D6	D5	D4	D3	D2	D1	D0
7	A7	A6	1	1	1	0	0	0
6	A7	A6	1	1	0	0	0	0
5	A7	A6	1	0	1	0	0	0
4	A7	A6	1	0	0	0	0	0
3	A7	A6	0	1	1	0	0	0
2	A7	A6	0	1	0	0	0	0
1	A7	A6	0	0	1	0	0	0
0	A7	A6	0	0	0	0	0	0

During the third INTA pulse the higher address of the appropriate service routine, which was programmed as byte 2 of the initialization sequence (A₈-A₁₅), is enabled onto the bus.

Content of Third Interrupt Vector Byte

D7	D6	D5	D4	D3	D2	D1	D0
A15	A14	A13	A12	A11	A10	A9	A8

8086, 8088

8086 mode is similar to MCS-80 mode except that only two interrupt acknowledge cycles are issued by the processor and no CALL opcode is sent to the processor. The first interrupt acknowledge cycle is similar to that of MCS-80, 85 systems in that the 8259A uses it to internally freeze the state of the interrupts for priority resolution and as a master it issues the interrupt code on the cascade lines at the end of the INTA pulse. On this first cycle it does not issue any data to the processor and leaves its data bus buffers disabled. On the second interrupt acknowledge cycle in 8086 mode the master (or slave if so programmed) will send a byte of data to the processor with the acknowledged interrupt code

composed as follows (note the state of the AD1 mode control is ignored and A₅-A₁₁ are unused in 8086 mode):

**Content of Interrupt Vector Byte
for 8086 System Mode**

	D7	D6	D5	D4	D3	D2	D1	D0
IR7	T7	T6	T5	T4	T3	1	1	1
IR6	T7	T6	T5	T4	T3	1	1	0
IR5	T7	T6	T5	T4	T3	1	0	1
IR4	T7	T6	T5	T4	T3	1	0	0
IR3	T7	T6	T5	T4	T3	0	1	1
IR2	T7	T6	T5	T4	T3	0	1	0
IR1	T7	T6	T5	T4	T3	0	0	1
IR0	T7	T6	T5	T4	T3	0	0	0

PROGRAMMING THE 8259A

The 8259A accepts two types of command words generated by the CPU:

1. *Initialization Command Words (ICWs)*: Before normal operation can begin, each 8259A in the system must be brought to a starting point—by a sequence of 2 to 4 bytes timed by WR pulses.

2. *Operation Command Words (OCWs)*: These are the command words which command the 8259A to operate in various interrupt modes. These modes are:

- a. Fully nested mode
- b. Rotating priority mode
- c. Special mask mode
- d. Polled mode

The OCWs can be written into the 8259A anytime after initialization.

INITIALIZATION COMMAND WORDS (ICWS)**General**

Whenever a command is issued with A₀ = 0 and D₄ = 1, this is interpreted as Initialization Command Word 1 (ICW1). ICW1 starts the initialization sequence during which the following automatically occur.

- a. The edge sense circuit is reset, which means that following initialization, an interrupt request (IR) input must make a low-to-high transition to generate an interrupt.

- b. The Interrupt Mask Register is cleared.
- c. IR7 input is assigned priority 7.
- d. The slave mode address is set to 7.
- e. Special Mask Mode is cleared and Status Read is set to IRR.
- f. If IC4 = 0, then all functions selected in ICW4 are set to zero. (Non-Buffered mode*, no Auto-EOI, MCS-80, 85 system).

***NOTE:**

Master/Slave in ICW4 is only used in the buffered mode.

Initialization Command Words 1 and 2 (ICW1, ICW2)

A₆-A₁₅: *Page starting address of service routines.* In an MCS 80/85 system, the 8 request levels will generate CALLS to 8 locations equally spaced in memory. These can be programmed to be spaced at intervals of 4 or 8 memory locations, thus the 8 routines will occupy a page of 32 or 64 bytes, respectively.

The address format is 2 bytes long (A₀-A₁₅). When the routine interval is 4, A₀-A₄ are automatically inserted by the 8259A, while A₅-A₁₅ are programmed externally. When the routine interval is 8, A₀-A₅ are automatically inserted by the 8259A, while A₆-A₁₅ are programmed externally.

The 8-byte interval will maintain compatibility with current software, while the 4-byte interval is best for a compact jump table.

In an 8086 system A₁₅-A₁₁ are inserted in the five most significant bits of the vectoring byte and the 8259A sets the three least significant bits according to the interrupt level. A₁₀-A₅ are ignored and ADI (Address interval) has no effect.

LTIM: If LTIM = 1, then the 8259A will operate in the level interrupt mode. Edge detect logic on the interrupt inputs will be disabled.

ADI: CALL address interval. ADI = 1 then interval = 4; ADI = 0 then interval = 8.

SNGL: Single. Means that this is the only 8259A in the system. If SNGL = 1 no ICW3 will be issued.

IC4: If this bit is set—ICW4 has to be read. If ICW4 is not needed, set IC4 = 0.

case SNGL = 0. It will load the 8-bit slave register. The functions of this register are:

- a. In the master mode (either when SP = 1, or in buffered mode when M/S = 1 in ICW4) a "1" is set for each slave in the system. The master then will release byte 1 of the call sequence (for MCS-80/85 system) and will enable the corresponding slave to release bytes 2 and 3 (for 8086 only byte 2) through the cascade lines.

- b. In the slave mode (either when \overline{SP} = 0, or if BUF = 1 and M/S = 0 in ICW4) bits 2-0 identify the slave. The slave compares its cascade input with these bits and, if they are equal, bytes 2 and 3 of the call sequence (or just byte 2 for 8086) are released by it on the Data Bus.

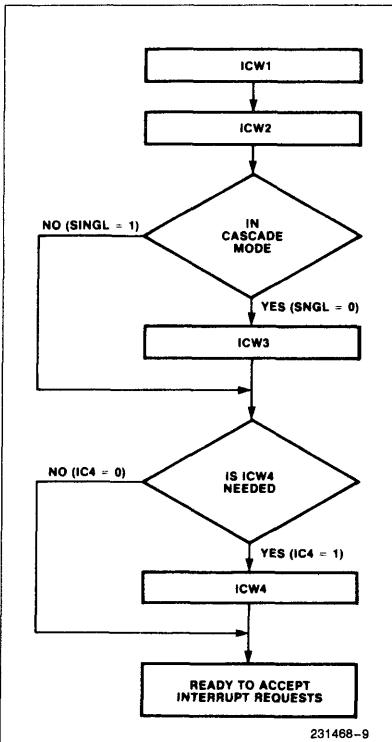


Figure 6. Initialization Sequence

Initialization Command Word 3 (ICW3)

This word is read only when there is more than one 8259A in the system and cascading is used, in which

Initialization Command Word 4 (ICW4)

SFNM: If SFNM = 1 the special fully nested mode is programmed.

BUF: If BUF = 1 the buffered mode is programmed. In buffered mode SP/EN becomes an enable output and the master/slave determination is by M/S.

M/S: If buffered mode is selected: M/S = 1 means the 8259A is programmed to be a

master, M/S = 0 means the 8259A is programmed to be a slave. If BUF = 0, M/S has no function.

AEOI: If AEOI = 1 the automatic end of interrupt mode is programmed.

μ PM: Microprocessor mode: μ PM = 0 sets the 8259A for MCS-80, 85 system operation, μ PM = 1 sets the 8259A for 8086 system operation.

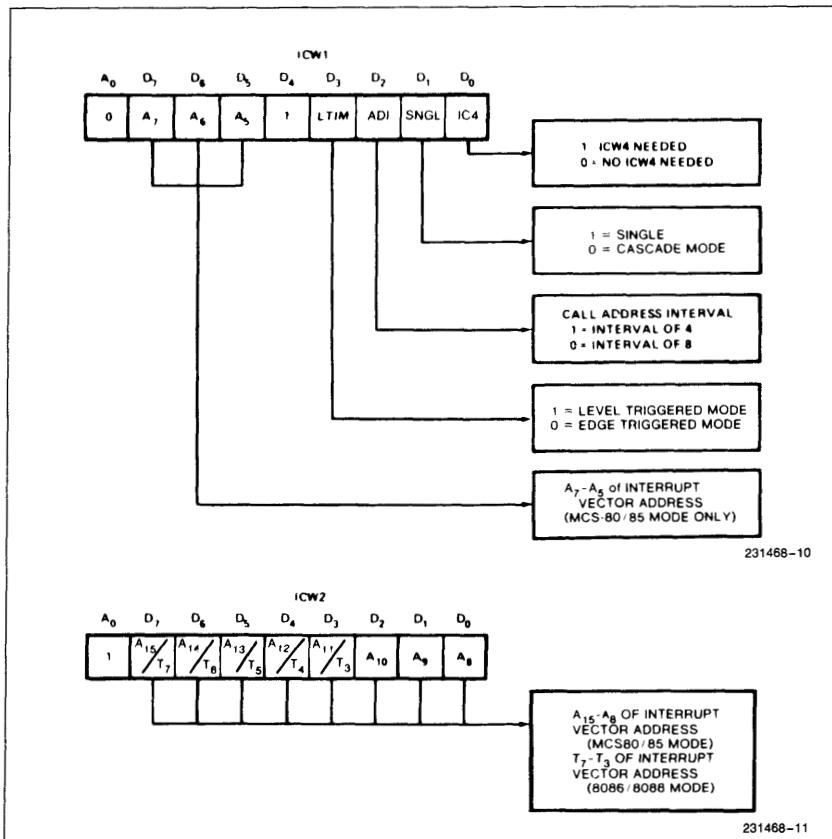
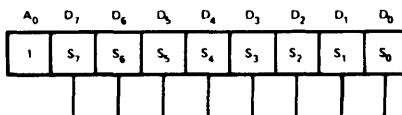


Figure 7. Initialization Command Word Format

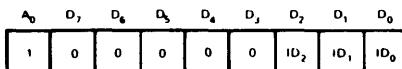
ICW3 (MASTER DEVICE)



1 = IR INPUT HAS A SLAVE
0 = IR INPUT DOES NOT HAVE
A SLAVE

231468-12

ICW3 (SLAVE DEVICE)

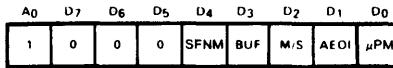


SLAVE ID[1]

0	1	2	3	4	5	6	7
0	1	0	1	0	1	0	1
0	0	1	1	0	0	1	1
0	0	0	0	1	1	1	1

231468-13

ICW4



1 = 8086/8088 MODE
0 = MCS-80/85 MODE

1 = AUTO EOI
0 = NORMAL EOI

0	X	-- NON BUFFERED MODE
1	0	-- BUFFERED MODE/SLAVE
1	1	-- BUFFERED MODE/MASTER

1 = SPECIAL FULLY NESTED
MODE
0 = NOT SPECIAL FULLY
NESTED MODE

231468-14

NOTE:

Slave ID is equal to the corresponding master IR input.

Figure 7. Initialization Command Word Format (Continued)

OPERATION COMMAND WORDS (OCWs)

After the Initialization Command Words (ICWs) are programmed into the 8259A, the chip is ready to accept interrupt requests at its input lines. However, during the 8259A operation, a selection of algorithms can command the 8259A to operate in various modes through the Operation Command Words (OCWs).

Operation Control Words (OCWs)

OCW1									
A0	D7	D6	D5	D4	D3	D2	D1	D0	
1	M7	M6	M5	M4	M3	M2	M1	M0	
OCW2									
0	R	SL	EOI	0	0	L2	L1	L0	
OCW3									
0	0	ESMM	SMM	0	1	P	RR	RIS	

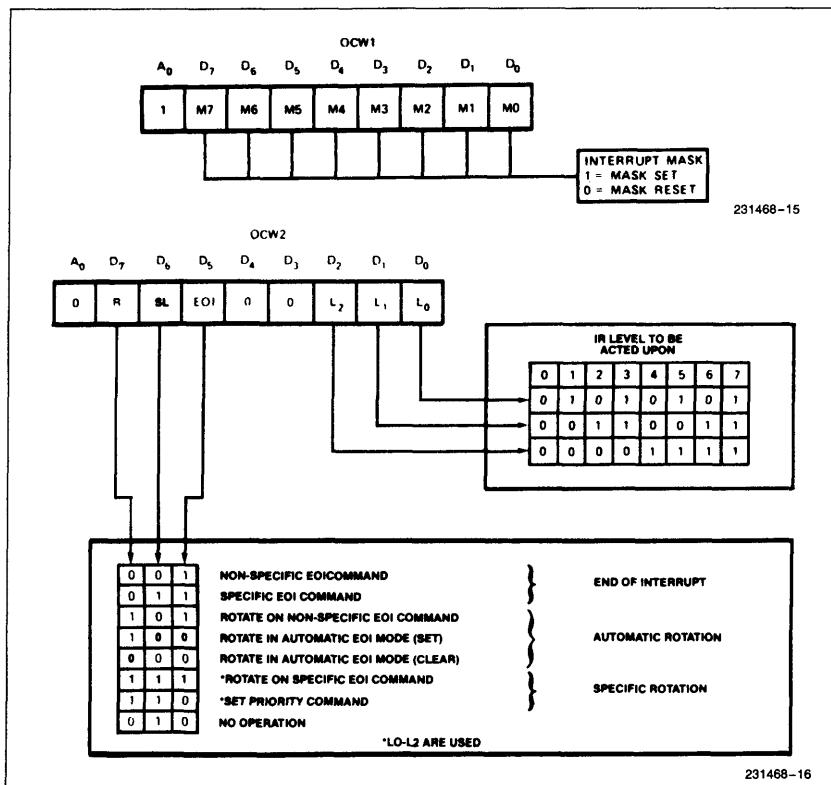


Figure 8. Operation Command Word Format

Operation Control Word 1 (OCW1)

OCW1 sets and clears the mask bits in the interrupt Mask Register (IMR). M₇-M₀ represent the eight mask bits. M = 1 indicates the channel is masked (inhibited), M = 0 indicates the channel is enabled.

Operation Control Word 2 (OCW2)

R, SL, EOI—These three bits control the Rotate and End of Interrupt modes and combinations of the two. A chart of these combinations can be found on the Operation Command Word Format.

L₂, L₁, L₀—These bits determine the interrupt level acted upon when the SL bit is active.

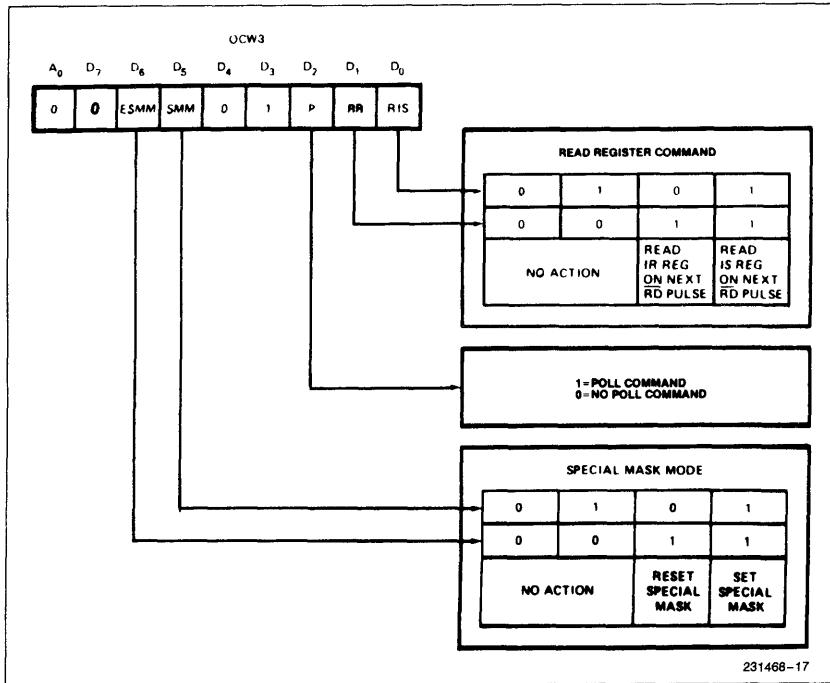


Figure 8. Operation Command Word Format (Continued)

231468-17

Operation Control Word 3 (OCW3)

ESMM—Enable Special Mask Mode. When this bit is set to 1 it enables the SMM bit to set or reset the Special Mask Mode. When ESMM = 0 the SMM bit becomes a "don't care".

SMM—Special Mask Mode. If ESMM = 1 and SMM = 1 the 8259A will enter Special Mask Mode. If ESMM = 1 and SMM = 0 the 8259A will revert to normal mask mode. When ESMM = 0, SMM has no effect.

Fully Nested Mode

This mode is entered after initialization unless another mode is programmed. The interrupt requests are ordered in priority from 0 through 7 (0 highest). When an interrupt is acknowledged the highest priority request is determined and its vector placed on the bus. Additionally, a bit of the Interrupt Service register (ISC-7) is set. This bit remains set until the microprocessor issues an End of Interrupt (EOI) command immediately before returning from the service routine, or if AEOI (Automatic End of Interrupt) bit is set, until the trailing edge of the last INTA. While the IS bit is set, all further interrupts of the same or lower priority are inhibited, while higher levels will generate an interrupt (which will be acknowledged only if the microprocessor internal Interrupt enable flip-flop has been re-enabled through software).

After the initialization sequence, IR0 has the highest priority and IR7 the lowest. Priorities can be changed, as will be explained, in the rotating priority mode.

End of Interrupt (EOI)

The In Service (IS) bit can be reset either automatically following the trailing edge of the last in sequence INTA pulse (when AEOI bit in ICW1 is set) or by a command word that must be issued to the 8259A before returning from a service routine (EOI command). An EOI command must be issued twice if in the Cascade mode, once for the master and once for the corresponding slave.

There are two forms of EOI command: Specific and Non-Specific. When the 8259A is operated in modes which preserve the fully nested structure, it can determine which IS bit to reset on EOI. When a Non-Specific EOI command is issued the 8259A will automatically reset the highest IS bit of those that are set, since in the fully nested mode the highest IS level was necessarily the last level acknowledged and serviced. A non-specific EOI can be issued with OCW2 (EOI = 1, SL = 0, R = 0).

When a mode is used which may disturb the fully nested structure, the 8259A may no longer be able to determine the last level acknowledged. In this case a Specific End of Interrupt must be issued which includes as part of the command the IS level to be reset. A specific EOI can be issued with OCW2 (EOI = 1, SL = 1, R = 0, and L0-L2 is the binary level of the IS bit to be reset).

It should be noted that an IS bit that is masked by an IMR bit will not be cleared by a non-specific EOI if the 8259A is in the Special Mask Mode.

Automatic End of Interrupt (AEOI) Mode

If AEOI = 1 in ICW4, then the 8259A will operate in AEOI mode continuously until reprogrammed by ICW4. In this mode the 8259A will automatically perform a non-specific EOI operation at the trailing edge of the last interrupt acknowledge pulse (third pulse in MCS-80/85, second in 8086). Note that from a system standpoint, this mode should be used only when a nested multilevel interrupt structure is not required within a single 8259A.

The AEOI mode can only be used in a master 8259A and not a slave.

Automatic Rotation (Equal Priority Devices)

In some applications there are a number of interrupting devices of equal priority. In this mode a device, after being serviced, receives the lowest priority, so a device requesting an interrupt will have to wait, in the worst case until each of 7 other devices are serviced at most once. For example, if the priority and "in service" status is:

Before Rotate (IR4 the highest priority requiring service)

IS7	IS6	IS5	IS4	IS3	IS2	IS1	IS0
0	1	0	1	0	0	0	0

"IS" Status 231468-18

Priority Status							
Lowest Priority				Highest Priority			
7	6	5	4	3	2	1	0

Priority Status 231468-19

After Rotate (IR4 was serviced, all other priorities rotated correspondingly)

IS7	IS6	IS5	IS4	IS3	IS2	IS1	IS0
0	1	0	0	0	0	0	0

"IS" Status 231468-20

Priority Status							
Highest Priority				Lowest Priority			
2	1	0	7	6	5	4	3

Priority Status 231468-21

There are two ways to accomplish Automatic Rotation using OCW2, the Rotation on Non-Specific EOI Command ($R = 1, SL = 0, EOI = 1$) and the Rotate in Automatic EOI Mode which is set by ($R = 1, SL = 0, EOI = 0$) and cleared by ($R = 0, SL = 0, EOI = 0$).

Specific Rotation (Specific Priority)

The programmer can change priorities by programming the bottom priority and thus fixing all other priorities; i.e., if IR5 is programmed as the bottom priority device, then IR6 will have the highest one.

The Set Priority command is issued in OCW2 where: $R = 1, SL = 1, L0-L2$ is the binary priority level code of the bottom priority device.

Observe that in this mode internal status is updated by software control during OCW2. However, it is independent of the End of Interrupt (EOI) command (also executed by OCW2). Priority changes can be executed during an EOI command by using the Rotate on Specific EOI command in OCW2 ($R = 1, SL = 1, EOI = 1$ and $L0-L2$ = IR level to receive bottom priority).

Interrupt Masks

Each Interrupt Request input can be masked individually by the Interrupt Mask Register (IMR) programmed through OCW1. Each bit in the IMR masks one interrupt channel if it is set (1). Bit 0 masks IR0, Bit 1 masks IR1 and so forth. Masking an IR channel does not affect the other channels operation.

Special Mask Mode

Some applications may require an interrupt service routine to dynamically alter the system priority struc-

ture during its execution under software control. For example, the routine may wish to inhibit lower priority requests for a portion of its execution but enable some of them for another portion.

The difficulty here is that if an Interrupt Request is acknowledged and an End of Interrupt command did not reset its IS bit (i.e., while executing a service routine), the 8259A would have inhibited all lower priority requests with no easy way for the routine to enable them.

That is where the Special Mask Mode comes in. In the special Mask Mode, when a mask bit is set in OCW1, it inhibits further interrupts at that level *and enables* interrupts from *all other* levels (lower as well as higher) that are not masked.

Thus, any interrupts may be selectively enabled by loading the mask register.

The special Mask Mode is set by OWC3 where: SSSM = 1, SMM = 1, and cleared where SSSM = 1, SMM = 0.

Poll Command

In this mode the INT output is not used or the microprocessor internal Interrupt Enable flip-flop is reset, disabling its interrupt input. Service to devices is achieved by software using a Poll command.

The Poll command is issued by setting P = '1' in OWC3. The 8259A treats the next RD pulse to the 8259A (i.e., $\overline{RD} = 0, \overline{CS} = 0$) as an interrupt acknowledge, sets the appropriate IS bit if there is a request, and reads the priority level. Interrupt is frozen from WR to RD.

The word enabled onto the data bus during RD is:

D7	D6	D5	D4	D3	D2	D1	D0
----	----	----	----	----	----	----	----

I	—	—	—	—	W2	W1	W0
---	---	---	---	---	----	----	----

W0-W2: Binary code of the highest priority level requesting service.
I: Equal to "1" if there is an interrupt.

This mode is useful if there is a routine command common to several levels so that the INTA sequence is not needed (saves ROM space). Another application is to use the poll mode to expand the number of priority levels to more than 64.

Reading the 8259A Status

The input status of several internal registers can be read to update the user information on the system.

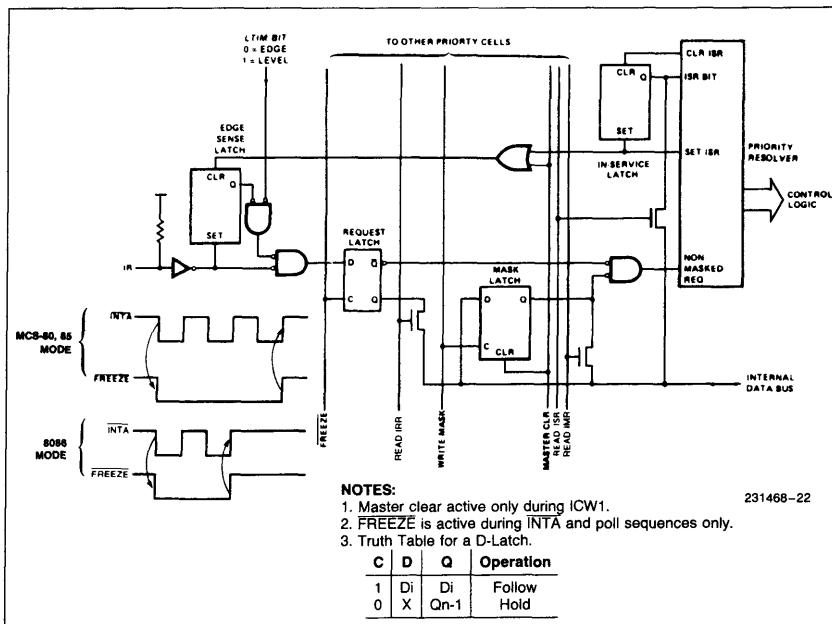


Figure 9. Priority Cell—Simplified Logic Diagram

The following registers can be read via OCW3 (IRR and ISR or OCW1 [IMR]).

Interrupt Request Register (IRR): 8-bit register which contains the levels requesting an interrupt to be acknowledged. The highest request level is reset from the IRR when an interrupt is acknowledged. (Not affected by IMR.)

In-Service Register (ISR): 8-bit register which contains the priority levels that are being serviced. The ISR is updated when an End of Interrupt Command is issued.

Interrupt Mask Register: 8-bit register which contains the interrupt request lines which are masked.

The IRR can be read when, prior to the RD pulse, a Read Register Command is issued with OCW3 (RR = 1, RIS = 0.)

The ISR can be read, when, prior to the RD pulse, a Read Register Command is issued with OCW3 (RR = 1, RIS = 1).

There is no need to write an OCW3 before every status read operation, as long as the status read corresponds with the previous one; i.e., the 8259A "remembers" whether the IRR or ISR has been previously selected by the OCW3. This is not true when poll is used.

After initialization the 8259A is set to IRR.

For reading the IMR, no OCW3 is needed. The output data bus will contain the IMR whenever RD is active and A0 = 1 (OCW1).

Polling overrides status read when P = 1, RR = 1 in OCW3.

Edge and Level Triggered Modes

This mode is programmed using bit 3 in ICW1.

If LTIM = '0', an interrupt request will be recognized by a low to high transition on an IR input. The IR input can remain high without generating another interrupt.

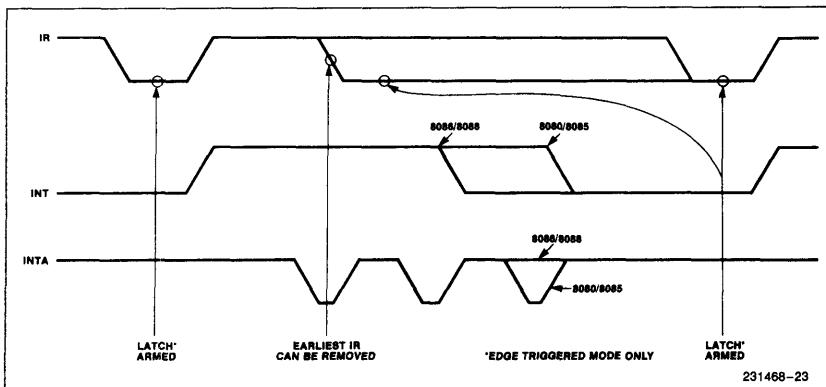


Figure 10. IR Triggering Timing Requirements

If LTIM = '1', an interrupt request will be recognized by a 'high' level on IR input, and there is no need for an edge detection. The interrupt request must be removed before the EOI command is issued or the CPU interrupts is enabled to prevent a second interrupt from occurring.

The priority cell diagram shows a conceptual circuit of the level sensitive and edge sensitive input circuitry of the 8259A. Be sure to note that the request latch is a transparent D type latch.

In both the edge and level triggered modes the IR inputs must remain high until after the falling edge of the first INTA. If the IR input goes low before this time a DEFAULT IR7 will occur when the CPU acknowledges the interrupt. This can be a useful safeguard for detecting interrupts caused by spurious noise glitches on the IR inputs. To implement this feature the IR7 routine is used for "clean up" simply executing a return instruction, thus ignoring the interrupt. If IR7 is needed for other purposes a default IR7 can still be detected by reading the ISR. A normal IR7 interrupt will set the corresponding ISR bit, a default IR7 won't. If a default IR7 routine occurs during a normal IR7 routine, however, the ISR will remain set. In this case it is necessary to keep track of whether or not the IR7 routine was previously entered. If another IR7 occurs it is a default.

The Special Fully Nest Mode

This mode will be used in the case of a big system where cascading is used, and the priority has to be conserved within each slave. In this case the fully nested mode will be programmed to the master (us-

ing ICW4). This mode is similar to the normal nested mode with the following exceptions:

- When an interrupt request from a certain slave is in service this slave is not locked out from the master's priority logic and further interrupt requests from higher priority IR's within the slave will be recognized by the master and will initiate interrupts to the processor. (In the normal nested mode a slave is masked out when its request is in service and no higher requests from the same slave can be serviced.)
- When exiting the Interrupt Service routine the software has to check whether the interrupt serviced was the only one from that slave. This is done by sending a non-specific End of Interrupt (EOI) command to the slave and then reading its In-Service register and checking for zero. If it is empty, a non-specific EOI can be sent to the master too. If not, no EOI should be sent.

Buffered Mode

When the 8259A is used in a large system where bus driving buffers are required on the data bus and the cascading mode is used, there exists the problem of enabling buffers.

The buffered mode will structure the 8259A to send an enable signal on SP/EN to enable the buffers. In this mode, whenever the 8259A's data bus outputs are enabled, the SP/EN output becomes active.

This modification forces the use of software programming to determine whether the 8259A is a master or a slave. Bit 3 in ICW4 programs the buffered mode, and bit 2 in ICW4 determines whether it is a master or a slave.

CASCADE MODE

The 8259A can be easily interconnected in a system of one master with up to eight slaves to handle up to 64 priority levels.

The master controls the slaves through the 3 line cascade bus. The cascade bus acts like chip selects to the slaves during the INTA sequence.

In a cascade configuration, the slave interrupt outputs are connected to the master interrupt request inputs. When a slave request line is activated and afterwards acknowledged, the master will enable the corresponding slave to release the device routine address during bytes 2 and 3 of INTA. (Byte 2 only for 8086/8088).

The cascade bus lines are normally low and will contain the slave address code from the trailing edge of the first INTA pulse to the trailing edge of the third pulse. Each 8259A in the system must follow a separate initialization sequence and can be programmed to work in a different mode. An EOI command must be issued twice: once for the master and once for the corresponding slave. An address decoder is required to activate the Chip Select (CS) input of each 8259A.

The cascade lines of the Master 8259A are activated only for slave inputs, non-slave inputs leave the cascade line inactive (low).

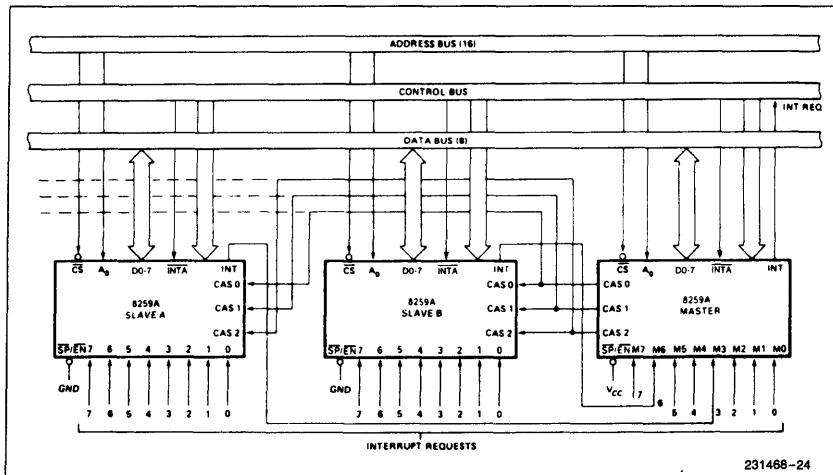


Figure 11. Cascading the 8259A

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias 0°C to 70°C
 Storage Temperature -65°C to +150°C
 Voltage on Any Pin
 with Respect to Ground -0.5V to +7V
 Power Dissipation 1W

**Notice: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.*

D. C. CHARACTERISTICS ($T_A = 0^\circ\text{C}$ to 70°C , $V_{CC} = 5\text{ V} \pm 5\%$ (8259A-8), $V_{CC} = 5\text{ V} \pm 10\%$ (8259A, 8259A-2))

Symbol	Parameter	Min	Max	Units	Test Conditions
V_{IL}	Input Low Voltage	-0.5	0.8	V	
V_{IH}	Input High Voltage	2.0*	$V_{CC} + 0.5\text{V}$	V	
V_{OL}	Output Low Voltage		0.45	V	$I_{OL} = 2.2\text{ mA}$
V_{OH}	Output High Voltage	2.4		V	$I_{OH} = -400\text{ }\mu\text{A}$
$V_{OH(INT)}$	Interrupt Output High Voltage	3.5		V	$I_{OH} = -100\text{ }\mu\text{A}$
		2.4		V	$I_{OH} = -400\text{ }\mu\text{A}$
I_{LI}	Input Load Current	-10	+10	μA	$0\text{V} \leq V_{IN} \leq V_{CC}$
I_{LOL}	Output Leakage Current	-10	+10	μA	$0.45\text{V} \leq V_{OUT} \leq V_{CC}$
I_{CC}	V_{CC} Supply Current		85	mA	
I_{LIR}	IR Input Load Current		-300	μA	$V_{IN} = 0$
			10	μA	$V_{IN} = V_{CC}$

***NOTE:**

For Extended Temperature EXPRESS $V_{IH} = 2.3\text{V}$.

CAPACITANCE $T_A = 25^\circ\text{C}$; $V_{CC} = \text{GND} = 0\text{V}$

Symbol	Parameter	Min	Typ	Max	Unit	Test Conditions
C_{IN}	Input Capacitance			10	pF	$f_c = 1\text{ MHz}$
$C_{I/O}$	I/O Capacitance			20	pF	Unmeasured Pins Returned to V_{SS}

A.C. CHARACTERISTICS

$T_A = 0^\circ\text{C}$ to 70°C , $V_{CC} = 5\text{V} \pm 5\%$ (8259 A-8), $V_{CC} = 5\text{V} \pm 10\%$ (8259A, 8259A-2)

TIMING REQUIREMENTS

Symbol	Parameter	8259A-8		8259A		8259A-2		Units	Test Conditions
		Min	Max	Min	Max	Min	Max		
TAHRL	AO/CS Setup to RD/INTA ↓	50		0		0		ns	
TRHAX	AO/CS Hold after RD/INTA ↑	5		0		0		ns	
TRLRH	RD Pulse Width	420		235		160		ns	
TAHWL	AO/CS Setup to WR ↓	50		0		0		ns	
TWHAX	AO/CS Hold after WR ↑	20		0		0		ns	
TWLWH	WR Pulse Width	400		290		190		ns	
TDVWH	Data Setup to WR ↑	300		240		160		ns	
TWHDX	Data Hold after WR ↑	40		0		0		ns	
TJLJH	Interrupt Request Width (Low)	100		100		100		ns	See Note 1
TCVIAL	Cascade Setup to Second or Third INTA ↓ (Slave Only)	55		55		40		ns	
TRHRL	End of RD to Next RD End of INTA to Next INTA within an INTA Sequence Only	160		160		160		ns	
TWHWL	End of WR to Next WR	190		190		190		ns	
*TCHCL	End of Command to Next Command (Not Same Command Type)	500		500		500		ns	
	End of INTA Sequence to Next INTA Sequence.								

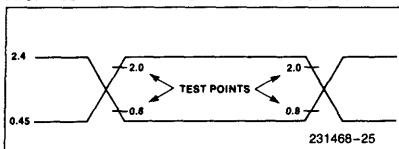
*Worst case timing for TCHCL in an actual microprocessor system is typically much greater than 500 ns (i.e. 8085A = 1.6 μs , 8085A-2 = 1 μs , 8086 = 1 μs , 8086-2 = 625 ns)

NOTE:

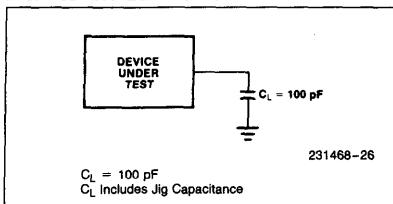
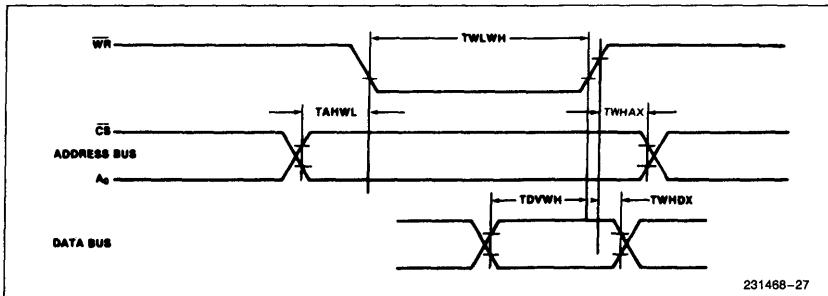
This is the low time required to clear the input latch in the edge triggered mode.

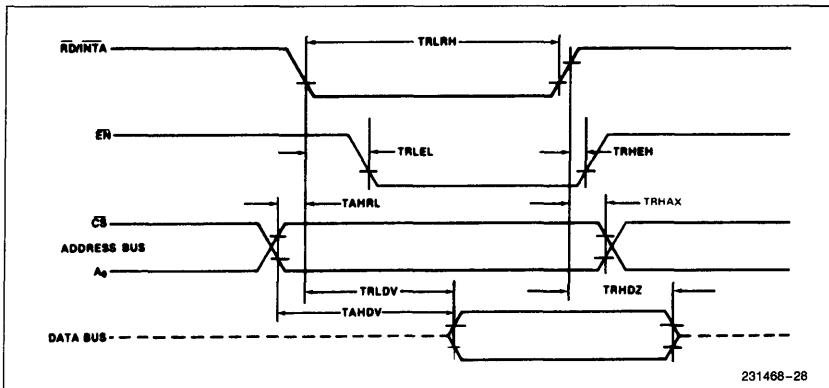
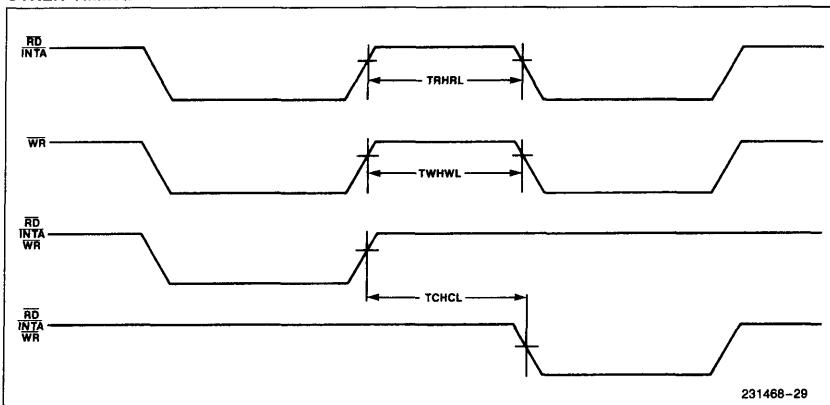
TIMING RESPONSES

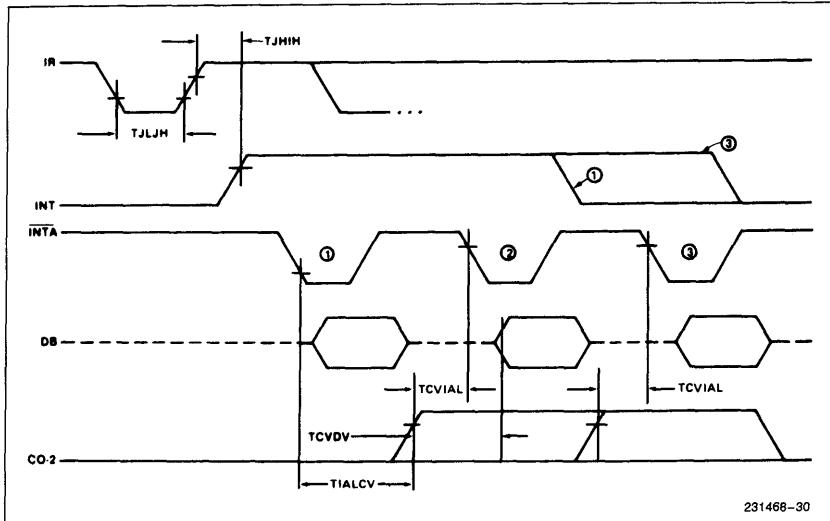
Symbol	Parameter	8259A-8		8259A		8259A-2		Units	Test Conditions
		Min	Max	Min	Max	Min	Max		
TRLDV	Data Valid from RD/INTA ↓		300		200		120	ns	C of Data Bus = 100 pF
TRHDZ	Data Float after RD/INTA ↑	10	200	10	100	10	85	ns	C of Data Bus
TJIIH	Interrupt Output Delay		400		350		300	ns	Max Test C = 100 pF Min Test C = 15 pF
TIALCV	Cascade Valid from First INTA ↓ (Master Only)		565		565		360	ns	C _{INT} = 100 pF
TRLEL	Enable Active from RD ↓ or INTA ↓		160		125		100	ns	CCASCADE = 100 pF
TRHEH	Enable Inactive from RD ↑ or INTA ↑		325		150		150	ns	
TAHDV	Data Valid from Stable Address		350		200		200	ns	
TCVDV	Cascade Valid to Valid Data		300		300		200	ns	

A.C. TESTING INPUT/OUTPUT WAVEFORM

A.C. Testing: Inputs are driven at 2.4V for a logic "1" and 0.45V for a logic "0". Timing measurements are made at 2.0V for a logic "1" and 0.8V for a logic "0".

A.C. TESTING LOAD CIRCUIT**WAVEFORMS****WRITE**

WAVEFORMS (Continued)**READ/INTA****OTHER TIMING**

WAVEFORMS (Continued)**INTA SEQUENCE****NOTES:**

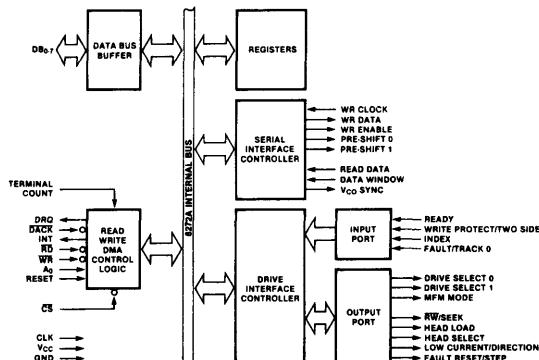
- Interrupt output must remain HIGH at least until leading edge of first INTA.
1. Cycle 1 in 8086, 8088 systems, the Data Bus is not active.



8272A SINGLE/DOMBLE DENSITY FLOPPY DISK CONTROLLER

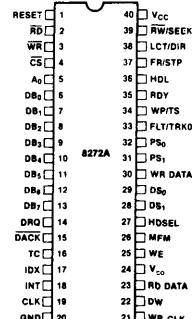
- IBM Compatible in Both Single and Double Density Recording Formats
- Programmable Data Record Lengths: 128, 256, 512, or 1024 Bytes/Sector
- Multi-Sector and Multi-Track Transfer Capability
- Drives Up to 4 Floppy or Mini-Floppy Disks
- Controls 8", 5 1/4" and 3 1/2" Floppy Disk Drives
- Data Transfers in DMA or Non-DMA Mode
- Parallel Seek Operations on Up to Four Drives
- Compatible with all Intel and Most Other Microprocessors
- Single-Phase 8 MHz Clock
- Single +5V Power Supply ($\pm 10\%$)
- Plastic 40 Pin DIP or 40 Pin CERDIP Packages

The 8272A is an LSI Floppy Disk Controller (FDC) Chip, which contains the circuitry and control functions for interfacing a processor to 4 Floppy Disk Drives. It is capable of supporting either IBM 3740 single density format (FM), or IBM System 34 Double Density format (MFM) including double sided recording. The 8272A provides control signals which simplify the design of an external phase locked loop and write precompensation circuitry. The FDC simplifies and handles most of the burdens associated with implementing a Floppy Disk Drive Interface. The 8272A is a pin-compatible upgrade to the 8272.



210606-1

Figure 1. 8272A Internal Block Diagram



210606-2

Figure 2. Pin Configuration

Table 1. Pin Description

Symbol	Pin No.	Type	Connec-tion To	Name and Function
RESET	1	I	μ P	RESET: Places FDC in idle state. Resets output lines to FDD to "0" (low). Does not clear the last specify command.
\overline{RD}	2	I(1)	μ P	READ: Control signal for transfer of data from FDC to Data Bus, when "0" (low).
\overline{WR}	3	I(1)	μ P	WRITE: Control signal for transfer of data to FDC via Data Bus, when "0" (low).
\overline{CS}	4	I	μ P	CHIP SELECT: IC selected when "0" (low) allowing \overline{RD} and \overline{WR} to be enabled.
A_0	5	I(1)	μ P	DATA/STATUS REGISTER SELECT: Selects Data Reg ($A_0 = 1$) or Status Reg ($A_0 = 0$) contents to be sent to Data Bus.
DB_0-DB_7	6-13	I/O(1)	μ P	DATA BUS: Bidirectional 8-Bit Data Bus.
DRQ	14	O	DMA	DATA DMA REQUEST: DMA Request is being made by FDC when DRQ "1".(3)
DACK	15	I	DMA	DMA ACKNOWLEDGE: DMA cycle is active when "0" (low) and Controller is performing DMA transfer.
TC	16	I	DMA	TERMINAL COUNT: Indicates the termination of a DMA transfer when "1" (high)(2).
IDX	17	I	FDD	INDEX: Indicates the beginning of a disk track.
INT	18	O	μ P	INTERRUPT: Interrupt Request Generated by FDC.
CLK	19	I		CLOCK: Single Phase 8 MHz (4 MHz for mini floppies) Squarewave Clock.
GND	20			GROUND: D.C. Power Return.
V_{CC}	40			D.C. POWER: + 5V
RW/SEEK	39	O	FDD	READ WRITE/SEEK: When "1" (high) Seek mode selected and when "0" (low) Read/Write mode selected.
LCT/DIR	38	O	FDD	LOW CURRENT/DIRECTION: Lowers Write current on inner tracks in Read/Write mode, determines direction head will step in Seek mode.
FR/STP	37	O	FDD	FAULT RESET/STEP: Resets fault FF in FDD in Read/Write mode, provides step pulses to move head to another cylinder in Seek mode.
HDL	36	O	FDD	HEAD LOAD: Command which causes Read/Write head in FDD to contact diskette.
RDY	35	I	FDD	READY: Indicates FDD is ready to send or receive data. Must be tied high (gated by the index pulse) for mini floppies which do not normally have a Ready line.
WP/TS	34	I	FDD	WRITE PROTECT/TWO-SIDE: Senses Write Protect status in Read/Write mode, and Two Side Media in Seek mode.
FLT/TRK0	33	I	FDD	FAULT/TRACK 0: Senses FDD fault condition in Read/Write mode and Track 0 condition in Seek mode.
PS ₁ , PS ₀	31, 32	O	FDD	PRECOMPENSATION (PRE-SHIFT): Write precompensation status during MFM mode. Determines early, late, and normal times.
WR DATA	30	O	FDD	WRITE DATA: Serial clock and data bits to FDD.
DS ₁ , DS ₀	28, 29	O	FDD	DRIVE SELECT: Selects FDD unit.

Table 1. Pin Description (Continued)

Symbol	Pin No.	Type	Connec-tion To	Name and Function
HDSEL	27	O	FDD	HEAD SELECT: Head 1 selected when "1" (high) Head 0 selected when "0" (low).
MFM	26	O	PLL	MFM MODE: MFM mode when "1," FM mode when "0".
WE	25	O	FDD	WRITE ENABLE: Enables write data into FDD.
VCO	24	O	PLL	VCO SYNC: Inhibits VCO in PLL when "0" (low), enables VCO when "1."
RD DATA	23	I	FDD	READ DATA: Read data from FDD, containing clock and data bits.
DW	22	I	PLL	DATA WINDOW: Generated by PLL, and used to sample data from FDD.
WR CLK	21	I		WRITE CLOCK: Write data rate to FDD FM = 500 kHz, MFM = 1 MHz, with a pulse width of 250 ns for both FM and MFM. Must be enabled for all operations, both Read and Write.

NOTES:

1. Disabled when CS = 1.
2. TC must be activated to terminate the Execution Phase of any command.
3. DRQ is also an input for certain test modes. It should have a 5 kΩ pull-up resistor to prevent activation.

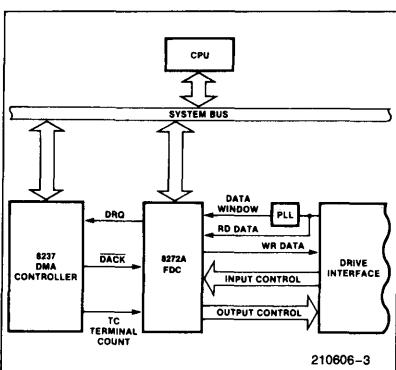


Figure 3. 8272A System Block Diagram

DESCRIPTION

Hand-shaking signals are provided in the 8272A which make DMA operation easy to incorporate with the aid of an external DMA Controller chip, such as the 8237A. The FDC will operate in either DMA or Non-DMA mode. In the Non-DMA mode, the FDC generates interrupts to the processor for every transfer of a data byte between the CPU and the 8272A. In the DMA mode, the processor need only

load a command into the FDC and all data transfers occur under control of the 8272A and DMA controller.

There are 15 separate commands which the 8272A will execute. Each of these commands require multiple 8-bit bytes to fully specify the operation which the processor wishes the FDC to perform. The following commands are available.

Read Data	Write Data
Read ID	Format a Track
Read Deleted Data	Write Deleted Data
Read a Track	Seek
Scan Equal	Recalibrate (Restore to Track 0)
Scan High or Equal	Sense Interrupt Status
Scan Low or Equal	Sense Drive Status
Specify	

For more information see the Intel Application Notes AP-116 and AP-121.

FEATURES

Address mark detection circuitry is internal to the FDC which simplifies the phase locked loop and read electronics. The track stepping rate, head load time, and head unload time may be programmed by the user. The 8272A offers many additional features such as multiple sector transfers in both read and write modes with a single command, and full IBM compatibility in both single (FM) and double density (MFM) modes.

8272A ENHANCEMENTS

On the 8272A, after detecting the Index Pulse, the VCO Sync output stays low for a shorter period of time. See Figure 4a.

On the 8272 there can be a problem reading data when Gap 4A is 00 and there is no IAM. This occurs on some older floppy formats. The 8272A cures this problem by adjusting the VCO Sync timing so that it is not low during the data field. See Figure 4b.

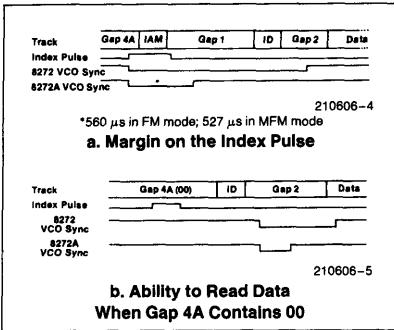


Figure 4. 8272A Enhancements over the 8272

8272A REGISTERS—CPU INTERFACE

The 8272A contains two registers which may be accessed by the main system processor; a Status Register and a Data Register. The 8-bit Main Status Register contains the status information of the FDC, and may be accessed at any time. The 8-bit Data Register (actually consists of several registers in a stack with only one register presented to the data bus at a time), stores data, commands, parameters, and FDD status information. Data bytes are read out of, or written into, the Data Register in order to program or obtain the results after execution of a command. The Status Register may only be read and is used to facilitate the transfer of data between the processor and 8272A.

The relationship between the Status/Down registers and the signals RD, WR, and A₀ is shown in Table 2.

Table 2. A₀, RD, WR Decoding for the Selection of Status/Data Register Functions.

A ₀	RD	WR	Function
0	0	1	Read Main Status Register
0	1	0	Illegal(1)
0	0	0	Illegal(1)
1	0	0	Illegal(1)
1	0	1	Read from Data Register
1	1	0	Write into Data Register

NOTE:

- Design must guarantee that the 8272A is not subjected to illegal inputs.

The Main Status Register bits are defined in Table 3.

Table 3. Main Status Register Bit Description

Bit Number	Name	Symbol	Description
D ₀	FDD 0 Busy	D ₀ B	FDD number 0 is in the Seek mode.
D ₁	FDD 1 Busy	D ₁ B	FDD number 1 is in the Seek mode.
D ₂	FDD 2 Busy	D ₂ B	FDD number 2 is in the Seek mode.
D ₃	FDD 3 Busy	D ₃ B	FDD number 3 is in the Seek mode.
D ₄	FDC Busy	CB	A read or write command is in process.
D ₅	Non-DMA Mode	NDM	The FDC is in the non-DMA mode. This bit is set only during the execution phase in non-DMA mode. Transition to "0" state indicates execution phase has ended.
D ₆	Data Input/Output	DIO	Indicates direction of data transfer between FDC and Data Register. If DIO = "1" then transfer is from Data Register to the Processor. If DIO = "0", then transfer is from the Processor to Data Register.
D ₇	Request for Master	RQM	Indicates Data Register is ready to send or receive data to or from the Processor. Both bits DIO and RQM should be used to perform the handshaking functions of "ready" and "direction" to the processor.

The DIO and RQM bits in the Status Register indicate when Data is ready and in which direction data will be transferred on the Data Bus.

NOTE:

There is a 12 μ s or 24 μ s RQM flag delay when using an 8 or 4 MHz clock respectively.

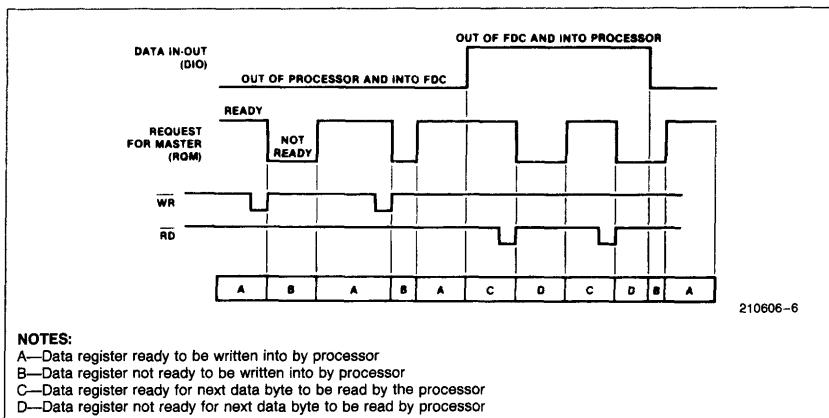


Figure 5. Status Register Timing

The 8272A is capable of executing 15 different commands. Each command is initiated by a multi-byte transfer from the processor, and the result after execution of the command may also be a multi-byte transfer back to the processor. Because of this multi-byte interchange of information between the 8272A and the processor, it is convenient to consider each command as consisting of three phases:

Command Phase: The FDC receives all information required to perform a particular operation from the processor.

Execution Phase: The FDC performs the operation it was instructed to do.

Result Phase: After completion of the operation, status and other house-keeping information are made available to the processor.

During Command or Result Phases the Main Status Register (described in Table 3) must be read by the processor before each byte of information is written into or read from the Data Register. Bits D6 and D7 in the Main Status Register must be in a 0 and 1 state, respectively, before each byte of the command word may be written into the 8272A. Many of the commands require multiple bytes, and as a result the Main Status Register must be read prior to each byte transfer to the 8272A. On the other hand, during the Result Phase, D6 and D7 in the Main Status Register must both be 1's (D6 = 1 and D7 = 1) before reading each byte from the Data Register.

NOTE:

This reading of the Main Status Register before each byte transfer to the 8272A is required in only the Command and Result Phases, and NOT during the Execution Phase.

During the Execution Phase, the Main Status Register need not be read. If the 8272A is in the non-DMA Mode, then the receipt of each data byte (if 8272A is reading data from FDD) is indicated by an interrupt signal on pin 18 (INT = 1). The generation of a Read signal (RD = 0) will reset the interrupt as well as output the Data onto the Data Bus. For example, if the processor cannot handle Interrupts fast enough (every 13 μ s for MFM mode) then it may poll the Main Status Register and then bit D7 (RQM) functions just like the Interrupt signal. If a Write Command is in process, then the WR signal performs the reset to the interrupt signal.

The 8272A always operates in a multi-sector transfer mode. It continues to transfer data until the TC input is active. In Non-DMA Mode, the system must supply the TC input.

If the 8272A is in the DMA Mode, no Interrupts are generated during the Execution Phase. The 8272A generates DREQ's (DMA Requests) when each byte of data is available. The DMA Controller responds to this request with both a $\overline{DACK} = 0$ (DMA Acknowledge) and a RD = 0 (Read signal). When the DMA Acknowledge signal goes low ($\overline{DACK} = 0$) then the DMA Request is reset (DREQ = 0). If a Write Command has been programmed then a WR signal will appear instead of RD. After the Execution Phase has been completed (Terminal Count has occurred) then an Interrupt will occur (INT = 1). This signifies the beginning of the Result Phase. When the first byte of data is read during the Result Phase, the Interrupt is automatically reset (INT = 0).

It is important to note that during the Result Phase all bytes shown in the Command Table must be read. The Read Data Command, for example has seven bytes of data in the Result Phase. All seven bytes must be read in order to successfully complete the Read Data Command. The 8272A will not accept a new command until all seven bytes have been read. Other commands may require fewer bytes to be read during the Result Phase.

The 8272A contains five Status Registers. The Main Status Register mentioned above may be read by the processor at any time. The other four Status Registers (ST0, ST1, ST2, and ST3) are only available during the Result Phase, and may be read only after successfully completing a command. The particular command which has been executed determines how many of the Status Registers will be read.

The bytes of data which are sent to the 8272A to form the Command Phase, and are read out of the 8272A in the Result Phase, must occur in the order shown in the Table 4. That is, the Command Code must be sent first and the other bytes sent in the prescribed sequence. No foreshortening of the Command or Result Phases are allowed. After the last byte of data in the Command Phase is sent to the 8272A, the Execution Phase automatically starts. In a similar fashion, when the last byte of

Table 4. 8272A Command Set

Phase	R/W	Data Bus								Remarks
		D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	
READ DATA										
Command	W	MT	MFM	SK	0	0	1	1	0	Command Codes Sector ID Information Prior to Command Execution
	W	0	0	0	0	0	HDS	DS1	DS0	
	W	_____	_____	_____	C	_____	_____	_____	_____	
	W	_____	_____	_____	H	_____	_____	_____	_____	
	W	_____	_____	_____	R	_____	_____	_____	_____	
	W	_____	_____	_____	N	_____	_____	_____	_____	
	W	_____	_____	_____	EOT	_____	_____	_____	_____	
	W	_____	_____	_____	GPL	_____	_____	_____	_____	
	W	_____	_____	_____	DTL	_____	_____	_____	_____	
Execution										Data Transfer Between the FDD and Main-System
Result	R				ST 0					Status Information After Command Execution
	R				ST 1					
	R				ST 2					
	R				C					
	R				H					
	R				R					
	R				N					
READ DELETED DATA										
Command	W	MT	MFM	SK	0	1	1	0	0	Command Codes Sector ID Information Prior to Command Execution
	W	0	0	0	0	0	HDS	DS1	DS0	
	W	_____	_____	_____	C	_____	_____	_____	_____	
	W	_____	_____	_____	H	_____	_____	_____	_____	
	W	_____	_____	_____	R	_____	_____	_____	_____	
	W	_____	_____	_____	N	_____	_____	_____	_____	
	W	_____	_____	_____	EOT	_____	_____	_____	_____	
	W	_____	_____	_____	GPL	_____	_____	_____	_____	
	W	_____	_____	_____	DTL	_____	_____	_____	_____	
Execution										Data Transfer Between the FDD and Main-System
Result	R				ST 0					Status Information After Command Execution
	R				ST 1					
	R				ST 2					
	R				C					
	R				H					
	R				R					
	R				N					

Table 4. 8272A Command Set (Continued)

Phase	R/W	Data Bus								Remarks
		D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	
WRITE DATA										
Command	W	MT	MFM	0	0	0	1	0	1	Command Codes
	W	0	0	0	0	0	HDS	DS1	DS0	Sector ID Information
	W	_____	_____	C	_____	_____	_____	_____	_____	Prior to Command
	W	_____	_____	H	_____	_____	_____	_____	_____	Execution
	W	_____	_____	R	_____	_____	_____	_____	_____	
	W	_____	_____	N	_____	_____	_____	_____	_____	
	W	_____	_____	EOT	_____	_____	_____	_____	_____	
	W	_____	_____	GPL	_____	_____	_____	_____	_____	
Execution	W	_____	_____	DTL	_____	_____	_____	_____	_____	Data Transfer
	W	_____	_____	_____	_____	_____	_____	_____	_____	Between the Main-System and FDD
	R	_____	_____	ST 0	_____	_____	_____	_____	_____	Status Information
	R	_____	_____	ST 1	_____	_____	_____	_____	_____	After Command
	R	_____	_____	ST 2	_____	_____	_____	_____	_____	Execution
	R	_____	_____	C	_____	_____	_____	_____	_____	
	R	_____	_____	H	_____	_____	_____	_____	_____	
	R	_____	_____	R	_____	_____	_____	_____	_____	
Result	R	_____	_____	N	_____	_____	_____	_____	_____	Sector ID Information
	R	_____	_____	_____	_____	_____	_____	_____	_____	After Command
	R	_____	_____	_____	_____	_____	_____	_____	_____	Execution
	R	_____	_____	_____	_____	_____	_____	_____	_____	
	R	_____	_____	_____	_____	_____	_____	_____	_____	
	R	_____	_____	_____	_____	_____	_____	_____	_____	
	R	_____	_____	_____	_____	_____	_____	_____	_____	
	R	_____	_____	_____	_____	_____	_____	_____	_____	
WRITE DELETED DATA										
Command	W	MT	MFM	0	0	1	0	0	1	Command Codes
	W	0	0	0	0	0	HDS	DS1	DS0	Sector ID Information
	W	_____	_____	C	_____	_____	_____	_____	_____	Prior to Command
	W	_____	_____	H	_____	_____	_____	_____	_____	Execution
	W	_____	_____	R	_____	_____	_____	_____	_____	
	W	_____	_____	N	_____	_____	_____	_____	_____	
	W	_____	_____	EOT	_____	_____	_____	_____	_____	
	W	_____	_____	GPL	_____	_____	_____	_____	_____	
Execution	W	_____	_____	DTL	_____	_____	_____	_____	_____	Data Transfer
	W	_____	_____	_____	_____	_____	_____	_____	_____	Between the FDD and Main-System
	R	_____	_____	ST 0	_____	_____	_____	_____	_____	Status Information
	R	_____	_____	ST 1	_____	_____	_____	_____	_____	After Command
	R	_____	_____	ST 2	_____	_____	_____	_____	_____	Execution
	R	_____	_____	C	_____	_____	_____	_____	_____	
	R	_____	_____	H	_____	_____	_____	_____	_____	
	R	_____	_____	R	_____	_____	_____	_____	_____	
Result	R	_____	_____	N	_____	_____	_____	_____	_____	Sector ID Information
	R	_____	_____	_____	_____	_____	_____	_____	_____	After Command
	R	_____	_____	_____	_____	_____	_____	_____	_____	Execution
	R	_____	_____	_____	_____	_____	_____	_____	_____	
	R	_____	_____	_____	_____	_____	_____	_____	_____	
	R	_____	_____	_____	_____	_____	_____	_____	_____	
	R	_____	_____	_____	_____	_____	_____	_____	_____	
	R	_____	_____	_____	_____	_____	_____	_____	_____	

Table 4. 8272A Command Set (Continued)

Phase	R/W	Data Bus								Remarks
		D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	
READ A TRACK										
Command	W	0	MFM	SK	0	0	0	1	0	Command Codes Sector ID Information Prior to Command Execution
	W	0	0	0	0	0	HDS	DS1	DS0	
	W	—	—	—	—	C	—	—	—	
	W	—	—	—	—	H	—	—	—	
	W	—	—	—	—	R	—	—	—	
	W	—	—	—	—	N	—	—	—	
	W	—	—	—	—	EOT	—	—	—	
	W	—	—	—	—	GPL	—	—	—	
	W	—	—	—	—	DTL	—	—	—	
Execution										Data Transfer Between the FDD and Main-System. FDC Reads all of Cylinders Contents from Index Hole to EOT
Result	R	—	—	—	ST 0	—	—	—	—	Status Information After Command Execution
	R	—	—	—	ST 1	—	—	—	—	
	R	—	—	—	ST 2	—	—	—	—	
	R	—	—	—	C	—	—	—	—	
	R	—	—	—	H	—	—	—	—	
	R	—	—	—	R	—	—	—	—	
	R	—	—	—	N	—	—	—	—	
READ ID										
Command	W	0	MFM	0	0	1	0	1	0	Commands The First Correct ID Information on the Cylinder is Stored in Data Register
	W	0	0	0	0	0	HDS	DS1	DS0	
Execution										Status Information After Command Execution
Result	R	—	—	—	ST 0	—	—	—	—	Sector ID Information During Execution Phase
	R	—	—	—	ST 1	—	—	—	—	
	R	—	—	—	ST 2	—	—	—	—	
	R	—	—	—	C	—	—	—	—	
	R	—	—	—	H	—	—	—	—	
	R	—	—	—	R	—	—	—	—	
	R	—	—	—	N	—	—	—	—	

Table 4. 8272A Command Set (Continued)

Phase	R/W	Data Bus									Remarks
		D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀		
FORMAT A TRACK											
Command	W	0	MFM	0	0		1	1	0	1	Command Codes
	W	0	0	0	0		0	HDS	DS1	DS0	Bytes/Sector
	W					N					Sectors/Cylinder
	W					SC					Gap 3
	W					GPL					Filler Byte
	W					D					FDC Formats an Entire Cylinder
Execution											
Result	R					ST 0					Status Information After Command Execution
	R					ST 1					
	R					ST 2					
	R					C					
	R					H					
	R					R					
	R					N					In This Case, the ID Information has no Meaning

Table 4. 8272A Command Set (Continued)

Phase	R/W	Data Bus									Remarks	
		D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀			
SCAN EQUAL												
Command	W	MT	MFM	SK	1	0	0	0	1	Command Codes Sector ID Information Prior to Command Execution		
	W	0	0	0	0	0	HDS	DS1	DS0			
	W				C							
	W				H							
	W				R							
	W				N							
	W				EOT							
	W				GPL							
	W				STP							
Execution										Data Compared Between the FDD and Main-System		
Result	R				ST 0					Status Information After Command Execution		
	R				ST 1							
	R				ST 2							
	R				C							
	R				H							
	R				R							
	R				N							
SCAN LOW OR EQUAL												
Command	W	MT	MFM	SK	1	1	0	0	1	Command Codes Sector ID Information Prior to Command Execution		
	W	0	0	0	0	0	HDS	DS1	DS0			
	W				C							
	W				H							
	W				R							
	W				N							
	W				EOT							
	W				GPL							
	W				STP							
Execution										Data Compared Between the FDD and Main-System		
Result	R				ST 0					Status Information After Command Execution		
	R				ST 1							
	R				ST 2							
	R				C							
	R				H							
	R				R							
	R				N							

Table 4. 8272A Command Set (Continued)

Phase	R/W	Data Bus								Remarks
		D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	
SCAN HIGH OR EQUAL										
Command	W	MT	MFM	SK	1	1	1	0	1	Command Codes
	W	0	0	0	0	0	HDS	DS1	DS0	Sector ID Information Prior to Command Execution
	W					C				
	W					H				
	W					R				
	W					N				
	W				EOT					
	W					GPL				
	W					STP				
Execution										Data Compared Between the FDD and Main-System
Result	R				ST 0					Status Information After Command Execution
	R				ST 1					
	R				ST 2					
	R				C					
	R				H					
	R				R					
	R				N					
RECALIBRATE										
Command	W	0	0	0	0	0	1	1	1	Command Codes
	W	0	0	0	0	0	0	DS1	DS0	Head Retracted to Track 0
Execution										
SENSE INTERRUPT STATUS										
Command	W	0	0	0	0	1	0	0	0	Command Codes
Result	R				ST 0					Status Information at the End of Each Seek Operation About the FDC
	R				PCN					



8272A

Table 4. 8272A Command Set (Continued)

Phase	R/W	Data Bus								Remarks
		D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	
SPECIFY										
Command	W	0	0	0	0	0	0	1	1	Command Codes
	W	—	SRT	————→		←————	HUT	—		
	W	—	HLT	————→		————→		ND		
SENSE DRIVE STATUS										
Command	W	0	0	0	0	0	1	0	0	Command Codes
	W	0	0	0	0	0	HDS	DS1	DS0	
Result	R	————	————	ST 3	————	————	————	————	————	Status Information about FDD
SEEK										
Command	W	0	0	0	0	1	1	1	1	Command Codes
	W	0	0	0	0	0	HDS	DS1	DS0	
Execution	W	————	————	NCN	————	————	————	————	————	Head is Positioned Over Proper Cylinder on Diskette
INVALID										
Command	W	————	————	Invalid Codes	————	————	————	————	————	Invalid Command Codes (NoOp—FDC Goes Into Standby State)
		————	————		————	————	————	————	————	ST 0 = 80 (16)
Result	R	————	————	ST 0	————	————	————	————	————	

Table 5. Command Mnemonics

Symbol	Name	Description
A ₀	Address Line 0	A ₀ controls selection of Main Status Register (A ₀ = 0) or Data Register (A ₀ = 1).
C	Cylinder Number	C stands for the current selected Cylinder track number 0 through 76 of the medium.
D	Data	D stands for the data pattern which is going to be written into a Sector.
D ₇ -D ₀	Data Bus	8-bit Data bus where D ₇ is the most significant bit, and D ₀ is the least significant bit.
DS0, DS1	Drive Select	DS stands for a selected drive number 0 or 1.
DTL	Data Length	When N is defined as 00, DTL stands for the data length which users are going to read out or write into the Sector.
EOT	End of Track	EOT stands for the final Sector number of a Cylinder.
GPL	Gap Length	GPL stands for the length of Gap 3 (spacing between Sectors excluding VCO Sync Field).
H	Head Address	H stands for head number 0 or 1, as specified in ID field.
HDS	Head Select	HDS stands for a selected head number 0 or 1 (H = HDS in all command words).
HLT	Head Load Time	HLT stands for the head load time in the FDD (2 to 254 ms in 2 ms increments).
HUT	Head Unload Time	HUT stands for the head unload time after a read or write operation has occurred (16 to 240 ms in 16 ms increments).
MFM	FM or MFM Mode	If MF is low, FM mode is selected and if it is high, MFM mode is selected.
MT	Multi-Track	If MT is high, a multi-track operation is to be performed (a cylinder under both HDO and HD1 will be read or written).
N	Number	N stands for the number of data bytes written in a Sector.
NCN	New Cylinder Number	NCN stands for a new Cylinder number, which is going to be reached as a result of the Seek operation. Desired position of Head.
ND	Non-DMA Mode	ND stands for operation in the Non-DMA Mode.
PCN	Present Cylinder Number	PCN stands for the Cylinder number at the completion of SENSE INTERRUPT STATUS Command. Position of Head at present time.
R	Record	R stands for the Sector number, which will be read or written.
R/W	Read/Write	R/W stands for either Read (R) or Write (W) signal.
SC	Sector	SC indicates the number of Sectors per Cylinder.
SK	Skip	SK stands for Skip Deleted Data Address Mark.
SRT	Step Rate Time	SRT stands for the Stepping Rate for the FDD (1 to 16 ms in 1 ms increments). The same Stepping Rate applies to all drives (F = 1 ms, E = 2 ms, etc.).

Table 5. Command Mnemonics (Continued)

Symbol	Name	Description
ST 0	Status 0	
ST 1	Status 1	
ST 2	Status 2	
ST 3	Status 3	ST 0-3 stand for one of four registers which store the status information after a command has been executed. This information is available during the result phase after command execution. These registers should not be confused with the main status register (selected by A ₀ = 0). ST 0-3 may be read only after a command has been executed and contain information relevant to that particular command.
STP		During a Scan operation, if STP = 1, the data in contiguous sectors is compared byte by byte with data sent from the processor (or DMA), and if STP = 2, then alternate sectors are read and compared.

data is read out in the Result Phase, the command is automatically ended and the 8272A is ready for a new command. A command may be aborted by simply sending a Terminal Count signal to pin 16 (TC=1). This is a convenient means of ensuring that the processor may always get the 8272A's attention even if the disk system hangs up in an abnormal manner.

POLLING FEATURE OF THE 8272A

After power-up RESET, the Drive Select Lines DS0 and DS1 will automatically go into a polling mode. In between commands (and between step pulses in the SEEK command) the 8272A polls all four FDDs looking for a change in the Ready line from any of the drives. If the Ready line changes state (usually due to a door opening or closing) then the 8272A will generate an interrupt. When Status Register 0 (ST0) is read (after Sense Interrupt Status is issued), Not Ready (NR) will be indicated. The polling of the Ready line by the 8272A occurs continuously between instructions, thus notifying the processor which drives are on or off line. Approximate scan timing is shown in Table 6.

Table 6. Scan Timing

DS1	DS0	Approximate Scan Timing
0	0	220 µs
0	1	220 µs
1	0	220 µs
1	1	440 µs

COMMAND DESCRIPTIONS

During the Command Phase, the Main Status Register must be polled by the CPU before each byte is

written into the Data Register. The DIO (DB6) and RQM (BD7) bits in the Main Status Register must be in the "0" and "1" states respectively, before each byte of the command may be written into the 8272A. The beginning of the execution phase for any of these commands will cause DIO and RQM to switch to "1" and "0" states respectively.

READ DATA

A set of nine (9) byte words are required to place the FDC into the Read Data Mode. After the Read Data command has been issued the FDC loads the head (if it is in the unloaded state), waits the specified head settling time (defined in the Specify Command), and begins reading ID Address Marks and ID fields. When the current sector number ("R") stored in the ID Register (IDR) compares with the sector number read off the diskette, then the FDC outputs data (from the data field) byte-by-byte to the main system via the data bus.

After completion of the read operation from the current sector, the Sector Number is incremented by one, and the data from the next sector is read and output on the data bus. This continuous function is called a "Multi-Sector Read Operation." The Read Data Command must be terminated by the receipt of a Terminal Count signal. Upon receipt of this signal, the FDC stops outputting data to the processor, but will continue to read data from the current sector, check CRC (Cyclic Redundancy Count) bytes, and then at the end of the sector terminate the Read Data Command.

The amount of data which can be handled with a single command to the FDC depends upon MT (multi-track), MFM (MFM/FM), and N (Number of Bytes/Sector). Table 7 on the next page shows the Transfer Capacity.

Table 7. Transfer Capacity

Multi-Track MT	MFM/FM MFM	Bytes/Sector N	Maximum Transfer Capacity (Bytes/Sector)(Number of Sectors)	Final Sector Read from Diskette
0	0	00	(128) (26) = 3,328	26 at Side 0 or 26 at Side 1
	1	01	(256) (26) = 6,656	
1	0	00	(128) (52) = 6,656	26 at Side 1
	1	01	(256) (52) = 13,312	
0	0	01	(256) (15) = 3,840	15 at Side 0 or 15 at Side 1
	1	02	(512) (15) = 7,680	
1	0	01	(256) (30) = 7,680	15 at Side 1
	1	02	(512) (30) = 15,360	
0	0	02	(512) (8) = 4,096	8 at Side 0 or 8 at Side 1
	1	03	(1024) (8) = 8,192	
1	0	02	(512) (16) = 8,192	8 at Side 1
	1	03	(1024) (16) = 16,384	

The "multi-track" function (MT) allows the FDC to read data from both sides of the diskette. For a particular cylinder, data will be transferred starting at Sector 1, Side 0 and completing at Sector L, Side 1 (Sector L = last sector on the side).

NOTE:

This function pertains to only one cylinder (the same track) on each side of the diskette.

When N = 0, then DTL defines the data length which the FDC must treat as a sector. If DTL is smaller than the actual data length in a Sector, the data beyond DTL in the Sector is not sent to the Data Bus. The FDC reads (internally) the complete Sector performing the CRC check, and depending upon the manner of command termination, may perform a Multi-Sector Read Operation. When N is non-zero, then DTL has no meaning and should be set to 0FFH.

At the completion of the Read Data Command, the head is not unloaded until after Head Unload Time Interval (specified in the Specify Command) has elapsed. If the processor issues another command before the head unloads then the head settling time may be saved between subsequent reads. This time out is particularly valuable when a diskette is copied from one drive to another.

If the FDC detects the Index Hole twice without finding the right sector, (indicated in "R"), then the FDC sets the ND (No Data) flag in Status Register 1 to a 1 (high), and terminates the Read Data Command. (Status Register 0 also has bits 7 and 6 set to 0 and 1 respectively.)

After reading the ID and Data Fields in each sector, the FDC checks the CRC bytes. If a read error is detected (incorrect CRC in ID field), the FDC sets the DE (Data Error) flag in Status Register 1 to a 1 (high), and if a CRC error occurs in the Data Field the FDC also sets the DD (Data Error in Data Field) flag in Status Register 2 to a 1 (high), and terminates the Read Data Command. (Status Register 0 also has bits 7 and 6 set to 0 and 1 respectively.)

If the FDC reads a Deleted Data Address Mark off the diskette, and the SK bit (bit D5 in the first Command Word) is not set (SK = 0), then the FDC sets the CM (Control Mark) flag in Status Register 2 to a 1 (high), and terminates the Read Data Command, after reading all the data in the Sector. If SK = 1, the FDC skips the sector with the Deleted Data Address Mark and reads the next sector.

During disk data transfers between the FDC and the processor, via the data bus, the FDC must be serviced by the processor every 27 μ s in the FM Mode, and every 13 μ s in the MFM Mode, or the FDC sets the OR (Over Run) flag in Status Register 1 to a 1 (high), and terminates the Read Data Command.

If the processor terminates a read (or write) operation in the FDC, then the ID Information in the Result Phase is dependent upon the state of the MT bit and EOT byte. Table 5 shows the values for C, H, R, and N, when the processor terminates the Command.

Table 8. ID Information When Processor Terminates Command

MT	EOT	Final Sector Transferred to Processor	ID Information at Result Phase			
			C	H	R	N
0	1A 0F 08	Sector 1 to 25 at Side 0 Sector 1 to 14 at Side 0 Sector 1 to 7 at Side 0	NC	NC	R + 1	NC
	1A 0F 08	Sector 26 at Side 0 Sector 15 at Side 0 Sector 8 at Side 0	C + 1	NC	R = 01	NC
	1A 0F 08	Sector 1 to 25 at Side 1 Sector 1 to 14 at Side 1 Sector 1 to 7 at Side 1	NC	NC	R + 1	NC
	1A 0F 08	Sector 26 at Side 1 Sector 15 at Side 1 Sector 8 at Side 1	C + 1	NC	R = 01	NC
1	1A 0F 08	Sector 1 to 25 at Side 0 Sector 1 to 14 at Side 0 Sector 1 to 7 at Side 0	NC	NC	R + 1	NC
	1A 0F 08	Sector 26 at Side 0 Sector 15 at Side 0 Sector 8 at Side 0	NC	LSB	R = 01	NC
	1A 0F 08	Sector 1 to 25 at Side 1 Sector 1 to 14 at Side 1 Sector 1 to 7 at Side 1	NC	NC	R + 1	NC
	1A 0F 08	Sector 26 at Side 1 Sector 15 at Side 1 Sector 8 at Side 1	C + 1	LSB	R = 01	NC

NOTES:

1. NC (No Change): The same value as the one at the beginning of command execution.
2. LSB (Least Significant Bit): The least significant bit of H is complemented.

WRITE DATA

A set of nine (9) bytes are required to set the FDC into the Write Data mode. After the Write Data command has been issued the FDC loads the head (if it is in the unloaded state), waits the specified head settling time (defined in the Specify Command), and begins reading ID Fields. When the current sector number ("R"), stored in the ID Register (IDR) compares with the sector number read off the diskette, then the FDC takes data from the processor byte-by-byte via the data bus, and outputs it to the FDD.

After writing data into the current sector, the Sector Number stored in "R" is incremented by one, and the next data field is written into. The FDC continues this "Multi-Sector Write Operation" until the issu-

ance of a Terminal Count signal. If a Terminal Count signal is sent to the FDC it continues writing into the current sector to complete the data field. If the Terminal Count signal is received while a data field is being written then the remainder of the data field is filled with 00 (zeros).

The FDC reads the ID field of each sector and checks the CRC bytes. If the FDC detects a read error (incorrect CRC) in one of the ID Fields, it sets the DE (Data Error) flag of Status Register 1 to a 1 (high), and terminates the Write Data Command. (Status Register 0 also has bits 7 and 6 set to 0 and 1 respectively.)

The Write Command operates in much the same manner as the Read Command. The following items

are the same; refer to the Read Data Command for details:

- Transfer Capacity
- EN (End of Cylinder) Flag
- ND (No Data) Flag
- Head Unload Time Interval
- ID Information when the processor terminates command (see Table 2)
- Definition of DTL when $N = 0$ and when $N \neq 0$

In the Write Data mode, data transfers between the processor and FDC must occur every $31 \mu s$ in the FM mode, and every $15 \mu s$ in the MFM mode. If the time interval between data transfers is longer than this then the FDC sets the OR (Over Run) flag in Status Register 1 to a 1 (high), and terminates the Write Data Command.

For mini-floppies, multiple track writes are usually not permitted. This is because of the turn-off time of the erase head coils—the head switches tracks before the erase head turns off. Therefore the system should typically wait 1.3 ms before attempting to step or change sides.

WRITE DELETED DATA

This command is the same as the Write Data Command except a Deleted Data Address Mark is written at the beginning of the Data Field instead of the normal Data Address Mark.

READ DELETED DATA

This command is the same as the Read Data Command except that when the FDC detects a Data Address Mark at the beginning of a Data Field (and SK = 0 (low)), it will read all the data in the sector and set the CM flag in Status Register 2 to a 1 (high), and then terminate the command. If SK = 1, then the FDC skips the sector with the Data Address Mark and reads the next sector.

READ A TRACK

This command is similar to READ DATA Command except that the entire data field is read continuously from each of the sectors of a track. Immediately after encountering the INDEX HOLE, the FDC starts reading all data fields on the track as continuous blocks of data. If the FDC finds an error in the ID or DATA CRC check bytes, it continues to read data from the track. The FDC compares the ID information read from each sector with the value stored in the IDR, and sets the ND flag of Status Register 1 to

a 1 (high) if there is no comparison. Multi-track or skip operations are not allowed with this command.

This command terminates when EOT number of sectors have been read. If the FDC does not find an ID Address Mark on the diskette after it encounters the INDEX HOLE for the second time, then it sets the MA (missing address mark) flag in Status Register 1 to a 1 (high), and terminates the command. (Status Register 0 has bits 7 and 6 set to 0 and 1 respectively.)

READ ID

The READ ID Command is used to give the present position of the recording head. The FDC stores the values from the first ID Field it is able to read. If no proper ID Address Mark is found on the diskette, before the INDEX HOLE is encountered for the second time then the MA (Missing Address Mark) flag in Status Register 1 is set to a 1 (high), and if no data is found then the ND (No Data) flag is also set in Status Register 1 to a 1 (high) and the command is terminated.

FORMAT A TRACK

The Format Command allows an entire track to be formatted. After the INDEX HOLE is detected, Data is written on the Diskette: Gaps, Address Marks, ID Fields and Data Fields, all per the IBM System 34 (Double Density) or System 3740 (Single Density) Format are recorded. The particular format which will be written is controlled by the values programmed into N (number of bytes/sector), SC (sectors/cylinder), GPL (Gap Length), and D (Data Pattern) which are supplied by the processor during the Command Phase. The Data Field is filled with the Byte of data stored in D. The ID Field for each sector is supplied by the processor; that is, four data requests per sector are made by the FDC for C (Cylinder Number), H(Head Number), R(Sector Number) and N(Number of Bytes/Sector). This allows the diskette to be formatted with nonsequential sector numbers, if desired.

After formatting each sector, the processor must send new values for C, H, R, and N to the 8272A for each sector on the track. The contents of the R Register is incremented by one after each sector is formatted, thus, the R register contains a value of $R + 1$ when it is read during the Result Phase. This incrementing and formatting continues for the whole track until the FDC encounters the INDEX HOLE for the second time, whereupon it terminates the command.

If a FAULT signal is received from the FDD at the end of a write operation, then the FDC sets the EC flag of Status Register 0 to a 1 (high), and terminates the command after setting bits 7 and 6 of Status Register 0 to 0 and 1 respectively. Also the loss of a READY signal at the beginning of a com-

mand execution phase causes command termination.

Table 9 shows the relationship between N, SC, and GPL for various sector sizes:

Table 9. Sector Size Relationships

Format	Bytes/ Sector	8" Floppy				Bytes/ Sector	5 1/4" Floppy				Bytes/ Sector	3 1/2" Mini Floppy			
		N	SC	GPL(1)	GPL(2)		N	SC	GPL(1)	GPL(2)		N	SC	GPL(1)	GPL(2)
FM Mode	128	00	1A	07	1B	128	00	12	07	09	128	0	0F	07	1B
	256	01	0F	0E	2A	128	00	10	10	19	—	—	—	—	—
	512	02	08	1B	3A	256	01	08	18	30	256	1	09	0F	2A
	1024	03	04	47	8A	512	02	04	46	87	512	2	05	1B	3A
	2048	04	02	C8	FF	1024	03	02	C8	FF	—	—	—	—	—
	4096	05	01	C8	FF	2048	04	01	C8	FF	—	—	—	—	—
MPM Mode	256	01	1A	0E	36	256	01	12	0A	0C	256	1	0F	CE	36
	512	02	0F	1B	54	256	01	10	20	32	—	—	—	—	—
	1024	03	08	35	74	512	02	08	2A	50	512	2	09	1B	54
	2048	04	04	99	FF	1024	03	04	80	F0	1024	3	05	35	74
	4096	05	02	C8	FF	2048	04	02	C8	FF	—	—	—	—	—
	8192	06	01	C8	FF	4096	05	01	C8	FF	—	—	—	—	—

NOTES:

- Suggested values of GPL in Read or Write Commands to avoid splice point between data field and ID field of contiguous sections.
- Suggested values of GPL in format command.

SCAN COMMANDS

The SCAN Commands allow data which is being read from the diskette to be compared against data which is being supplied from the main system (Processor in NON-DMA mode, and DMA Controller in DMA mode). The FDC compares the data on a byte-by-byte basis, and looks for a sector of data which meets the conditions of $D_{FDD} = D_{Processor}$, $D_{FDD} \leq D_{Processor}$, or $D_{FDD} \geq D_{Processor}$. Ones complement arithmetic is used for comparison (FF = largest number, 00 = smallest number). After a whole

sector of data is compared, if the conditions are not met, the sector number is incremented ($R + STP \rightarrow R$), and the scan operation is continued. The scan operation continues until one of the following conditions occur; the conditions for scan are met (equal, low, or high), the last sector on the track is reached (EOT), or the terminal count signal is received.

If the conditions for scan are met then the FDC sets the SH (Scan Hit) flag of Status Register 2 to a 1 (high), and terminates the Scan Command. If the

Table 10. Scan Status Codes

Command	Status Register 2		Comments
	Bit 2 = SN	Bit 3 = SH	
Scan Equal	0	1	$D_{FDD} = D_{Processor}$
	1	0	$D_{FDD} \neq D_{Processor}$
Scan Low or Equal	0	1	$D_{FDD} = D_{Processor}$
	0	0	$D_{FDD} < D_{Processor}$
	1	0	$D_{FDD} \geq D_{Processor}$
Scan High or Equal	0	1	$D_{FDD} = D_{Processor}$
	0	0	$D_{FDD} > D_{Processor}$
	1	0	$D_{FDD} \leq D_{Processor}$

conditions for scan are not met between the starting sector (as specified by R) and the last sector on the cylinder (EOT), then the FDC sets the SN (Scan Not Satisfied) flag of Status Register 2 to a 1 (high), and terminates the Scan Command. The receipt of a TERMINAL COUNT signal from the Processor or DMA Controller during the scan operation will cause the FDC to complete the comparison of the particular byte which is in process, and then to terminate the command. Table 10 shows the status of bits SH and SN under various conditions of SCAN.

If the FDC encounters a Deleted Data Address Mark on one of the sectors (and SK = 0), then it regards the sector as the last sector on the cylinder, sets CM (Control Mark) flag of Status Register 2 to a 1 (high) and terminates the command. If SK = 1, the FDC skips the sector with the Deleted Address Mark, and reads the next sector. In the second case (SK = 1), the FDC sets the CM (Control Mark) flag of Status Register 2 to a 1 (high) in order to show that a Deleted Sector had been encountered.

When either the STP (contiguous sectors STP = 01, or alternate sectors STP = 02 sectors are read) or the MT (Multi-Track) are programmed, it is necessary to remember that the last sector on the track must be read. For example, if STP = 02, MT = 0, the sectors are numbered sequentially 1 through 26, and we start the Scan Command at sector 21; the following will happen. Sectors 21, 23, and 25 will be read, then the next sector (26) will be skipped and the Index Hole will be encountered before the EOT value of 26 can be read. This will result in an abnormal termination of the command. If the EOT had been set at 25 or the scanning started at sector 20, then the Scan Command would be completed in a normal manner.

During the Scan Command data is supplied by either the processor or DMA Controller for comparison against the data read from the diskette. In order to avoid having the OR (Over Run) flag set in Status Register 1, it is necessary to have the data available in less than 27 μ s (FM Mode) or 13 μ s (MFM Mode). If an Overrun occurs the FDC terminates the command.

SEEK

The read/write within the FDD is moved from cylinder to cylinder under control of the Seek Command. The FDC compares the PCN (Present Cylinder Number) which is the current head position with the NCN

(New Cylinder Number), and performs the following operation if there is a difference:

PCN < NCN: Direction signal to FDD set to a 1 (high), and Step Pulses are issued. (Step In.)

PCN > NCN: Direction signal to FDD set to a 0 (low), and Step Pulses are issued. (Step Out.)

The rate at which Step Pulses are issued is controlled by SRT (Stepping Rate Time) in the SPECIFY Command. After each Step Pulse is issued NCN is compared against PCN, and when NCN = PCN, then the SE (Seek End) flag is set in Status Register 0 to a 1 (high), and the command is terminated.

During the Command Phase of the Seek operation the FDC is in the FDC BUSY state, but during the Execution Phase it is in the NON BUSY state. While the FDC is in the NON BUSY state, another Seek Command may be issued, and in this manner parallel seek operations may be done on up to 4 Drives at once.

If an FDD is in a NOT READY state at the beginning of the command execution phase or during the seek operation, then the NR (NOT READY) flag is set in Status Register 0 to a 1 (high), and the command is terminated.

Note that the 8272A Read and Write Commands do not have implied Seekes. Any R/W command should be preceded by: 1) Seek Command; 2) Sense Interrupt Status; and 3) Read ID.

RECALIBRATE

This command causes the read/write head within the FDD to retract to the Track 0 position. The FDC clears the contents of the PCN counter, and checks the status of the Track 0 signal from the FDD. As long as the Track 0 signal is low, the Direction signal remains 1 (high) and Step Pulses are issued. When the Track 0 signal goes high, the SE (SEEK END) flag in Status Register 0 is set to a 1 (high) and the command is terminated. If the Track 0 signal is still low after 77 Step Pulses have been issued, the FDC sets the SE (SEEK END) and EC (EQUIPMENT CHECK) flags of Status Register 0 to both 1s (highs), and terminates the command.

The ability to overlap RECALIBRATE Commands to multiple FDDs, and the loss of the READY signal, as described in the SEEK Command, also applies to the RECALIBRATE Command.

SENSE INTERRUPT STATUS

An Interrupt signal is generated by the FDC for one of the following reasons:

- 1) Upon entering the Result Phase of:
 - a) Read Data Command
 - b) Read a Track Command
 - c) Read ID Command
 - d) Read Deleted Data Command
 - e) Write Data Command
 - f) Format a Cylinder Command
 - g) Write Deleted Data Command
 - h) Scan Commands
- 2) Ready Line of FDD changes state
- 3) End of Seek or Recalibrate Command
- 4) During Execution Phase in the NON-DMA Mode

Interrupts caused by reasons 1 and 4 above occur during normal command operations and are easily discernible by the processor. However, interrupts caused by reasons 2 and 3 above may be uniquely identified with the aid of the Sense Interrupt Status Command. This command when issued resets the interrupt signal and via bits 5, 6, and 7 of Status Register 0 identifies the cause of the interrupt.

Neither the Seek or Recalibrate Command have a Result Phase. Therefore, it is mandatory to use the Sense Interrupt Status Command after these commands to effectively terminate them and to provide verification of the head position (PCN).

Table 11. Seek, Interrupt Codes

Seek End Bit 5	Interrupt Code		Cause
	Bit 6	Bit 7	
0	1	1	Ready Line Changed State, Either Polarity
1	0	0	Normal Termination of Seek or Recalibrate Command
1	1	0	Abnormal Termination of Seek or Recalibrate Command

SPECIFY

The Specify Command sets the initial values for each of the three internal timers. The HUT (Head Unload Time) defines the time from the end of the Execution Phase of one of the Read/Write Com-

mands to the head unload state. This timer is programmable from 16 to 240 ms in increments of 16 ms (01 = 16 ms, 02 = 32 ms . . . OF = 240 ms). The SRT (Step Rate Time) defines the time interval between adjacent step pulses. This timer is programmable from 1 to 16 ms in increments of 1 ms (F = 1 ms, E = 2 ms, D = 3 ms, etc.). The HLT (Head Load Time) defines the time between when the Head Load signal goes high and when the Read/Write operation starts. This timer is programmable from 2 to 254 ms in increments of 2 ms (01 = 2 ms, 02 = 4 ms, 03 = 6 ms . . . FE = 254 ms).

The step rate should be programmed 1 ms longer than the minimum time required by the drive.

The time intervals mentioned above are a direct function of the clock (CLK on pin 19). Times indicated above are for an 8 MHz clock, if the clock was reduced to 4 MHz (mini-floppy application) then all time intervals are increased by a factor of 2.

The choice of DMA or NON-DMA operation is made by the ND (NON-DMA) bit. When this bit is high (ND = 1) the NON-DMA mode is selected, and when ND = 0 the DMA mode is selected.

SENSE DRIVE STATUS

This command may be used by the processor whenever it wishes to obtain the status of the FDDs. Status Register 3 contains the Drive Status information.

INVALID

If an invalid command is sent to the FDC (a command not defined above), then the FDC will terminate the command. No interrupt is generated by the 8272A during this condition. Bit 6 and bit 7 (DIO and RQM) in the Main Status Register are both high ("1") indicating to the processor that the 8272A is in the Result Phase and the contents of Status Register 0 (ST0) must be read. When the processor reads Status Register 0 it will find an 80H indicating an invalid command was received.

A Sense Interrupt Status Command must be sent after a Seek or Recalibrate interrupt, otherwise the FDC will consider the next command to be an invalid Command.

In some applications the user may wish to use this command as a No-Op command, to place the FDC in a standby or no operation state.

Table 12. Status Registers

Bit			Description
No.	Name	Symbol	
STATUS REGISTER 0			
D ₇	Interrupt Code	IC	D ₇ = 0 and D ₆ = 0 Normal Termination of Command, (NT). Command was completed and properly executed.
			D ₇ = 0 and D ₆ = 1 Abnormal Termination of Command, (AT). Execution of Command was started, but was not successfully completed.
			D ₇ = 1 and D ₆ = 0 Invalid Command issue, (IC). Command which was issued was never started.
			D ₇ = 1 and D ₆ = 1 Abnormal Termination because during command execution the ready signal from FDD changed state.
D ₅	Seek End	SE	When the FDC completes the SEEK Command, this flag is set to 1 (high).
D ₄	Equipment Check	EC	If a fault Signal is received from the FDD, or if the Track 0 Signal fails to occur after 77 Step Pulses (Recalibrate Command) then this flag is set.
D ₃	Not Ready	NR	When the FDD is in the not-ready state and a read or write command is issued, this flag is set. If a read or write command is issued to Side 1 of a single sided drive, then this flag is set.
D ₂	Head Address	HD	This flag is used to indicate the state of the head at Interrupt.
D ₁	Unit Select 1	US 1	These flags are used to indicate a Drive Unit Number at Interrupt.
D ₀	Unit Select 0	US 0	
STATUS REGISTER 1			
D ₇	End of Cylinder	EN	When the FDC tries to access a Sector beyond the final Sector of a Cylinder, this flag is set.
D ₆			Not used. This bit is always 0 (low).
D ₅	Data Error	DE	When the FDC detects a CRC error in either the ID field or the data field, this flag is set.
D ₄	Over Run	OR	If the FDC is not serviced by the main-systems during data transfers, within a certain time interval, this flag is set.
D ₃			Not used. This bit always 0 (low).
D ₂	No Data	ND	During execution of READ DATA, WRITE DELETED DATA or SCAN Command, if the FDC cannot find the Sector specified in the IDR Register, this flag is set.
			During executing the READ ID Command, if the FDC cannot read the ID field without an error, then this flag is set.
			During the execution of the READ A Cylinder Command, if the starting sector cannot be found, then this flag is set.

Table 12. Status Register (Continued)

Bit			Description
No.	Name	Symbol	
STATUS REGISTER 1 (Continued)			
D ₁	Not Writable	NW	During execution of WRITE DATA, WRITE DELETED DATA or Format A Cylinder Command, if the FDC detects a write protect signal from the FDD, then this flag is set.
D ₀	Missing Address Mark	MA	If the FDC cannot detect the ID Address Mark after encountering the index hole twice, then this flag is set.
			If the FDC cannot detect the Data Address Mark or Deleted Data Address Mark, this flag is set. Also at the same time, the MD (Missing Address Mark in Data Field) of Status Register 2 is set.
STATUS REGISTER 2			
D ₇			Not used. This bit is always 0 (low).
D ₆	Control Mark	CM	During executing the READ DATA or SCAN Command, if the FDC encounters a Sector which contains a Deleted Data Address Mark, this flag is set.
D ₅	Data Error in Data Field	DD	If the FDC detects a CRC error in the data field then this flag is set.
D ₄	Wrong Cylinder	WC	This bit is related with the ND bit, and when the contents of C on the medium is different from that stored in the IDR, this flag is set.
D ₃	Scan Equal Hit	SH	During execution, the SCAN Command, if the condition of "equal" is satisfied, this flag is set.
D ₂	Scan Not Satisfied	SN	During executing the SCAN Command, if the FDC cannot find a Sector on the cylinder which meets the condition, then this flag is set.
D ₁	Bad Cylinder	BC	This bit is related with the ND bit, and when the content of C on the medium is different from that stored in the IDR and the content of C is FF, then this flag is set.
D ₀	Missing Address Mark in Data Field	MD	When data is read from the medium, if the FDC cannot find a Data Address Mark or Deleted Data Address Mark, then this flag is set.
STATUS REGISTER 3			
D ₇	Fault	FT	This bit is used to indicate the status of the Fault signal from the FDD.
D ₆	Write Protected	WP	This bit is used to indicate the status of the Write Protected signal from the FDD.
D ₅	Ready	RDY	This bit is used to indicate the status of the Ready signal from the FDD.
D ₄	Track 0	T0	This bit is used to indicate the status of the Track 0 signal from the FDD.
D ₃	Two Side	TS	This bit is used to indicate the status of the Two Side signal from the FDD.
D ₂	Head Address	HD	This bit is used to indicate the status of Side Select signal to the FDD.
D ₁	Unit Select 1	US 1	This bit is used to indicate the status of the Unit Select 1 signal to the FDD.
D ₀	Unit Select 0	US 0	This bit is used to indicate the status of the Unit Select 0 signal to the FDD.

ABSOLUTE MAXIMUM RATINGS*

Operating Temperature	0°C to +70°C
Storage Temperature	-40°C to +125°C
All Output Voltages	-0.5 to +7V
All Input Voltages	-0.5 to +7V
Supply Voltage V _{CC}	-0.5 to +7V
Power Dissipation	1 Watt

T_A = 25°C

*Notice: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

D.C. CHARACTERISTICS T_A = 0°C to +70°C, V_{CC} = +5V ± 10%

Symbol	Parameter	Limits		Unit	Test Conditions
		Min	Max		
V _{IL}	Input Low Voltage	-0.5	0.8	V	
V _{IH}	Input High Voltage	2.0	V _{CC} + 0.5	V	
V _{OL}	Output Low Voltage		0.45	V	I _{OL} = 2.0 mA
V _{OH}	Output High Voltage	2.4	V _{CC}	V	I _{OH} = -400 μA
I _{CC}	V _{CC} Supply Current		120	mA	
I _{IL}	Input Load Current (All Input Pins)		10 -10	μA μA	V _{IN} = V _{CC} V _{IN} = 0V
I _{LOH}	High Level Output Leakage Current		10	μA	V _{OUT} = V _{CC}
I _{OFL}	Output Float Leakage Current		±10	μA	0.45C ≤ V _{OUT} ≤ V _{CC}

CAPACITANCE T_A = 25°C, f_c = 1 MHz, V_{CC} = 0V

Symbol	Parameter	Limits		Unit	Test Conditions
		Min	Max		
C _{IN(ϕ)}	Clock Input Capacitance		20	pF	
C _{IN}	Input Capacitance		10	pF	
C _{I/O}	Input/Output Capacitance		20	pF	All Pins Except Pin Under Test Tied to AC Ground

A.C. CHARACTERISTICS $T_A = 0^\circ\text{C}$ to $+70^\circ\text{C}$, $V_{CC} = +5.0V \pm 10\%$

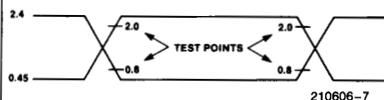
Symbol	Parameter	Typ ⁽¹⁾	Min	Max	Unit	Notes
CLOCK TIMING						
t_{CY}	Clock Period		120	500	ns	(Note 5)
t_{CH}	Clock High Period		40		ns	(Note 4, 5)
t_{RST}	Reset Width		14		t_{CY}	
READ CYCLE						
t_{AR}	Select Setup to $\overline{RD} \downarrow$		0		ns	
t_{RA}	Select Hold from $\overline{RD} \uparrow$		0		ns	
t_{RR}	\overline{RD} Pulse Width		250		ns	
t_{RD}	Data Delay from $\overline{RD} \downarrow$			200	ns	
t_{DF}	Output Float Delay		20	100	ns	
WRITE CYCLE						
t_{AW}	Select Setup to $\overline{WR} \downarrow$		0		ns	
t_{WA}	Select Hold from $\overline{WR} \uparrow$		0		ns	
t_{WW}	\overline{WR} Pulse Width		250		ns	
t_{DW}	Data Setup to $\overline{WR} \uparrow$		150		ns	
t_{WD}	Data Hold from $\overline{WR} \uparrow$		5		ns	
INTERRUPTS						
t_{RI}	INT Delay from $\overline{RD} \uparrow$			500	ns	(Note 6)
t_{WI}	INT Delay from $WR \uparrow$			500	ns	(Note 6)
DMA						
t_{RQCY}	DRQ Cycle Period		13		μs	(Note 6)
t_{AKRQ}	$\overline{DACK} \downarrow$ to DRQ \downarrow			200	ns	
t_{RQR}	DRQ \uparrow to $\overline{RD} \downarrow$		800		ns	(Note 6)
t_{RQW}	DRQ \uparrow to $\overline{WR} \downarrow$		250		ns	(Note 6)
t_{RQRW}	DRQ \uparrow to $\overline{RD} \uparrow$ or $\overline{WR} \uparrow$			12	μs	(Note 6)

A.C. CHARACTERISTICS $T_A = 0^\circ\text{C}$ to $+70^\circ\text{C}$, $V_{CC} = +5.0\text{V} \pm 10\%$ (Continued)

Symbol	Parameter	Typ ⁽¹⁾	Min	Max	Unit	Notes
FDD INTERFACE						
t_{WCY}	WCK Cycle Time	2 or 4 1 or 2			μs	MFM = 0 MFM = 1 (Note 2)
t_{WCH}	WCK High Time	250	80	350	ns	
t_{CP}	Pre-Shift Delay from WCK \uparrow		20	100	ns	
t_{CD}	WDA Delay from WCK \uparrow		20	100	ns	
t_{WDD}	Write Data Width		$t_{WCH} - 50$		ns	
t_{WE}	WE \uparrow to WCK \uparrow or WE \downarrow to WCK \downarrow Delay		20	100	ns	
t_{WWCY}	Window Cycle Time	2 1			μs	MM = 0 MFM = 1
t_{WRD}	Window Setup to RDD \uparrow		15		ns	
t_{RDW}	Window Hold from RDD \downarrow		15		ns	
t_{RDD}	RDD Active Time (HIGH)		40		ns	
FDD SEEK/DIRECTION/STEP						
t_{US}	US _{0,1} Setup to RW/SEEK \uparrow		12		μs	(Note 6)
t_{SU}	US _{0,1} Hold after RW/SEEK \downarrow		15		μs	(Note 6)
t_{SD}	RW/SEEK Setup to LCT/DIR		7		μs	(Note 6)
t_{DS}	RW/SEEK Hold from LCT/DIR		30		μs	(Note 6)
t_{DST}	LCT/DIR Setup to FR/STEP \uparrow		1		μs	(Note 6)
t_{STD}	LCT/DIR Hold from FR/STEP \downarrow		24		μs	(Note 6)
t_{STU}	DS _{2,1} Hold from FR/Step \downarrow		5		μs	(Note 6)
t_{STP}	STEP Active Time (High)	5			μs	(Note 6)
t_{SC}	STEP Cycle Time		33		μs	(Note 3, 6)
t_{FR}	FAULT RESET Active Time (High)		8	10	μs	(Note 6)
t_{IDX}	INDEX Pulse Width	10			t_{CY}	
t_{TC}	Terminal Count Width		1		t_{CY}	

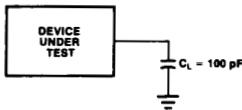
NOTES:

1. Typical values for $T_A = 25^\circ\text{C}$ and nominal supply voltage.
2. The former values are used for standard floppy and the latter values are used for mini-floppies.
3. $t_{SC} = 33 \mu\text{s}$ min. is for different drive units. In the case of same unit, t_{SC} can be ranged from 1 ms to 16 ms with 8 MHz clock period, and 2 ms to 32 ms with 4 MHz clock, under software control.
4. From 2.0V to +2.0V.
5. At 4 MHz, the clock duty cycle may range from 16% to 76%. Using an 8 MHz clock the duty cycle can range from 32% to 52%. Duty cycle is defined as: D.C. = $100 (t_{CH} / t_{CY})$ with typical rise and fall times of 5 ns.
6. The specified values listed are for an 8 MHz clock period. Multiply timings by 2 when using a 4 MHz clock period.

A.C. TESTING INPUT, OUTPUT WAVEFORM

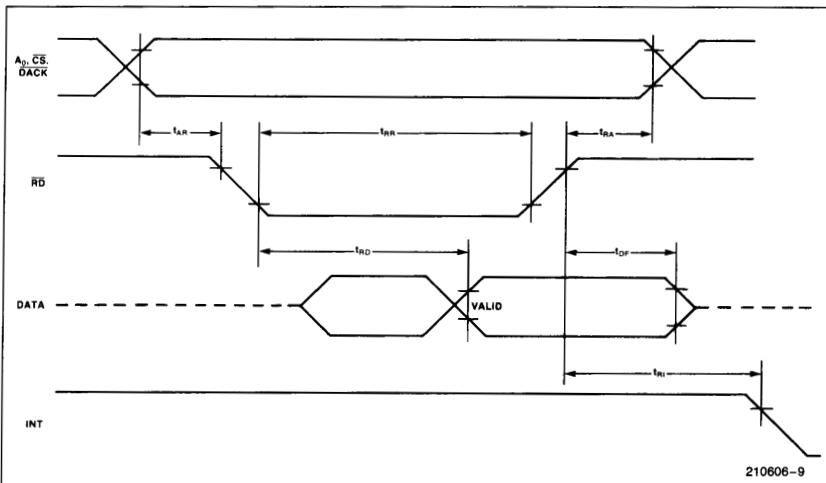
210606-7

A.C. Testing: Inputs are driven at 2.4V for a Logic "1" and 0.45V for a Logic "0". Timing measurements are made at 2.0V for a Logic "1" and 0.8V for a Logic "0".

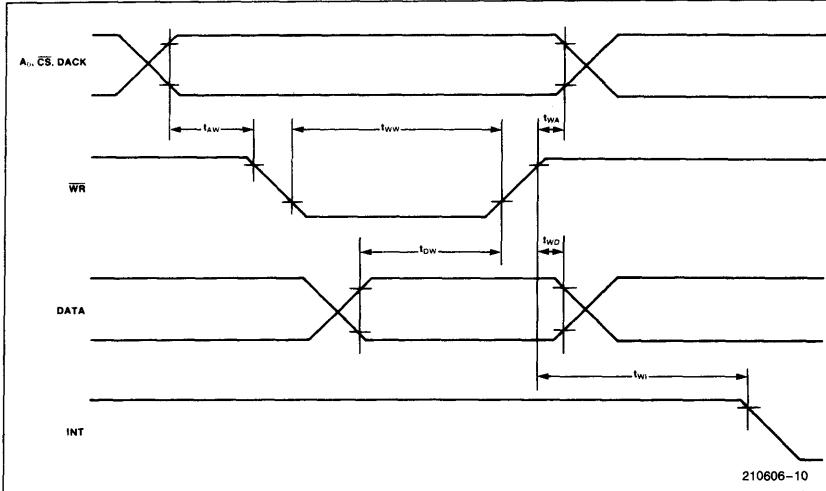
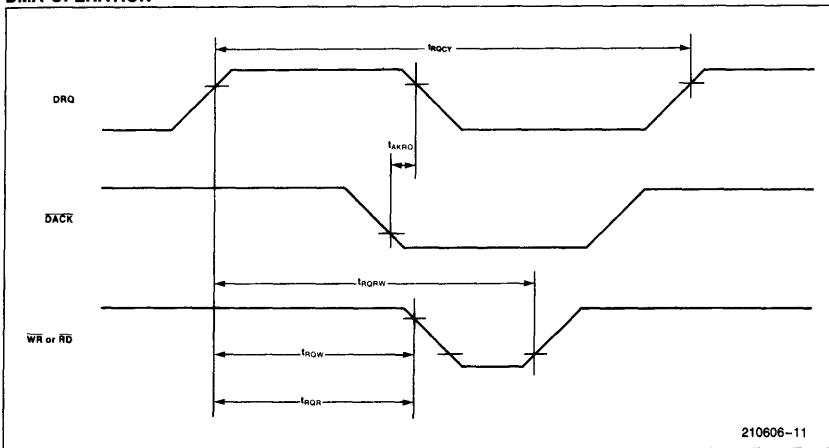
A.C. TESTING LOAD CIRCUIT

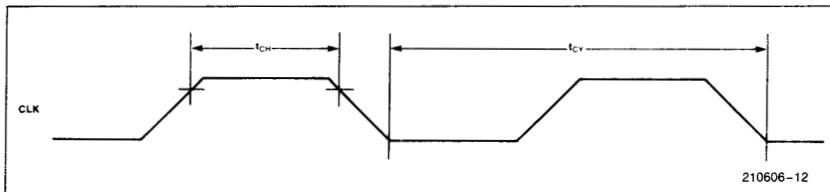
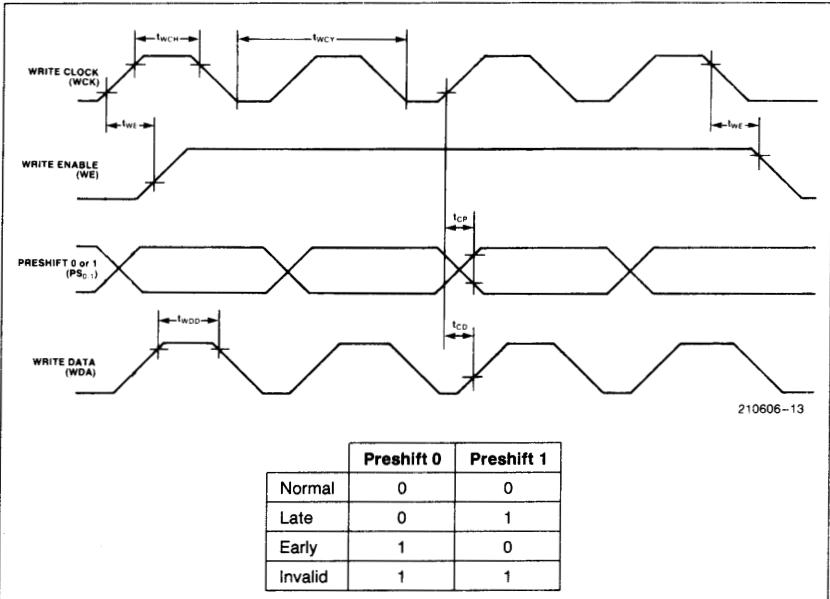
210606-8

$C_L = 100 \text{ pF}$
 C_L Includes Jig Capacitance

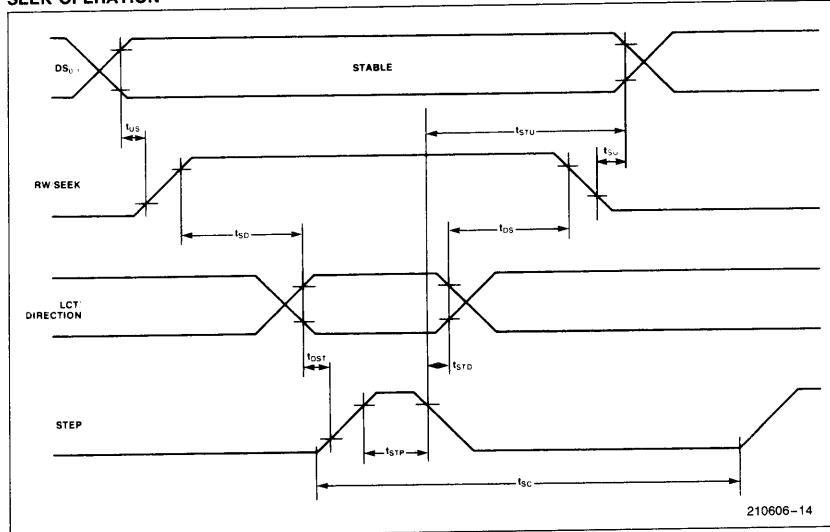
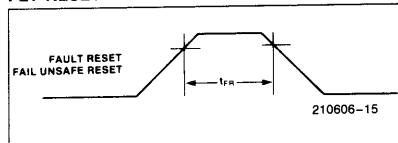
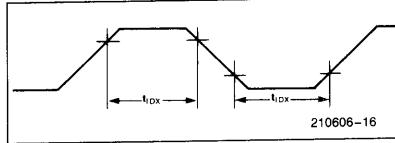
WAVEFORMS**PROCESSOR READ OPERATION**

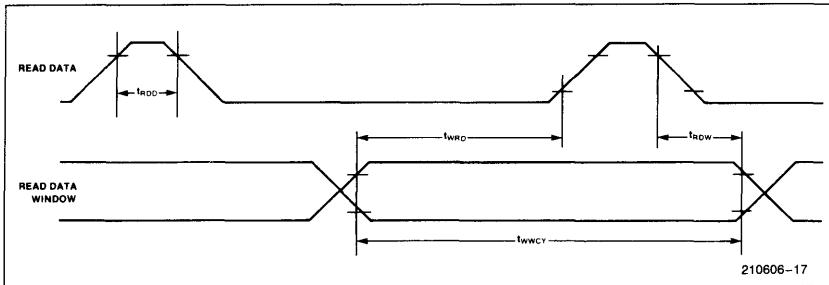
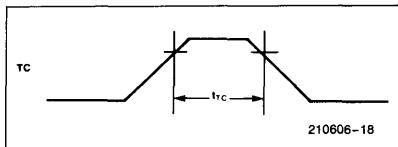
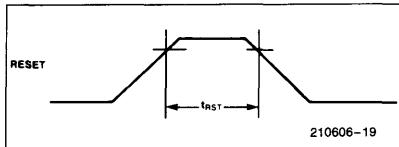
210606-9

WAVEFORMS (Continued)**PROCESSOR WRITE OPERATION****DMA OPERATION**

WAVEFORMS (Continued)**CLOCK TIMING****FDD WRITE OPERATION**

	Preshift 0	Preshift 1
Normal	0	0
Late	0	1
Early	1	0
Invalid	1	1

WAVEFORMS (Continued)**SEEK OPERATION****FLT RESET****INDEX**

WAVEFORMS (Continued)**FDD READ OPERATION****TERMINAL COUNT****RESET**



80286 High Performance Microprocessor with Memory Management and Protection

(80286-12, 80286-10, 80286-8, 80286-6)

- High Performance Processor (Up to six times 8086)
- Large Address Space:
 - 16 Megabytes Physical
 - 1 Gigabyte Virtual per Task
- Integrated Memory Management, Four-Level Memory Protection and Support for Virtual Memory and Operating Systems
- Two 8086 Upward Compatible Operating Modes:
 - 8086 Real Address Mode
 - Protected Virtual Address Mode
- Optional Processor Extension:
 - 80287 High Performance 80-bit Numeric Data Processor
- Available in EXPRESS:
 - Standard Temperature Range
- Range of clock rates
 - 12.5 MHz for 80286-12
 - 10 MHz for 80286-10
 - 8 MHz for 80286-8
 - 6 MHz for 80286-6
- Complete System Development Support:
 - Development Software: Assembler, PL/M, Pascal, FORTRAN, and System Utilities
 - In-Circuit-Emulator (ICE™-286)
- High Bandwidth Bus Interface (12.5 Megabyte/Sec)
- Available in 68 Pin Ceramic LCC (Leadless Chip Carrier) and PGA (Pin Grid Array) Packages

(See Packaging Spec., Order #231369)

The 80286 is an advanced, high-performance microprocessor with specially optimized capabilities for multiple user and multi-tasking systems. The 80286 has built-in memory protection that supports operating system and task isolation as well as program and data privacy within tasks. A 10 MHz 80286 provides five times or more throughput than the standard 5 MHz 8086. The 80286 includes memory management capabilities that map 2^{30} (one gigabyte) of virtual address space per task into 2^{24} bytes (16 megabytes) of physical memory.

The 80286 is upward compatible with 8086 and 88 software. Using 8086 real address mode, the 80286 is object code compatible with existing 8086, 88 software. In protected virtual address mode, the 80286 is source code compatible with 8086, 88 software and may require upgrading to use virtual addresses supported by the 80286's integrated memory management and protection mechanism. Both modes operate at full 80286 performance and execute a superset of the 8086 and 88 instructions.

The 80286 provides special operations to support the efficient implementation and execution of operating systems. For example, one instruction can end execution of one task, save its state, switch to a new task, load its state, and start execution of the new task. The 80286 also supports virtual memory systems by providing a segment-not-present exception and restartable instructions.

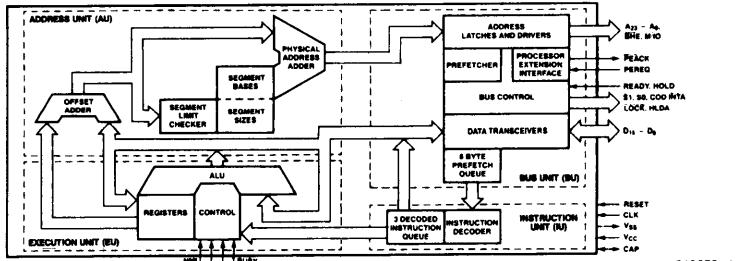
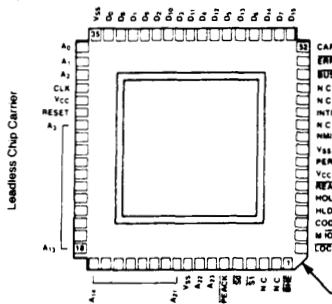


Figure 1. 80286 Internal Block Diagram

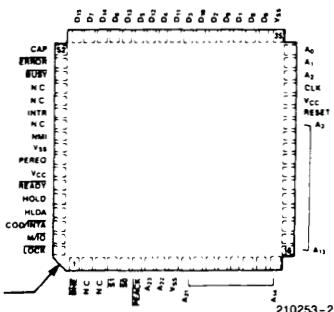
210253-1

Component Pad Views—As viewed from underside of component when mounted on the board.

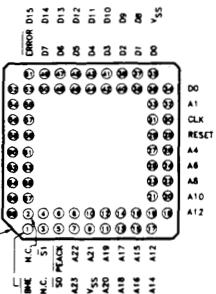
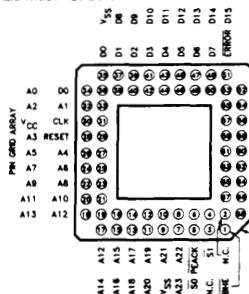


NOTE:
N.C. signals must not be connected

P.C. Board Views—As viewed from the component side of the P.C. board.



210253-2



210253-3

FIGURE 2. 80286 Pin Configuration

Table 1. Pin Description

The following pin function descriptions are for the 80286 microprocessor:

Symbol	Type	Name and Function
CLK	I	SYSTEM CLOCK provides the fundamental timing for 80286 systems. It is divided by two inside the 80286 to generate the processor clock. The internal divide-by-two circuitry can be synchronized to an external clock generator by a LOW to HIGH transition on the RESET input.
D ₁₅ -D ₀	I/O	DATA BUS inputs data during memory, I/O, and interrupt acknowledge read cycles; outputs data during memory and I/O write cycles. The data bus is active HIGH and floats to 3-state OFF during bus hold acknowledgement.
A ₂₃ -A ₀	O	ADDRESS BUS outputs physical memory and I/O port addresses. A ₀ is LOW when data is to be transferred on pins D ₇ -0. A ₂₃ -A ₁₆ are LOW during I/O transfers. The address bus is active HIGH and floats to 3-state OFF during bus hold acknowledgement.
BHE	O	BUS HIGH ENABLE indicates transfer or data on the upper byte of the data bus. D ₁₅ -8. Eight-bit oriented devices assigned to the upper byte of the data bus would normally use BHE to condition chip select functions. BHE is active LOW and floats to 3-state OFF during bus hold acknowledgement.
		BHE and A0 Encodings
		BHE Value A0 Value Function
		0 0 Word transfer
		0 1 Byte transfer on upper half of data bus (D ₁₅ -D ₈)
		1 0 Byte transfer on lower half of data bus (D ₇ -0)
		1 1 Will never occur



80286

Table I. Pin Description (Continued)

Symbol	Type	Name and Function				
S1, S0	O	BUS CYCLE STATUS indicates initiation of a bus cycle and, along with M/I _O and COD/INTA, defines the type of bus cycle. The bus is in a T _s state whenever one or both are LOW, S1 and S0 are active LOW and float to 3-state OFF during bus hold acknowledge.				
80286 Bus Cycle Status Definition						
COD/INTA	M/I _O	S1	S0	Bus Cycle Initiated		
0 (LOW)	0	0	0	Interrupt acknowledge		
0	0	0	1	Will not occur		
0	0	1	0	Will not occur		
0	0	1	1	None; not a status cycle		
0	1	0	0	IF A1 = 1 then halt; else shutdown		
0	1	0	1	Memory data read		
0	1	1	0	Memory data write		
0	1	1	1	None; not a status cycle		
1 (HIGH)	0	0	0	Will not occur		
1	0	0	1	I/O read		
1	0	1	0	I/O write		
1	0	1	1	None; not a status cycle		
1	1	0	0	Will not occur		
1	1	0	1	Memory instruction read		
1	1	1	0	Will not occur		
1	1	1	1	None; not a status cycle		
M/I _O	O	MEMORY I/O SELECT distinguishes memory access from I/O access. If HIGH during T _s , a memory cycle or a halt/shutdown cycle is in progress. If LOW, an I/O cycle or an interrupt acknowledge cycle is in progress. M/I _O floats to 3-state OFF during bus hold acknowledgement.				
COD/INTA	O	CODE/INTERRUPT ACKNOWLEDGE distinguishes instruction fetch cycles from memory data read cycles. Also distinguishes interrupt acknowledge cycles from I/O cycles. COD/INTA floats to 3-state OFF during bus hold acknowledgement. Its timing is the same as M/I _O .				
LOCK	O	BUS LOCK indicates that other system bus masters are not to gain control of the system bus for the current and the following bus cycle. The LOCK signal may be activated explicitly by the "LOCK" instruction prefix or automatically by 80286 hardware during memory XCHG instructions, interrupt acknowledge, or descriptor table access. LOCK is active LOW and floats to 3-state OFF during bus hold acknowledgement.				
READY	I	BUS READY terminates a bus cycle. Bus cycles are extended without limit until terminated by READY LOW. READY is an active LOW synchronous input requiring setup and hold times relative to the system clock to be met for correct operation. READY is ignored during bus hold acknowledgement.				
HOLD HLDA	I O	HOLD REQUEST AND HOLD ACKNOWLEDGE control ownership of the 80286 local bus. The HOLD input allows another local bus master to request control of the local bus. When control is granted, the 80286 will float its bus drivers to 3-state OFF and then activate HLDA, thus entering the bus hold acknowledgement condition. The local bus will remain granted to the requesting master until HOLD becomes inactive which results in the 80286 deactivating HLDA and regaining control of the local bus. This terminates the bus hold acknowledgement condition. HOLD may be asynchronous to the system clock. These signals are active HIGH.				
INTR	I	INTERRUPT REQUEST requests the 80286 to suspend its current program execution and service a pending external request. Interrupt requests are masked whenever the interrupt enable bit in the flag word is cleared. When the 80286 responds to an interrupt request, it performs two interrupt acknowledge bus cycles to read an 8-bit interrupt vector that identifies the source of the interrupt. To assure program interruption, INTR must remain active until the first interrupt acknowledgement cycle is completed. INTR is sampled at the beginning of each processor cycle and must be active HIGH at least two processor cycles before the current instruction ends in order to interrupt before the next instruction. INTR is level sensitive, active HIGH, and may be asynchronous to the system clock.				
NMI	I	NON-MASKABLE INTERRUPT REQUEST interrupts the 80286 with an internally supplied vector value of 2. No interrupt acknowledgement cycles are performed. The interrupt enable bit in the 80286 flag word does not affect this input. The NMI input is active HIGH, may be asynchronous to the system clock, and is edge triggered after internal synchronization. For proper recognition, the input must have been previously LOW for at least four system clock cycles and remain HIGH for at least four system clock cycles.				



80286

Table 1. Pin Description (Continued)

Symbol	Type	Name and Function				
PEREQ PEACK	I O	PROCESSOR EXTENSION OPERAND REQUEST AND ACKNOWLEDGE extend the memory management and protection capabilities of the 80286 to processor extensions. The PEREQ input requests the 80286 to perform a data operand transfer for a processor extension. The PEACK output signals the processor extension when the requested operand is being transferred. PEREQ is active HIGH and floats to 3-state OFF during bus hold acknowledge. PEACK may be asynchronous to the system clock. PEACK is active LOW.				
BUSY ERROR	I I	PROCESSOR EXTENSION BUSY AND ERROR indicate the operating condition of a processor extension to the 80286. An active BUSY input stops 80286 program execution on WAIT and some ESC instructions until BUSY becomes inactive (HIGH). The 80286 may be interrupted while waiting for BUSY to become inactive. An active ERROR input causes the 80286 to perform a processor extension interrupt when executing WAIT or some ESC instructions. These inputs are active LOW and may be asynchronous to the system clock.				
RESET	I	SYSTEM RESET clears the internal logic of the 80286 and is active HIGH. The 80286 may be reinitialized at any time with a LOW to HIGH transition on RESET which remains active for more than 16 system clock cycles. During RESET active, the output pins of the 80286 enter the state shown below: 80286 Pin State During Reset <table border="1"><thead><tr><th>Pin Value</th><th>Pin Names</th></tr></thead><tbody><tr><td>1 (HIGH) 0 (LOW) 3-state OFF</td><td>S0, S1, PEACK, A23-A0, BHE, LOCK M/I/O, COD/INTA, HLDA (Note 1) D15-D0</td></tr></tbody></table> Operation of the 80286 begins after a HIGH to LOW transition on RESET. The HIGH to LOW transition of RESET must be synchronous to the system clock. Approximately 38 CLK cycles from the trailing edge of RESET are required by the 80286 for internal initialization before the first bus cycle, to fetch code from the power-on execution address, occurs. A LOW to HIGH transition of RESET synchronous to the system clock will end a processor cycle at the second HIGH to LOW transition of the system clock. The LOW to HIGH transition of RESET may be asynchronous to the system clock; however, in this case it cannot be predetermined which phase of the processor clock will occur during the next system clock period. Synchronous LOW to HIGH transitions of RESET are required only for systems where the processor clock must be phase synchronous to another clock.	Pin Value	Pin Names	1 (HIGH) 0 (LOW) 3-state OFF	S0, S1, PEACK, A23-A0, BHE, LOCK M/I/O, COD/INTA, HLDA (Note 1) D15-D0
Pin Value	Pin Names					
1 (HIGH) 0 (LOW) 3-state OFF	S0, S1, PEACK, A23-A0, BHE, LOCK M/I/O, COD/INTA, HLDA (Note 1) D15-D0					
V _{SS}	I	SYSTEM GROUND: 0 Volts.				
V _{CC}	I	SYSTEM POWER: +5 Volt Power Supply.				
CAP	I	SUBSTRATE FILTER CAPACITOR: a 0.047 μ F \pm 20% 12V capacitor must be connected between this pin and ground. This capacitor filters the output of the internal substrate bias generator. A maximum DC leakage current of 1 μ A is allowed through the capacitor. For correct operation of the 80286, the substrate bias generator must charge this capacitor to its operating voltage. The capacitor chargeup time is 5 milliseconds (max.) after V _{CC} and CLK reach their specified AC and DC parameters. RESET may be applied to prevent spurious activity by the CPU during this time. After this time, the 80286 processor clock can be synchronized to another clock by pulsing RESET LOW synchronous to the system clock.				

NOTE:

1. HLDA is only Low if HOLD is inactive (Low).

FUNCTIONAL DESCRIPTION

Introduction

The 80286 is an advanced, high-performance microprocessor with specially optimized capabilities for multiple user and multi-tasking systems. Depending on the application, an 8 MHz 80286's performance is up to six times faster than the standard 5 MHz 8086's, while providing complete upward software compatibility with Intel's 8086, 88, and 186 family of CPU's.

The 80286 operates in two modes: 8086 real address mode and protected virtual address mode. Both modes execute a superset of the 8086 and 88 instruction set.

In 8086 real address mode programs use real addresses with up to one megabyte of address space. Programs use virtual addresses in protected virtual address mode, also called protected mode. In protected mode, the 80286 CPU automatically maps 1 gigabyte of virtual addresses per task into a 16 megabyte real address space. This mode also provides memory protection to isolate the operating system and ensure privacy of each tasks' programs and data. Both modes provide the same base instruction set, registers, and addressing modes.

The following Functional Description describes first, the base 80286 architecture common to both modes, second, 8086 real address mode, and third, protected mode.

80286 BASE ARCHITECTURE

The 8086, 88, 186, and 286 CPU family all contain the same basic set of registers, instructions, and

addressing modes. The 80286 processor is upward compatible with the 8086, 8088, and 80186 CPU's.

Register Set

The 80286 base architecture has fifteen registers as shown in Figure 3. These registers are grouped into the following four categories:

General Registers: Eight 16-bit general purpose registers used to contain arithmetic and logical operands. Four of these (AX, BX, CX, and DX) can be used either in their entirety as 16-bit words or split into pairs of separate 8-bit registers.

Segment Registers: Four 16-bit special purpose registers select, at any given time, the segments of memory that are immediately addressable for code, stack, and data. (For usage, refer to Memory Organization.)

Base and Index Registers: Four of the general purpose registers may also be used to determine offset addresses of operands in memory. These registers may contain base addresses or indexes to particular locations within a segment. The addressing mode determines the specific registers used for operand address calculations.

Status and Control Registers: The 3 16-bit special purpose registers in figure 3A record or control certain aspects of the 80286 processor state including the Instruction Pointer, which contains the offset address of the next sequential instruction to be executed.

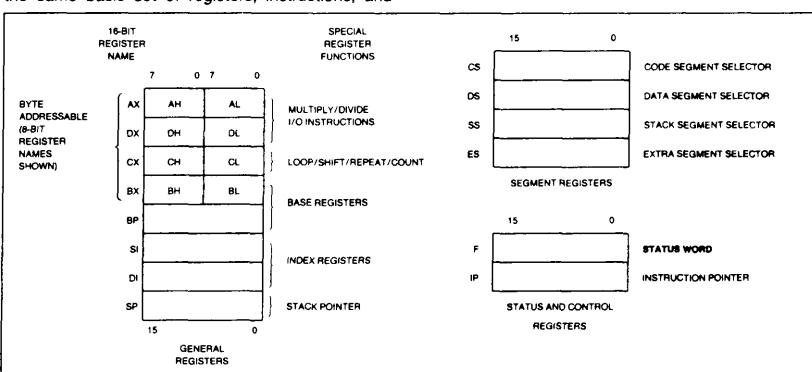


Figure 3. Register Set

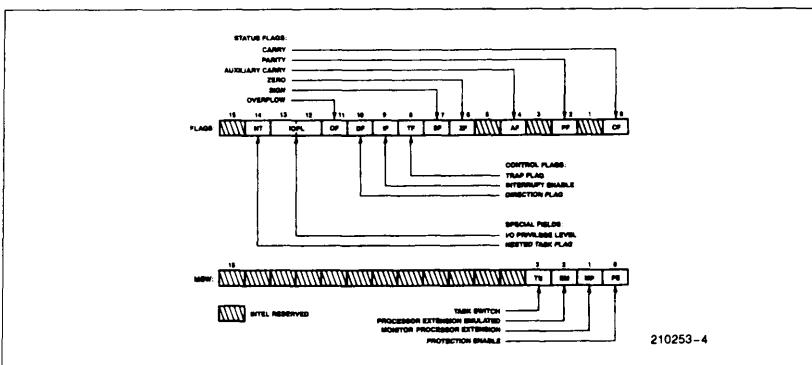


Figure 3a. Status and Control Register Bit Functions

Flags Word Description

The Flags word (Flags) records specific characteristics of the result of logical and arithmetic instructions (bits 0, 2, 4, 6, 7, and 11) and controls the operation of the 80286 within a given operating mode (bits 8 and 9). Flags is a 16-bit register. The function of the flag bits is given in Table 2.

Instruction Set

The instruction set is divided into seven categories: data transfer, arithmetic, shift/rotate/logical, string manipulation, control transfer, high level instructions, and processor control. These categories are summarized in Figure 4.

An 80286 instruction can reference zero, one, or two operands; where an operand resides in a register, in the instruction itself, or in memory. Zero-operand instructions (e.g. NOP and HLT) are usually one byte long. One-operand instructions (e.g. INC and DEC) are usually two bytes long but some are encoded in only one byte. One-operand instructions may reference a register or memory location. Two-operand instructions permit the following six types of instruction operations:

- Register to Register
- Memory to Register
- Immediate to Register
- Memory to Memory
- Register to Memory
- Immediate to Memory

Table 2. Flags Word Bit Functions

Bit Position	Name	Function
0	CF	Carry Flag—Set on high-order bit carry or borrow; cleared otherwise
2	PF	Parity Flag—Set if low-order 8 bits of result contain an even number of 1-bits; cleared otherwise
4	AF	Set on carry from or borrow to the low order four bits of AL; cleared otherwise
6	ZF	Zero Flag—Set if result is zero; cleared otherwise
7	SF	Sign Flag—Set equal to high-order bit of result (0 if positive, 1 if negative)
11	OF	Overflow Flag—Set if result is a too-large positive number or a too-small negative number (excluding sign-bit) to fit in destination operand; cleared otherwise
8	TF	Single Step Flag—Once set, a single step interrupt occurs after the next instruction executes. TF is cleared by the single step interrupt.
9	IF	Interrupt-enable Flag—When set, maskable interrupts will cause the CPU to transfer control to an interrupt vector specified location.
10	DF	Direction Flag—Causes string instructions to auto decrement the appropriate index registers when set. Clearing DF causes auto increment.

Two-operand instructions (e.g. MOV and ADD) are usually three to six bytes long. Memory to memory operations are provided by a special class of string instructions requiring one to three bytes. For detailed instruction formats and encodings refer to the instruction set summary at the end of this document.

For detailed operation and usage of each instruction, see Appendix of 80286 Programmer's Reference Manual (Order No. 210498)

GENERAL PURPOSE	
MOV	Move byte or word
PUSH	Push word onto stack
POP	Pop word off stack
PUSHA	Push all registers on stack
POPA	Pop all registers from stack
XCHG	Exchange byte or word
XLAT	Translate byte
INPUT/OUTPUT	
IN	Input byte or word
OUT	Output byte or word
ADDRESS OBJECT	
LEA	Load effective address
LDS	Load pointer using DS
LES	Load pointer using ES
FLAG TRANSFER	
LAHF	Load AH register from flags
SAHF	Store AH register in flags
PUSHF	Push flags onto stack
POPF	Pop flags off stack

Figure 4a. Data Transfer Instructions

MOVS	Move byte or word string
INS	Input bytes or word string
OUTS	Output bytes or word string
CMPS	Compare byte or word string
SCAS	Scan byte or word string
LODS	Load byte or word string
STOS	Store byte or word string
REP	Repeat
REPE/REPZ	Repeat while equal/zero
REPNE/REPNZ	Repeat while not equal/not zero

Figure 4c. String Instructions

ADDITION	
ADD	Add byte or word
ADC	Add byte or word with carry
INC	Increment byte or word by 1
AAA	ASCII adjust for addition
DAA	Decimal adjust for addition
SUBTRACTION	
SUB	Subtract byte or word
SBB	Subtract byte or word with borrow
DEC	Decrement byte or word by 1
NEG	Negate byte or word
CMP	Compare byte or word
AAS	ASCII adjust for subtraction
DAS	Decimal adjust for subtraction
MULTIPLICATION	
MUL	Multiple byte or word unsigned
IMUL	Integer multiply byte or word
AAM	ASCII adjust for multiply
DIVISION	
DIV	Divide byte or word unsigned
IDIV	Integer divide byte or word
AAD	ASCII adjust for division
CBW	Convert byte to word
CWD	Convert word to doubleword

Figure 4b. Arithmetic Instructions

LOGICALS	
NOT	"Not" byte or word
AND	"And" byte or word
OR	"Inclusive or" byte or word
XOR	"Exclusive or" byte or word
TEST	"Test" byte or word
SHIFTS	
SHL/SAL	Shift logical/arithmetic left byte or word
SHR	Shift logical right byte or word
SAR	Shift arithmetic right byte or word
ROTATES	
ROL	Rotate left byte or word
ROR	Rotate right byte or word
RCL	Rotate through carry left byte or word
RCR	Rotate through carry right byte or word

Figure 4d. Shift/Rotate Logical Instructions

CONDITIONAL TRANSFERS		UNCONDITIONAL TRANSFERS	
JA/JNBE	Jump if above/not below nor equal	CALL	Call procedure
JAE/JNB	Jump if above or equal/not below	RET	Return from procedure
JB/JNAE	Jump if below/not above nor equal	JMP	Jump
JBE/JNA	Jump if below or equal/not above		
JC	Jump if carry	ITERATION CONTROLS	
JE/JZ	Jump if equal/zero	LOOP	Loop
JG/JNLE	Jump if greater/not less nor equal	LOOPE/LOOPZ	Loop if equal/zero
JGE/JNL	Jump if greater or equal/not less	LOOPNE/LOOPNZ	Loop if not equal/not zero
JL/JNGE	Jump if less/not greater nor equal	JCXZ	Jump if register CX = 0
JLE/JNG	Jump if less or equal/not greater		
JNC	Jump if not carry	INTERRUPTS	
JNE/JNZ	Jump if not equal/not zero	INT	Interrupt
JNO	Jump if not overflow	INTO	Interrupt if overflow
JNP/JPO	Jump if not parity/parity odd		
JNS	Jump if not sign	IRET	Interrupt return
JO	Jump if overflow		
JP/JPE	Jump if parity/parity even		
JS	Jump if sign		

Figure 4e. Program Transfer Instructions

FLAG OPERATIONS	
STC	Set carry flag
CLC	Clear carry flag
CMC	Complement carry flag
STD	Set direction flag
CLD	Clear direction flag
STI	Set interrupt enable flag
CLI	Clear interrupt enable flag
EXTERNAL SYNCHRONIZATION	
HLT	Halt until interrupt or reset
WAIT	Wait for BUSY not active
ESC	Escape to extension processor
LOCK	Lock bus during next instruction
NO OPERATION	
NOP	No operation
EXECUTION ENVIRONMENT CONTROL	
LMSW	Load machine status word
SMSW	Store machine status word

Figure 4f. Processor Control Instructions

ENTER	Format stack for procedure entry
LEAVE	Restore stack for procedure exit
BOUND	Detects values outside prescribed range

Figure 4g. High Level Instructions

Memory Organization

Memory is organized as sets of variable length segments. Each segment is a linear contiguous sequence of up to 64K (2¹⁶) 8-bit bytes. Memory is addressed using a two component address (a pointer) that consists of a 16-bit segment selector, and a 16-bit offset. The segment selector indicates the desired segment in memory. The offset component indicates the desired byte address within the segment.

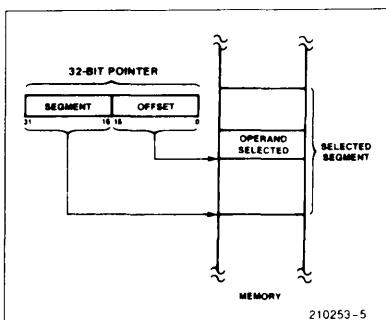


Figure 5. Two Component Address

Table 3. Segment Register Selection Rules

Memory Reference Needed	Segment Register Used	Implicit Segment Selection Rule
Instructions	Code (CS)	Automatic with instruction prefetch
Stack	Stack (SS)	All stack pushes and pops. Any memory reference which uses BP as a base register.
Local Data	Data (DS)	All data references except when relative to stack or string destination
External (Global) Data	Extra (ES)	Alternate data segment and destination of string operation

All instructions that address operands in memory must specify the segment and the offset. For speed and compact instruction encoding, segment selectors are usually stored in the high speed segment registers. An instruction need specify only the desired segment register and an offset in order to address a memory operand.

Most instructions need not explicitly specify which segment register is used. The correct segment register is automatically chosen according to the rules of Table 3. These rules follow the way programs are written (see Figure 6) as independent modules that require areas for code and data, a stack, and access to external data areas.

Special segment override instruction prefixes allow the implicit segment register selection rules to be overridden for special cases. The stack, data, and extra segments may coincide for simple programs. To access operands not residing in one of the four immediately available segments, a full 32-bit pointer or a new segment selector must be loaded.

Addressing Modes

The 80286 provides a total of eight addressing modes for instructions to specify operands. Two addressing modes are provided for instructions that operate on register or immediate operands:

Register Operand Mode: The operand is located in one of the 8 or 16-bit general registers.

Immediate Operand Mode: The operand is included in the instruction.

Six modes are provided to specify the location of an operand in a memory segment. A memory operand address consists of two 16-bit components: segment selector and offset. The segment selector is supplied by a segment register either implicitly chosen by the addressing mode or explicitly chosen by a segment override prefix. The offset is calculated by summing any combination of the following three address elements:

the **displacement** (an 8 or 16-bit immediate value contained in the instruction)

the **base** (contents of either the BX or BP base registers)

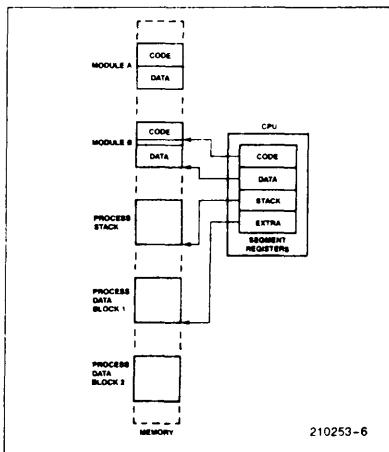


Figure 6. Segmented Memory Helps Structure Software

the **index** (contents of either the SI or DI index registers)

Any carry out from the 16-bit addition is ignored. Eight-bit displacements are sign extended to 16-bit values.

Combinations of these three address elements define the six memory addressing modes, described below.

Direct Mode: The operand's offset is contained in the instruction as an 8 or 16-bit displacement element.

Register Indirect Mode: The operand's offset is in one of the registers SI, DI, BX, or BP.

Based Mode: The operand's offset is the sum of an 8 or 16-bit displacement and the contents of a base register (BX or BP).

Indexed Mode: The operand's offset is the sum of an 8 or 16-bit displacement and the contents of an index register (SI or DI).

Based Indexed Mode: The operand's offset is the sum of the contents of a base register and an index register.

Based Indexed Mode with Displacement: The operand's offset is the sum of a base register's contents, an index register's contents, and an 8 or 16-bit displacement.

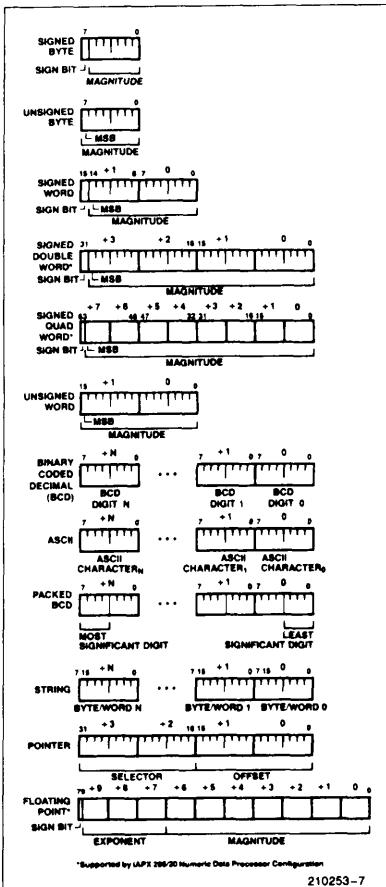
Data Types

The 80286 directly supports the following data types:

Integer:	A signed binary numeric value contained in an 8-bit byte or a 16-bit word. All operations assume a 2's complement representation. Signed 32 and 64-bit integers are supported using the Numeric Data Processor, the 80287.
Ordinal:	An unsigned binary numeric value contained in an 8-bit byte or 16-bit word.
Pointer:	A 32-bit quantity, composed of a segment selector component and an offset component. Each component is a 16-bit word.
String:	A contiguous sequence of bytes or words. A string may contain from 1 byte to 64K bytes.
ASCII:	A byte representation of alphanumeric and control characters using the ASCII standard of character representation.
BCD:	A byte (unpacked) representation of the decimal digits 0-9.
Packed BCD:	A byte (packed) representation of two decimal digits 0-9 storing one digit in each nibble of the byte.
Floating Point:	A signed 32, 64, or 80-bit real number representation. (Floating point operands are supported using the 80287 Numeric Processor).

Figure 7 graphically represents the data types supported by the 80286.

either an 8-bit port address, specified in the instruction, or a 16-bit port address in the DX register. 8-bit port addresses are zero extended such that A₁₅-A₈ are LOW. I/O port addresses 00F8(H) through 0OFF(H) are reserved.



*Supported by iAPX 386/32 Numeric Data Processor Configuration

210253-7

Figure 7. 80286 Supported Data Types

I/O Space

The I/O space consists of 64K 8-bit or 32K 16-bit ports. I/O instructions address the I/O space with

Table 4. Interrupt Vector Assignments

Function	Interrupt Number	Related Instructions	Does Return Address Point to Instruction Causing Exception?
Divide error exception	0	DIV, IDIV	Yes
Single step interrupt	1	All	
NMI interrupt	2	INT 2 or NMI pin	
Breakpoint interrupt	3	INT 3	
INTO detected overflow exception	4	INTO	No
BOUND range exceeded exception	5	BOUND	Yes
Invalid opcode exception	6	Any undefined opcode	Yes
Processor extension not available exception	7	ESC or WAIT	Yes
Intel reserved—do not use	8-15		
Processor extension error interrupt	16	ESC or WAIT	
Intel reserved—do not use	17-31		
User defined	32-255		

Interrupts

An interrupt transfers execution to a new program location. The old program address (CS:IP) and machine state (Flags) are saved on the stack to allow resumption of the interrupted program. Interrupts fall into three classes: hardware initiated, INT instructions, and instruction exceptions. Hardware initiated interrupts occur in response to an external input and are classified as non-maskable or maskable. Programs may cause an interrupt with an INT instruction. Instruction exceptions occur when an unusual condition, which prevents further instruction processing, is detected while attempting to execute an instruction. The return address from an exception will always point at the instruction causing the exception and include any leading instruction prefixes.

A table containing up to 256 pointers defines the proper interrupt service routine for each interrupt. Interrupts 0–31, some of which are used for instruction exceptions, are reserved. For each interrupt, an 8-bit vector must be supplied to the 80286 which identifies the appropriate table entry. Exceptions supply the interrupt vector internally. INT instructions contain or imply the vector and allow access to all 256 interrupts. Maskable hardware initiated interrupts supply the 8-bit vector to the CPU during an interrupt acknowledge bus sequence. Non-maskable hardware interrupts use a predefined internally supplied vector.

MASKABLE INTERRUPT (INTR)

The 80286 provides a maskable hardware interrupt request pin, INTR. Software enables this input by

setting the interrupt flag bit (IF) in the flag word. All 224 user-defined interrupt sources can share this input, yet they can retain separate interrupt handlers. An 8-bit vector read by the CPU during the interrupt acknowledge sequence (discussed in System Interface section) identifies the source of the interrupt.

Further maskable interrupts are disabled while servicing an interrupt by resetting the IF but as part of the response to an interrupt or exception. The saved flag word will reflect the enable status of the processor prior to the interrupt. Until the flag word is restored to the flag register, the interrupt flag will be zero unless specifically set. The interrupt return instruction includes restoring the flag word, thereby restoring the original status of IF.

NON-MASKABLE INTERRUPT REQUEST (NMI)

A non-maskable interrupt input (NMI) is also provided. NMI has higher priority than INTR. A typical use of NMI would be to activate a power failure routine. The activation of this input causes an interrupt with an internally supplied vector value of 2. No external interrupt acknowledge sequence is performed.

While executing the NMI servicing procedure, the 80286 will service neither further NMI requests, INTR requests, nor the processor extension segment overrun interrupt until an interrupt return (IRET) instruction is executed or the CPU is reset. If NMI occurs while currently servicing an NMI, its presence will be saved for servicing after executing the first IRET instruction. IF is cleared at the beginning of an NMI interrupt to inhibit INTR interrupts.

SINGLE STEP INTERRUPT

The 80286 has an internal interrupt that allows programs to execute one instruction at a time. It is called the single step interrupt and is controlled by the single step flag bit (TF) in the flag word. Once this bit is set, an internal single step interrupt will occur after the next instruction has been executed. The interrupt clears the TF bit and uses an internally supplied vector of 1. The IRET instruction is used to set the TF bit and transfer control to the next instruction to be single stepped.

Interrupt Priorities

When simultaneous interrupt requests occur, they are processed in a fixed order as shown in Table 5. Interrupt processing involves saving the flags, return address, and setting CS:I to point at the first instruction of the interrupt handler. If other interrupts remain enabled they are processed before the first instruction of the current interrupt handler is executed. The last interrupt processed is therefore the first one serviced.

Table 5. Interrupt Processing Order

Order	Interrupt
1	Instruction exception
2	Single step
3	NMI
4	Processor extension segment overrun
5	INTR
6	INT instruction

Initialization and Processor Reset

Processor initialization or start up is accomplished by driving the RESET input pin HIGH. RESET forces the 80286 to terminate all execution and local bus activity. No instruction or bus activity will occur as long as RESET is active. After RESET becomes inactive and an internal processing interval elapses, the 80286 begins execution in real address mode with the instruction at physical location FFFF0(H). RESET also sets some registers to predefined values as shown in Table 6.

Table 6. 80286 Initial Register State after RESET

Flag word	0002(H)
Machine Status Word	FFFO(H)
Instruction pointer	FFFO(H)
Code segment	F000(H)
Data segment	0000(H)
Extra segment	0000(H)
Stack segment	0000(H)

HOLD must not be active during the time from the leading edge of the initial RESET to 34 CLKs after the trailing edge of the initial RESET of an 80286 system.

Machine Status Word Description

The machine status word (MSW) records when a task switch takes place and controls the operating mode of the 80286. It is a 16-bit register of which the lower four bits are used. One bit places the CPU into protected mode, while the other three bits, as shown in Table 7, control the processor extension interface. After RESET, this register contains FFF0(H) which places the 80286 in 8086 real address mode.

Table 7. MSW Bit Functions

Bit Position	Name	Function
0	PE	Protected mode enable places the 80286 into protected mode and cannot be cleared except by RESET.
1	MP	Monitor processor extension allows WAIT instructions to cause a processor extension not present exception (number 7).
2	EM	Emulate processor extension causes a processor extension not present exception (number 7) on ESC instructions to allow emulating a processor extension.
3	TS	Task switched indicates the next instruction using a processor extension will cause exception 7, allowing software to test whether the current processor extension context belongs to the current task.

The LMSW and SMSW instructions can load and store the MSW in real address mode. The recommended use of TS, EM, and MP is shown in Table 8.

Table 8. Recommended MSW Encodings For Processor Extension Control

TS	MP	EM	Recommended Use	Instructions Causing Exception 7
0	0	0	Initial encoding after RESET. 80286 operation is identical to 8086, 88.	None
0	0	1	No processor extension is available. Software will emulate its function.	ESC
1	0	1	No processor extension is available. Software will emulate its function. The current processor extension context may belong to another task.	ESC
0	1	0	A processor extension exists.	None
1	1	0	A processor extension exists. The current processor extension context may belong to another task. The Exception 7 on WAIT allows software to test for an error pending from a previous processor extension operation.	ESC or WAIT

Halt

The HLT instruction stops program execution and prevents the CPU from using the local bus until restarted. Either NMI, INTR with IF = 1, or RESET will force the 80286 out of halt. If interrupted, the saved CS:IP will point to the next instruction after the HLT.

8086 REAL ADDRESS MODE

The 80286 executes a fully upward-compatible superset of the 8086 instruction set in real address mode. In real address mode the 80286 is object code compatible with 8086 and 8088 software. The real address mode architecture (registers and addressing modes) is exactly as described in the 80286 Base Architecture section of this Functional Description.

Memory Size

Physical memory is a contiguous array of up to 1,048,576 bytes (one megabyte) addressed by pins A₀ through A₁₉ and BHE. A₂₀ through A₂₃ should be ignored.

Memory Addressing

In real address mode physical memory is a contiguous array of up to 1,048,576 bytes (one megabyte) addressed by pins A₀ through A₁₉ and BHE. Address bits A₂₀-A₂₃ may not always be zero in real mode. A₂₀-A₂₃ should not be used by the system while the 80286 is operating in Real Mode.

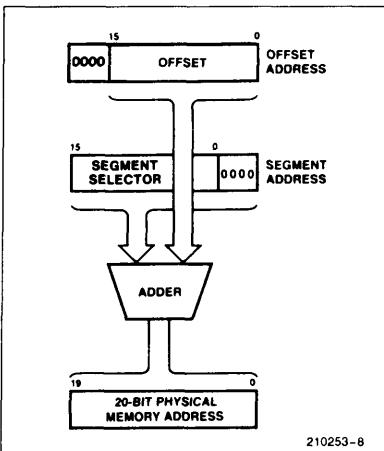
The selector portion of a pointer is interpreted as the upper 16 bits of a 20-bit segment address. The lower four bits of the 20-bit segment address are always zero. Segment addresses, therefore, begin on multiples of 16 bytes. See Figure 8 for a graphic representation of address information.

All segments in real address mode are 64K bytes in size and may be read, written, or executed. An exception or interrupt can occur if data operands or instructions attempt to wrap around the end of a segment (e.g. a word with its low order byte at offset FFFF(H) and its high order byte at offset 0000(H)). If, in real address mode, the information contained in a segment does not use the full 64K bytes, the unused end of the segment may be overlaid by another segment to reduce physical memory requirements.

Reserved Memory Locations

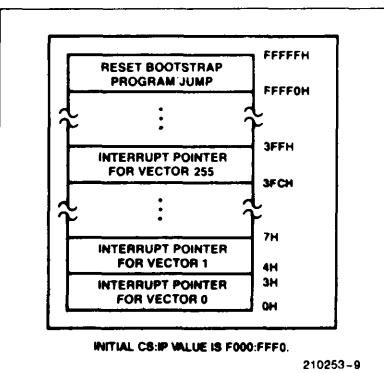
The 80286 reserves two fixed areas of memory in real address mode (see Figure 9); system initializa-

tion area and interrupt table area. Locations from addresses FFFF0(H) through FFFF(H) are reserved for system initialization. Initial execution begins at location FFFF0(H). Locations 00000(H) through 003FF(H) are reserved for interrupt vectors.



210253-8

Figure 8. 8086 Real Address Mode Address Calculation



210253-9

Figure 9. 8086 Real Address Mode Initially Reserved Memory Locations

Table 9. Real Address Mode Addressing Interrupts

Function	Interrupt Number	Related Instructions	Return Address Before Instruction?
Interrupt table limit too small exception	8	INT vector is not within table limit	Yes
Processor extension segment overrun interrupt	9	ESC with memory operand extending beyond offset FFFF(H)	No
Segment overrun exception	13	Word memory reference with offset = FFFF(H) or an attempt to execute past the end of a segment	Yes

Interrupts

Table 9 shows the interrupt vectors reserved for exceptions and interrupts which indicate an addressing error. The exceptions leave the CPU in the state existing before attempting to execute the failing instruction (except for PUSH, POP, PUSHA, or POPA). Refer to the next section on protected mode initialization for a discussion on exception 8.

Protected Mode Initialization

To prepare the 80286 for protected mode, the LIDT instruction is used to load the 24-bit interrupt table base and 16-bit limit for the protected mode interrupt table. This instruction can also set a base and limit for the interrupt vector table in real address mode. After reset, the interrupt table base is initialized to 000000(H) and its size set to 03FF(H). These values are compatible with 8086, 88 software. LIDT should only be executed in preparation for protected mode.

Shutdown

Shutdown occurs when a severe error is detected that prevents further instruction processing by the CPU. Shutdown and halt are externally signalled via a halt bus operation. They can be distinguished by A₁ HIGH for halt and A₁ LOW for shutdown. In real address mode, shutdown can occur under two conditions:

- Exceptions 8 or 13 happen and the IDT limit does not include the interrupt vector.
- A CALL INT or PUSH instruction attempts to wrap around the stack segment when SP is not even.

An NMI input can bring the CPU out of shutdown if the IDT limit is at least 000F(H) and SP is greater than 0005(H), otherwise shutdown can only be exited via the RESET input.

PROTECTED VIRTUAL ADDRESS MODE

The 80286 executes a fully upward-compatible subset of the 8086 instruction set in protected virtual address mode (protected mode). Protected mode also provides memory management and protection mechanisms and associated instructions.

The 80286 enters protected virtual address mode from real address mode by setting the PE (Protection Enable) bit of the machine status word with the Load Machine Status Word (LMSW) instruction. Protected mode offers extended physical and virtual memory address space, memory protection mechanisms, and new operations to support operating systems and virtual memory.

All registers, instructions, and addressing modes described in the 80286 Base Architecture section of this Functional Description remain the same. Programs for the 8086, 88, 186, and real address mode 80286 can be run in protected mode; however, embedded constants for segment selectors are different.

Memory Size

The protected mode 80286 provides a 1 gigabyte virtual address space per task mapped into a 16 megabyte physical address space defined by the address pin A_{23..A₀} and BHE. The virtual address space may be larger than the physical address space since any use of an address that does not map to a physical memory location will cause a restartable exception.

Memory Addressing

As in real address mode, protected mode uses 32-bit pointers, consisting of 16-bit selector and offset components. The selector, however, specifies an index into a memory resident table rather than the upper 16-bits of a real memory address. The 24-bit

base address of the desired segment is obtained from the tables in memory. The 16-bit offset is added to the segment base address to form the physical address as shown in Figure 10. The tables are automatically referenced by the CPU whenever a segment register is loaded with a selector. All 80286 instructions which load a segment register will reference the memory based tables without additional software. The memory based tables contain 8 byte values called descriptors.

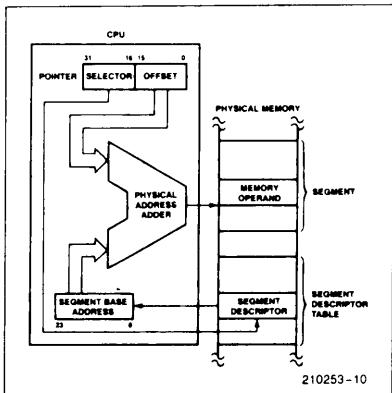


Figure 10. Protected Mode Memory Addressing

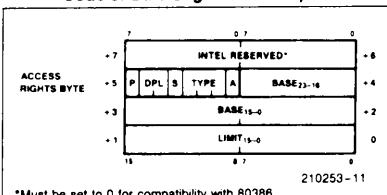
DESCRIPTORS

Descriptors define the use of memory. Special types of descriptors also define new functions for transfer of control and task switching. The 80286 has segment descriptors for code, stack and data segments, and system control descriptors for special system data segments and control transfer operations. Descriptor accesses are performed as locked bus operations to assure descriptor integrity in multi-processor systems.

CODE AND DATA SEGMENT DESCRIPTORS (S = 1)

Besides segment base addresses, code and data descriptors contain other segment attributes including segment size (1 to 64K bytes), access rights (read only, read/write, execute only, and execute/read), and presence in memory (for virtual memory systems) (See Figure 11). Any segment usage violating a segment attribute indicated by the segment descriptor will prevent the memory cycle and cause an exception or interrupt.

Code or Data Segment Descriptor



*Must be set to 0 for compatibility with 80386

Access Rights Byte Definition

Bit Position		Name	Function						
Type Field Definition	7	Present (P)	P = 1	Segment is mapped into physical memory					
	6-5	Descriptor Privilege Level (DPL)	P = 0	No mapping to physical memory exists, base and limit are not used.					
	4	Segment Descriptor (S)	S = 1	Code or Data (includes stacks) segment descriptor					
	3	Executable (E)	E = 0	Data segment descriptor type is:	If Data Segment (S = 1, E = 0)				
	2	Expansion Direction (ED)	ED = 0	Expand up segment, offsets must be \leq limit.					
	1	Writable (W)	ED = 1	Expand down segment, offsets must be $>$ limit.					
	3	Executable (E)	W = 0	Data segment may not be written into.					
	2	Conforming (C)	W = 1	Data segment may be written into.					
	1	Readable (R)	E = 1	Code Segment Descriptor type is:	If Code Segment (S = 1, E = 1)				
	0	Accessed (A)	C = 1	Code segment may only be executed when CPL \geq DPL and CPL remains unchanged.					
			R = 0	Code segment may not be read					
			R = 1	Code segment may be read.					

Figure 11. Code and Data Segment Descriptor Formats

Code and data (including stack data) are stored in two types of segments: code segments and data segments. Both types are identified and defined by segment descriptors ($S = 1$). Code segments are identified by the executable (E) bit set to 1 in the descriptor access rights byte. The access rights byte of both code and data segment descriptor types have three fields in common: present (P) bit, Descriptor Privilege Level (DPL), and accessed (A) bit. If $P = 0$, any attempted use of this segment will cause a not-present exception. DPL specifies the privilege level of the segment descriptor. DPL controls when the descriptor may be used by a task (refer to privilege discussion below). The A bit shows whether the segment has been previously accessed for usage profiling, a necessity for virtual memory systems. The CPU will always set this bit when accessing the descriptor.

Data segments ($S = 1, E = 0$) may be either read-only or read-write as controlled by the W bit of the access rights byte. Read-only ($W = 0$) data segments may not be written into. Data segments may grow in two directions, as determined by the Expansion Direction (ED) bit: upwards ($ED = 0$) for data segments, and downwards ($ED = 1$) for a segment containing a stack. The limit field for a data segment descriptor is interpreted differently depending on the ED bit (see Figure 11).

A code segment ($S = 1, E = 1$) may be execute-only or execute/read as determined by the Readable (R) bit. Code segments may never be written into and execute-only code segments ($R = 0$) may not be read. A code segment may also have an attribute called conforming (C). A conforming code segment may be shared by programs that execute at different privilege levels. The DPL of a conforming code segment defines the range of privilege levels at which the segment may be executed (refer to privilege discussion below). The limit field identifies the last byte of a code segment.

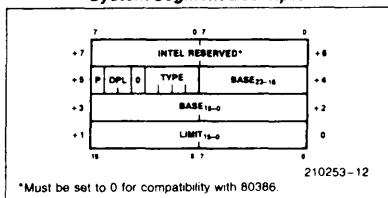
SYSTEM SEGMENT DESCRIPTORS ($S = 0$, TYPE = 1-3)

In addition to code and data segment descriptors, the protected mode 80286 defines System Segment Descriptors. These descriptors define special system data segments which contain a table of descriptors (Local Descriptor Table Descriptor) or segments which contain the execution state of a task (Task State Segment Descriptor).

Figure 12 gives the formats for the special system data segment descriptors. The descriptors contain a 24-bit base address of the segment and a 16-bit limit. The access byte defines the type of descriptor, its state and privilege level. The descriptor contents are valid and the segment is in physical memory if $P = 1$. If $P = 0$, the segment is not valid. The DPL field is only used in Task State Segment descriptors and indicates the privilege level at which the descriptor

may be used (see Privilege). Since the Local Descriptor Table descriptor may only be used by a special privileged instruction, the DPL field is not used. Bit 4 of the access byte is 0 to indicate that it is a system control descriptor. The type field specifies the descriptor type as indicated in Figure 12.

System Segment Descriptor



System Segment Descriptor Fields

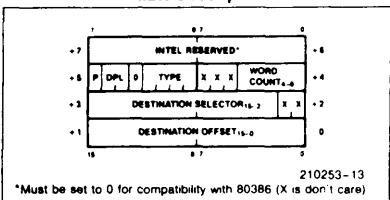
Name	Value	Description
TYPE	1	Available Task State Segment (TSS)
	2	Local Descriptor Table
	3	Busy Task State Segment (TSS)
P	0	Descriptor contents are not valid
	1	Descriptor contents are valid
DPL	0-3	Descriptor Privilege Level
BASE	24-bit number	Base Address of special system data segment in real memory
LIMIT	16-bit number	Offset of last byte in segment

Figure 12. System Segment Descriptor Format

GATE DESCRIPTORS ($S = 0$, TYPE = 4-7)

Gates are used to control access to entry points within the target code segment. The gate descriptors are call gates, task gates, interrupt gates and trap gates. Gates provide a level of indirection between the source and destination of the control transfer. This indirection allows the CPU to automatically perform protection checks and control entry point of the destination. Call gates are used to change privilege levels (see Privilege), task gates are used to perform a task switch, and interrupt and trap gates are used to specify interrupt service routines. The interrupt gate disables interrupts (resets IF) while the trap gate does not.

Gate Descriptor



Gate Descriptor Fields

Name	Value	Description
TYPE	4	-Call Gate
	5	-Task Gate
	6	-Interrupt Gate
	7	-Trap Gate
P	0	-Descriptor Contents are not valid
	1	-Descriptor Contents are valid
DPL	0-3	Descriptor Privilege Level
WORD COUNT	0-31	Number of words to copy from callers stack to called procedures stack. Only used with call gate.
DESTINATION SELECTOR	16-bit selector	Selector to the target code segment (Call, Interrupt or Trap Gate) Selector to the target task state segment (Task Gate)
DESTINATION OFFSET	16-bit offset	Entry point within the target code segment

Figure 13. Gate Descriptor Format

Figure 13 shows the format of the gate descriptors. The descriptor contains a destination pointer that points to the descriptor of the target segment and the entry point offset. The destination selector in an interrupt gate, trap gate, and call gate must refer to a code segment descriptor. These gate descriptors contain the entry point to prevent a program from constructing and using an illegal entry point. Task gates may only refer to a task state segment. Since task gates invoke a task switch, the destination offset is not used in the task gate.

Exception 13 is generated when the gate is used if a destination selector does not refer to the correct descriptor type. The word count field is used in the call gate descriptor to indicate the number of parameters (0-31 words) to be automatically copied from the caller's stack to the stack of the called routine when a control transfer changes privilege levels. The word count field is not used by any other gate descriptor.

The access byte format is the same for all gate descriptors. P = 1 indicates that the gate contents are valid. P = 0 indicates the contents are not valid and causes exception 11 if referenced. DPL is the de-

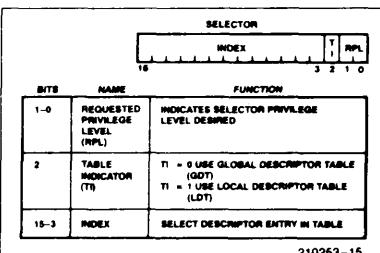
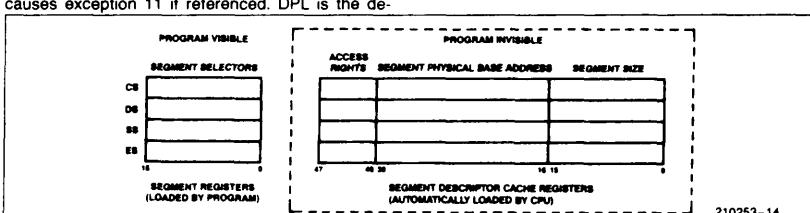
scriptor privilege level and specifies when this descriptor may be used by a task (refer to privilege discussion below). Bit 4 must equal 0 to indicate a system control descriptor. The type field specifies the descriptor type as indicated in Figure 13.

SEGMENT DESCRIPTOR CACHE REGISTERS

A segment descriptor cache register is assigned to each of the four segment registers (CS, SS, DS, ES). Segment descriptors are automatically loaded (cached) into a segment descriptor cache register (Figure 14) whenever the associated segment register is loaded with a selector. Only segment descriptors may be loaded into segment descriptor cache registers. Once loaded, all references to that segment of memory use the cached descriptor information instead of reaccessing the descriptor. The descriptor cache registers are not visible to programs. No instructions exist to store their contents. They only change when a segment register is loaded.

SELECTOR FIELDS

A protected mode selector has three fields: descriptor entry index, local or global descriptor table indicator (TI), and selector privilege (RPL) as shown in Figure 15. These fields select one of two memory based tables of descriptors, select the appropriate table entry and allow highspeed testing of the selector's privilege attribute (refer to privilege discussion below).

**Figure 15. Selector Fields****Figure 14. Descriptor Cache Registers**

LOCAL AND GLOBAL DESCRIPTOR TABLES

Two tables of descriptors, called descriptor tables, contain all descriptors accessible by a task at any given time. A descriptor table is a linear array of up to 8192 descriptors. The upper 13 bits of the selector value are an index into a descriptor table. Each table has a 24-bit base register to locate the descriptor table in physical memory and a 16-bit limit register that confine descriptor access to the defined limits of the table as shown in Figure 16. A restorable exception (13) will occur if an attempt is made to reference a descriptor outside the table limits.

One table, called the Global Descriptor table (GDT), contains descriptors available to all tasks. The other table, called the Local Descriptor Table (LDT), contains descriptors that can be private to a task. Each task may have its own private LDT. The GDT may contain all descriptor types except interrupt and trap descriptors. The LDT may contain only segment, task gate, and call gate descriptors. A segment cannot be accessed by a task if its segment descriptor does not exist in either descriptor table at the time of access.

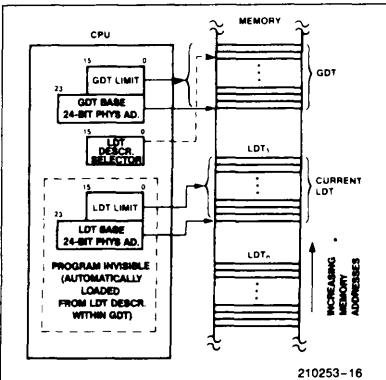
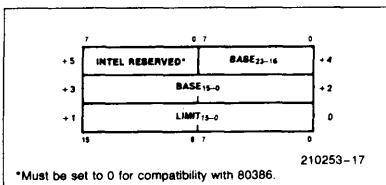


Figure 16. Local and Global Descriptor Table Definition

The LGDT and LLDT instructions load the base and limit of the global and local descriptor tables. LGDT and LLDT are privileged, i.e. they may only be executed by trusted programs operating at level 0. The LGDT instruction loads a six byte field containing the 16-bit table limit and 24-bit physical base address of the Global Descriptor Table as shown in Figure 17. The LDT instruction loads a selector which refers to a Local Descriptor Table descriptor containing the

base address and limit for an LDT, as shown in Figure 12.



*Must be set to 0 for compatibility with 80386.

Figure 17. Global Descriptor Table and Interrupt Descriptor Table Data Type

INTERRUPT DESCRIPTOR TABLE

The protected mode 80286 has a third descriptor table, called the Interrupt Descriptor Table (IDT) (see Figure 18), used to define up to 256 interrupts. It may contain only task gates, interrupt gates and trap gates. The IDT (Interrupt Descriptor Table) has a 24-bit physical base and 16-bit limit register in the CPU. The privileged LIDT instruction loads these registers with a six byte value of identical form to that of the LGDT instruction (see Figure 17 and Protected Mode Initialization).

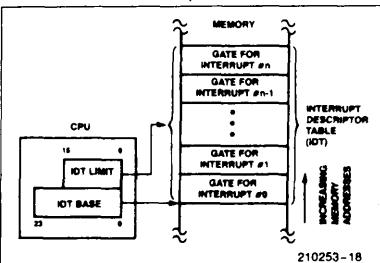
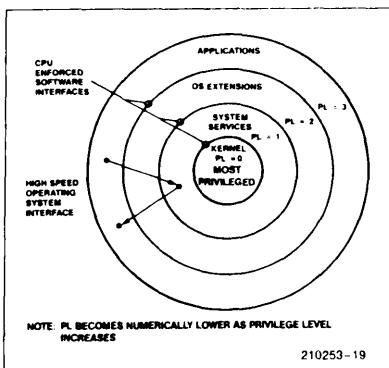


Figure 18. Interrupt Descriptor Table Definition

References to IDT entries are made via INT instructions, external interrupt vectors, or exceptions. The IDT must be at least 256 bytes in size to allocate space for all reserved interrupts.

Privilege

The 80286 has a four-level hierarchical privilege system which controls the use of privileged instructions and access to descriptors (and their associated segments) within a task. Four-level privilege, as shown in Figure 19, is an extension of the user/supervisor mode commonly found in minicomputers. The privilege levels are numbered 0 through 3. Level 0 is the



most privileged level. Privilege levels provide protection within a task. (Tasks are isolated by providing private LDT's for each task.) Operating system routines, interrupt handlers, and other system software can be included and protected within the virtual address space of each task using the four levels of privilege. Each task in the system has a separate stack for each of its privilege levels.

Tasks, descriptors, and selectors have a privilege level attribute that determines whether the descriptor may be used. Task privilege effects the use of instructions and descriptors. Descriptor and selector privilege only effect access to the descriptor.

TASK PRIVILEGE

A task always executes at one of the four privilege levels. The task privilege level at any specific instant is called the Current Privilege Level (CPL) and is defined by the lower two bits of the CS register. CPL cannot change during execution in a single code segment. A task's CPL may only be changed by control transfers through gate descriptors to a new code segment (See Control Transfer). Tasks begin executing at the CPL value specified by the code segment selector within TSS when the task is initiated via a task switch operation (See Figure 20). A task executing at Level 0 can access all data segments defined in the GDT and the task's LDT and is considered the most trusted level. A task executing at a Level 3 has the most restricted access to data and is considered the least trusted level.

DESCRIPTOR PRIVILEGE

Descriptor privilege is specified by the Descriptor Privilege Level (DPL) field of the descriptor access byte. DPL specifies the least trusted task privilege level (CPL) at which a task may access the descrip-

tor. Descriptors with DPL = 0 are the most protected. Only tasks executing at privilege level 0 (CPL = 0) may access them. Descriptors with DPL = 3 are the least protected (i.e. have the least restricted access) since tasks can access them when CPL = 0, 1, 2, or 3. This rule applies to all descriptors, except LDT descriptors.

SELECTOR PRIVILEGE

Selector privilege is specified by the Requested Privilege Level (RPL) field in the least significant two bits of a selector. Selector RPL may establish a less trusted privilege level than the current privilege level for the use of a selector. This level is called the task's effective privilege level (EPL). RPL can only reduce the scope of a task's access to data with this selector. A task's effective privilege is the numeric maximum of RPL and CPL. A selector with RPL = 0 imposes no additional restriction on its use while a selector with RPL = 3 can only refer to segments at privilege Level 3 regardless of the task's CPL. RPL is generally used to verify that pointer parameters passed to a more trusted procedure are not allowed to use data at a more privileged level than the caller (refer to pointer testing instructions).

Descriptor Access and Privilege Validation

Determining the ability of a task to access a segment involves the type of segment to be accessed, the instruction used, the type of descriptor used and CPL, RPL, and DPL. The two basic types of segment accesses are control transfer (selectors loaded into CS) and data (selectors loaded into DS, ES or SS).

DATA SEGMENT ACCESS

Instructions that load selectors into DS and ES must refer to a data segment descriptor or readable code segment descriptor. The CPL of the task and the RPL of the selector must be the same as or more privileged (numerically equal to or lower than) than the descriptor DPL. In general, a task can only access data segments at the same or less privileged levels than the CPL or RPL (whichever is numerically higher) to prevent a program from accessing data it cannot be trusted to use.

An exception to the rule is a readable conforming code segment. This type of code segment can be read from any privilege level.

If the privilege checks fail (e.g. DPL is numerically less than the maximum of CPL and RPL) or an incorrect type of descriptor is referenced (e.g. gate de-

scriptor or execute only code segment) exception 13 occurs. If the segment is not present, exception 11 is generated.

Instructions that load selectors into SS must refer to data segment descriptors for writable data segments. The descriptor privilege (DPL) and RPL must equal CPL. All other descriptor types or a privilege level violation will cause exception 13. A not present fault causes exception 12.

CONTROL TRANSFER

Four types of control transfer can occur when a selector is loaded into CS by a control transfer operation (see Table 10). Each transfer type can only occur if the operation which loaded the selector references the correct descriptor type. Any violation of these descriptor usage rules (e.g. JMP through a call gate or RET to a Task State Segment) will cause exception 13.

The ability to reference a descriptor for control transfer is also subject to rules of privilege. A CALL or JUMP instruction may only reference a code segment descriptor with DPL equal to the task CPL or a conforming segment with DPL of equal or greater privilege than CPL. The RPL of the selector used to reference the code descriptor must have as much privilege as CPL.

RET and IRET instructions may only reference code segment descriptors with descriptor privilege equal to or less privileged than the task CPL. The selector loaded into CS is the return address from the stack. After the return, the selector RPL is the task's new CPL. If CPL changes, the old stack pointer is popped after the return address.

When a JMP or CALL references a Task State Segment descriptor, the descriptor DPL must be the same or less privileged than the task's CPL. Refer-

ence to a valid Task State Segment descriptor causes a task switch (see Task Switch Operation). Reference to a Task State Segment descriptor at a more privileged level than the task's CPL generates exception 13.

When an instruction or interrupt references a gate descriptor, the gate DPL must have the same or less privilege than the task CPL. If DPL is at a more privileged level than CPL, exception 13 occurs. If the destination selector contained in the gate references a code segment descriptor, the code segment descriptor DPL must be the same or more privileged than the task CPL. If not, Exception 13 is issued. After the control transfer, the code segment descriptors DPL is the task's new CPL. If the destination selector in the gate references a task state segment, a task switch is automatically performed (see Task Switch Operation).

The privilege rules on control transfer require:

- JMP or CALL direct to a code segment (code segment descriptor) can only be to a conforming segment with DPL of equal or greater privilege than CPL or a non-conforming segment at the same privilege level.
- interrupts within the task or calls that may change privilege levels, can only transfer control through a gate at the same or a less privileged level than CPL to a code segment at the same or more privileged level than CPL.
- return instructions that don't switch tasks can only return control to a code segment at the same or less privileged level.
- task switch can be performed by a call, jump or interrupt which references either a task gate or task state segment at the same or less privileged level.

Table 10. Descriptor Types Used for Control Transfer

Control Transfer Types	Operation Types	Descriptor References	Descriptor Table
Intersegment within the same privilege level	JMP, CALL, RET, IRET*	Code Segment	GDT/LDT
Intersegment to the same or higher privilege level Interrupt within task may change CPL	CALL	Call Gate	GDT/LDT
	Interrupt Instruction, Exception, External Interrupt	Trap or Interrupt Gate	IDT
Intersegment to a lower privilege level (changes task CPL)	RET, IRET*	Code Segment	GDT/LDT
	CALL, JMP	Task State Segment	GDT
Task Switch	CALL, JMP	Task Gate	GDT/LDT
	IRET** Interrupt Instruction, Exception, External Interrupt	Task Gate	IDT

*NT (Nested Task bit of flag word) = 0

**NT (Nested Task bit of flag word) = 1

PRIVILEGE LEVEL CHANGES

Any control transfer that changes CPL within the task, causes a change of stacks as part of the operation. Initial values of SS:SP for privilege levels 0, 1, and 2 are kept in the task state segment (refer to Task Switch Operation). During a JMP or CALL control transfer, the new stack pointer is loaded into the SS and SP registers and the previous stack pointer is pushed onto the new stack.

When returning to the original privilege level, its stack is restored as part of the RET or IRET instruction operation. For subroutine calls that pass parameters on the stack and cross privilege levels, a fixed number of words, as specified in the gate, are copied from the previous stack to the current stack. The inter-segment RET instruction with a stack adjustment value will correctly restore the previous stack pointer upon return.

Protection

The 80286 includes mechanisms to protect critical instructions that affect the CPU execution state (e.g. HLT) and code or data segments from improper usage. These protection mechanisms are grouped into three forms:

Restricted usage of segments (e.g. no write allowed to read-only data segments). The only segments available for use are defined by descriptors in the Local Descriptor Table (LDT) and Global Descriptor Table (GDT).

Restricted access to segments via the rules of privilege and descriptor usage.

Privileged instructions or operations that may only be executed at certain privilege levels as determined by the CPL and I/O Privilege Level (IOPL). The IOPL is defined by bits 14 and 13 of the flag word.

These checks are performed for all instructions and can be split into three categories: segment load checks (Table 11), operand reference checks (Table 12), and privileged instruction checks (Table 13). Any violation of the rules shown will result in an exception. A not-present exception related to the stack segment causes exception 12.

The IRET and POPF instructions do not perform some of their defined functions if CPL is not of sufficient privilege (numerically small enough). Precisely these are:

- The IF bit is not changed if CPL > IOPL.
- The IOPL field of the flag word is not changed if CPL > 0.

No exceptions or other indication are given when these conditions occur.

Table 11
Segment Register Load Checks

Error Description	Exception Number
Descriptor table limit exceeded	13
Segment descriptor not-present	11 or 12
Privilege rules violated	13
Invalid descriptor/segment type segment register load: —Read only data segment load to SS —Special Control descriptor load to DS, ES, SS —Execute only segment load to DS, ES, SS —Data segment load to CS —Read/Execute code segment load to SS	13

Table 12. Operand Reference Checks

Error Description	Exception Number
Write into code segment	13
Read from execute-only code segment	13
Write to read-only data segment	13
Segment limit exceeded ¹	12 or 13

NOTE:

Carry out in offset calculations is ignored.

Table 13. Privileged Instruction Checks

Error Description	Exception Number
CPL ≠ 0 when executing the following instructions: LIDT, LLDT, LGDT, LTR, LMSW, CTS, HLT	13
CPL > IOPL when executing the following instructions: INS, IN, OUTS, OUT, STI, CLI, LOCK	13

EXCEPTIONS

The 80286 detects several types of exceptions and interrupts, in protected mode (see Table 14). Most are restartable after the exceptional condition is removed. Interrupt handlers for most exceptions can read an error code, pushed on the stack after the return address, that identifies the selector involved (0 if none). The return address normally points to the failing instruction, including all leading prefixes. For a processor extension segment overrun exception, the return address will not point at the ESC instruction that caused the exception; however, the processor extension registers may contain the address of the failing instruction.

Table 14. Protected Mode Exceptions

Interrupt Vector	Function	Return Address At Falling Instruction?	Always Restartable?	Error Code on Stack?
8	Double exception detected	Yes	No ²	Yes
9	Processor extension segment overrun	No	No ²	No
10	Invalid task state segment	Yes	Yes	Yes
11	Segment not present	Yes	Yes	Yes
12	Stack segment overrun or stack segment not present	Yes	Yes ¹	Yes
13	General protection	Yes	No ²	Yes

NOTE:

- When a PUSHA or POPA instruction attempts to wrap around the stack segment, the machine state after the exception will not be restartable because stack segment wrap around is not permitted. This condition is identified by the value of the saved SP being either 0000(H), 0001(H), FFFE(H), or FFFF(H).
- These exceptions indicate a violation to privilege rules or usage rules has occurred. Restart is generally not attempted under those conditions.

These exceptions indicate a violation to privilege rules or usage rules has occurred. Restart is generally not attempted under those conditions.

All these checks are performed for all instructions and can be split into three categories: segment load checks (Table 11), operand reference checks (Table 12), and privileged instruction checks (Table 13). Any violation of the rules shown will result in an exception. A not-present exception causes exception 11 or 12 and is restartable.

Special Operations

TASK SWITCH OPERATION

The 80286 provides a built-in task switch operation which saves the entire 80286 execution state (registers, address space, and a link to the previous task), loads a new execution state, and commences execution in the new task. Like gates, the task switch operation is invoked by executing an inter-segment JMP or CALL instruction which refers to a Task State Segment (TSS) or task gate descriptor in the GDT or LDT. An INT n instruction, exception, or external interrupt may also invoke the task switch operation by selecting a task gate descriptor in the associated IDT descriptor entry.

The TSS descriptor points at a segment (see Figure 20) containing the entire 80286 execution state while a task gate descriptor contains a TSS selector. The limit field of the descriptor must be > 002B(H).

Each task must have a TSS associated with it. The current TSS is identified by a special register in the 80286 called the Task Register (TR). This register contains a selector referring to the task state segment descriptor that defines the current TSS. A hidden base and limit register associated with TR are loaded whenever TR is loaded with a new selector.

The IRET instruction is used to return control to the task that called the current task or was interrupted. Bit 14 in the flag register is called the Nested Task (NT) bit. It controls the function of the IRET instruction. If NT = 0, the IRET instruction performs the regular current task by popping values off the stack; when NT = 1, IRET performs a task switch operation back to the previous task.

When a CALL, JMP, or INT instruction initiates a task switch, the old (except for case of JMP) and new TSS will be marked busy and the back link field of the new TSS set to the old TSS selector. The NT bit of the new task is set by CALL or INT initiated task switches. An interrupt that does not cause a task switch will clear NT. NT may also be set or cleared by POPF or IRET instructions.

The task state segment is marked busy by changing the descriptor type field from Type 1 to Type 3. Use of a selector that references a busy task state segment causes Exception 13.

PROCESSOR EXTENSION CONTEXT SWITCHING

The context of a processor extension (such as the 80287 numerics processor) is not changed by the task switch operation. A processor extension context need only be changed when a different task attempts to use the processor extension (which still contains the context of a previous task). The 80286 detects the first use of a processor extension after a task switch by causing the processor extension not present exception (7). The interrupt handler may then decide whether a context change is necessary.

Whenever the 80286 switches tasks, it sets the Task Switched (TS) bit of the MSW. TS indicates that a processor extension context may belong to a different task than the current one. The processor extension not present exception (7) will occur when attempting to execute an ESC or WAIT instruction if TS=1 and a processor extension is present (MP=1 in MSW).

POINTER TESTING INSTRUCTIONS

The 80286 provides several instructions to speed pointer testing and consistency checks for maintaining system integrity (see Table 15). These instruc-

tions use the memory management hardware to verify that a selector value refers to an appropriate segment without risking an exception. A condition flag (ZF) indicates whether use of the selector or segment will cause an exception.

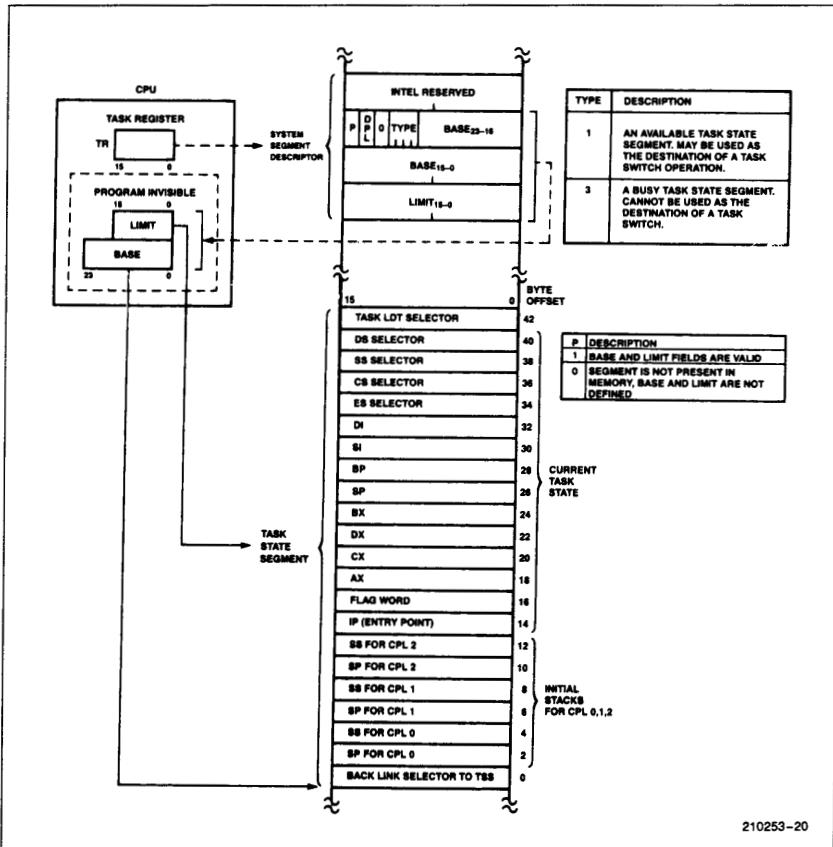


Figure 20. Task State Segment and TSS Registers

Table 15. 80286 Pointer Test Instructions

Instruction	Operands	Function
ARPL	Selector, Register	Adjust Requested Privilege Level: adjusts the RPL of the selector to the numeric maximum of current selector RPL value and the RPL value in the register. Set zero flag if selector RPL was changed by ARPL.
VERR	Selector	VERify for Read: sets the zero flag if the segment referred to by the selector can be read.
VERW	Selector	VERify for Writer: sets the zero flag if the segment referred to by the selector can be written.
LSL	Register, Selector	Load Segment Limit: reads the segment limit into the register if privilege rules and descriptor type allow. Set zero flag if successful.
LAR	Register, Selector	Load Access Rights: reads the descriptor access rights byte into the register if privilege rules allow. Set zero flag if successful.

DOUBLE FAULT AND SHUTDOWN

If two separate exceptions are detected during a single instruction execution, the 80286 performs the double fault exception (8). If an execution occurs during processing of the double fault exception, the 80286 will enter shutdown. During shutdown no further instructions or exceptions are processed. Either NMI (CPU remains in protected mode) or RESET (CPU exits protected mode) can force the 80286 out of shutdown. Shutdown is externally signalled via a HALT bus operation with A₁ LOW.

PROTECTED MODE INITIALIZATION

The 80286 initially executes in real address mode after RESET. To allow initialization code to be placed at the top of physical memory, A₂₃-A₂₀ will be HIGH when the 80286 performs memory references relative to the CS register until CS is changed. A₂₃-A₂₀ will be zero for references to the DS, ES, or SS segments. Changing CS in real address mode will force A₂₃-A₂₀ LOW whenever CS is used again. The initial CS:IP value of F000:FFF0 provides 64K bytes of code space for initialization code without changing CS.

Protected mode operation requires several registers to be initialized. The GDT and IDT base registers must refer to a valid GDT and IDT. After executing the LMSW instruction to set PE, the 80286 must im-

mediately execute an intra-segment JMP instruction to clear the instruction queue of instructions decoded in real address mode.

To force the 80286 CPU registers to match the initial protected mode state assumed by software, execute a JMP instruction with a selector referring to the initial TSS used in the system. This will load the task register, local descriptor table register, segment registers and initial general register state. The TR should point at a valid TSS since any task switch operation involves saving the current task state.

SYSTEM INTERFACE

The 80286 system interface appears in two forms: a local bus and a system bus. The local bus consists of address, data, status, and control signals at the pins of the CPU. A system bus is any buffered version of the local bus. A system bus may also differ from the local bus in terms of coding of status and control lines and/or timing and loading of signals. The 80286 family includes several devices to generate standard system buses such as the IEEE 796 standard MULTIBUS.

Bus Interface Signals and Timing

The 80286 microsystem local bus interfaces the 80286 to local memory and I/O components. The interface has 24 address lines, 16 data lines, and 8 status and control signals.

The 80286 CPU, 82284 clock generator, 82288 bus controller, 82289 bus arbiter, transceivers, and latches provide a buffered and decoded system bus interface. The 82284 generates the system clock and synchronizes READY and RESET. The 82288 converts bus operation status encoded by the 80286 into command and bus control signals. The 82289 bus arbiter generates Multibus bus arbitration signals. These components can provide the timing and electrical power drive levels required for most system bus interfaces including the Multibus.

Physical Memory and I/O Interface

A maximum of 16 megabytes of physical memory can be addressed in protected mode. One megabyte can be addressed in real address mode. Memory is accessible as bytes or words. Words consist of any two consecutive bytes addressed with the least significant byte stored in the lowest address.

Byte transfers occur on either half of the 16-bit local data bus. Even bytes are accessed over D₇₋₀ while odd bytes are transferred over D₁₅₋₈. Even-addressed words are transferred over D₁₅₋₀ in one bus cycle, while odd-addressed word require two bus operations. The first transfers data on D₁₅₋₈, and the second transfers data on D₇₋₀. Both byte data transfers occur automatically, transparent to software.

Two bus signals, A_0 and \overline{BHE} , control transfers over the lower and upper halves of the data bus. Even address byte transfers are indicated by A_0 LOW and \overline{BHE} HIGH. Odd address byte transfers are indicated by A_0 HIGH and \overline{BHE} LOW. Both A_0 and BHE are LOW for even address word transfers.

The I/O address space contains 64K addresses in both modes. The I/O space is accessible as either bytes or words, as is memory. Byte wide peripheral devices may be attached to either the upper or lower byte of the data bus. Byte-wide I/O devices attached to the upper data byte (D_{15-8}) are accessed with odd I/O addresses. Devices on the lower data byte are accessed with even I/O addresses. An interrupt controller such as Intel's 8259A must be connected to the lower data byte (D_{7-0}) for proper return of the interrupt vector.

Bus Operation

The 80286 uses a double frequency system clock (CLK input) to control bus timing. All signals on the local bus are measured relative to the system CLK input. The CPU divides the system clock by 2 to produce the internal processor clock, which determines bus state. Each processor clock is composed of two system clock cycles named phase 1 and phase 2. The 82284 clock generator output (PCLK) identifies the next phase of the processor clock. (See Figure 21.)

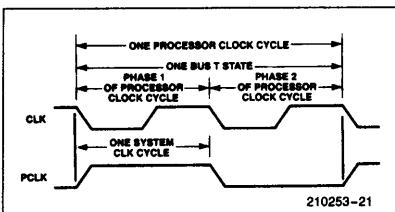


Figure 21. System and Processor Clock Relationships

Six types of bus operations are supported; memory read, memory write, I/O read, I/O write, interrupt acknowledgement, and halt/shutdown. Data can be transferred at a maximum rate of one word per two processor clock cycles.

The 80286 bus has three basic states: idle (T_i), send status (T_s), and perform command (T_c). The 80286 CPU also has a fourth local bus state called hold (T_h). T_h indicates that the 80286 has surrendered control of the local bus to another bus master in response to a HOLD request.

Each bus state is one processor clock long. Figure 22 shows the four 80286 local bus states and allowed transitions.

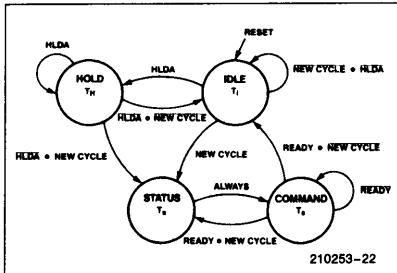


Figure 22. 80286 Bus States

Bus States

The idle (T_i) state indicates that no data transfers are in progress or requested. The first active state T_s is signaled by status line S_1 or S_0 going LOW and identifying phase 1 of the processor clock. During T_s , the command encoding, the address, and data (for a write operation) are available on the 80286 output pins. The 82286 bus controller decodes the status signals and generates Multibus compatible read/write command and local transceiver control signals.

After T_s , the perform command (T_c) state is entered. Memory or I/O devices respond to the bus operation during T_c , either transferring read data to the CPU or accepting write data. T_c states may be repeated as often as necessary to assure sufficient time for the memory or I/O device to respond. The READY signal determines whether T_c is repeated. A repeated T_c state is called a wait state.

During hold (T_h), the 80286 will float all address, data, and status output pins enabling another bus master to use the local bus. The 80286 HOLD input signal is used to place the 80286 into the T_h state. The 80286 HLDA output signal indicates that the CF \bullet has entered T_h .

Pipelined Addressing

The 80286 uses a local bus interface with pipelined timing to allow as much time as possible for data access. Pipelined timing allows a new bus operation to be initiated every two processor cycles, while allowing each individual bus operation to last for three processor cycles.

The timing of the address outputs is pipelined such that the address of the next bus operation becomes available during the current bus operation. Or in other words, the first clock of the next bus operation is overlapped with the last clock of the current bus operation. Therefore, address decode and routing logic can operate in advance of the next bus operation.

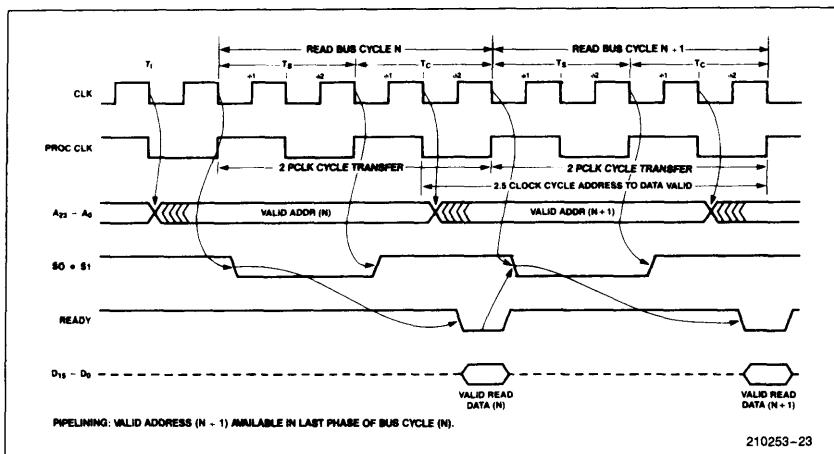


Figure 23. Basic Bus Cycle

210253-23

External address latches may hold the address stable for the entire bus operation, and provide additional AC and DC buffering.

The 80286 does not maintain the address of the current bus operation during all T_c states. Instead, the address for the next bus operation may be emitted during phase 2 of any T_c . The address remains valid during phase 1 of the first T_c to guarantee hold time, relative to ALE, for the address latch inputs.

Bus Control Signals

The 82288 bus controller provides control signals; address latch enable (ALE), Read/Write commands, data transmit/receive (DT/R), and data enable (DEN) that control the address latches, data transceivers, write enable, and output enable for memory and I/O systems.

The Address Latch Enable (ALE) output determines when the address may be latched. ALE provides at least one system CLK period of address hold time from the end of the previous bus operation until the address for the next bus operation appears at the latch outputs. This address hold time is required to support MULTIBUS and common memory systems.

The data bus transceivers are controlled by 82288 outputs Data Enable (DEN) and Data Transmit/Receive (DT/R). DEN enables the data transceivers; while DT/R controls transceiver direction. DEN and DT/R are timed to prevent bus contention between the bus master, data bus transceivers, and system data bus transceivers.

Command Timing Controls

Two system timing customization options, command extension and command delay, are provided on the 80286 local bus.

Command extension allows additional time for external devices to respond to a command and is analogous to inserting wait states on the 8086. External logic can control the duration of any bus operation such that the operation is only as long as necessary. The READY input signal can extend any bus operation for as long as necessary.

Command delay allows an increase of address or write data setup time to system bus command active for any bus operation by delaying when the system bus command becomes active. Command delay is controlled by the 82288 CMDLY input. After T_S , the bus controller samples CMDLY at each falling edge of CLK. If CMDLY is HIGH, the 82288 will not activate the command signal. When CMDLY is LOW, the 82288 will activate the command signal. After the command becomes active, the CMDLY input is not sampled.

When a command is delayed, the available response time from command active to return read data or accept write data is less. To customize system bus timing, an address decoder can determine which bus operations require delaying the command. The CMDLY input does not affect the timing of ALE, DEN, or DT/R.

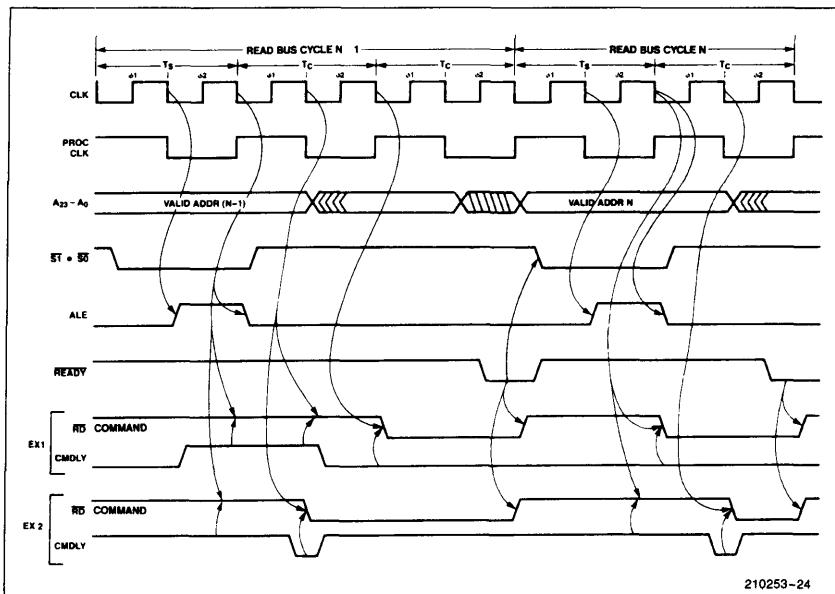


Figure 24. CMDLY Controls the Leading Edge of Command Signal

Figure 24 illustrates four uses of CMDLY. Example 1 shows delaying the read command two system CLKs for cycle N-1 and no delay for cycle N, and example 2 shows delaying the read command one system CLK for cycle N-1 and one system CLK delay for cycle N.

Bus Cycle Termination

At maximum transfer rates, the 80286 bus alternates between t_S status and command states. The bus status signals become inactive after t_S so that they may correctly signal the start of the next bus operation after the completion of the current cycle. No external indication of t_C exists on the 80286 local bus. The bus master and bus controller enter t_C directly after t_S and continue executing t_C cycles until terminated by READY.

READY Operation

The current bus master and 82288 bus controller terminate each bus operation simultaneously to achieve maximum bus operation bandwidth. Both are informed in advance by READY active (open-collector output from 82284) which identifies the last t_C cycle of the current bus operation. The bus master and bus controller must see the same sense of

the READY signal, thereby requiring READY be synchronous to the system clock.

Synchronous Ready

The 82284 clock generator provides READY synchronization from both synchronous and asynchronous sources (see Figure 25). The synchronous ready input (SRDY) of the clock generator is sampled with the falling edge of CLK at the end of phase 1 of each t_C. The state of SRDY is then broadcast to the bus master and bus controller via the READY output line.

Asynchronous Ready

Many systems have devices or subsystems that are asynchronous to the system clock. As a result, their ready outputs cannot be guaranteed to meet the 82284 SRDY setup and hold time requirements. But the 82284 asynchronous ready input (ARDY) is designed to accept such signals. The ARDY input is sampled at the beginning of each t_C cycle by 82284 synchronization logic. This provides one system CLK cycle time to resolve its value before broadcasting it to the bus master and bus controller.

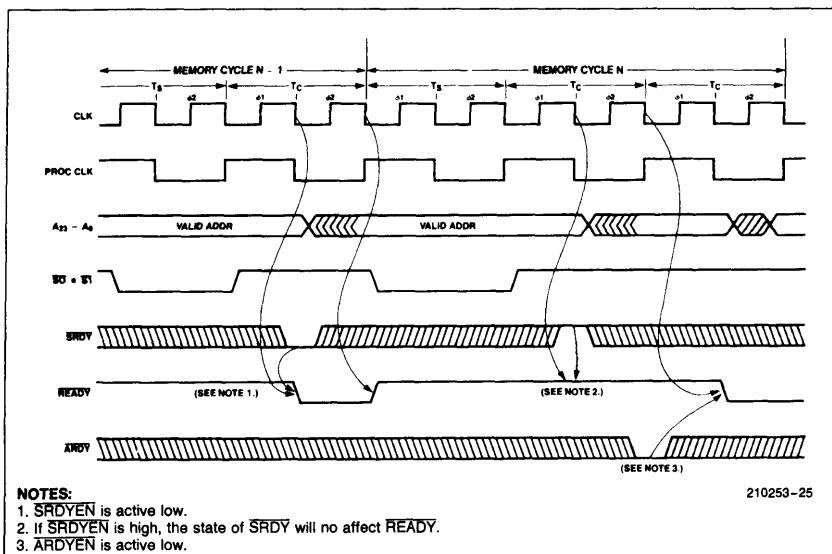


Figure 25. Synchronous and Asynchronous Ready

ARDY or ARDYEN must be HIGH at the end of T_s . ARDY cannot be used to terminate bus cycle with no wait states.

Each ready input of the 82284 has an enable pin (SRDYEN and ARDYEN) to select whether the current bus operation will be terminated by the synchronous or asynchronous ready. Either of the ready inputs may terminate a bus operation. These enable inputs are active low and have the same timing as their respective ready inputs. Address decode logic usually selects whether the current bus operation should be terminated by ARDY or SRDY.

The data bus is driven with write data during the second phase of T_s . The delay in write data timing allows the read data drivers, from a previous read cycle, sufficient time to enter 3-state OFF before the 80286 CPU begins driving the local data bus for write operations. Write data will always remain valid for one system clock past the last T_c to provide sufficient hold time for Multibus or other similar memory or I/O systems. During write-read or write-idle sequences the data bus enters 3-state OFF during the second phase of the processor cycle after the last T_c . In a write-write sequence the data bus does not enter 3-state OFF between T_c and T_s .

Data Bus Control

Figures 26, 27, and 28 show how the DT/R, DEN, data bus, and address signals operate for different combinations of read, write, and idle bus operations. DT/R goes active (LOW) for a read operation. DT/R remains HIGH before, during, and between write operations.

Bus Usage

The 80286 local bus may be used for several functions: instruction data transfers, data transfers by other bus masters, instruction fetching, processor extension data transfers, interrupt acknowledge, and halt/shutdown. This section describes local bus activities which have special signals or requirements.

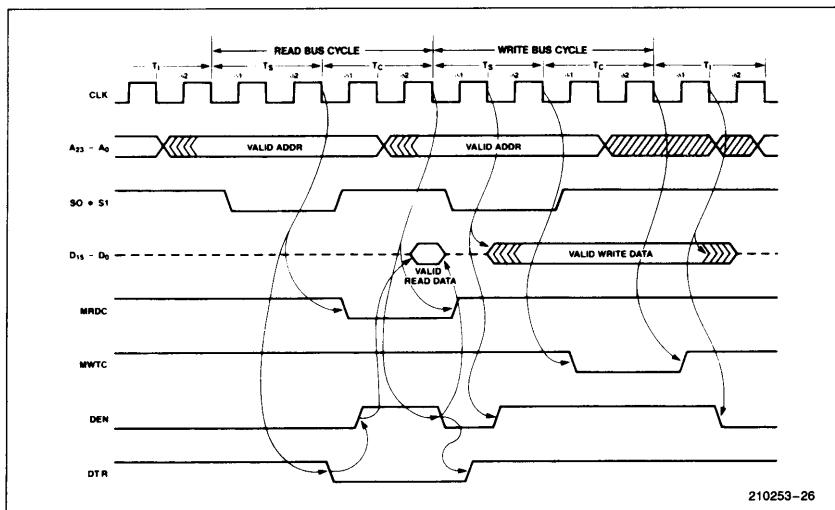


Figure 26. Back to Back Read-Write Cycles

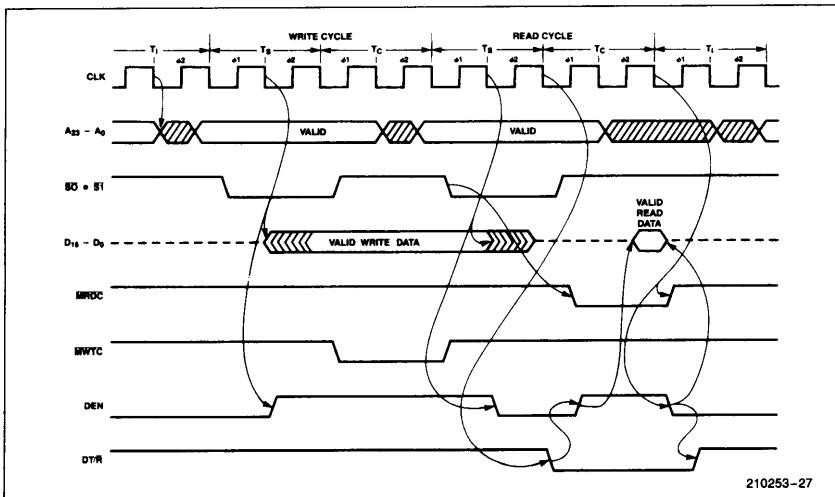


Figure 27. Back to Back Write-Read Cycles

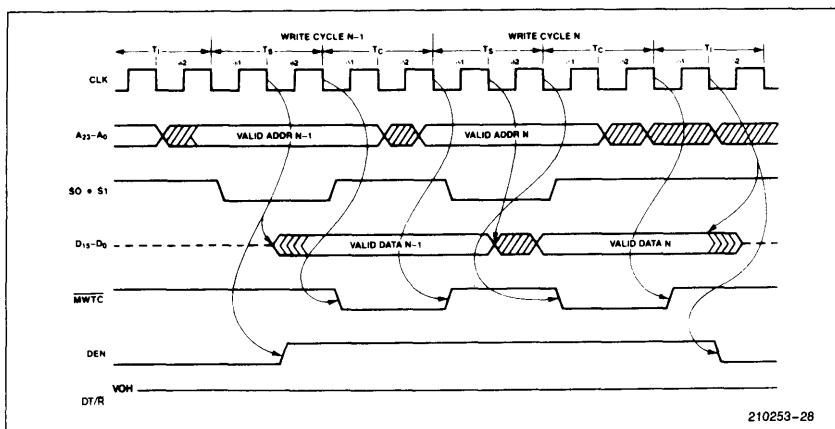


Figure 28. Back to Back Write-Write Cycles

HOLD and HLDA

HOLD AND HLDA allow another bus master to gain control of the local bus by placing the 80286 bus into the T_h state. The sequence of events required to pass control between the 80286 and another local bus master are shown in Figure 29.

In this example, the 80286 is initially in the T_h state as signaled by HLDA being active. Upon leaving T_h, as signaled by HLDA going inactive, a write operation is started. During the write operation another local bus master requests the local bus from the 80286 as shown by the HOLD signal. After completing the write operation, the 80286 performs one T_i bus cycle, to guarantee write data hold time, then enters T_h as signaled by HLDA going active.

The CMDLY signal and ARDY ready are used to start and stop the write bus command, respectively. Note that SRDY must be inactive or disabled by SRDYEN to guarantee ARDY will terminate the cycle.

HOLD must not be active during the time from the leading edge of RESET until 34 CLks following the trailing edge of RESET unless the 80286 is in the Halt condition. To insure that the 80286 remains in the Halt condition until the processor Reset operation is complete, no interrupts should occur after the execution of HLT until 34 CLks after the trailing edge of the RESET pulse.

Lock

The CPU asserts an active lock signal during interrupt-Acknowledge cycles, the XCHG instruction, and during some descriptor accesses. Lock is also asserted when the LOCK prefix is used. The LOCK

prefix may be used with the following ASM-286 assembly instructions; MOVS, INS, and OUTS. For bus cycles other than Interrupt-Acknowledge cycles, Lock will be active for the first and subsequent cycles of a series of cycles to be locked. Lock will not be shown active during the last cycle to be locked. For the next-to-last cycle, Lock will become inactive at the end of the first T_c regardless of the number of wait-states inserted. For Interrupt-Acknowledge cycles, Lock will be active for each cycle, and will become inactive at the end of the first T_c for each cycle regardless of the number of wait-states inserted.

Instruction Fetching

The 80286 Bus Unit (BU) will fetch instructions ahead of the current instruction being executed. This activity is called prefetching. It occurs when the local bus would otherwise be idle and obeys the following rules:

A prefetch bus operation starts when at least two bytes of the 6-byte prefetch queue are empty.

The prefetcher normally performs word prefetches independent of the byte alignment of the code segment base in physical memory.

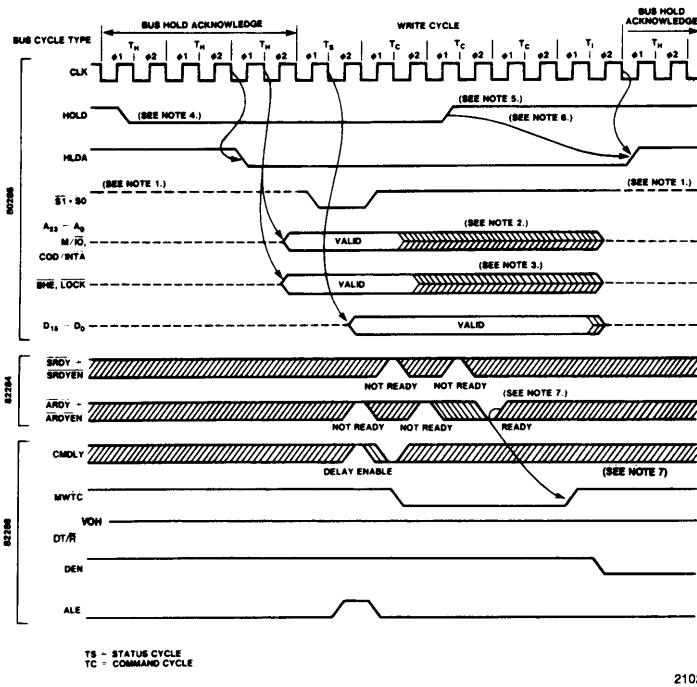
The prefetcher will perform only a byte code fetch operation for control transfers to an instruction beginning on a numerically odd physical address.

Prefetching stops whenever a control transfer or HLT instruction is decoded by the IU and placed into the instruction queue.

In real address mode, the prefetcher may fetch up to 6 bytes beyond the last control transfer or HLT instruction in a code segment.

In protected mode, the prefetcher will never cause a segment overrun exception. The prefetcher stops at the last physical memory word of the code segment. Exception 13 will occur if the program attempts to execute beyond the last full instruction in the code segment.

If the last byte of a code segment appears on an even physical memory address, the prefetcher will read the next physical byte of memory (perform a word code fetch). The value of this byte is ignored and any attempt to execute it causes exception 13.



210253-29

NOTES:

1. Status lines are not driven by 80286, yet remain high due to pullup resistors in 82288 and 82289 during HOLD state.
2. Address, M/I/O and COD/INTA may start floating during any T_C depending on when internal 80286 bus arbiter decides to release bus to external HOLD. The float starts in ϕ_2 of T_C .
3. BHE and LOCK may start floating after the end of any T_C depending on when internal 80286 bus arbiter decides to release bus to external HOLD. The float starts in ϕ_1 of T_C .
4. The minimum HOLD to HMDA time is shown. Maximum is one T_H longer.
5. The earliest HOLD time is shown. It will always allow a subsequent memory cycle if pending is shown.
6. The minimum HOLD to HMDA time is shown. Maximum is a function of the instruction, type of bus cycle and other machine state (i.e., Interrupts, Waits, Lock, etc.).
7. Asynchronous ready allows termination of the cycle. Synchronous ready does not signal ready in this example. Synchronous ready state is ignored after ready is signaled via the asynchronous input.

Figure 29. MULTIBUS® Write Terminated by Asynchronous Ready with Bus Hold

Processor Extension Transfers

The processor extension interface uses I/O port addresses 00F8(H), 00FA(H), and 00FC(H) which are part of the I/O port address range reserved by Intel. An ESC instruction with Machine Status Word bits EM = 0 and TS = 0 will perform I/O bus operations to one or more of these I/O port addresses independent of the value of IOPL and CPL.

ESC instructions with memory references enable the CPU to accept PEREQ inputs for processor extension operand transfers. The CPU will determine the operand starting address and read/write status of the instruction. For each operand transfer, two or three bus operations are performed, one word transfer with I/O port address 00FA(H) and one or two bus operations with memory. Three bus operations are required for each word operand aligned on an odd byte address.

Interrupt Acknowledge Sequence

Figure 30 illustrates an interrupt acknowledge sequence performed by the 80286 in response to an INTR input. An interrupt acknowledge sequence consists of two INTA bus operations. The first allows a master 8259A Programmable Interrupt Controller (PIC) to determine which if any of its slaves should return the interrupt vector. An eight bit vector is read on D0-D7 of the 80286 during the second INTA bus operation to select an interrupt handler routine from the interrupt table.

The Master Cascade Enable (MCE) signal of the 82288 is used to enable the cascade address drivers, during INTA bus operations (See Figure 30), onto the local address bus for distribution to slave interrupt controllers via the system address bus. The 80286 emits the LOCK signal (active LOW) during T_s of the first INTA bus operation. A local bus "hold" request will not be honored until the end of the second INTA bus operation.

Three idle processor clocks are provided by the 80286 between INTA bus operations to allow for the minimum INTA to INTA time and CAS (cascade address) out delay of the 8259A. The second INTA bus operation must always have at least one extra T_c state added via logic controlling READY. This is needed to meet the 8259A minimum INTA pulse width.

Local Bus Usage Priorities

The 80286 local bus is shared among several internal units and external HOLD requests. In case of simultaneous requests, their relative priorities are:

- (Highest) Any transfers which assert LOCK either explicitly (via the LOCK instruction prefix) or implicitly (i.e. some segment descriptor accesses, interrupt acknowledge sequence, or an XCHG with memory).
- The second of the two byte bus operations required for an odd aligned word operand.
- The second or third cycle of a processor extension data transfer.
- Local bus request via HOLD input.
- Processor extension data operand transfer via PEREQ input.
- Data transfer performed by EU as part of an instruction.

- (Lowest) An instruction prefetch request from BU. The EU will inhibit prefetching two processor clocks in advance of any data transfers to minimize waiting by EU for a prefetch to finish.

Halt or Shutdown Cycles

The 80286 externally indicates halt or shutdown conditions as a bus operation. These conditions occur due to a HLT instruction or multiple protection exceptions while attempting to execute one instruction. A halt or shutdown bus operation is signalled when S1, S0 and COD/INTA are LOW and M/I/O is HIGH. A₁ HIGH indicates halt, and A₁ LOW indicates shutdown. The 82288 bus controller does not issue ALE, nor is READY required to terminate a halt or shutdown bus operation.

During halt or shutdown, the 80286 may service PEREQ or HOLD requests. A processor extension segment overrun exception during shutdown will inhibit further service of PEREQ. Either NMI or RESET will force the 80286 out of either halt or shutdown. An INTR, if interrupts are enabled, or a processor extension segment overrun exception will also force the 80286 out of halt.

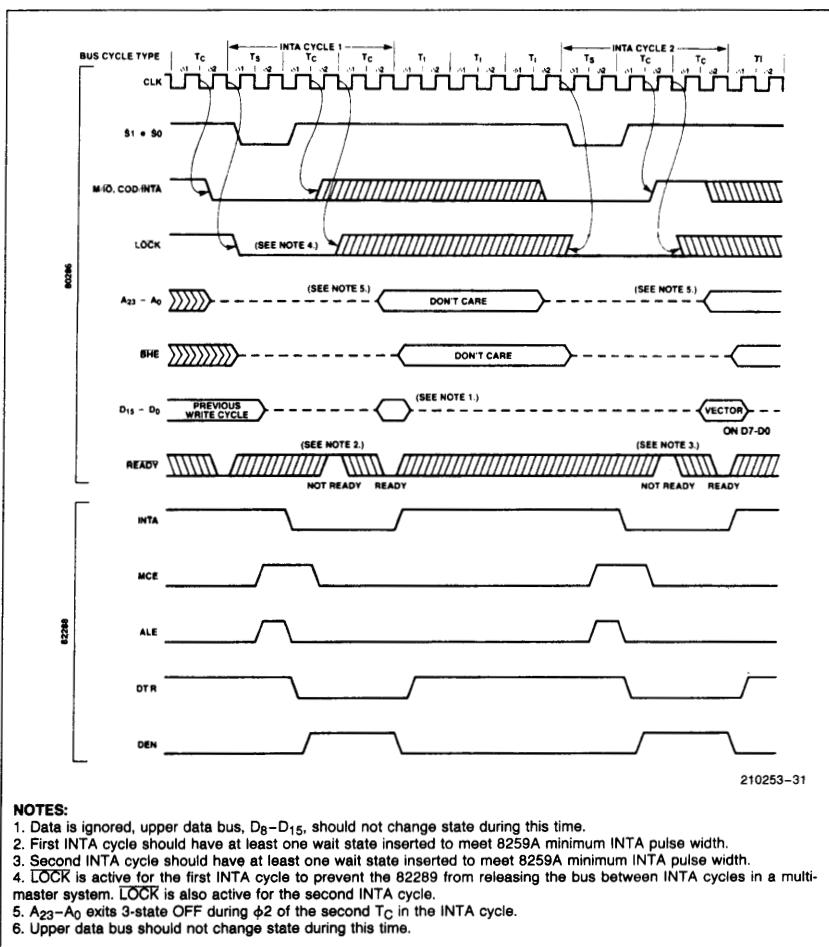


Figure 30. Interrupt Acknowledge Sequence

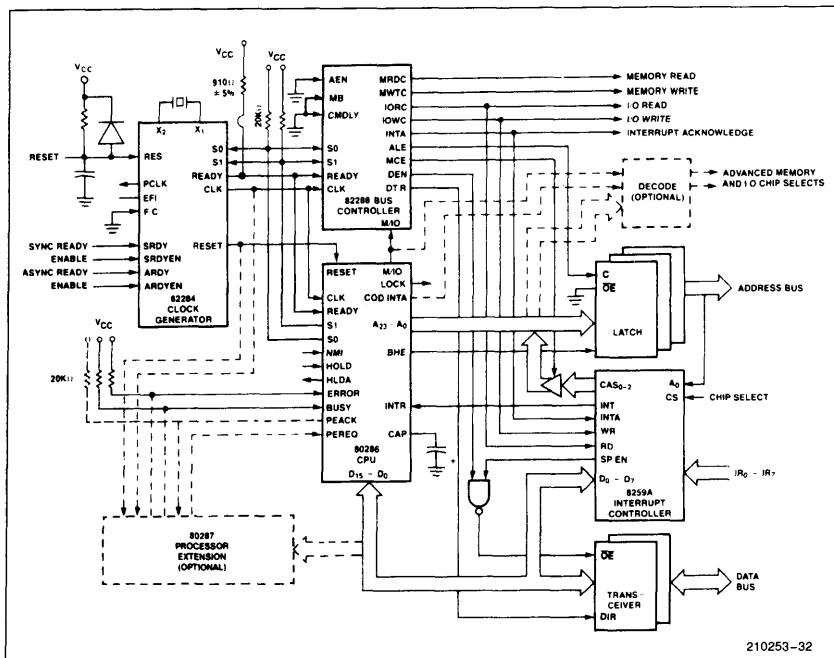


Figure 31. Basic 80286 System Configuration

SYSTEM CONFIGURATIONS

The versatile bus structure of the 80286 microsystem, with a full complement of support chips, allows flexible configuration of a wide range of systems. The basic configuration, shown in Figure 31, is similar to an 8086 maximum mode system. It includes the CPU plus an 8259A interrupt controller, 82284 clock generator, and the 82288 Bus Controller.

As indicated by the dashed lines in Figure 31, the ability to add processor extensions is an integral feature of 80286 microsystems. The processor extension interface allows external hardware to perform special functions and transfer data concurrent with CPU execution of other instructions. Full system integrity is maintained because the 80286 supervises all data transfers and instruction execution for the processor extension.

The 80287 has all the instructions and data types of an 8087. The 80287 NPX can perform numeric calculations and data transfers concurrently with CPU program execution. Numerics code and data have the same integrity as all other information protected by the 80286 protection mechanism.

The 80286 can overlap chip select decoding and address propagation during the data transfer for the previous bus operation. This information is latched by ALE during the middle of a T_s cycle. The latched chip select and address information remains stable during the bus operation while the next cycle's address is being decoded and propagated into the system. Decode logic can be implemented with a high speed bipolar PROM.

The optional decode logic shown in Figure 31 takes advantage of the overlap between address and data of the 80286 bus cycle to generate advanced memory and IO-select signals. This minimizes system

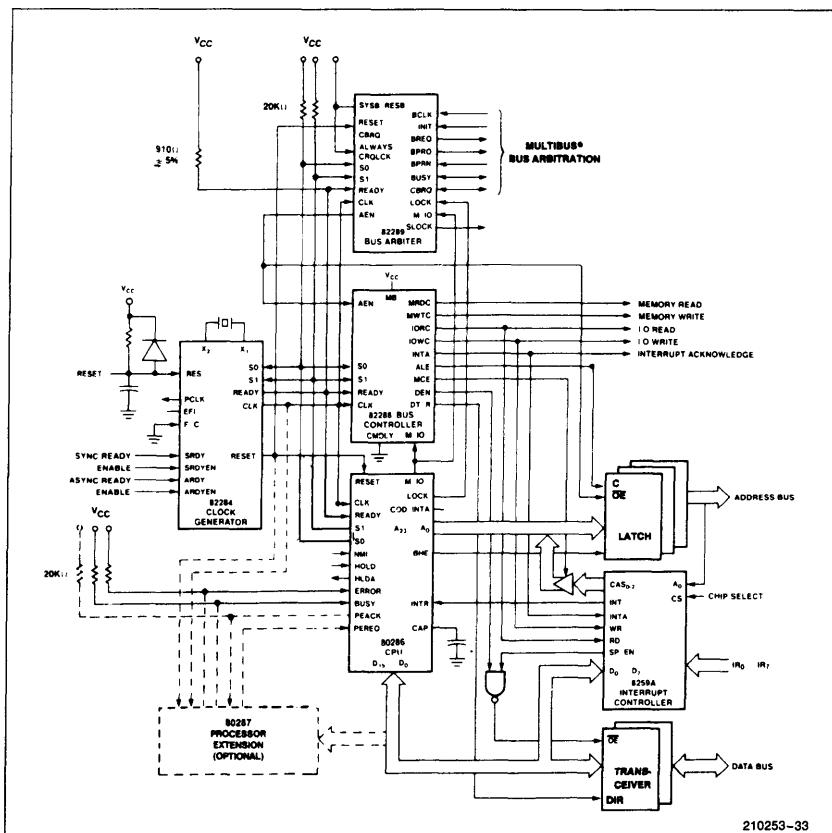


Figure 32. MULTIBUS® System Bus Interface

performance degradation caused by address propagation and decode delays. In addition to selecting memory and I/O, the advanced selects may be used with configurations supporting local and system buses to enable the appropriate bus interface for each bus cycle. The COD/INTA and M/I/O signals are applied to the decode logic to distinguish between interrupt, I/O, code and data bus cycles.

By adding the 82289 bus arbiter chip, the 80286 provides a MULTIBUS system bus interface as shown in Figure 32. The ALE output of the 82288 for the

MULTIBUS bus is connected to its CMDLY input to delay the start of commands one system CLK as required to meet MULTIBUS address and write data setup times. This arrangement will add at least one extra T_c state to each bus operation which uses the MULTIBUS.

A second 82288 bus controller and additional latches and transceivers could be added to the local bus of Figure 32. This configuration allows the 80286 to support an on-board bus for local memory and peripherals, and the MULTIBUS for system bus interfacing.

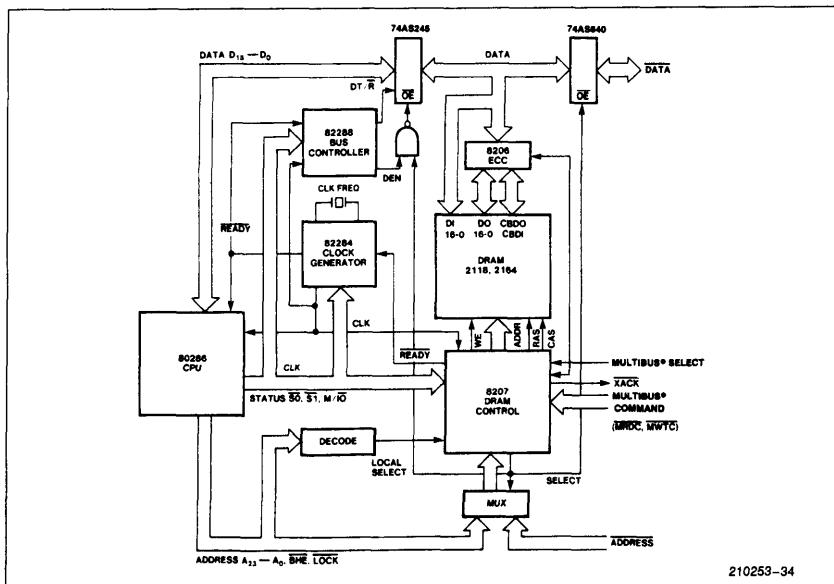


Figure 33. 80286 System Configuration with Dual-Ported Memory

210253-34

Figure 33 shows the addition of dual ported dynamic memory between the MULTIBUS system bus and the 80286 local bus. The dual port interface is provided by the 8207 Dual Port DRAM Controller. The 8207 runs synchronously with the CPU to maximize throughput for local memory references. It also arbitrates between requests from the local and system buses and performs functions such as refresh,

initialization of RAM, and read/modify/write cycles. The 8207 combined with the 8206 Error Checking and Correction memory controller provide for single bit error correction. The dual-ported memory can be combined with a standard MULTIBUS system bus interface to maximize performance and protection in multiprocessor system configurations.

Table 16. 80286 Systems Recommended Pull Up Resistor Values

80286 Pin and Name	Pullup Value	Purpose
4—S1	20 kΩ ± 10%	Pull S0, S1, and PEACK inactive during 80286 hold periods
5—S0	20 kΩ ± 10%	Pull S0, S1, and PEACK inactive during 80286 hold periods
6—PEACK	20 kΩ ± 10%	Pull ERROR and BUSY inactive when 80287 not present (or temporarily removed from socket)
53—ERROR	20 kΩ ± 10%	Pull ERROR and BUSY inactive when 80287 not present (or temporarily removed from socket)
54—BUSY	20 kΩ ± 10%	Pull ERROR and BUSY inactive when 80287 not present (or temporarily removed from socket)
63—READY	910Ω ± 5%	Pull READY inactive within required minimum time ($C_L = 150\text{ pF}$, $I_R \leq 7\text{ mA}$)

I²ICETM-286 System Design Considerations

One of the advantages of using the 80286 is that full in-circuit emulation debugging support is provided through the I²ICE system 80286 probe. To utilize this powerful tool it is necessary that the system designer be aware of a few minor parametric and

functional differences between the 80286 and I²ICE system 80286 probe. The I²ICE data sheet (I²ICE Integrated Instrumentation and In-Circuit Emulation System, order #210469) contains a detailed description of these design considerations. It is recommended that this document be reviewed by the 80286 system designer to determine whether or not these differences affect his design.

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias 0°C to + 70°C
 Storage Temperature -65°C to + 150°C
 Voltage on Any Pin with
 Respect to Ground -1.0V to + 7V
 Power Dissipation 3.3W

**Notice: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.*

D.C. CHARACTERISTICS ($V_{CC} = 5V \pm 5\%$, $T_{CASE} = 0^{\circ}\text{C}$ to $+ 85^{\circ}\text{C}$)

Symbol	Parameter	Min	Max	Unit	Test Condition
V_{IL}	Input LOW Voltage	-.5	.8	V	
V_{IH}	Input HIGH Voltage	2.0	$V_{CC} + .5$	V	
V_{ILC}	CLK Input LOW Voltage	-.5	.6	V	
V_{IHC}	CLK Input HIGH Voltage	3.8	$V_{CC} + .5$	V	

D.C. CHARACTERISTICS ($V_{CC} = 5V \pm 5\%$, $T_{CASE} = 0^{\circ}\text{C}$ to $+ 85^{\circ}\text{C}$)

Symbol	Parameter	Min	Max	Unit	Test Condition
V_{OL}	Output LOW Voltage		0.45	V	$I_{OL} = 2.0 \text{ mA}$
V_{OH}	Output HIGH Voltage	2.4		V	$I_{OH} = -400 \mu\text{A}$
I_{LI}	Input Leakage Current		± 10	μA	$0V \leq V_{IN} \leq V_{CC}$
I_{LCR}	Input CLK Leakage Current		± 10	μA	$0.45 \leq V_{IN} \leq V_{CC}$
I_{LCR}	Input CLK Leakage Current		± 1	mA	$0V \leq V_{IN} \leq 0.45V$
I_{IL}	Input Sustaining Current on BUSY and ERROR Pins	30	500	μA	$V_{IN} = 0V$
I_{LO}	Output Leakage Current		± 10	μA	$0.45V \leq V_{OUT} \leq V_{CC}$
I_{LO}	Output Leakage Current		± 1	mA	$0V \leq V_{OUT} \leq 0.45V$
I_{CC}	Supply Current		600	mA	25°C , I_{OH} Max
C_{CLK}	CLK Input Capacitance		20	pF	$F_C = 1 \text{ MHz}$
C_{IN}	Other Input Capacitance		10	pF	$F_C = 1 \text{ MHz}$
C_O	Input/Output Capacitance		20	pF	$F_C = 1 \text{ MHz}$

A.C. CHARACTERISTICS ($V_{CC} = 5V \pm 5\%$, $T_{CASE} = 0^\circ C$ to $+85^\circ C$)*

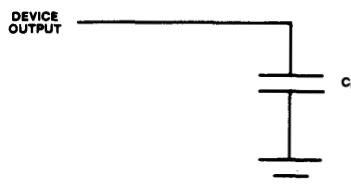
AC timings are referenced to 0.8V and 2.0V points of signals as illustrated in datasheet waveforms, unless otherwise noted.

Symbol	Parameter	6 MHz		8 MHz		10 MHz (Preliminary)		12.5 MHz (Preliminary)		Unit	Test Condition
		-6 Min	-6 Max	-8 Min	-8 Max	-10 Min	-10 Max	-12 Max	-12 Min		
1	System Clock (CLK) Period	83	250	62	250	50	250	40	250	ns	
2	System Clock (CLK) LOW Time	20	225	15	225	12	232	11	237	ns	at 1.0V
3	System Clock (CLK) HIGH Time	25	230	25	235	16	239	13	239	ns	at 3.6V
17	System Clock (CLK) Rise Time	10		10			8	—	8	ns	1.0V to 3.6V
18	System Clock (CLK) Fall Time	10		10			8	—	8	ns	3.6V to 1.0V
4	Asynch. Inputs Setup Time	30		20		20		15		ns	Note 1
5	Asynch. Inputs Hold Time	30		20		20		15		ns	Note 1
6	RESET Setup Time	33		28		23		18		ns	
7	RESET Hold Time	5		5		5		5		ns	
8	Read Data Setup Time	20		10		8		5		ns	
9	Read Data Hold Time	8		8		8		6		ns	
10	READY Setup Time	50		38		26		22		ns	
11	READY Hold Time	35		25		25		20		ns	
12	Status/PEACK Valid Delay	1	55	1	40	—	—	—	—	ns	Note 2 Note 3
12a	Status/PEACK Active Delay	—	—	—	—	1	22	3	18	ns	Note 2 Note 3
12b	Status/PEACK Inactive Delay	—	—	—	—	1	30	3	20	ns	Note 2 Note 3
13	Address Valid Delay	1	80	1	60	1	35	1	32	ns	Note 2 Note 3
14	Write Data Valid Delay	0	65	0	50	0	30	0	30	ns	Note 2 Note 3
15	Address/Status/Data Float Delay	0	80	0	50	0	47	0	32	ns	Note 2 Note 4
16	HLDA Valid Delay	0	80	0	50	0	47	0	25	ns	Note 2 Note 3
19	Address Valid To Status Valid Setup Time	—		38		27		22		ns	Note 3 Note 5 Note 6

* T_A is guaranteed as long as T_{CASE} is not exceeded.

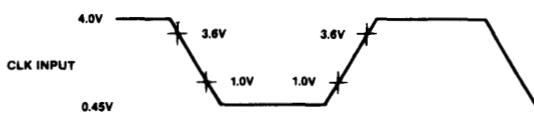
NOTES:

1. Asynchronous inputs are INTR, NMI, HOLD, PEREQ, ERROR, and BUSY. This specification is given only for testing purposes, to assure recognition at a specific CLK edge.
2. Delay from 1.0V on the CLK to 0.8V or 2.0V or float on the output as appropriate for valid or floating condition.
3. Output load: $C_L = 100 \text{ pF}$.
4. Float condition occurs when output current is less than I_{IO} in magnitude.
5. Delay measured from address either reaching 0.8V or 2.0V (valid) to status going active reaching 2.0V or status going inactive reaching 0.8V.
6. For load capacitance of 10 pF on STATUS/PEACK lines, subtract typically 7 ns for 8 MHz spec, and maximum 7 ns for 10 MHz spec.

A.C. CHARACTERISTICS (Continued)

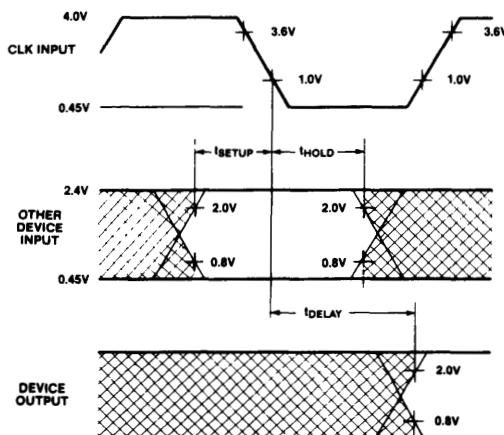
210253-37

NOTE 7:
AC Test Loading on Outputs



210253-38

NOTE 8:
AC Drive and Measurement Points—CLK Input



210253-39

NOTE 9:
AC Setup, Hold and Delay Time Measurement—General

A.C. CHARACTERISTICS (Continued)**82284 Timing Requirements**

Symbol	Parameter	82284-6		82284-8		82284-10 (Preliminary)		Units	Test Conditions
		Min	Max	Min	Max	Min	Max		
11	SRDY/SRDYEN Setup Time	25		17		15		ns	
12	SRDY/SRDYEN Hold Time	0		0		0		ns	
13	ARDY/ARDYEN Setup Time	5		0		0		ns	(Note 1)
14	ARDY/ARDYEN Hold Time	30		30		30		ns	(Note 1)
19	PCLK Delay	0	45	0	45	0	35	ns	$C_L = 75 \text{ pF}$ $I_{OL} = 5 \text{ mA}$ $I_{OH} = -1 \text{ mA}$

NOTE 1:

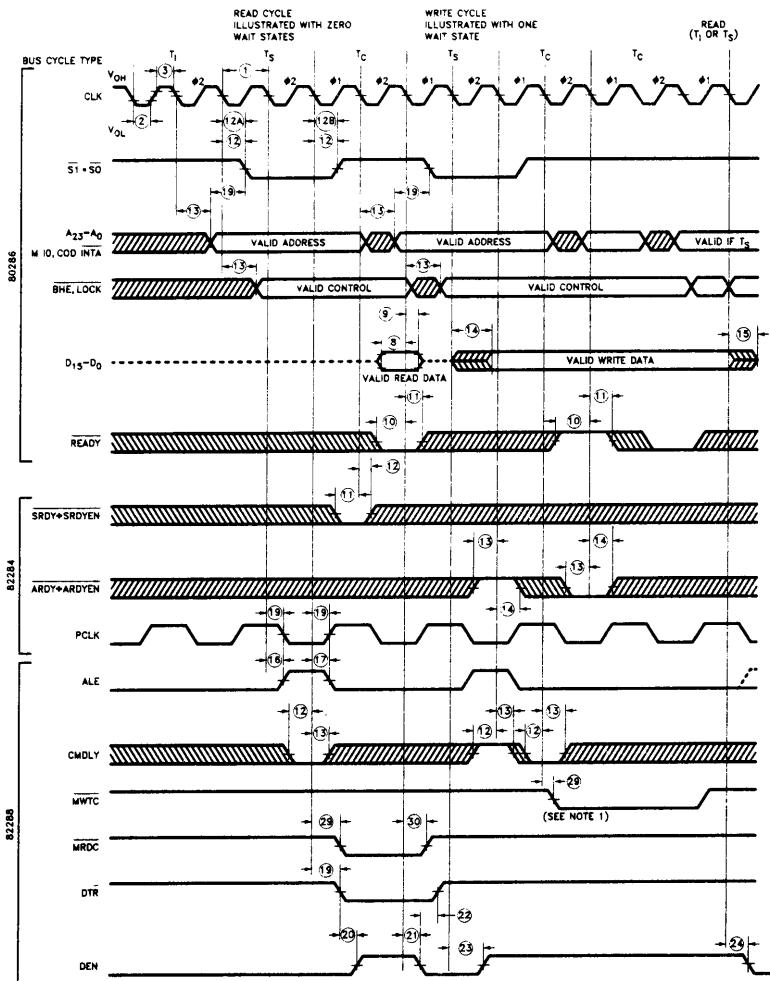
These times are given for testing purposes to assure a predetermined action.

82288 Timing Requirements

Symbol	Parameter	82288-6		82288-8		82288-10 (Preliminary)		Units	Test Conditions
		Min	Max	Min	Max	Min	Max		
12	CMDLY Setup Time	25		20		15		ns	
13	CMDLY Hold Time	1		1		1		ns	
30	Command Delay from CLK	5	30	5	25	5	20	ns	$C_L = 300 \text{ pF max}$ $I_{OL} = 32 \text{ mA max}$ $I_{OH} = -5 \text{ mA max}$
		3	40	3	25	3	21		
16	ALE Active Delay	3	25	3	20	3	16	ns	$C_L = 150 \text{ pF}$ $I_{OL} = 16 \text{ mA max}$ $I_{OH} = -1 \text{ mA max}$
17	ALE Inactive Delay		35		25		19	ns	
19	DT/R Read Active Delay		40		25		23	ns	
22	DT/R Read Inactive Delay	5	45	5	35	5	20	ns	
20	DEN Read Active Delay	5	50	5	35	5	21	ns	
21	DEN Read Inactive Delay	3	40	3	35	3	21	ns	
23	DEN Write Active Delay		35		30		23	ns	
24	DEN Write Inactive Delay	3	35	3	30	3	19	ns	

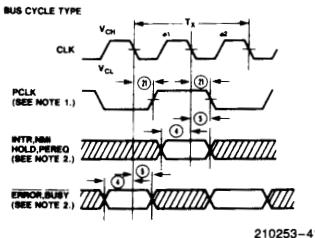
WAVEFORMS

MAJOR CYCLE TIMING



WAVEFORMS (Continued)

80286 ASYNCHRONOUS INPUT SIGNAL TIMING

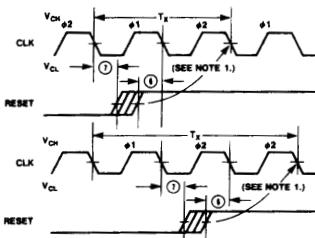


210253-41

NOTES:

1. PCLK indicates which processor cycle phase will occur on the next CLK. PCLK may not indicate the correct phase until the first bus cycle is performed.
2. These inputs are asynchronous. The setup and hold times shown assure recognition for testing purposes.

80286 RESET INPUT TIMING AND SUBSEQUENT PROCESSOR CYCLE PHASE

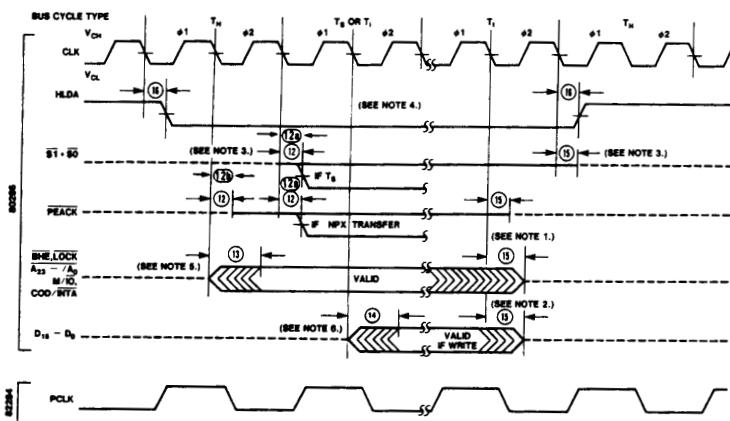


210253-42

NOTE:

- When RESET meets the setup time shown, the next CLK will start or repeat ϕ_2 of a processor cycle.

EXITING AND ENTERING HOLD



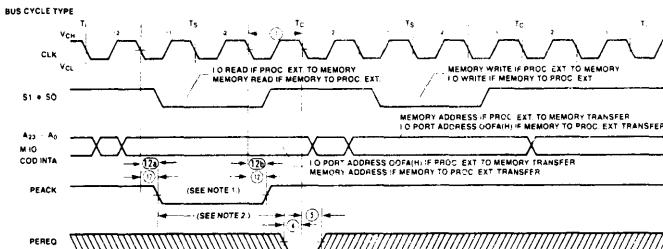
210253-43

NOTES:

1. These signals may not be driven by the 80286 during the time shown. The worst case in terms of latest float time is shown.
2. The data bus will be driven as shown if the last cycle before T_1 in the diagram was a write T_C .
3. The 80286 floats its status pins during T_H . External 20 K Ω resistors keep these signals high (see Table 16).
4. For HOLD request set up to HLDA, refer to Figure 29.
5. BHE and LOCK are driven at this time but will not become valid until T_S .
6. The data bus will remain in 3-state OFF if a read cycle is performed.

WAVEFORMS (Continued)

80286 PEREQ/PEACK TIMING FOR ONE TRANSFER ONLY

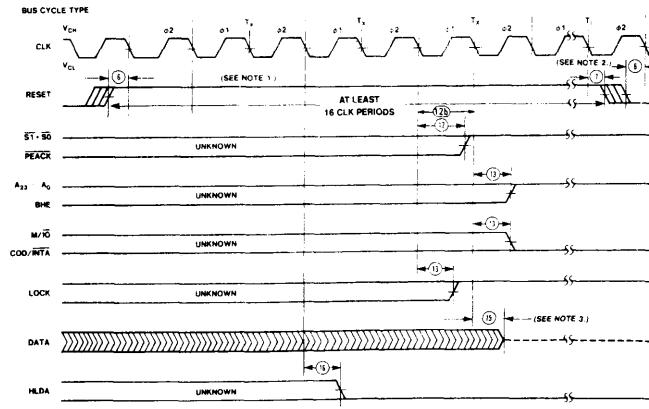


210253-44

NOTES:

1. PEACK always goes active during the first bus operation of a processor extension data operand transfer sequence. The first bus operation will be either a memory read at operand address or I/O read at port address COFA(H).
2. To prevent a second processor extension data operand transfer, the worst case maximum time (Shown above) is: $3 \times \textcircled{1} - 12a_{\max} - \textcircled{2}_{\min}$. The actual, configuration dependent, maximum time is: $3 \times \textcircled{1} - 12a_{\max} - \textcircled{2}_{\min} + A \times 2 \times \textcircled{1}$. A is the number of extra T_C states added to either the first or second bus operation of the processor extension data operand transfer sequence.

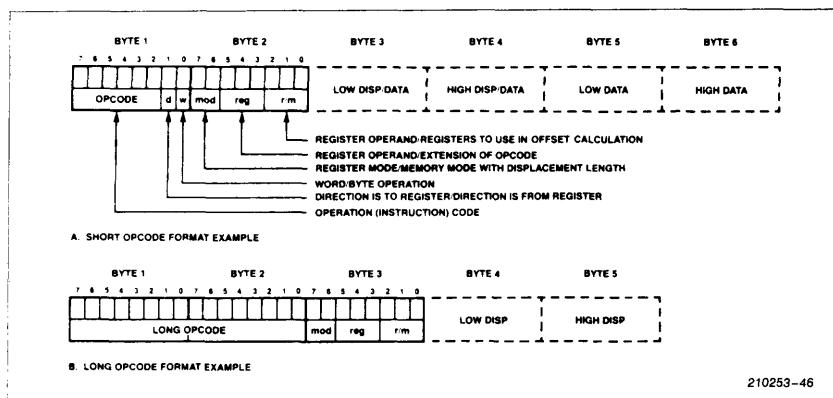
INITIAL 80286 PIN STATE DURING RESET



210253-45

NOTES:

1. Setup time for RESET ↑ may be violated with the consideration that φ1 of the processor clock may begin one system CLK period later.
2. Setup and hold times for RESET ↓ must be met for proper operation, but RESET ↓ may occur during φ1 or φ2.
3. The data bus is only guaranteed to be in 3-state OFF at the time shown.



210253-46

Figure 35. 80286 Instruction Format Examples

80286 INSTRUCTION SET SUMMARY

Instruction Timing Notes

The instruction clock counts listed below establish the maximum execution rate of the 80286. With no delays in bus cycles, the actual clock count of an 80286 program will average 5% more than the calculated clock count, due to instruction sequences which execute faster than they can be fetched from memory.

To calculate elapsed times for instruction sequences, multiply the sum of all instruction clock counts, as listed in the table below, by the processor clock period. An 8 MHz processor clock has a clock period of 125 nanoseconds and requires an 80286 system clock (CLK input) of 16 MHz.

Instruction Clock Count Assumptions

1. The instruction has been prefetched, decoded, and is ready for execution. Control transfer instruction clock counts include all time required to fetch, decode, and prepare the next instruction for execution.
2. Bus cycles do not require wait states.
3. There are no processor extension data transfer or local bus HOLD requests.
4. No exceptions occur during instruction execution.

Instruction Set Summary Notes

Addressing displacements selected by the MOD field are not shown. If necessary they appear after the instruction fields shown.

Above/below refers to unsigned value

Greater refers to positive signed value

Less refers to less positive (more negative) signed values

if d = 1 then to register; if d = 0 then from register

if w = 1 then word instruction; if w = 0 then byte instruction

if s = 0 then 16-bit immediate data form the operand

if s = 1 then an immediate data byte is sign-extended to form the 16-bit operand

x don't care

z used for string primitives for comparison with ZF FLAG

If two clock counts are given, the smaller refers to a register operand and the larger refers to a memory operand

* = add one clock if offset calculation requires summing 3 elements

n = number of times repeated

m = number of bytes of code in next instruction

Level (L)—Lexical nesting level of the procedure

The following comments describe possible exceptions, side effects, and allowed usage for instructions in both operating modes of the 80286.

REAL ADDRESS MODE ONLY

1. This is a protected mode instruction. Attempted execution in real address mode will result in an undefined opcode exception (6).
2. A segment overrun exception (13) will occur if a word operand reference at offset FFFF(H) is attempted.
3. This instruction may be executed in real address mode to initialize the CPU for protected mode.
4. The IOPL and NT fields will remain 0.
5. Processor extension segment overrun interrupt (9) will occur if the operand exceeds the segment limit.

EITHER MODE

6. An exception may occur, depending on the value of the operand.
7. LOCK is automatically asserted regardless of the presence or absence of the LOCK instruction prefix.
8. LOCK does not remain active between all operand transfers.

PROTECTED VIRTUAL ADDRESS MODE ONLY

9. A general protection exception (13) will occur if the memory operand cannot be used due to either a segment limit or access rights violation. If a stack segment limit is violated, a stack segment overrun exception (12) occurs.
10. For segment load operations, the CPL, RPL, and DPL must agree with privilege rules to avoid an exception. The segment must be present to avoid a not-present exception (11). If the SS register is the destination, and a segment not-present violation occurs, a stack exception (12) occurs.

11. All segment descriptor accesses in the GDT or LDT made by this instruction will automatically assert LOCK to maintain descriptor integrity in multiprocessor systems.
12. JMP, CALL, INT, RET, IRET instructions referring to another code segment will cause a general protection exception (13) if any privilege rule is violated.
13. A general protection exception (13) occurs if CPL \neq 0.
14. A general protection exception (13) occurs if CPL $>$ IOPL.
15. The IF field of the flag word is not updated if CPL $>$ IOPL. The IOPL field is updated only if CPL = 0.
16. Any violation of privilege rules as applied to the selector operand do not cause a protection exception; rather, the instruction does not return a result and the zero flag is cleared.
17. If the starting address of the memory operand violates a segment limit, or an invalid access is attempted, a general protection exception (13) will occur before the ESC instruction is executed. A stack segment overrun exception (12) will occur if the stack limit is violated by the operand's starting address. If a segment limit is violated during an attempted data transfer then a processor extension segment overrun exception (9) occurs.
18. The destination of an INT, JMP, CALL, RET or IRET instruction must be in the defined limit of a code segment or a general protection exception (13) will occur.

80286 INSTRUCTION SET SUMMARY

FUNCTION	FORMAT	CLOCK COUNT		COMMENTS					
		Real Address Mode	Protected Virtual Address Mode	Real Address Mode	Protected Virtual Address Mode				
DATA TRANSFER									
MOV = Move:									
Register to Register/Memory	<table border="1"><tr><td>1000100w</td><td>mod reg</td><td>r/m</td></tr></table>	1000100w	mod reg	r/m	2,3*	2,3*	2	9	
1000100w	mod reg	r/m							
Register/memory to register	<table border="1"><tr><td>1000101w</td><td>mod reg</td><td>r/m</td></tr></table>	1000101w	mod reg	r/m	2,5*	2,5*	2	9	
1000101w	mod reg	r/m							
Immediate to register/memory	<table border="1"><tr><td>1100011w</td><td>mod 000 r/m</td><td>data</td><td>data if w = 1</td></tr></table>	1100011w	mod 000 r/m	data	data if w = 1	2,3*	2,3*	2	9
1100011w	mod 000 r/m	data	data if w = 1						
Immediate to register	<table border="1"><tr><td>1011w reg</td><td>data</td><td>data if w = 1</td></tr></table>	1011w reg	data	data if w = 1	2	2			
1011w reg	data	data if w = 1							
Memory to accumulator	<table border="1"><tr><td>1010000w</td><td>addr-low</td><td>addr-high</td></tr></table>	1010000w	addr-low	addr-high	5	5	2	9	
1010000w	addr-low	addr-high							
Accumulator to memory	<table border="1"><tr><td>1010001w</td><td>addr-low</td><td>addr-high</td></tr></table>	1010001w	addr-low	addr-high	3	3	2	9	
1010001w	addr-low	addr-high							
Register/memory to segment register	<table border="1"><tr><td>10001110</td><td>mod 0 reg</td><td>r/m</td></tr></table>	10001110	mod 0 reg	r/m	2,5*	17,19*	2	9,10,11	
10001110	mod 0 reg	r/m							
Segment register to register/memory	<table border="1"><tr><td>10001100</td><td>mod 0 reg</td><td>r/m</td></tr></table>	10001100	mod 0 reg	r/m	2,3*	2,3*	2	9	
10001100	mod 0 reg	r/m							
PUSH = Push:									
Memory	<table border="1"><tr><td>11111111</td><td>mod 110 r/m</td></tr></table>	11111111	mod 110 r/m	5*	5*	2	9		
11111111	mod 110 r/m								
Register	<table border="1"><tr><td>01010 reg</td></tr></table>	01010 reg	3	3	2	9			
01010 reg									
Segment register	<table border="1"><tr><td>000reg110</td></tr></table>	000reg110	3	3	2	9			
000reg110									
Immediate	<table border="1"><tr><td>01101000</td><td>imm</td><td>data if a=0</td></tr></table>	01101000	imm	data if a=0	3	3	1	1	
01101000	imm	data if a=0							
POPA = Push All:	01100000	17	17	2	9				
POP = Pop:									
Memory	<table border="1"><tr><td>10001111</td><td>mod 000 r/m</td></tr></table>	10001111	mod 000 r/m	5*	5*	2	9		
10001111	mod 000 r/m								
Register	<table border="1"><tr><td>01011 reg</td></tr></table>	01011 reg	5	5	2	9			
01011 reg									
Segment register	<table border="1"><tr><td>000reg111</td><td>(reg#01)</td></tr></table>	000reg111	(reg#01)	5	20	2	9,10,11		
000reg111	(reg#01)								
POPA = Pop All:	01100001	18	18	2	9				
XCHG = Exchange:									
Register/memory with register	<table border="1"><tr><td>1000011w</td><td>mod reg</td><td>r/m</td></tr></table>	1000011w	mod reg	r/m	3,5*	3,5*	2,7	7,9	
1000011w	mod reg	r/m							
Register with accumulator	<table border="1"><tr><td>10010 reg</td></tr></table>	10010 reg	3	3					
10010 reg									
IN = Input from:									
Fixed port	<table border="1"><tr><td>1110010w</td><td>port</td></tr></table>	1110010w	port	5	5		14		
1110010w	port								
Variable port	<table border="1"><tr><td>1110110w</td></tr></table>	1110110w	5	5		14			
1110110w									
OUT = Output to:									
Fixed port	<table border="1"><tr><td>1110011w</td><td>port</td></tr></table>	1110011w	port	3	3		14		
1110011w	port								
Variable port	<table border="1"><tr><td>1110111w</td></tr></table>	1110111w	3	3		14			
1110111w									
XLAT = Translate byte to AL	<table border="1"><tr><td>11010111</td></tr></table>	11010111	5	5		9			
11010111									
LEA = Load EA to register	<table border="1"><tr><td>10001101</td><td>mod reg</td><td>r/m</td></tr></table>	10001101	mod reg	r/m	3*	3*			
10001101	mod reg	r/m							
LDS = Load pointer to DS	<table border="1"><tr><td>11000101</td><td>mod reg</td><td>r/m</td></tr></table>	11000101	mod reg	r/m	(mod#11)	7*	21*	2	9,10,11
11000101	mod reg	r/m							
LES = Load pointer to ES	<table border="1"><tr><td>11000100</td><td>mod reg</td><td>r/m</td></tr></table>	11000100	mod reg	r/m	(mod#1)	7*	21*	2	9,10,11
11000100	mod reg	r/m							

Shaded areas indicate instructions not available in 8086, 88 microsystems.

All mnemonics copyright Intel Corporation 1987.

80286 INSTRUCTION SET SUMMARY (Continued)

FUNCTION	FORMAT	CLOCK COUNT		COMMENTS					
		Real Address Mode	Protected Virtual Address Mode	Real Address Mode	Protected Virtual Address Mode				
DATA TRANSFER (Continued)									
LAHF = Load AH with flags	<table border="1"><tr><td>10011111</td></tr></table>	10011111	2	2					
10011111									
SAHF = Store AH into flags	<table border="1"><tr><td>10011110</td></tr></table>	10011110	2	2					
10011110									
PUSHF = Push flags	<table border="1"><tr><td>10011100</td></tr></table>	10011100	3	3	2	9			
10011100									
POPF = Pop flags	<table border="1"><tr><td>10011101</td></tr></table>	10011101	5	5	2.4	9.15			
10011101									
ARITHMETIC									
ADD = Add:									
Reg/memory with register to either	<table border="1"><tr><td>000000dw</td><td>mod reg r/m</td></tr></table>	000000dw	mod reg r/m	2,7*	2,7*	2	9		
000000dw	mod reg r/m								
Immediate to register/memory	<table border="1"><tr><td>100000sw</td><td>mod 000 r/m</td><td>data</td><td>data if s w = 01</td></tr></table>	100000sw	mod 000 r/m	data	data if s w = 01	3,7*	3,7*	2	9
100000sw	mod 000 r/m	data	data if s w = 01						
Immediate to accumulator	<table border="1"><tr><td>0000010w</td><td>data</td><td>data if w = 1</td></tr></table>	0000010w	data	data if w = 1	3	3			
0000010w	data	data if w = 1							
ADC = Add with carry:									
Reg/memory with register to either	<table border="1"><tr><td>000100dw</td><td>mod reg r/m</td></tr></table>	000100dw	mod reg r/m	2,7*	2,7*	2	9		
000100dw	mod reg r/m								
Immediate to register/memory	<table border="1"><tr><td>100000sw</td><td>mod 010 r/m</td><td>data</td><td>data if s w = 01</td></tr></table>	100000sw	mod 010 r/m	data	data if s w = 01	3,7*	3,7*	2	9
100000sw	mod 010 r/m	data	data if s w = 01						
Immediate to accumulator	<table border="1"><tr><td>0001010w</td><td>data</td><td>data if w = 1</td></tr></table>	0001010w	data	data if w = 1	3	3			
0001010w	data	data if w = 1							
INC = Increment:									
Register/memory	<table border="1"><tr><td>1111111w</td><td>mod 000 r/m</td></tr></table>	1111111w	mod 000 r/m	2,7*	2,7*	2	9		
1111111w	mod 000 r/m								
Register	<table border="1"><tr><td>01000reg</td></tr></table>	01000reg	2	2					
01000reg									
SUB = Subtract:									
Reg/memory and register to either	<table border="1"><tr><td>001010dw</td><td>mod reg r/m</td></tr></table>	001010dw	mod reg r/m	2,7*	2,7*	2	9		
001010dw	mod reg r/m								
Immediate from register/memory	<table border="1"><tr><td>100000sw</td><td>mod 101 r/m</td><td>data</td><td>data if s w = 01</td></tr></table>	100000sw	mod 101 r/m	data	data if s w = 01	3,7*	3,7*	2	9
100000sw	mod 101 r/m	data	data if s w = 01						
Immediate from accumulator	<table border="1"><tr><td>0010110w</td><td>data</td><td>data if w = 1</td></tr></table>	0010110w	data	data if w = 1	3	3			
0010110w	data	data if w = 1							
SBB = Subtract with borrow:									
Reg/memory and register to either	<table border="1"><tr><td>000110dw</td><td>mod reg r/m</td></tr></table>	000110dw	mod reg r/m	2,7*	2,7*	2	9		
000110dw	mod reg r/m								
Immediate from register/memory	<table border="1"><tr><td>100000sw</td><td>mod 011 r/m</td><td>data</td><td>data if s w = 01</td></tr></table>	100000sw	mod 011 r/m	data	data if s w = 01	3,7*	3,7*	2	9
100000sw	mod 011 r/m	data	data if s w = 01						
Immediate from accumulator	<table border="1"><tr><td>0001110w</td><td>data</td><td>data if w = 1</td></tr></table>	0001110w	data	data if w = 1	3	3			
0001110w	data	data if w = 1							
DEC = Decrement									
Register/memory	<table border="1"><tr><td>1111111w</td><td>mod 001 r/m</td></tr></table>	1111111w	mod 001 r/m	2,7*	2,7*	2	9		
1111111w	mod 001 r/m								
Register	<table border="1"><tr><td>01001 reg</td></tr></table>	01001 reg	2	2					
01001 reg									
CMP = Compare									
Register/memory with register	<table border="1"><tr><td>0011101w</td><td>mod reg r/m</td></tr></table>	0011101w	mod reg r/m	2,6*	2,6*	2	9		
0011101w	mod reg r/m								
Register with register/memory	<table border="1"><tr><td>0011100w</td><td>mod reg r/m</td></tr></table>	0011100w	mod reg r/m	2,7*	2,7*	2	9		
0011100w	mod reg r/m								
Immediate with register/memory	<table border="1"><tr><td>100000sw</td><td>mod 111 r/m</td><td>data</td><td>data if s w = 01</td></tr></table>	100000sw	mod 111 r/m	data	data if s w = 01	3,6*	3,6*	2	9
100000sw	mod 111 r/m	data	data if s w = 01						
Immediate with accumulator	<table border="1"><tr><td>0011110w</td><td>data</td><td>data if w = 1</td></tr></table>	0011110w	data	data if w = 1	3	3			
0011110w	data	data if w = 1							
NEG = Change sign									
	<table border="1"><tr><td>1111011w</td><td>mod 011 r/m</td></tr></table>	1111011w	mod 011 r/m	2	7*	2	9		
1111011w	mod 011 r/m								
AAA = ASCII adjust for add	<table border="1"><tr><td>00110111</td></tr></table>	00110111	3	3					
00110111									
DAA = Decimal adjust for add	<table border="1"><tr><td>00100111</td></tr></table>	00100111	3	3					
00100111									

All mnemonics copyright Intel Corporation 1987.

80286 INSTRUCTION SET SUMMARY (Continued)

FUNCTION	FORMAT	CLOCK COUNT		COMMENTS	
		Real Address Mode	Protected Virtual Address Mode	Real Address Mode	Protected Virtual Address Mode
ARITHMETIC (Continued)					
AAS = ASCII adjust for subtract	00111111	3	3		
DAS = Decimal adjust for subtract	00101111	3	3		
MUL = Multiply (unsigned):	11110111 w mod 100 r/m				
Register-Byte		13	13		
Register-Word		21	21		
Memory-Byte		16*	16*	2	9
Memory-Word		24*	24*	2	9
IMUL = integer multiply (signed):	11110111 w mod 101 r/m				
Register-Byte		13	13		
Register-Word		21	21		
Memory-Byte		16*	16*	2	9
Memory-Word		24*	24*	2	9
IMUL = Integer immediate multiply (signed)	01101010 s1 mod reg r/m data data Ns = 0	21,24*	21,24*	2	9
DIV = Divide (unsigned)	11110111 w mod 110 r/m				
Register-Byte		14	14	6	6
Register-Word		22	22	6	6
Memory-Byte		17*	17*	2,6	6,9
Memory-Word		25*	25*	2,6	6,9
IDIV = Integer divide (signed)	11110111 w mod 111 r/m				
Register-Byte		17	17	6	6
Register-Word		25	25	6	6
Memory-Byte		20*	20*	2,6	6,9
Memory-Word		28*	28*	2,6	6,9
AAM = ASCII adjust for multiply	11010100 00001010	16	16		
AAD = ASCII adjust for divide	11010101 00001010	14	14		
CBW = Convert byte to word	10011000	2	2		
CWD = Convert word to double word	10011001	2	2		
LOGIC					
Shift/Rotate Instructions:					
Register/Memory by 1	1101000 w mod TTT r/m	2,7*	2,7*	2	9
Register/Memory by CL	1101001 w mod TTT r/m	5+n,8+n*	5+n,8+n*	2	9
Register/Memory by Count	1100000 w mod TTT r/m count	5+n,8+n*	5+n,8+n*	2	9
	TTT Instruction				
	000 ROL				
	001 ROR				
	010 RCL				
	011 RCR				
	100 SHL/SAL				
	101 SHR				
	111 SAR				

Shaded areas indicate instructions not available in 8086, 88 microsystems.

All mnemonics copyright Intel Corporation 1987.



80286

80286 INSTRUCTION SET SUMMARY (Continued)

FUNCTION	FORMAT	CLOCK COUNT		COMMENTS					
		Real Address Mode	Protected Virtual Address Mode	Real Address Mode	Protected Virtual Address Mode				
ARITHMETIC (Continued)									
AND = And:									
Reg/memory and register to either	<table border="1"><tr><td>001000 d w</td><td>mod reg r/m</td></tr></table>	001000 d w	mod reg r/m	2,7*	2,7*	2	9		
001000 d w	mod reg r/m								
Immediate to register/memory	<table border="1"><tr><td>10000000 w</td><td>mod 100 r/m</td><td>data</td><td>data if w = 1</td></tr></table>	10000000 w	mod 100 r/m	data	data if w = 1	3,7*	3,7*	2	9
10000000 w	mod 100 r/m	data	data if w = 1						
Immediate to accumulator	<table border="1"><tr><td>0010010 w</td><td>data</td><td>data if w = 1</td></tr></table>	0010010 w	data	data if w = 1	3	3			
0010010 w	data	data if w = 1							
TEST = And function to flags, no result:									
Register/memory and register	<table border="1"><tr><td>1000010 w</td><td>mod reg r/m</td></tr></table>	1000010 w	mod reg r/m	2,6*	2,6*	2	9		
1000010 w	mod reg r/m								
Immediate data and register/memory	<table border="1"><tr><td>1111011 w</td><td>mod 000 r/m</td><td>data</td><td>data if w = 1</td></tr></table>	1111011 w	mod 000 r/m	data	data if w = 1	3,6*	3,6*	2	9
1111011 w	mod 000 r/m	data	data if w = 1						
Immediate data and accumulator	<table border="1"><tr><td>1010100 w</td><td>data</td><td>data if w = 1</td></tr></table>	1010100 w	data	data if w = 1	3	3			
1010100 w	data	data if w = 1							
OR = Or:									
Reg/memory and register to either	<table border="1"><tr><td>0000100 d w</td><td>mod reg r/m</td></tr></table>	0000100 d w	mod reg r/m	2,7*	2,7*	2	9		
0000100 d w	mod reg r/m								
Immediate to register/memory	<table border="1"><tr><td>10000000 w</td><td>mod 001 r/m</td><td>data</td><td>data if w = 1</td></tr></table>	10000000 w	mod 001 r/m	data	data if w = 1	3,7*	3,7*	2	9
10000000 w	mod 001 r/m	data	data if w = 1						
Immediate to accumulator	<table border="1"><tr><td>0000110 w</td><td>data</td><td>data if w = 1</td></tr></table>	0000110 w	data	data if w = 1	3	3			
0000110 w	data	data if w = 1							
XOR = Exclusive or:									
Reg/memory and register to either	<table border="1"><tr><td>0011000 d w</td><td>mod reg r/m</td></tr></table>	0011000 d w	mod reg r/m	2,7*	2,7*	2	9		
0011000 d w	mod reg r/m								
Immediate to register/memory	<table border="1"><tr><td>10000000 w</td><td>mod 110 r/m</td><td>data</td><td>data if w = 1</td></tr></table>	10000000 w	mod 110 r/m	data	data if w = 1	3,7*	3,7*	2	9
10000000 w	mod 110 r/m	data	data if w = 1						
Immediate to accumulator	<table border="1"><tr><td>0011010 w</td><td>data</td><td>data if w = 1</td></tr></table>	0011010 w	data	data if w = 1	3	3			
0011010 w	data	data if w = 1							
NOT = Invert register/memory	<table border="1"><tr><td>1111011 w</td><td>mod 010 r/m</td></tr></table>	1111011 w	mod 010 r/m	2,7*	2,7*	2	9		
1111011 w	mod 010 r/m								
STRING MANIPULATION:									
MOV\$ = Move byte/word	<table border="1"><tr><td>1010010 w</td></tr></table>	1010010 w	5	5	2	9			
1010010 w									
CMP\$ = Compare byte/word	<table border="1"><tr><td>1010011 w</td></tr></table>	1010011 w	8	8	2	9			
1010011 w									
SCAS\$ = Scan byte/word	<table border="1"><tr><td>1010111 w</td></tr></table>	1010111 w	7	7	2	9			
1010111 w									
LODS\$ = Load byte/wd to AL/A	<table border="1"><tr><td>1010110 w</td></tr></table>	1010110 w	5	5	2	9			
1010110 w									
STOS\$ = Stor byte/wd from AL/A	<table border="1"><tr><td>1010101 w</td></tr></table>	1010101 w	3	3	2	9			
1010101 w									
IN\$ = Input byte/wd from DX port	<table border="1"><tr><td>0110110 w</td></tr></table>	0110110 w	5	5	2	9,14			
0110110 w									
OUT\$ = Output byte/wd to DX port	<table border="1"><tr><td>0110111 w</td></tr></table>	0110111 w	5	5	2	9,14			
0110111 w									
Repeated by count in CX									
MOV\$ = Move string	<table border="1"><tr><td>11110011 1010010 w</td></tr></table>	11110011 1010010 w	5+4n	5+4n	2	9			
11110011 1010010 w									
CMP\$ = Compare string	<table border="1"><tr><td>1111001 z 1010011 w</td></tr></table>	1111001 z 1010011 w	5+9n	5+9n	2,8	8,9			
1111001 z 1010011 w									
SCAS\$ = Scan string	<table border="1"><tr><td>1111001 z 1010111 w</td></tr></table>	1111001 z 1010111 w	5+8n	5+8n	2,8	8,9			
1111001 z 1010111 w									
LODS\$ = Load string	<table border="1"><tr><td>11110011 1010110 w</td></tr></table>	11110011 1010110 w	5+4n	5+4n	2,8	8,9			
11110011 1010110 w									
STOS\$ = Store string	<table border="1"><tr><td>11110011 1010101 w</td></tr></table>	11110011 1010101 w	4+3n	4+3n	2,8	8,9			
11110011 1010101 w									
IN\$ = Input string	<table border="1"><tr><td>11110011 0110110 w</td></tr></table>	11110011 0110110 w	5+4n	5+4n	2	9,14			
11110011 0110110 w									
OUT\$ = Output string	<table border="1"><tr><td>11110011 0110111 w</td></tr></table>	11110011 0110111 w	5+4n	5+4n	2	9,14			
11110011 0110111 w									

Shaded areas indicate instructions not available in 8086, 88 microsystems.

All mnemonics copyright Intel Corporation 1987.

80286 INSTRUCTION SET SUMMARY (Continued)

FUNCTION	FORMAT	CLOCK COUNT		COMMENTS				
		Real Address Mode	Protected Virtual Address Mode	Real Address Mode	Protected Virtual Address Mode			
CONTROL TRANSFER								
CALL Call:								
Direct within segment	<table border="1"><tr><td>11101000</td><td>disp-low</td><td>disp-high</td></tr></table>	11101000	disp-low	disp-high	7 + m	7 + m	2	16
11101000	disp-low	disp-high						
Register/memory indirect within segment	<table border="1"><tr><td>11111111</td><td>mod 0 10 r/m</td><td></td></tr></table>	11111111	mod 0 10 r/m		7 + m, 11 + m*	7 + m, 11 + m*	2,8	8,9,18
11111111	mod 0 10 r/m							
Direct intersegment	<table border="1"><tr><td>10011010</td><td>segment offset</td><td></td></tr></table>	10011010	segment offset		13 + m	26 + m	2	11,12,18
10011010	segment offset							
Protected Mode Only (Direct intersegment):	<table border="1"><tr><td></td><td>segment selector</td><td></td></tr></table>		segment selector					
	segment selector							
Via call gate to same privilege level			41 + m		8,11,12,18			
Via call gate to different privilege level, no parameters			82 + m		8,11,12,18			
Via call gate to different privilege level, x parameters			86 + 4x + m		8,11,12,18			
Via TSS			177 + m		8,11,12,18			
Via task gate			182 + m		8,11,12,18			
Indirect intersegment	<table border="1"><tr><td>11111111</td><td>mod 0 1 r/m</td><td>(mod > 11)</td></tr></table>	11111111	mod 0 1 r/m	(mod > 11)	16 + m	29 + m*	2	8,9,11,12,18
11111111	mod 0 1 r/m	(mod > 11)						
Protected Mode Only (Indirect Intersegment):								
Via call gate to same privilege level			44 + m*		8,9,11,12,18			
Via call gate to different privilege level, no parameters			93 + m*		8,9,11,12,18			
Via call gate to different privilege level, x parameters			90 + 4x + m*		8,9,11,12,18			
Via TSS			180 + m*		8,9,11,12,18			
Via task gate			185 + m*		8,9,11,12,18			
JMP = Unconditional jump:								
Short/long	<table border="1"><tr><td>11101011</td><td>disp-low</td><td></td></tr></table>	11101011	disp-low		7 + m	7 + m		18
11101011	disp-low							
Direct within segment	<table border="1"><tr><td>11101001</td><td>disp-low</td><td>disp-high</td></tr></table>	11101001	disp-low	disp-high	7 + m	7 + m		18
11101001	disp-low	disp-high						
Register/memory indirect within segment	<table border="1"><tr><td>11111111</td><td>mod 100 r/m</td><td></td></tr></table>	11111111	mod 100 r/m		7 + m, 11 + m*	7 + m, 11 + m*	2	9,18
11111111	mod 100 r/m							
Direct intersegment	<table border="1"><tr><td>11101010</td><td>segment offset</td><td></td></tr></table>	11101010	segment offset		11 + m	23 + m		11,12,18
11101010	segment offset							
Protected Mode Only (Direct intersegment):	<table border="1"><tr><td></td><td>segment selector</td><td></td></tr></table>		segment selector					
	segment selector							
Via call gate to same privilege level			38 + m		8,11,12,18			
Via TSS			175 + m		8,11,12,18			
Via task gate			180 + m		8,11,12,18			
Indirect intersegment	<table border="1"><tr><td>11111111</td><td>mod 101 r/n</td><td>(mod > 11)</td></tr></table>	11111111	mod 101 r/n	(mod > 11)	15 + m*	26 + m*	2	8,9,11,12,18
11111111	mod 101 r/n	(mod > 11)						
Protected Mode Only (Indirect intersegment):								
Via call gate to same privilege level			41 + m*		8,9,11,12,18			
Via TSS			178 + m*		8,9,11,12,18			
Via task gate			183 + m*		8,9,11,12,18			
RET = Return from CALL:								
Within segment	<table border="1"><tr><td>11000011</td><td></td><td></td></tr></table>	11000011			11 + m	11 + m	2	8,9,18
11000011								
Within seg adding immed to SP	<table border="1"><tr><td>11000010</td><td>data-low</td><td>data-high</td></tr></table>	11000010	data-low	data-high	11 + m	11 + m	2	8,9,18
11000010	data-low	data-high						
Intersegment	<table border="1"><tr><td>11001011</td><td></td><td></td></tr></table>	11001011			15 + m	25 + m	2	8,9,11,12,18
11001011								
Intersegment adding immediate to SP	<table border="1"><tr><td>11001010</td><td>data-low</td><td>data-high</td></tr></table>	11001010	data-low	data-high	15 + m		2	8,9,11,12,18
11001010	data-low	data-high						
Protected Mode Only (RET):				55 + m	9,11,12,18			
To different privilege level								

All mnemonics copyright Intel Corporation 1987.



80286

80286 INSTRUCTION SET SUMMARY (Continued)

FUNCTION	FORMAT	CLOCK COUNT		COMMENTS	
		Real Address Mode	Protected Virtual Address Mode	Real Address Mode	Protected Virtual Address Mode
CONTROL TRANSFER (Continued)					
JE/JZ = Jump on equal zero	01110100 disp	7 + m or 3	7 + m or 3		18
JL/JNGE = Jump on less/not greater or equal	01111100 disp	7 + m or 3	7 + m or 3		18
JLE/JNG = Jump on less or equal/not greater	01111110 disp	7 + m or 3	7 + m or 3		18
JB/JNAE = Jump on below/not above or equal	01110010 disp	7 + m or 3	7 + m or 3		18
JBE/JNA = Jump on below or equal/not above	01110110 disp	7 + m or 3	7 + m or 3		18
JP/JPE = Jump on parity/parity even	01111010 disp	7 + m or 3	7 + m or 3		18
JO = Jump on overflow	01110000 disp	7 + m or 3	7 + m or 3		18
JS = Jump on sign	01111000 disp	7 + m or 3	7 + m or 3		18
JNE/JNZ = Jump on not equal/not zero	01110101 disp	7 + m or 3	7 + m or 3		18
JNL/JQE = Jump on not less/greater or equal	01111101 disp	7 + m or 3	7 + m or 3		18
JNLE/JQ = Jump on not less or equal/greater	01111111 disp	7 + m or 3	7 + m or 3		18
JNB/JAE = Jump on not below/above or equal	01110011 disp	7 + m or 3	7 + m or 3		18
JNBE/JA = Jump on not below or equal/above	01110111 disp	7 + m or 3	7 + m or 3		18
JNP/JPO = Jump on not par/par odd	01111011 disp	7 + m or 3	7 + m or 3		18
JNO = Jump on not overflow	01110001 disp	7 + m or 3	7 + m or 3		18
JNS = Jump on not sign	01111001 disp	7 + m or 3	7 + m or 3		18
LOOP = Loop CX times	11100010 disp	8 + m or 4	8 + m or 4		18
LOOPZ/LOOPE = Loop while zero/equal	11100001 disp	8 + m or 4	8 + m or 4		18
LOOPNZ/LOOPNE = Loop while not zero/equal	11100000 disp	8 + m or 4	8 + m or 4		18
JCXZ = Jump on CX zero	11100011 disp	8 + m or 4	8 + m or 4		18
ENTER = Enter Procedure	11001000 date-low date-high t			2,8	8,9
L=0		11	11		
L=1		16	15	2,8	8,9
L > 1		16+4(L-1)	16+4(L-1)	2,8	8,9
LEAVE = Leave Procedure	11001001	6	6		
INT = Interrupt:					
Type specified	11001101 type	23+m		2,7,8	
Type 3	11001100	23+m		2,7,8	
INTO = Interrupt on overflow	11001110	24+m or 3 (3 if no interrupt)		2,6,8	

Shaded areas indicate instructions not available in 8086, 88 microsystems.

All mnemonics copyright Intel Corporation 1987.

80286 INSTRUCTION SET SUMMARY (Continued)

FUNCTION	FORMAT	CLOCK COUNT		COMMENTS					
		Real Address Mode	Protected Virtual Address Mode	Real Address Mode	Protected Virtual Address Mode				
CONTROL TRANSFER (Continued)									
Protected Mode Only:									
Via interrupt or trap gate to same privilege level			40 + m		7,8,11,12,18				
Via interrupt or trap gate to fit different privilege level*			78 + m		7,8,11,12,18				
Via Task Gate			167 + m		7,8,11,12,18				
IRET = Interrupt return	<table border="1"><tr><td>11001111</td></tr></table>	11001111	17 + m	31 + m	2,4	8,9,11,12,15,18			
11001111									
Protected Mode Only:									
To different privilege level			55 + m		8,9,11,12,15,18				
To different task (NT = 1)			169 + m		8,9,11,12,18				
BOUND = Detect value out of range	<table border="1"><tr><td>01100010</td><td>mod reg</td><td>r/m</td></tr></table>	01100010	mod reg	r/m	13*	13*	2,6	8,8,9,11,12,18	
01100010	mod reg	r/m							
PROCESSOR CONTROL									
CLC = Clear carry	<table border="1"><tr><td>11111000</td></tr></table>	11111000	2	2					
11111000									
CMC = Complement carry	<table border="1"><tr><td>11110101</td></tr></table>	11110101	2	2					
11110101									
STC = Set carry	<table border="1"><tr><td>11111001</td></tr></table>	11111001	2	2					
11111001									
CLD = Clear direction	<table border="1"><tr><td>11111100</td></tr></table>	11111100	2	2					
11111100									
STD = Set direction	<table border="1"><tr><td>11111101</td></tr></table>	11111101	2	2					
11111101									
CLI = Clear interrupt	<table border="1"><tr><td>11111010</td></tr></table>	11111010	3	3		14			
11111010									
STI = Set interrupt	<table border="1"><tr><td>11111011</td></tr></table>	11111011	2	2		14			
11111011									
HLT = Halt	<table border="1"><tr><td>11110100</td></tr></table>	11110100	2	2		13			
11110100									
WAIT = Wait	<table border="1"><tr><td>10011011</td></tr></table>	10011011	3	3					
10011011									
LOCK = Bus lock prefix	<table border="1"><tr><td>11110000</td></tr></table>	11110000	0	0		14			
11110000									
CTB = Clear task switched flag	<table border="1"><tr><td>00001111</td><td>00000110</td></tr></table>	00001111	00000110	2	2	3	13		
00001111	00000110								
ESC = Processor Extension Escape	<table border="1"><tr><td>11011TTT</td><td>mod LLL</td><td>r/m</td></tr></table>	11011TTT	mod LLL	r/m	9-20*	9-20*	5,8	8,17	
11011TTT	mod LLL	r/m							
(TTT LLL are opcode to processor extension)									
SEG = Segment Override Prefix	<table border="1"><tr><td>001 reg 110</td></tr></table>	001 reg 110	0	0					
001 reg 110									
PROTECTION CONTROL									
LGDT = Load global descriptor table register	<table border="1"><tr><td>00001111</td><td>00000001</td><td>mod 010</td><td>r/m</td></tr></table>	00001111	00000001	mod 010	r/m	11*	11*	2,3	9,13
00001111	00000001	mod 010	r/m						
SGDT = Store global descriptor table register	<table border="1"><tr><td>00001111</td><td>00000001</td><td>mod 000</td><td>r/m</td></tr></table>	00001111	00000001	mod 000	r/m	11*	11*	2,3	9
00001111	00000001	mod 000	r/m						
LIDT = Load interrupt descriptor table register	<table border="1"><tr><td>00001111</td><td>00000001</td><td>mod 011</td><td>r/m</td></tr></table>	00001111	00000001	mod 011	r/m	12*	12*	2,3	9,13
00001111	00000001	mod 011	r/m						
SIDT = Store interrupt descriptor table register	<table border="1"><tr><td>00001111</td><td>00000001</td><td>mod 001</td><td>r/m</td></tr></table>	00001111	00000001	mod 001	r/m	12*	12*	2,3	9
00001111	00000001	mod 001	r/m						
LLDT = Load local descriptor table register from register memory	<table border="1"><tr><td>00001111</td><td>00000000</td><td>mod 010</td><td>r/m</td></tr></table>	00001111	00000000	mod 010	r/m		17,19*	1	8,11,13
00001111	00000000	mod 010	r/m						
BLDT = Store local descriptor table register to register/memory	<table border="1"><tr><td>00001111</td><td>00000000</td><td>mod 000</td><td>r/m</td></tr></table>	00001111	00000000	mod 000	r/m		2,3*	1	9
00001111	00000000	mod 000	r/m						

Shaded areas indicate instructions not available in 8086, 88 microsystems.

All mnemonics copyright Intel Corporation 1987.

80286 INSTRUCTION SET SUMMARY (Continued)

FUNCTION	FORMAT	CLOCK COUNT		COMMENTS	
		Real Address Mode	Protected Address Mode	Real Address Mode	Protected Address Mode
PROTECTION CONTROL (Continued)					
LTR – Local task register from register/memory	00001111 00000000 mod 011 r/m			17,19*	1 3
STR – Store task register to register/memory	00001111 00000000 mod 001 r/m			2,3*	1 9
LMSW – Load machine status word from register/memory	00001111 00000001 mod 110 r/m			3,5*	2,9 6,10
MWSW – Store machine status word	00001111 00000001 mod 100 r/m			2,3*	2,3 9
LAR – Load access rights from register/memory	00001111 00000010 mod reg r/m			14,16*	1 6,11,16
LSL – Load segment limit from register/memory	00001111 00000011 mod reg r/m			14,16*	1 6,15,16
APPL – Adjust requested privilege level: from register/memory	01100011 mod reg r/m			10*,11*	2 6,9,10
VERR – Verify read access: register/memory	00001111 00000000 mod 100 r/m			14,16*	1 6,11,16
VERW – Verify write access:	00001111 00000000 mod 101 r/m			14,16*	1 6,11,16

Shaded areas indicate instructions not available in 8086, 88 microsystems.

Footnotes

The Effective Address (EA) of the memory operand is computed according to the mod and r/m fields:

if mod = 11 then r/m is treated as a REG field
 if mod = 00 then DISP = 0*, disp-low and disp-high are absent
 if mod = 01 then DISP = disp-low sign-extended to 16 bits, disp-high is absent
 if mod = 10 then DISP = disp-high: disp-low
 if r/m = 000 then EA = (BX) + (SI) + DISP
 if r/m = 001 then EA = (BX) + (DI) + DISP
 if r/m = 010 then EA = (BP) + (SI) + DISP
 if r/m = 011 then EA = (BP) + (DI) + DISP
 if r/m = 100 then EA = (SI) + DISP
 if r/m = 101 then EA = (DI) + DISP
 if r/m = 110 then EA = (BP) + DISP*
 if r/m = 111 then EA = (BX) + DISP

DISP follows 2nd byte of instruction (before data if required)

*except if mod = 00 and r/m = 110 then EQ = disp-high: disp-low.

REG is assigned according to the following table:

16-Bit (w = 1)		8-Bit (w = 0)	
000	AX	000	AL
001	CX	001	CL
010	DX	010	DL
011	BX	011	BL
100	SP	100	AH
101	BP	101	CH
101	SI	110	DH
111	DI	111	BH

The physical addresses of all operands addressed by the BP register are computed using the SS segment register. The physical addresses of the destination operands of the string primitive operations (those addressed by the DI register) are computed using the ES segment, which may not be overridden.

SEGMENT OVERRIDE PREFIX

0	0	1	reg	1	1	0
---	---	---	-----	---	---	---

reg is assigned according to the following:

reg	Segment Register
00	ES
01	CS
10	SS
11	DC



80287 80-BIT HMOS NUMERIC PROCESSOR EXTENSION (80287-3, 80287-6, 80287-8, 80287-10)

- High Performance 80-Bit Internal Architecture
- Implements Proposed IEEE Floating Point Standard 754
- Expands 80286 Data types to Include 32-, 64-, 80-Bit Floating Point, 32-, 64-Bit Integers and 18-Digit BCD Operands
- Object Code Compatible with 8087
- Built-In Exception Handling
- Operates In Both Real and Protected Mode 80286 Systems
- 8x80-Bit, Individually Addressable, Numeric Register Stack
- Protected Mode Operation Completely Conforms to the 80286 Memory Management and Protection Mechanisms
- Directly Extends 80286 Instruction Set to Trigonometric, Logarithmic, Exponential and Arithmetic Instructions for All Data types
- Operates with 80386 CPU without Software Modification
- Available in EXPRESS—Standard Temperature Range
- Available In 40 pin-CERDIP package
(see Packaging Spec: Order #231369)

The Intel 80287 is a high performance numerics processor extension that extends the 80286 architecture with floating point, extended integer and BCD data types. The 80286/80287 computing system fully conforms to the proposed IEEE Floating Point Standard. Using a numerics oriented architecture, the 80287 adds over fifty mnemonics to the 80286/80287 instruction set, making the 80286/80287 a complete solution for high performance numeric processing. The 80287 is implemented in N-channel, depletion load, silicon gate technology (HMOS) and packaged in a 40-pin cerdip package. The 80286/80287 is object code compatible with the 8086/8087 and 8088/8087.

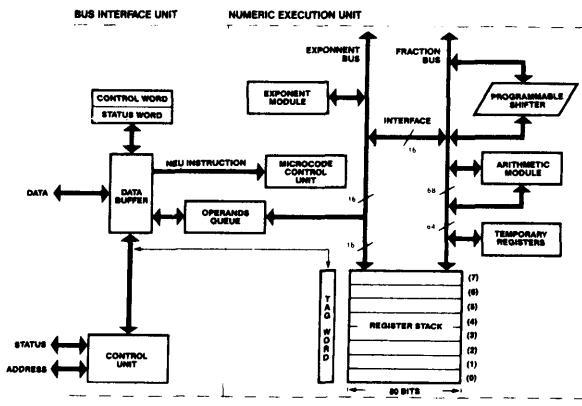


Figure 1. 80287 Block Diagram

N/C	1	40	N/C
N/C	2	39	CKM
N/C	3	38	N/C
N.C.	4	37	N/C
D15	5	36	PEACK
D14	6	35	RESET
D13	7	34	NPS1
D12	8	33	NPS2
Vcc	9	32	CLK
Vss	10	31	CM01
D11	11	30	Vss
D10	12	29	CM02
N/C	13	28	CM0R
D9	14	27	NPWR
D8	15	26	NPRD
D7	16	25	ERROR
D6	17	24	BUSY
D5	18	23	PREQ
D4	19	22	00
D3	20	21	01
			02

NOTE:
N/C Pins should not be connected

Figure 2.
80287 Pin Configuration

Table 1. 80287 Pin Description

Symbols	Type	Name and Function
CLK	I	CLOCK INPUT: this clock provides the basic timing for internal 80287 operations. Special MOS level inputs are required. The 82284 or 8284A CLK outputs are compatible to this input.
CKM	I	CLOCK MODE SIGNAL: indicates whether CLK input is to be divided by 3 or used directly. A HIGH input will cause CLK to be used directly. This input must be connected to V _{CC} or V _{SS} as appropriate. This input must be either HIGH or LOW 20 CLK cycles before RESET goes LOW.
RESET	I	SYSTEM RESET: causes the 80287 to immediately terminate its present activity and enter a dormant state. RESET is required to be HIGH for more than 4 80287 CLK cycles. For proper initialization the HIGH-LOW transition must occur no sooner than 50 µs after V _{CC} and CLK meet their D.C. and A.C. specifications.
D15-D0	I/O	DATA: 1-bit bidirectional data bus. Inputs to these pins may be applied asynchronous to the 80287 clock.
BUSY	O	BUSY STATUS: asserted by the 80287 to indicate that it is currently executing a command.
ERROR	O	ERROR STATUS: reflects the ES bit of the status word. This signal indicates that an unmasked error condition exists.
PEREQ	O	PROCESSOR EXTENSION DATA CHANNEL OPERAND TRANSFER REQUEST: a HIGH on this output indicates that the 80287 is ready to transfer data. PEREQ will be disabled upon assertion of PEACK or upon actual data transfer, whichever occurs first, if no more transfers are required.
PEACK	I	PROCESSOR EXTENSION DATA CHANNEL OPERAND TRANSFER ACKNOWLEDGE: acknowledges that the request signal (PEREQ) has been recognized. Will cause the request (PEREQ) to be withdrawn in case there are no more transfers required. PEACK may be asynchronous to the 80287 clock.
NPRD	I	NUMERIC PROCESSOR READ: Enables transfer of data from the 80287. This input may be asynchronous to the 80287 clock.
NPWR	I	NUMERIC PROCESSOR READ: Enables transfer of data from the 80287. This input may be asynchronous to the 80287 clock.
NPS1, NPS2	I	NUMERIC PROCESSOR SELECTS: indicate the CPU is performing an ESCAPE instruction. Concurrent assertion of these signals (i.e., NPS1 is LOW and NPS2 is HIGH) enables the 80287 to perform floating point instructions. No data transfers involving the 80287 will occur unless the device is selected via these lines. These inputs may be asynchronous to the 80287 clock.
CMD1, CMD0	I	COMMAND LINES: These, along with select inputs, allow the CPU to direct the operation of the 80287. These inputs may be asynchronous to the 80287 clock.

Table 1. 80187 Pin Description (Continued)

Symbols	Type	Name and Function
V _{SS}	I	System ground, both pins must be connected to ground.
V _{CC}	I	+5V supply

FUNCTIONAL DESCRIPTION

The 80287 Numeric Processor Extension (NPX) provides arithmetic instructions for a variety of numeric data types in 80286/80287 systems. It also executes numerous built-in transcendental functions (e.g., tangent and log functions). The 80287 executes instructions in parallel with an 80286. It effec-

tively extends the register and instruction set of an 80286 system for existing 80286 data types and adds several new data types as well. Figure 3 presents the program visible register model of the 80286/80287. Essentially, the 80287 can be treated as an additional resource or an extension to the 80286 that can be used as a single unified system, the 80286/80287.

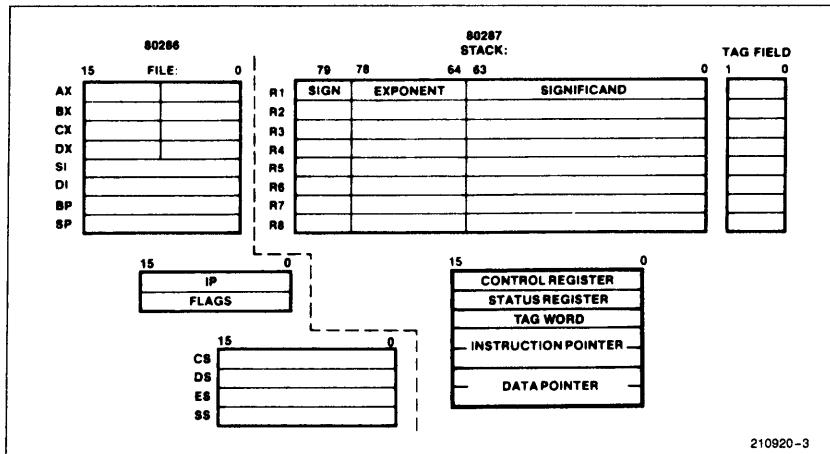


Figure 3. 80286/80287 Architecture

The 80287 has two operating modes similar to the two modes of the 80286. When reset, 80287 is in the real address mode. It can be placed in the protected virtual address mode by executing the SETPM ESC instruction. The 80287 cannot be switched back to the real address mode except by reset. In the real address mode, the 80286/80287 is completely software compatible with 8086/8087 and 8088/8087.

Once in protected mode, all references to memory for numerics data or status information, obey the 80286 memory management and protection rules giving a fully protected extension of the 80286 CPU. In the protected mode, 80286/80287 numerics software is also completely compatible with 8086/8087 and 8088/8087.

SYSTEM CONFIGURATION WITH 80286

As a processor extension to an 80286, the 80287 can be connected to the CPU as shown in Figure 4A. The data channel control signals (PEREQ, PEACK), the BUSY signal and the NPROD, NPWR signals, allow the NPX to receive instructions and data from the CPU. When in the protected mode, all information received by the NPX is validated by the 80286 memory management and protection unit. Once started, the 80287 can process in parallel with and independent of the host CPU. When the NPX detects an error or exception, it will indicate this to the CPU by asserting the ERROR signal.

The NPX uses the processor extension request and acknowledge pins of the 80286 CPU to implement data transfers with memory under the protection model of the CPU. The full virtual and physical address space of the 80286 is available. Data for the 80287 in memory is addressed and represented in the same manner as for an 8087.

The 80287 can operate either directly from the CPU clock or with a dedicated clock. For operation with the CPU clock ($CKM = 0$), the 80287 works at one-third the frequency of the system clock (i.e., for an 8 MHz 80286, the 16 MHz system clock is divided down to 5.3 MHz). The 80287 provides a capability to internally divide the CPU clock by three to produce the required internal clock (33% duty cycle). To use a higher performance 80287 (8 MHz), an 8284A clock driver and appropriate crystal may be used to directly drive the 80287 with a $\frac{1}{3}$ duty cycle clock on the CLK input ($CKM = 1$). The following table describes the relationship between the clock speed and the 287 operating speed as a function of the CKM state.

287 Speed	CLK Speed	
	CKM = 0	CKM = 1
5 MHz	12 MHz	5 MHz
6 MHz	16 MHz	6 MHz
8 MHz	20 MHz	8 MHz
10 MHz	25 MHz	10 MHz

SYSTEM CONFIGURATION WITH 80386

The 80287 can also be connected as a processor extension to the 80386 CPU as shown in Figure 4b. All software written for 8086/8087 and 80286/80287 is object code compatible with 80386/80287 and can benefit from the increased speed of the 80386 CPU.

Note that the PEACK input pin is pulled high. This is because the 80287 is not required to keep track of the number of words transferred during an operand transfer when it is connected to the 80386 CPU. Unlike the 80286 CPU, the 80386 CPU knows the exact length of the operand being transferred to/from the 80287. After an ESC instruction has been sent to the 80287, the 80386 processor extension data channel will initiate the data transfer as soon as it receives the PEREQ signal from the 80287. The transfer is automatically terminated by the 80386 CPU as soon as all the words of the operand have been transferred.

Because of the very high speed local bus of the 80386 CPU, the 80287 cannot reside directly on the CPU local bus. A local bus controller logic is used to generate the necessary read and write cycle timings as well as the chip select timings for the 80287. The 80386 CPU uses I/O addresses 800000F0 through 800000FF to communicate with the 80287. This is beyond the normal I/O address space of the CPU and makes it easier to generate the chip select signals using A31 and M/I/O. It may also be noted that the 80386 CPU automatically generates 16-bit bus cycles whenever it communicates with the 80287.

HARDWARE INTERFACE

Communication of instructions and data operands between the 80286 and 80287 is handled by the CMD0, CMD1, NPST, NPS2, NPROD, and NPWR signals. I/O port addresses 00FBH, 00FAH, and 00FCH are used by the 80286 for this communication. When any of these addresses are used, the NPS1 input must be LOW and NPS2 input HIGH. The IORC and IOWC outputs of the 82288 identify I/O space transfers (see Figure 4A). CMD0 should be connected to latched 80286 A1 and CMD1 should be connected to latched 80286 A2.

I/O ports 00FBH to 00FFH are reserved for the 80286/80287 interface. To guarantee correct operation of the 80287, programs must not perform any I/O operations to these ports.

The PEREQ, PEACK, BUSY, and ERROR signals of the 80287 are connected to the same-named 80286 input. The data pins of the 80287 should be directly connected to the 80286 data bus. Note that all bus drivers connected to the 80286 local bus must be inhibited when the 80286 reads from the 80287. The use of M/I/O in the decoder prevents INTA bus cycles from disabling the data transceivers.

PROGRAMMING INTERFACE

Table 2 lists the seven data types the 80287 supports and presents the format for each type. These

values are stored in memory with the least significant digits at the lowest memory address. Programs retrieve these values by generating the lowest address. All values should start at even addresses for maximum system performance.

Internally the 80287 holds all numbers in the temporary real format. Load instructions automatically convert operands represented in memory as 16-, 32-, or 64-bit integers, 32- or 64-bit floating point number or

18-digit packed BCD numbers into temporary real format. Store instructions perform the reverse type conversion.

80287 computations use the processor's register stack. These eight 80-bit registers provide the equivalent capacity of 40 16-bit registers. The 80287 register set can be accessed as a stack, with instructions operating on the top one or two stack elements, or as a fixed register set, with instructions operating on explicitly designated registers.

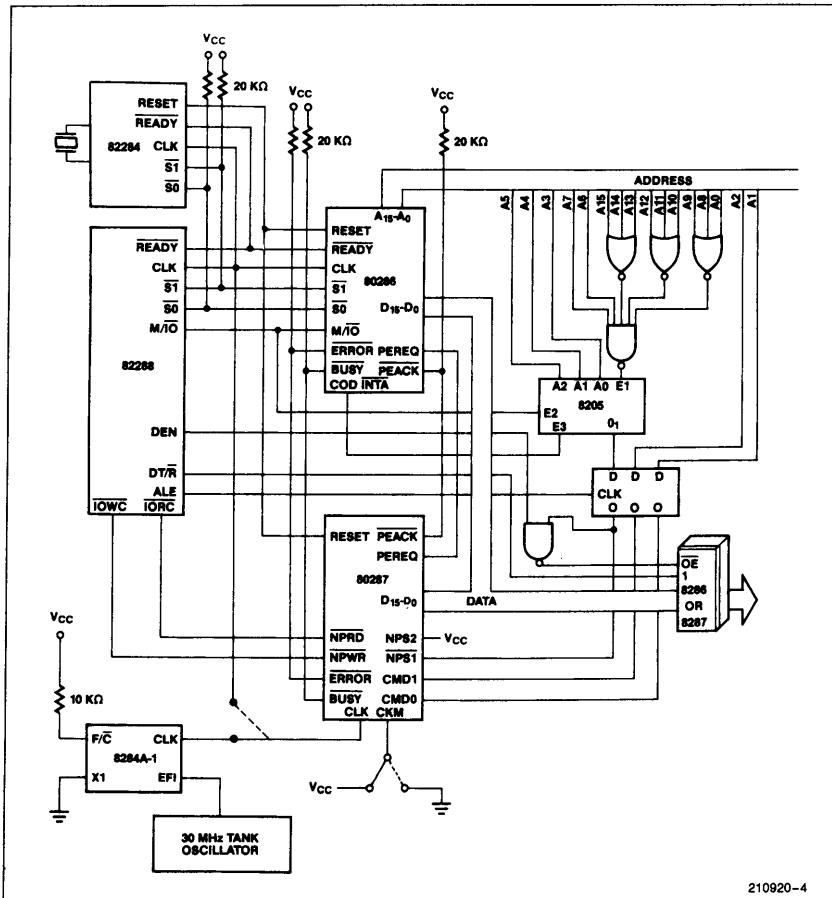


Figure 4A. 80286/80287 System Configuration

210920-4

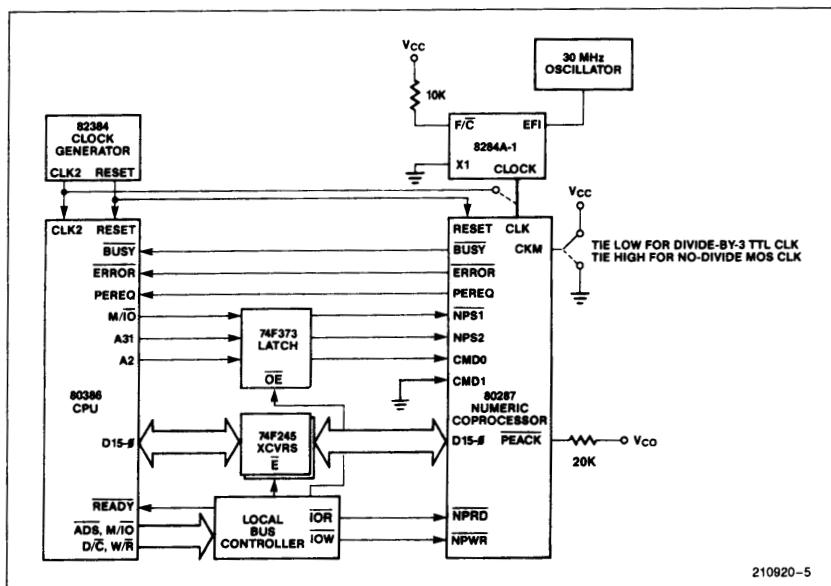


Figure 4B. 80386/80287 System Configuration

Table 2. 80287 Data Type Representation in Memory

Data Formats	Range	Precision	Most Significant Byte								HIGHEST ADDRESSED BYTE							
			7	0	7	0	7	0	7	0	7	0	7	0	7	0	7	0
Word Integer	10^4	16 Bits																
Short Integer	10^9	32 Bits																
Long Integer	10^{19}	64 Bits																
Packed BCD	10^{18}	18 Digits	S	X	d_{17}	d_{16}	d_{15}	d_{14}	d_{13}	d_{12}	d_{11}	d_{10}	d_9	d_8	d_7	d_6	d_5	d_4
Short Real	$10^{\pm 38}$	24 Bits	S	BIASED EXPONENT	SIGNIFICAND													
Long Real	$10^{\pm 308}$	53 Bits	S	BIASED EXPONENT	SIGNIFICAND													
Temporary Real	$10^{\pm 4932}$	64 Bits	S	BIASED EXPONENT	I	SIGNIFICAND												

NOTES:

1. S = Sign bit (0 = positive, 1 = negative)
2. d_n = Decimal digit (two per byte)
3. X = Bits have no significance; 8087 ignores when loading, zeros when storing.
4. ▲ = Position of implicit binary point
5. I = Integer bit of significant; stored in temporary real, implicit in short and long real.
6. Exponent Bias (normalized values):
 - Short Real: 127 (7FH)
 - Long Real: 1023 (3FFH)
 - Temporary Real: 16383 (3FFFH)
7. Packed BCD: $(-1)^s (D_{17} \dots D_0)$
8. Real: $(-1)^s (2^{E-BIAS}) (F_0 F_1 \dots)$

210920-6

Table 6 lists the 80287's instructions by class. No special programming tools are necessary to use the 80287 since all new instructions and data types are directly supported by the 80286 assembler and

appropriate high level languages. All 8086/8088 development tools which support the 8087 can also be used to develop software for the 80286/80287 in real address mode.

SOFTWARE INTERFACE

The 80286/80287 is programmed as a single processor. All communication between the 80286 and the 80287 is transparent to software. The CPU automatically controls the 80287 whenever a numeric instruction is executed. All memory addressing modes, physical memory, and virtual memory of the CPU are available for use by the NPX.

Since the NPX operates in parallel with the CPU, any errors detected by the NPX may be reported after the CPU has executed the ESCAPE instruction which caused it. To allow identification of the failing numeric instruction, the NPX contains two pointer registers which identify the address of the failing numeric instruction and the numeric memory operand if appropriate for the instruction encountering this error.

INTERRUPT DESCRIPTION

Several interrupts of the 80286 are used to report exceptional conditions while executing numeric programs in either real or protected mode. The interrupts and their functions are shown in Table 3.

PROCESSOR ARCHITECTURE

As shown in Figure 1, the NPX is internally divided into two processing elements, the bus interface unit (BIU) and the numeric execution unit (NEU). The NEU executes all numeric instructions, while the BIU receives and decodes instructions, requests operand transfers to and from memory and executes processor control instructions. The two units are able to operate independently of one another allowing the BIU to maintain asynchronous communication with the CPU while the NEU is busy processing a numeric instruction.

BUS INTERFACE UNIT

The BIU decodes the ESC instruction executed by the CPU. If the ESC code defines a math instruction, the BIU transmits the formatted instruction to the NEU. If the ESC code defines an administrative instruction, the BIU executes it independently of the NEU. The parallel operation of the NPX with the CPU is normally transparent to the user. The BIU generates the BUSY and ERROR signals for 80286/80287 processor synchronization and error notification, respectively.

The 80287 executes a single numeric instruction at a time. When executing most ESC instructions, the

Table 3. 80286 Interrupt Vectors Reserved for NPX

Interrupt Number	Interrupt Function
7	An ESC instruction was encountered when EM or TS of the 80286 MSW was set. EM = 1 indicates that software emulation of the instruction is required. When TS is set, either an ESC or WAIT instruction will cause interrupt 7. This indicates that the current NPX context may not belong to the current task.
9	The second or subsequent words of a numeric operand in memory exceeded a segment's limit. This interrupt occurs after executing an ESC instruction. The saved return address will not point at the numeric instruction causing this interrupt. After processing the addressing error, the 80286 program can be restarted at the return address with IRET. The address of the failing numeric instruction and numeric operand are saved in the 80287. An interrupt handler for this interrupt <i>must</i> execute FNINIT before any other ESC or WAIT instruction.
13	The starting address of a numeric operand is not in the segment's limit. The return address will point at the ESC instruction, including prefixes, causing this error. The 80287 has not executed this instruction. The instruction and data address is 80287 refer to a previous, correctly executed, instruction.
16	The previous numeric instruction caused an unmasked numeric error. The address of the faulty numeric instruction or numeric data operand is stored in the 80287. Only ESC or WAIT instructions can cause this interrupt. The 80286 return address will point at a WAIT or ESC instruction, including prefixes, which may be restarted after clearing the error condition in the NPX.

80286 tests the **BUSY** pin and waits until the 80287 indicates that it is not busy before initiating the command. Once initiated, the 80286 continues program execution while the 80287 executes the ESC instruction. In 8086/8087 systems, this synchronization is achieved by placing a WAIT instruction before an ESC instruction. For most ESC instructions, the 80287 does not require a WAIT instruction before the ESC opcode. However, the 80287 will operate correctly with these WAIT instruction. In all cases, a WAIT or ESC instruction should be inserted after any 80287 store to memory (except FSTSW and FSTCW) or load from memory (except FDENV or FRSTOR) before the 80286 reads or changes the value to be sure the numeric value has already been written or read by the NPX.

Data transfers between memory and the 80287, when needed, are controlled by the PEREQ PEACK, NPROD, NPWR, NPST, NPS2 signals. The 80286 does the actual data transfer with memory through its processor extension data channel. Numeric data transfers with memory performed by the 80286 use the same timing as any other bus cycle. Control signal for the 80287 are generated by the 8086 as

shown in Figure 4a, and meet the timing requirements shown in the AC requirements section.

NUMERIC EXECUTION UNIT

The NEU executes all instructions that involve the register stack; these include arithmetic, logical, transcendental, constant and data transfer instructions. The data path in the NEU is 84 bits wide (68 significant bits, 15 exponent bits and a sign bit) which allows internal operand transfers to be performed at very high speeds.

When the NEU begins executing an instruction, it activates the BIU **BUSY** signal. This signal is used in conjunction with the CPU WAIT instruction or automatically with most of the ESC instructions to synchronize both processors.

REGISTER SET

The 80287 register set is shown in Figure 5. Each of the eight data registers in the 8087's register stack

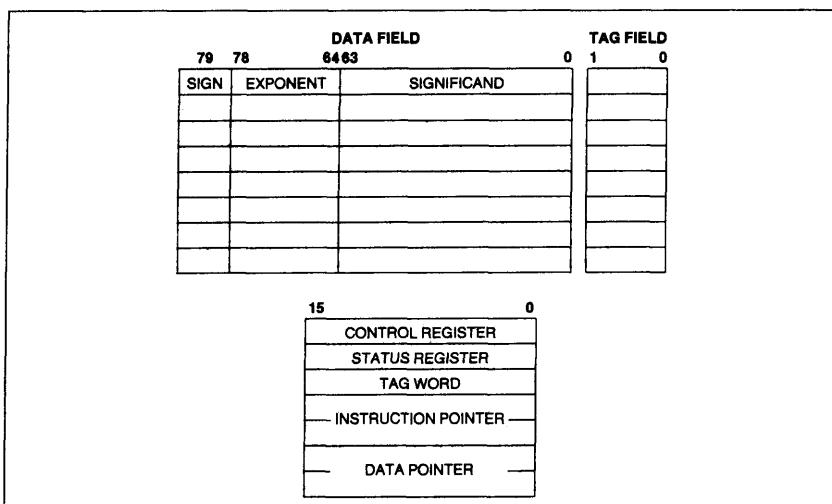


Figure 5. 80287 Register Set

is 80 bits wide and is divided into "fields" corresponding to the NPX's temporary real data type.

At a given point in time the TOP field in the status word identifies the current top-of-stack register. A "push" operation decrements TOP by 1 and loads a value into the new top register. A "pop" operation stores the value from the current top register and then increments TOP by 1. Like 80286 stacks in memory, the 80287 register stack grows "down" toward lower-addressed registers.

Instructions may address the data registers either implicitly or explicitly. Many instructions operate on the register at the TOP of the stack. These instructions implicitly address the register pointed by the TOP. Other instructions allow the programmer to explicitly specify the register which is to be used. This explicit register addressing is also "top-relative."

STATUS WORD

The 16-bit status word (in the status register) shown in Figure 6 reflects the overall state of the 80287. It may be read and inspected by CPU code. The busy bit (bit 15) indicates whether the NEU is executing an instruction ($B = 1$) or is idle ($B = 0$).

The instructions FSTSW, FSTSW AX, FSTENV, and FSAVE which store the status word are executed exclusively by the BIU and do not set the busy bit themselves or require the Busy bit be cleared in order to be executed.

The four numeric condition code bits ($C_0 - C_3$) are similar to the flags in a CPU: instructions that perform arithmetic operations update these bits to reflect the outcome of NPX operations. The effect of these instructions on the condition code is summarized in Tables 4a and 4b.

Bits 14–12 of the status word point to the 80287 register that is the current top-of-stack (TOP) as described above. Figure 6 shows the six error flags in bits 5–0 of the status word. Bits 5–0 are set to indicate that the NEU has detected an exception while executing an instruction. The section on exception handling explains how they are set and used.

Bit 7 is the error summary status bit. This bit is set if any unmasked exception bit is set and cleared otherwise. If this bit is set, the ERROR signal is asserted.

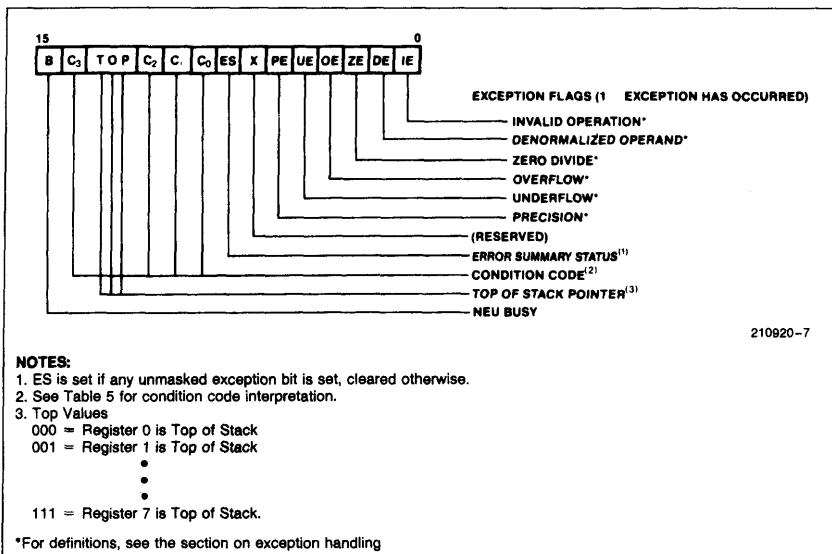


Figure 6. 80287 Status Word

TAG WORD

The tag word marks the content of each register as shown in Figure 7. The principal function of the tag word is to optimize the NPX's performance. The eight two-bit tags in the tag word can be used, however, to interpret the contents of 80287 registers.

32-bit virtual address used by the program which executed an ESC instruction. The same FLDENV/FSTENV/FSAVE/FRSTOR instructions as those of the 8087 are used to transfer these values between the 80287 registers and memory.

The saved instruction address in the 80287 will point at any prefixes which preceded the instruction. This is different than in the 8087 which only pointed at the ESCAPE instruction opcode.

INSTRUCTION AND DATA POINTERS

The instruction and data pointers (See Figures 8a and 8b) are provided for user-written error handlers. Whenever the 80287 executes a new instruction, the BIU saves the instruction address, the operand address (if present) and the instruction opcode. 80287 instructions can store this data into memory.

The instruction and data pointers appear in one of two formats depending on the operating mode of the 80287. In real mode, these values are the 20-bit physical address and 11-bit opcode formatted like the 8087. In protection mode, these values are the

CONTROL WORD

The NPX provides several processing options which are selected by loading a word from memory into the control word. Figure 9 shows the format and encoding of fields in the control word.

The low order byte of this control word configures the 80287 error and exception masking. Bits 5-0 of the control word contain individual masks for each of the six exceptions that the 80287 recognizes. The high order byte of the control word configures the

Table 4a. Condition Code Interpretation

Instruction Type	C ₃	C ₂	C ₁	C ₀	Interpretation
Compare, Test	0	0	X	0	ST > Source or 0 (FTST)
	0	0	X	1	ST < Source or 0 (FTST)
	1	0	X	0	ST = Source or 0 (FTST)
	1	1	X	1	ST is not comparable
Remainder	Q ₁	0	Q ₀	Q ₂	Complete reduction with three low bits of quotient (See Table 5b)
	U	1	U	U	Incomplete Reduction
Examine	0	0	0	0	Valid, positive unnormalized
	0	0	0	1	Invalid, positive, exponent = 0
	0	0	1	0	Valid, negative, unnormalized
	0	0	1	1	Invalid, negative, exponent = 0
	0	1	0	0	Valid, positive, normalized
	0	1	0	1	Infinity, positive
	0	1	1	0	Valid, negative, normalized
	0	1	1	1	Infinity, negative
	1	0	0	0	Zero, positive
	1	0	0	1	Empty
	1	0	1	0	Zero, Negative
	1	0	1	1	Empty
	1	1	0	0	Invalid, positive, exponent = 0
	1	1	0	1	Empty
	1	1	1	0	Invalid, negative, exponent = 0
	1	1	1	1	Empty

NOTES:

1. ST = Top of Stack
2. X = value is not affected by instruction
3. U = value is undefined following instruction
4. Q_n = Quotient bit n

Table 4b. Condition Code Interpretation after FPREM (See Note 1) Instruction as a Function of Dividend Value

Dividend Range	Q ₂	Q ₁	Q ₀
Dividend < 2 * Modulus	C ₃	C ₁	Q ₀
Dividend < 4 * Modulus	C ₃	Q ₁	Q ₀
Dividend ≥ 4 * Modulus	Q ₂	Q ₁	Q ₀

NOTE:

1. Previous value of indicated bit, not affected by FPREM instruction execution.

80287 operating mode including precision, rounding, and infinity control. The precision control bits (bits 9–8) can be used to set the 80287 internal operating precision at less than the default of temporary real (80-bit) precision. This can be useful in providing compatibility with the early generation arithmetic processors of smaller precision than the 80287. The rounding control bits (bits 11–10) provide for directed rounding and true chop as well as the unbiased round to nearest even mode specified in the IEEE standard. Control over closure of the number space at infinity is also provided (either affine closure: $\pm \infty$, or projective closure: ∞ , is treated as unsigned, may be specified).

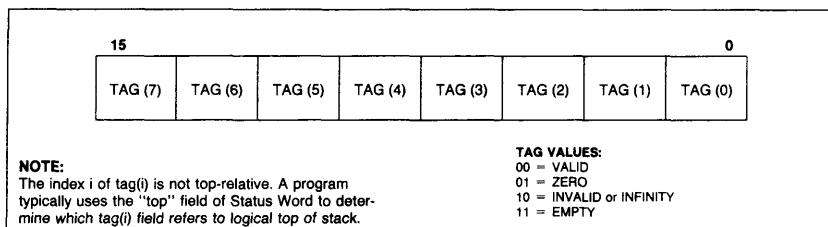


Figure 7. 80287 Tag Word

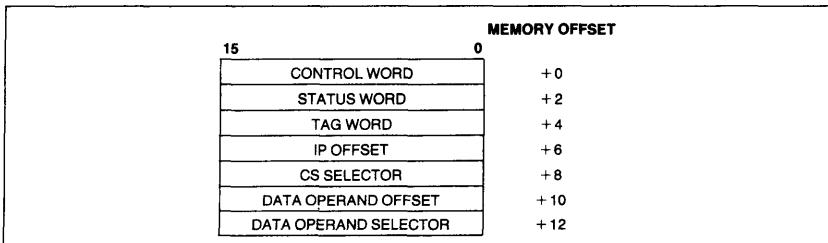


Figure 8a. Protected Mode 80287 Instruction and Data Pointer Image in Memory

EXCEPTION HANDLING

The 80287 detects six different exception conditions that can occur during instruction execution. Any or all exceptions will cause the assertion of external ERROR signal and ES bit of the Status Word if the appropriate exception masks are not set.

The exceptions that the 80287 detects and the 'default' procedures that will be carried out if the exception is masked, are as follows:

Invalid Operation: Stack overflow, stack underflow, indeterminate form (0/0, ∞ , $-\infty$, etc) or the use of a Non-Number (NAN) as an operand. An exponent value of all ones and non-zero significand is reserved to identify NaNs. If this exception is masked, the 80287 default response is to generate a specific

NAN called INDEFINITE, or to propagate already existing NaNs as the calculation result.

Overflow: The result is too large in magnitude to fit the specified format. The 80287 will generate an encoding for infinity if this exception is masked.

Zero Divisor: The divisor is zero while the dividend is a non-infinite, non-zero number. Again, the 80287 will generate an encoding for infinity if this exception is masked.

Underflow: The result is non-zero but too small in magnitude to fit in the specified format. If this exception is masked the 80287 will denormalize (shift right) the fraction until the exponent is in range. The process is called gradual underflow.

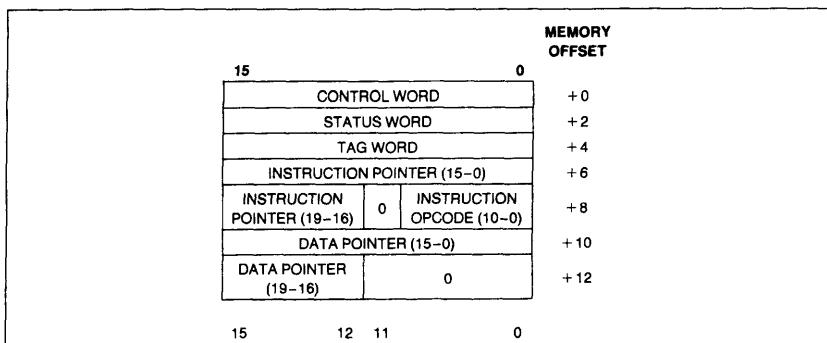


Figure 8b. Real Mode 80287 Instruction and Data Pointer Image in Memory

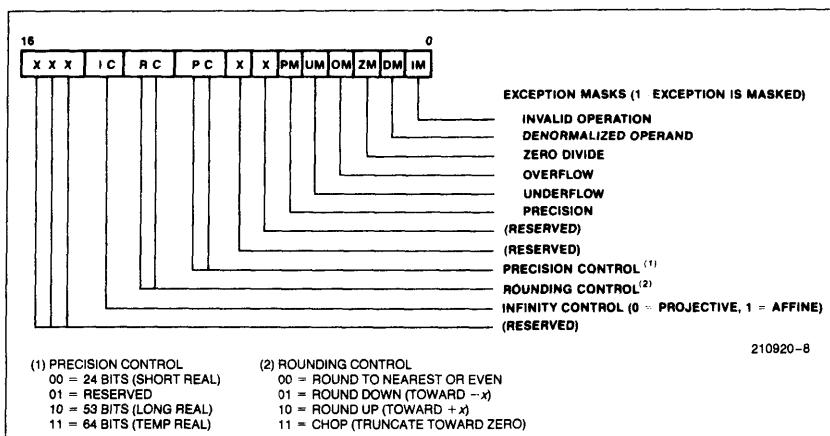


Figure 9. 80287 Control Word

Denormalized Operand: At least one of the operands is denormalized; it has the smallest exponent but a non-zero significand. Normal processing continues if this exception is masked off.

Inexact Result: The true result is not exactly representable in the specified format, the result is rounded according to the rounding mode, and this flag is set. If this exception is masked, processing will simply continue.

If the error is not masked, the corresponding error bit and the error status bit (ES) in the control word will be set, and the ERROR output signal will be asserted. If the CPU attempts to execute another ESC or WAIT instruction, exception 7 will occur.

The error condition must be resolved via an interrupt service routine. The 80287 saves the address of the floating point instruction causing the error as well as the address of the lowest memory location of any memory operand required by that instruction.

8086/8087 COMPATIBILITY:

The 80286/80287 supports portability of 8086/8087 programs when it is in the real address mode. However, because of differences in the numeric error handling techniques, error handling routines *may* need to be changed. The differences between an 80286/80287 and 8086/8087 are:

1. The NPX error signal does not pass through an interrupt controller (8087 INT signal does).

Therefore, any interrupt controller oriented instructions for the 8086/8087 may have to be deleted.

2. Interrupt vector 16 must point at the numeric error handler routine.
3. The saved floating point instruction address in the 80287 includes any leading prefixes before the ESCAPE opcode. The corresponding saved address of the 8087 does not include leading prefixes.
4. In protected mode, the format of the saved instruction and operand pointers is different than for the 8087. The instruction opcode is not saved—it must be read from memory if needed.
5. Interrupt 7 will occur when executing ESC instructions with either TS or EM or MSW = 1. If TS or MSW = 1 then WAIT will also cause interrupt 7. An interrupt handler should be added to handle this situation.
6. Interrupt 9 will occur if the second or subsequent words of a floating point operand fall outside a segment's size. Interrupt 13 will occur if the starting address of a numeric operand falls outside a segment's size. An interrupt handler should be added to report these programming errors.

In the protected mode, 8086/8087 application code can be directly ported via recompilation if the 80286 memory protection rules are not violated.

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias	0°C to 70°C
Storage Temperature	-65°C to +150°C
Case Temperature	0°C to 85°C
Voltage on any Pin with Respect to Ground	-1.0 to +7V
Power Dissipation	3.0 Watt

*Notice: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

D.C. CHARACTERISTICS $T_A = 0^\circ\text{C}$ to 70°C , $T_C = 0^\circ\text{C}$ to 85°C , $V_{CC} = 5\text{V} \pm 5\%$ **ALL SPEEDS SELECTIONS**

Symbol	Parameter	Min	Max	Unit	Test Conditions
V_{IL}	Input LOW Voltage	-0.5	0.8	V	
V_{IH}	Input HIGH Voltage	2.0	$V_{CC} + 0.5$	V	
V_{IHC}	Clock Input HIGH Voltage CKM = 1: CKM = 0:	2.0 3.8	$V_{CC} + 1$ $V_{CC} + 1$	V V	
V_{ILC}	Clock Input LOW Voltage CKM = 1 CKM = 0	-0.5 -0.5	0.8 0.6	V V	
V_{OL}	Output LOW Voltage		0.45	V	$I_{OL} = 3.0\text{ mA}$
V_{OH}	Output HIGH Voltage	2.4		V	$I_{OH} = -400\text{ }\mu\text{A}$
I_{LI}	Input Leakage Current	•	± 10	μA	$OV \leq V_{IN} \leq V_{CC}$
I_{LO}	Output Leakage Current	•	± 10	μA	$0.45V \leq V_{OUT} \leq V_{CC}$
I_{CC}	Power Supply Current		600 475 375	mA mA mA	$T_A = 0^\circ\text{C}$ $T_A = 25^\circ\text{C}$ $T_A = 70^\circ\text{C}$
C_{IN}	Input Capacitance	•	10	pF	$F_C = \text{MHz}$
C_O	Input/Output Capacitance (D0-D15)	•	20	pF	$V_C = 1\text{ MHz}$
C_{CLK}	CLK Capacitance	•	12	pF	$F_C = 1\text{ MHz}$

A.C. CHARACTERISTICS $T_A = 0^\circ\text{C}$ to 70°C , $T_{\text{CASE}} = 0^\circ\text{C}$ to 85°C , $V_{\text{CC}} = 5\text{V} \pm 5\%$ **TIMING REQUIREMENTS**

A.C. timings are referenced to 0.8V and 2.0V points on signals unless otherwise noted.

Symbol	Parameter	80287-3 5 MHz		80287-6 6 MHz		80287-8 8 MHz		80287-10 10 MHz Preliminary		Units	Test Conditions
		Min	Max	Min	Max	Min	Max	Min	Max		
T_{CLCL}	CLK Period CKM = 1: CKM = 0:	200 62.5	500 250	166 62.5	500 166	125 50	500 166	100 40	500 166	ns ns	
T_{CLCH}	CLK LOW Time CKM = 1: CKM = 0:	118 15	230	100 15	343 146	68 15	343 146	62 11	343 146	ns ns	At 0.8V At 0.6V
T_{CHCL}	CLK HIGH Time CKM = 1: CKM = 0:	69 20	235	50 20	230 151	43 20	230 151	28 18	230 151	ns ns	At 2.0V At 3.6V
T_{CH1CH2}	CLK Rise Time		10		10		10		10	ns	1.0V to 3.6V if CKM = 0
T_{CL2CL1}	CLK Fall Time		10		10		10		10	ns	3.6V to 1.0V if CKM = 0
T_{DYWH}	Data Setup to NPWR Inactive	75		75		75		75		ns	
T_{WHDX}	Data Hold from NPWR Inactive	30		30		18		18		ns	
T_{WLWH} T_{TRLH}	NPWR NPROD Active Time	95		95		90		90		ns	At 0.8V
T_{AVRL} T_{AVWL}	Command Valid to NPWR or NPRODActive	0		0		0		0		ns	
T_{MHRL}	Minimum Delay from PEREQ Active to NPROD Active	130		130		130		100		ns	
T_{KLKH}	PEAK Active Time	85		85		85		60		ns	At 0.8V
T_{KHKL}	PEAK Inactive Time	250		250		250		200		ns	At 2.0V
T_{KHCH}	PEAK Inactive to NPWR, NPROD Inactive	50		50		40		40		ns	
T_{CHKL}	NPWR, NPROD Inactive to PEAK Active	-30		-30		-30		-30		ns	
T_{WHAX} T_{RHAX}	Command Hold from NPWR, NPROD Inactive	30		30		30		22		ns	
T_{KLCL}	PEAK Active Setup to NPWR NPROD Active	50		50		40		40		ns	

A.C. CHARACTERISTICS $T_A = 0^\circ\text{C}$ to 70°C , $T_{\text{CASE}} = 0^\circ\text{C}$ to 85°C , $V_{\text{CC}} = 5\text{V} \pm 5\%$ (Continued)**TIMING REQUIREMENTS** (Continued)

A.C. timings are referenced to 0.8V and 2.0V points on signals unless otherwise noted.

Symbol	Parameter	80287-3 5 MHz		80287-6 6 MHz		80287-8 8 MHz		80287-10 10 MHz Preliminary		Units	Test Conditions
		Min	Max	Min	Max	Min	Max	Min	Max		
TIVCL	NPWR, NPROD to CLK Setup Time	70		70		70		53		ns	(Note 1)
TCLIH	NPWR, NPROD from CLK Hold Time	45		45		45		37		ns	(Note 1)
TRSCL	RESET to CLK Setup Time	20		20		20		20		ns	(Note 1)
TCLRS	RESET from CLK Hold Time	20		20		20		20		ns	(Note 1)

TIMING RESPONSES

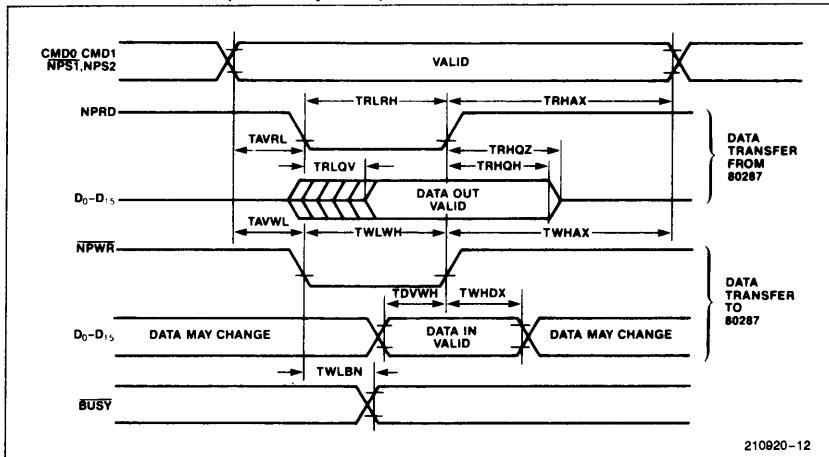
Symbol	Parameter	80287-3 5 MHz		80287-6 6 MHz		80287-8 8 MHz		80287-10 10 MHz Preliminary		Units	Test Conditions
		Min	Max	Min	Max	Min	Max	Min	Max		
TRHOZ	NPROD Inactive to Data Float		37.5		37.5		35		21	ns	(Note 2)
TRLOV	NPROD Active to Data Valid		60		60		60		60	ns	(Note 3)
TILBH	ERRR Active to BUSY Inactive	100		100		100		100		ns	(Note 4)
TWLBV	NPWR Active to BUSY Active		100		100		100		100	ns	(Note 5)
TKLML	PEAK Active to PEREQ Inactive		127		127		127		100	ns	(Note 6)
TCMDI	Command Inactive Time										
	Write-to-Write	95		95		95		75		ns	At 2.0V
	Read-to-Read	250		95		95		75		ns	At 2.0V
	Write-to-Read	105		95		95		75		ns	At 2.0V
	Read-to-Write	95		95		95		75		ns	At 2.0V
TRHOH	Data Hold from NPROD Inactive	5		3		3		3		ns	(Note 7)

NOTES:

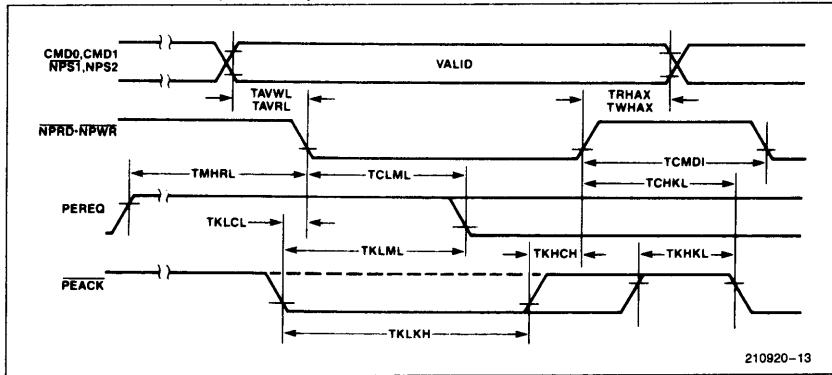
- This is an asynchronous input. This specification is given for testing purposes only, to assure recognition at a specific CLK edge.
- Float condition occurs when output current is less than I_{IO} on D0-D15.
- D0-D15 IoSINF: $XL = 100\text{ pF}$.
- BUSY loading: $CL = 100\text{ pF}$.
- BUSY loading: $CL = 100\text{ pF}$.
- On last data transfer on numeric instruction.
- D0-D15 loading: $CL = 100\text{ pF}$.

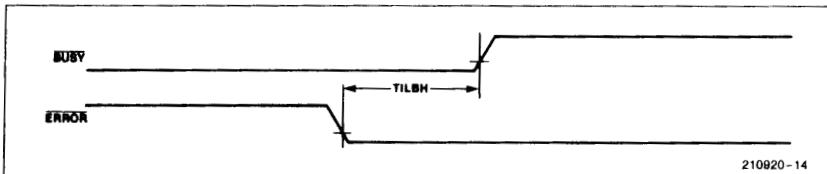
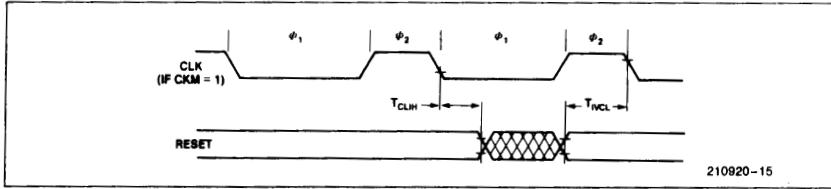
WAVEFORMS

DATA TRANSFER TIMING (Initiated by 80286)



DATA CHANNEL TIMING (Initiated by 80287)



WAVEFORMS (Continued)**ERROR OUTPUT TIMING****CLK, RESET TIMING (CKM = 1)****NOTE:**

Reset, NPWR, NRD are inputs asynchronous to CLK. Timing requirements on this page are given for testing purposes only, to assure recognition at a specific CLK edge.

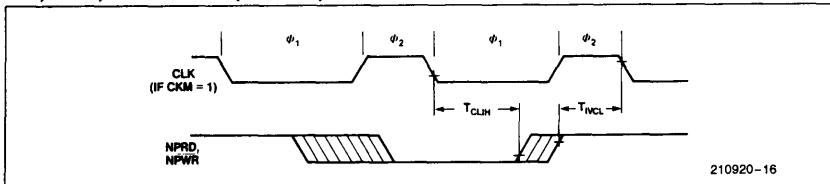
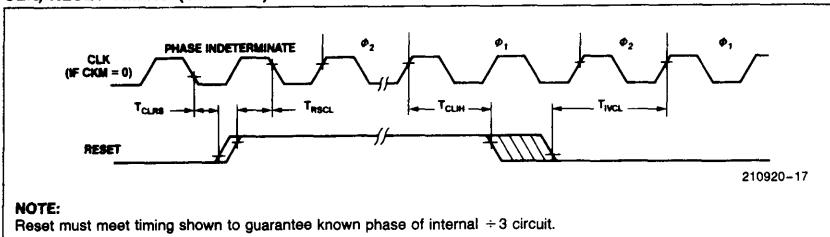
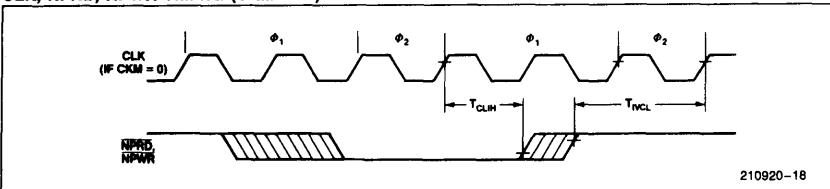
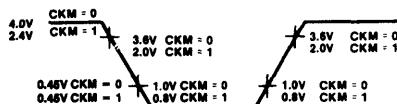
WAVEFORMS (Continued)**CLK, NPRD, NPWR TIMING (CKM = 1)****CLK, RESET TIMING (CKM = 0)****CLK, NPRD, NPWR TIMING (CKM = 0)**

Table 6. 80287 Extensions to the 80286 Instruction Set

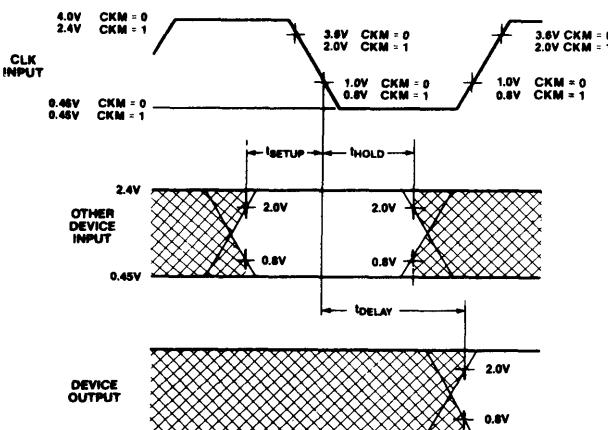
		Optional 8,16 Bit Displacement	Clock Count Range			
			32 Bit Real	32 Bit Integer	64 Bit Real	16 Bit Integer
Data Transfer						
FLD = LOAD	MF =		00	01	10	11
Integer/Real Memory to ST(0)	ESCAPE MF 1 MOD 0 0 0 R/M	DISP	38-56	52-60	40-60	46-54
Long Integer Memory to ST(0)	ESCAPE 1 1 1 MOD 1 0 1 R/M	DISP		60-68		
Temporary Real Memory to ST(0)	ESCAPE 0 1 1 MOD 1 0 1 R/M	DISP		53-65		
BCD Memory to ST(0)	ESCAPE 1 1 1 MOD 1 0 0 R/M	DISP		290-310		
ST(i) to ST(0)	ESCAPE 0 0 1 1 1 0 0 0 ST(i)			17-22		
FST = STORE						
ST(0) to Integer/Real Memory	ESCAPE MF 1 MOD 0 1 0 R/M	DISP	84-90	82-92	96-104	80-90
ST(0) to ST(i)	ESCAPE 1 0 1 1 1 0 1 0 ST(i)			15-22		
FSTP = STORE AND POP						
ST(0) to Integer/Real Memory	ESCAPE MF 1 MOD 0 1 1 R/M	DISP	86-92	84-94	98-106	82-92
ST(0) to Long Integer Memory	ESCAPE 1 1 1 MOD 1 1 1 R/M	DISP		94-105		
ST(0) to Temporary Real Memory	ESCAPE 0 1 1 MOD 1 1 1 R/M	DISP		52-58		
ST(0) to BCD Memory	ESCAPE 1 1 1 MOD 1 1 0 R/M	DISP		520-540		
ST(0) to ST(i)	ESCAPE 1 0 1 1 1 0 1 1 ST(i)			17-24		
FXCH = Exchange ST(i) and ST(0)	ESCAPE 0 0 1 1 1 0 0 1 ST(i)			10-15		
Comparison						
FCOM = Compare						
Integer/Real Memory to ST(0)	ESCAPE MF 0 MOD 0 1 0 R/M	DISP	60-70	78-91	65-75	72-86
ST(i) to ST(0)	ESCAPE 0 0 0 1 1 0 1 0 ST(i)			40-50		
FCOMP = Compare and Pop						
Integer/Real Memory to ST(0)	ESCAPE MF 0 MOD 0 1 1 R/M	DISP	63-73	80-93	67-77	74-88
ST(i) to ST(0)	ESCAPE 0 0 0 1 1 0 1 1 ST(i)			45-52		
FCOMPP = Compare ST(1) to ST(0) and Pop Twice	ESCAPE 1 1 0 1 1 0 1 1 0 0 1			45-55		
FTST = Test ST(0)	ESCAPE 0 0 1 1 1 1 0 0 1 0 0			38-48		
FXAM = Examine ST(0)	ESCAPE 0 0 1 1 1 1 0 0 1 0 1			12-23		
210920-19						

All mnemonics copyright Intel Corporation 1987.



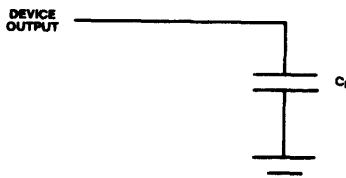
210920-9

AC Drive and Measurement Points—CLK Input



210920-10

AC Setup, Hold and Delay Time Measurement—General



210920-11

AC Test Loading on Outputs

Table 6. 80287 Extensions to the 80286 Instruction Set (Continued)

Constants			Optional 8,16 Bit Displacement	Clock Count Range			
	MF	-		32 Bit Real	32 Bit Integer	64 Bit Real	16 Bit Integer
FLDZ = LOAD + 0.0 into ST(0)	ESCAPE	0 0 1	1 1 1 0 1 1 1 0			00	01 10 11
FLD1 = LOAD + 1.0 into ST(0)	ESCAPE	0 0 1	1 1 1 0 1 0 0 0			11-17	
FLDP1 = LOAD π into ST(0)	ESCAPE	0 0 1	1 1 1 0 1 0 1 1			15-21	
FLDL2T = LOAD $\log_2 10$ into ST(0)	ESCAPE	0 0 1	1 1 1 0 1 0 0 1			16-22	
FLDL2E = LOAD $\log_2 e$ into ST(0)	ESCAPE	0 0 1	1 1 1 0 1 0 1 0			15-21	
FLDLG2 = LOAD $\log_{10} 2$ into ST(0)	ESCAPE	0 0 1	1 1 1 0 1 1 0 0			18-24	
FLDLN2 = LOAD $\log_2 e$ into ST(0)	ESCAPE	0 0 1	1 1 1 0 1 1 0 1			17-23	
Arithmetic							
FADD = Addition							
Integer/Real Memory with ST(0)	ESCAPE	MF 0	MOD 0 0 0 R/M	DISP		90-120	108-143 95-125 102-137
ST(i) and ST(0)	ESCAPE	d P 0	1 1 0 0 0 ST(i)			70-100	(Note 1)
FSUB = Subtraction							
Integer/Real Memory with ST(0)	ESCAPE	MF 0	MOD 1 0 R R/M	DISP		90-120	108-143 95-125 102-137
ST(i) and ST(0)	ESCAPE	d P 0	1 1 1 0 R R/M			70-100	(Note 1)
FMUL = Multiplication							
Integer/Real Memory with ST(0)	ESCAPE	MF 0	MOD 0 0 1 R/M	DISP		110-125	130-144 112-168 124-138
ST(i) and ST(0)	ESCAPE	d P 0	1 1 0 0 1 R/M			90-145	(Note 1)
FDIV = Division							
Integer/Real Memory with ST(0)	ESCAPE	MF 0	MOD 1 1 R R/M	DISP		215-225	230-243 220-230 224-238
ST(i) and ST(0)	ESCAPE	d P 0	1 1 1 1 R R/M			193-203	(Note 1)
FSQRT = Square Root of ST(0)	ESCAPE	0 0 1	1 1 1 1 1 0 1 0			180-186	
FSCALE = Scale ST(0) by ST(1)	ESCAPE	0 0 1	1 1 1 1 1 1 0 1			32-38	
FPREM = Partial Remainder of ST(0) + ST(1)	ESCAPE	0 0 1	1 1 1 1 1 1 0 0 0			15-190	
FRNDINT = Round ST(0) to Integer	ESCAPE	0 0 1	1 1 1 1 1 1 1 0 0			16-50	

210920-20

NOTE:

1. If P = 1 then add 5 clocks.

All mnemonics copyright Intel Corporation 1987.

Table 6. 80287 Extensions to the 80286 Instruction Set (Continued)

		Optional 8,16 Bit Displacement	Clock Count Range				
FXTRACT = Extract Components of ST(0)	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>0 0 1</td><td>1 1 1 1 0 1 0 0</td></tr></table>	ESCAPE	0 0 1	1 1 1 1 0 1 0 0		27-55	
ESCAPE	0 0 1	1 1 1 1 0 1 0 0					
FABS = Absolute Value of ST(0)	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>0 0 1</td><td>1 1 1 0 0 0 0 1</td></tr></table>	ESCAPE	0 0 1	1 1 1 0 0 0 0 1		10-17	
ESCAPE	0 0 1	1 1 1 0 0 0 0 1					
FCHS = Change Sign of ST(0)	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>0 0 1</td><td>1 1 1 0 0 0 0 0</td></tr></table>	ESCAPE	0 0 1	1 1 1 0 0 0 0 0		10-17	
ESCAPE	0 0 1	1 1 1 0 0 0 0 0					
Transcendental							
FPTAN = Partial Tangent of ST(0)	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>0 0 1</td><td>1 1 1 1 0 0 1 0</td></tr></table>	ESCAPE	0 0 1	1 1 1 1 0 0 1 0		30-540	
ESCAPE	0 0 1	1 1 1 1 0 0 1 0					
FPATAN = Partial Arc tangent of ST(0) - ST(1)	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>0 0 1</td><td>1 1 1 1 0 0 1 1</td></tr></table>	ESCAPE	0 0 1	1 1 1 1 0 0 1 1		250-800	
ESCAPE	0 0 1	1 1 1 1 0 0 1 1					
F2XM1 = $2^{ST(0)} - 1$	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>0 0 1</td><td>1 1 1 1 0 0 0 0</td></tr></table>	ESCAPE	0 0 1	1 1 1 1 0 0 0 0		310-630	
ESCAPE	0 0 1	1 1 1 1 0 0 0 0					
FYL2X = ST(1) • Log2 ST(0) 	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>0 0 1</td><td>1 1 1 1 0 0 0 1</td></tr></table>	ESCAPE	0 0 1	1 1 1 1 0 0 0 1		900-1100	
ESCAPE	0 0 1	1 1 1 1 0 0 0 1					
FYL2XP1 = ST(1) • Log2 ST(0) + 1 	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>0 0 1</td><td>1 1 1 1 1 0 0 1</td></tr></table>	ESCAPE	0 0 1	1 1 1 1 1 0 0 1		700-1000	
ESCAPE	0 0 1	1 1 1 1 1 0 0 1					
Processor Control							
FINIT = Initialize NPX	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>0 1 1</td><td>1 1 1 0 0 0 1 1</td></tr></table>	ESCAPE	0 1 1	1 1 1 0 0 0 1 1		2-8	
ESCAPE	0 1 1	1 1 1 0 0 0 1 1					
FSETPM = Enter Protected Mode	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>0 1 1</td><td>1 1 1 0 0 1 0 0</td></tr></table>	ESCAPE	0 1 1	1 1 1 0 0 1 0 0		2-8	
ESCAPE	0 1 1	1 1 1 0 0 1 0 0					
FSTSW AX = Store Control Word	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>1 1 1</td><td>1 1 1 0 0 0 0 0</td></tr></table>	ESCAPE	1 1 1	1 1 1 0 0 0 0 0		10-16	
ESCAPE	1 1 1	1 1 1 0 0 0 0 0					
FLDCW = Load Control Word	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>0 0 1</td><td>MOD 1 0 1 R/M</td><td style="border-left: none;">DISP</td></tr></table>	ESCAPE	0 0 1	MOD 1 0 1 R/M	DISP		7-14
ESCAPE	0 0 1	MOD 1 0 1 R/M	DISP				
FSTCW = Store Control Word	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>0 0 1</td><td>MOD 1 1 1 R/M</td><td style="border-left: none;">DISP</td></tr></table>	ESCAPE	0 0 1	MOD 1 1 1 R/M	DISP		12-18
ESCAPE	0 0 1	MOD 1 1 1 R/M	DISP				
FSTSW = Store Status Word	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>1 0 1</td><td>MOD 1 1 1 R/M</td><td style="border-left: none;">DISP</td></tr></table>	ESCAPE	1 0 1	MOD 1 1 1 R/M	DISP		12-18
ESCAPE	1 0 1	MOD 1 1 1 R/M	DISP				
FCLEX = Clear Exceptions	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>0 1 1</td><td>1 1 1 0 0 0 1 0</td></tr></table>	ESCAPE	0 1 1	1 1 1 0 0 0 1 0		2-8	
ESCAPE	0 1 1	1 1 1 0 0 0 1 0					
FSTENV = Store Environment	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>0 0 1</td><td>MOD 1 1 0 R/M</td><td style="border-left: none;">DISP</td></tr></table>	ESCAPE	0 0 1	MOD 1 1 0 R/M	DISP		40-50
ESCAPE	0 0 1	MOD 1 1 0 R/M	DISP				
FLDENV = Load Environment	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>0 0 1</td><td>MOD 1 0 0 R/M</td><td style="border-left: none;">DISP</td></tr></table>	ESCAPE	0 0 1	MOD 1 0 0 R/M	DISP		35-45
ESCAPE	0 0 1	MOD 1 0 0 R/M	DISP				
FSAVE = Save State	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>1 0 1</td><td>MOD 1 1 0 R/M</td><td style="border-left: none;">DISP</td></tr></table>	ESCAPE	1 0 1	MOD 1 1 0 R/M	DISP		205-215
ESCAPE	1 0 1	MOD 1 1 0 R/M	DISP				
FRSTOR = Restore State	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>1 0 1</td><td>MOD 1 0 0 R/M</td><td style="border-left: none;">DISP</td></tr></table>	ESCAPE	1 0 1	MOD 1 0 0 R/M	DISP		205-215
ESCAPE	1 0 1	MOD 1 0 0 R/M	DISP				
FINCSTP = Increment Stack Pointer	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>0 0 1</td><td>1 1 1 1 0 1 1 1</td></tr></table>	ESCAPE	0 0 1	1 1 1 1 0 1 1 1		6-12	
ESCAPE	0 0 1	1 1 1 1 0 1 1 1					
FDECSTP = Decrement Stack Pointer	<table border="1" style="display: inline-table;"><tr><td>ESCAPE</td><td>0 0 1</td><td>1 1 1 1 0 1 1 0</td></tr></table>	ESCAPE	0 0 1	1 1 1 1 0 1 1 0		6-12	
ESCAPE	0 0 1	1 1 1 1 0 1 1 0					

210920-21

All mnemonics copyright Intel Corporation 1987.

Table 6. 80287 Extensions to the 80286 Instruction Set (Continued)

		Clock Count Range											
FFREE = Free ST(i)	[<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>ESCAPE</td><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>ST(i)</td></tr></table>	ESCAPE	1	0	1	1	1	0	0	0	ST(i)	9-16	
ESCAPE	1	0	1	1	1	0	0	0	ST(i)				
FNOP = No Operation	[<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>ESCAPE</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td></tr></table>	ESCAPE	0	0	1	1	1	0	1	0	0	0	10-16 210920-22
ESCAPE	0	0	1	1	1	0	1	0	0	0			

NOTES:

1. if mod = 00 then DISP = 0*, disp-low and disp-high are absent
 if mod = 01 then DISP = disp-low sign-extended to 16-bits, disp-high is absent
 if mod = 10 then DISP = disp-high; disp-low
 if mod = 11 then r/m is treated as an ST(i) field
2. if r/m = 000 then EA = (BX) + (SI) + DISP
 if r/m = 001 then EA = (BX) + (DI) + DISP
 if r/m = 010 then EA = (BP) + (SI) + DISP
 if r/m = 011 then EA = (BP) + (DI) + DISP
 if r/m = 100 then EA = (SI) + DISP
 if r/m = 101 then EA = (DI) + DISP
 if r/m = 110 then EA = (BP) + DISP
 if r/m = 111 then EA = (BX) + DISP
 *except if mod = 000 and r/m = 110 then EA = disp-high; disp-low.
3. MF = Memory Format
 - 00—32-bit Real
 - 01—32-bit Integer
 - 10—64-bit Real
 - 11—16-bit Integer
4. ST(0) = Current stack top
 ST(i) = ith register below stack top
5. d = Destination
 - 0—Destination is ST(0)
 - 1—Destination is ST(i)
6. P = Pop
 - 0—No pop
 - 1—Pop ST(0)
7. R = Reverse: When d = 1 reverse the sense of R
 - 0—Destination (op) Source
 - 1—Source (op) Destination
8. For FSQRT: $-0 \leq ST(0) \leq +\infty$
 For FSCALE: $-2^{15} \leq ST(1) < +2^{15}$ and ST(1) integer
 For F2XM1: $0 \leq ST(0) \leq 2^{-1}$
 For FYL2X: $0 < ST(0) < \infty$
 $-\infty < ST(1) < +\infty$
 For FYL2XP1: $0 \leq |ST(0)| < (2 - \sqrt{2})/2$
 $-\infty < ST(1) < \infty$
 For FPATAN: $0 \leq ST(0) \leq \pi/4$
 For FPATAN: $0 \leq ST(0) < ST(1) < +\infty$
9. ESCAPE bit pattern is 11011.

All mnemonics copyright Intel Corporation 1987.

NS16450/INS8250A/NS16C450/INS82C50A Asynchronous Communications Element

General Description

The NS16450 is an improved specification version of the INS8250A Asynchronous Communications Element (ACE). The improved specifications ensure compatibility with the NS32016 and other state-of-the-art CPUs. Functionally, the NS16450 is equivalent to the INS8250A. The INS8250A is available in both $5V \pm 5\%$ and $5V \pm 10\%$ operating ranges. See ordering instructions on the last page. The ACE is fabricated using National Semiconductor's advanced scaled N-channel silicon-gate MOS process, XMOS.

The NS16C450 and INS82C50A are functionally equivalent to their XMOS counterparts, except that they are CMOS parts. (The CMOS parts will be available after June 1985.) It functions as a serial data input/output interface in a microcomputer system. The functional configuration of the ACE is programmed by the system software via a TRI-STATE® 8-bit bidirectional data bus; this includes the on-board baud rate generator.

The ACE performs serial-to-parallel conversion on data characters received from a peripheral device or a MODEM, and parallel-to-serial conversion on data characters received from the CPU. The CPU can read the complete status of the ACE at any time during the functional operation. Status information reported includes the type and condition of the transfer operations being performed by the ACE, as well as any error conditions (parity, overrun, framing, or break interrupt).

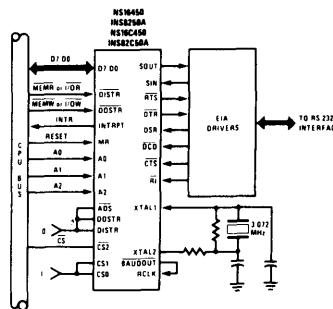
The ACE includes a programmable baud generator that is capable of dividing the timing reference clock input by divisors of 1 to $(2^{16} - 1)$, and producing a $16 \times$ clock for driving the internal transmitter logic. Provisions are also included to use this $16 \times$ clock to drive the receiver logic. Also included in the ACE is a complete MODEM-control capability, and a processor-interrupt system that may be software tailored to the user's requirements to minimize the computing required to handle the communications link.

Features

- Easily interfaces to most popular microprocessors.
- Adds or deletes standard asynchronous communication bits (start, stop, and parity) to or from serial data stream.
- Full double buffering eliminates need for precise synchronization.
- Independently controlled transmit, receive, line status, and data set interrupts.
- Programmable baud generator allows division of any input clock by 1 to $(2^{16} - 1)$ and generates the internal $16 \times$ clock.
- Independent receiver clock input.
- MODEM control functions (CTS, RTS, DSR, DTR, RI, and DCD).
- Fully programmable serial-interface characteristics:
 - 5-, 6-, 7-, or 8-bit characters
 - Even, odd, or no-parity bit generation and detection
 - 1-, $1\frac{1}{2}$ -, or 2-stop bit generation
 - Baud generation (DC to 56k baud).
- False start bit detection.
- Complete status reporting capabilities.
- TRI-STATE TTL drive capabilities for bidirectional data bus and control bus.
- Line break generation and detection.
- Internal diagnostic capabilities:
 - Loopback controls for communications link fault isolation
 - Break, parity, overrun, framing error simulation.
- Full prioritized interrupt system controls.

TRI-STATE® is a registered trademark of National Semiconductor Corp.

Basic Configuration


TL/C/8401-1

NS16450/INS8250A/NS16C450/INS82C50A Asynchronous Communications Element

Absolute Maximum Ratings

Temperature Under Bias	0°C to + 70°C
Storage Temperature	-65°C to + 150°C
All Input or Output Voltages with Respect to V _{SS}	- 0.5V to + 7.0V
Power Dissipation	700 mW

Note: Maximum ratings indicate limits beyond which permanent damage may occur. Continuous operation at these limits is not intended and should be limited to those conditions specified under DC electrical characteristics.

DC Electrical Characteristics

T_A = 0°C to + 70°C, V_{CC} = + 5V ± 5%, V_{SS} = 0V, unless otherwise specified.

Symbol	Parameter	Conditions	NS16450 NS16C450 (Note 1)		INS8250A,AWN INS82C50A (Note 1)		Units
			Min	Max	Min	Max	
V _{ILX}	Clock Input Low Voltage		- 0.5	0.8	- 0.5	0.8	V
V _{IHX}	Clock Input High Voltage		2.0	V _{CC}	2.0	V _{CC}	V
V _{IL}	Input Low Voltage		- 0.5	0.8	- 0.5	0.8	V
V _{IH}	Input High Voltage		2.0	V _{CC}	2.0	V _{CC}	V
V _{OL}	Output Low Voltage	I _{OL} = 1.6 mA on all *		0.4		0.4	V
V _{OH}	Output High Voltage	I _{OH} = ~ 1.0 mA *	2.4		2.4		V
I _{CC(AV)}	Avg. Power Supply Current (V _{CC})	V _{CC} = 5.25V, T _A = 25°C No Loads on output SIN, DSR, RLSD, CTS, RI = 2.0V All other inputs = 0.8V		120		95	mA
I _{CC(AV)}	Avg. Power Supply Current (V _{CC}) CMOS Parts Only	V _{CC} = 5.25V, T _A = 25°C No Loads on output SIN, DSR, RLSD, CTS, RI = 2.0V All other inputs = 0.8V Baud Rate Generator is 4 MHz Baud Rate is 56k		10		10	mA
I _{IL}	Input Leakage	V _{CC} = 5.25V, V _{SS} = 0V All other pins floating. V _{IN} = 0V, 5.25V		± 10		± 10	µA
I _{CL}	Clock Leakage			± 10		± 10	µA
I _{OZ}	TRI-STATE Leakage	V _{CC} = 5.25V, V _{SS} = 0V V _{OUT} = 0V, 5.25V 1) Chip deselected 2) WRITE mode, chip selected		± 20		± 20	µA
V _{ILMR}	MR Schmitt V _{IL}			0.8		0.8	V
V _{IHMR}	MR Schmitt V _{IH}		2.0		2.0		V

* Does not apply to XTAL²

Capacitance T_A = 25°C, V_{CC} = V_{SS} = 0V

Symbol	Parameter	Conditions	Min	Typ	Max	Units
C _{XTAL₂}	Clock Input Capacitance	f _c = 1 MHz		15	20	pF
C _{XTAL₁}	Clock Output Capacitance			20	30	pF
C _{IN}	Input Capacitance	Unmeasured pins returned to V _{SS}		6	10	pF
C _{OUT}	Output Capacitance			10	20	pF

Note 1: All specifications for CMOS parts are preliminary.

AC Electrical Characteristics $T_A = 0^\circ\text{C}$ to $+70^\circ\text{C}$, $V_{CC} = +5V \pm 5\%$

Symbol	Parameter	Conditions	NS16450		INS8250A, AWN INS82C50A (Note 1)		Units
			Min	Max	Min	Max	
t_{AW}	Address Strobe Width		60		90		ns
t_{AS}	Address Setup Time		60		90		ns
t_{AH}	Address Hold Time		0		0		ns
t_{CS}	Chip Select Setup Time		60		90		ns
t_{CH}	Chip Select Hold Time		0		0		ns
t_{DIW}	DISTR/DISTR Strobe Width		125		175		ns
t_{RC}	Ready Cycle Delay		175		500		ns
RC	$t_{AR} + t_{DIW} + t_{RC}$		360		755		ns
t_{DD}	DISTR/DISTR to Driver Disable Delay	@100 pF loading***		60		75	ns
t_{DDD}	Delay from DISTR/DISTR to Data	@100 pF loading		125		175	ns
t_{HZ}	DISTR/DISTR to Floating Data Delay	@100 pF loading***	0	100	100		ns
t_{DOW}	DOSTR/DOSTR Strobe Width		100		175		ns
t_{WC}	Write Cycle Delay		200		500		ns
WC	$t_{AW} + t_{DOW} + t_{WC}$		360		755		ns
t_{DS}	Data Setup Time		40		90		ns
t_{DH}	Data Hold Time		40		60		ns
t_{CSC^*}	Chip Select Output Delay from Select	@100 pF loading		100		125	ns
t_{RA^*}	Address Hold Time from DISTR/DISTR		20		20		ns
t_{RCS^*}	Chip Select Hold Time from DISTR/DISTR		20		20		ns
t_{AR^*}	DISTR/DISTR Delay from Address		60		80		ns
t_{CSR^*}	DISTR/DISTR Delay from Chip Select		50		80		ns
t_{WA^*}	Address Hold Time from DOSTR/DOSTR		20		20		ns
t_{WCS^*}	Chip Select Hold Time from DOSTR/DOSTR		20		20		ns
t_{AW^*}	DOSTR/DOSTR Delay from Address		60		80		ns
t_{CSW^*}	DOSTR/DOSTR Delay from Select		50		80		ns
t_{MRW}	Master Reset Pulse Width		5		10		μs
t_{XH}	Duration of Clock High Pulse	External Clock (3.1 MHz Max.)	140		140		ns
t_{XL}	Duration of Clock Low Pulse	External Clock (3.1 MHz Max.)	140		140		ns

Baud Generator

N	Baud Divisor		1	$2^{16}-1$	1	$2^{16}-1$	
t_{BLD}	Baud Output Negative Edge Delay	100 pF Load		125		250	ns
t_{BHD}	Baud Output Positive Edge Delay	100 pF Load		125		250	ns
t_{LW}	Baud Output Down Time	$f_X = 2 \text{ MHz}, \div 2, 100 \text{ pF Load}$	425		425		ns
t_{HW}	Baud Output Up Time	$f_X = 3 \text{ MHz}, \div 3, 100 \text{ pF Load}$	330		330		ns

Receiver

t_{SCD}	Delay from RCLK to Sample Time			2		2	μs
t_{SINT}	Delay from Stop to Set Interrupt		1	1	1	1	RCLK** Cycles
t_{RINT}	Delay from DISTR/DISTR (RD RBR/RDLSR) to Reset Interrupt	100 pF Load		1		1	μs

*Applicable only when A0S is tied low

**RCLK is equal to t_{LW} and t_{HW}

***Charge and discharge time is determined by V_{OL} , V_{OH} and the external loading.

Note 1: All specifications for CMOS parts are preliminary.

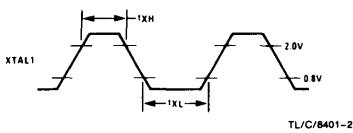
AC Electrical Characteristics (Continued)

AC Electrical Characteristics (Continued)							
Symbol	Parameter	Conditions	NS16450 NS16C450 (Note 1)		INS8250A, AWN INS82C50A (Note 1)		Units
			Min	Max	Min	Max	
Transmitter							
t _{HR}	Delay from DOSTR/DOSTR (WR THR) to Reset Interrupt	100 pF Load		175		1000	ns
t _{IRS}	Delay from Initial INTR Reset to Transmit Start		8	24	8	24	RCLK Cycles
t _{SI}	Delay from Initial Write to interrupt		16	32	16	32	RCLK Cycles
t _{STI}	Delay from Stop to Interrupt (THRE)		8	8	8	8	RCLK Cycles
t _{IR}	Delay from DISTR/DISTR (RD IIR) to Reset Interrupt (THRE)	100 pF Load		250		1000	ns
Modem Control							
t _{MDO}	Delay from DOSTR/DOSTR (WR MCR) to Output	100 pF Load		200		1000	ns
t _{SIM}	Delay to Set Interrupt from MODEM Input	100 pF Load				1000	ns
t _{RIM}	Delay to Reset Interrupt from DISTR/DISTR (RD MSR)	100 pF Load		250		1000	ns

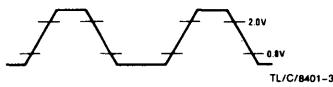
Note 1: All specifications for CMOS parts are preliminary.

Timing Waveforms (All timings are referenced to valid 0 and valid 1)

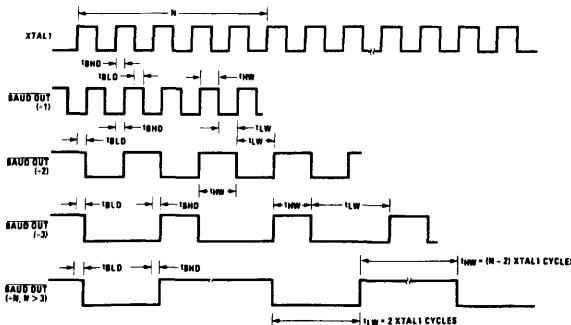
External Clock Input (3.1 MHz Max.)



AC Test Points

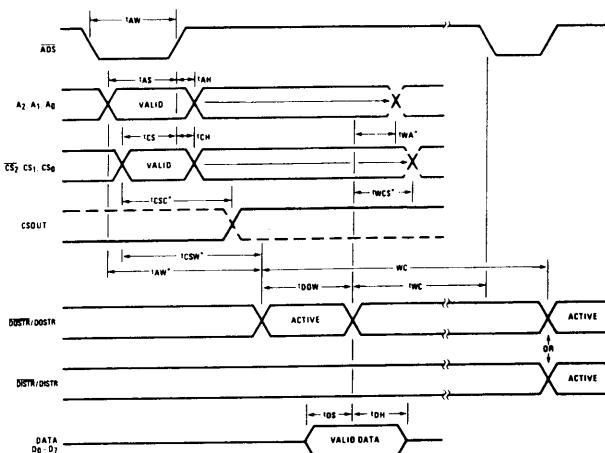


BAUDOUT Timing



Timing Waveforms (Continued)

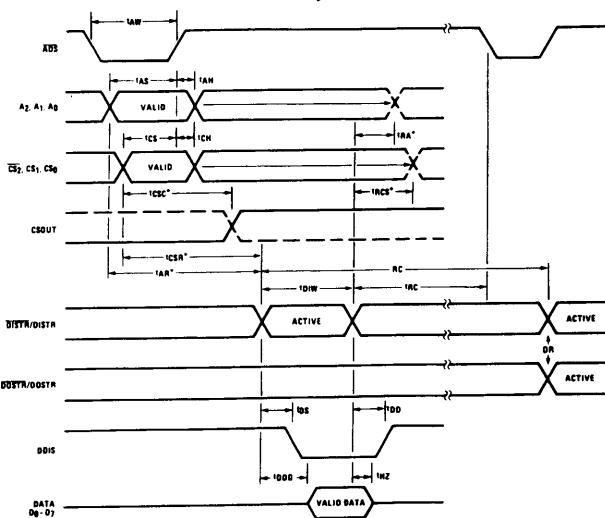
Write Cycle



TL/C/8401-5

*Applicable Only When \bar{ADS} is Tied Low.

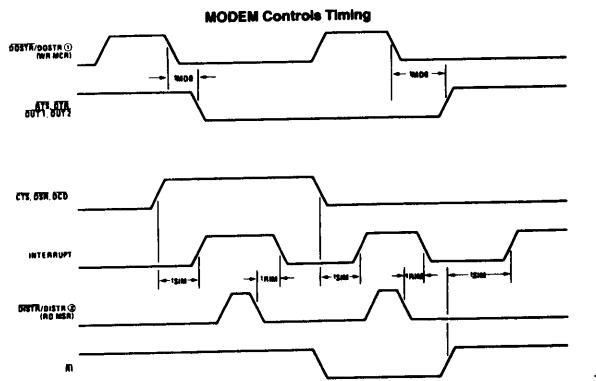
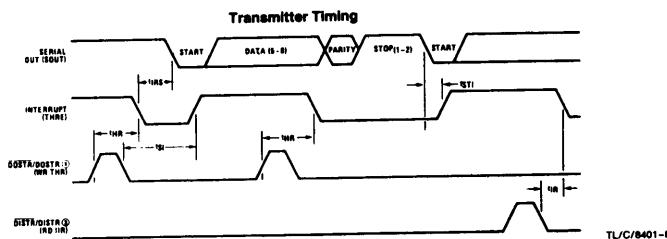
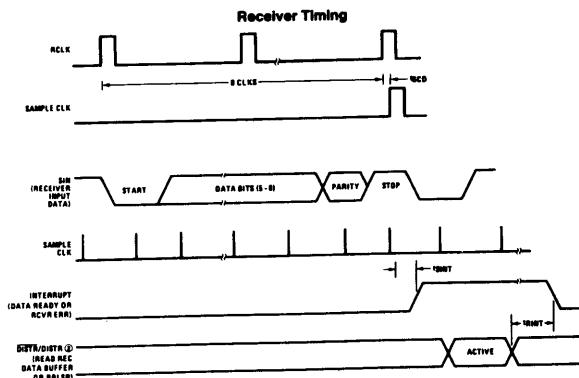
Read Cycle



TL/C/8401-6

*Applicable Only When \bar{ADS} is Tied Low.

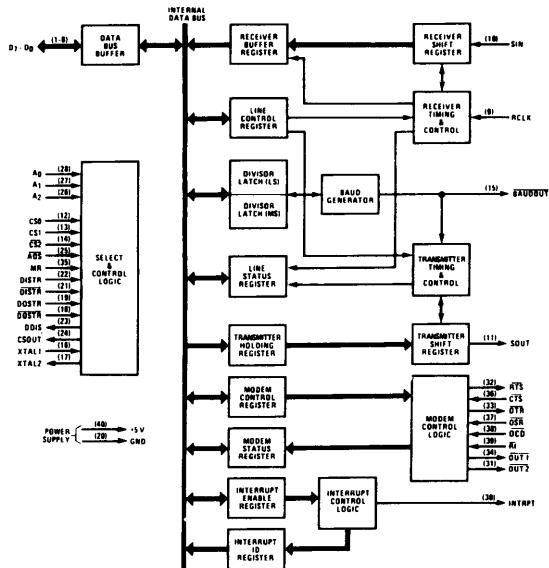
Timing Waveforms (Continued)



Note 1: See Write Cycle Timing

Note 2: See Read Cycle Timing

Block Diagram



TL/C/8401-10

Note: Applicable pinout numbers are included within parenthesis.

Functional Pin Description

The following describes the function of all NS16450, NS16C450 and INS8250A, INS82C50A input and output pins. Some of these descriptions reference internal circuits. Note: In the following descriptions, a low represents a logic 0 (0V nominal) and a high represents a logic 1 (+2.4V nominal).

INPUT SIGNALS

Chip Select (CS0, CS1, CS2), Pins 12–14: When CS0 and CS1 are high and CS2 is low, the chip is selected. Chip selection is complete when the decoded chip select signal is latched with an active (low) Address Strobe (ADS) input. This enables communication between the ACE and the CPU.

Data Input Strobe (DISTR, DISTR), Pins 22 and 21: When DISTR is high or DISTR is low while the chip is selected, it allows the CPU to read status information or data from a selected register of the ACE.

Note: Only an active DISTR or DISTR input is required to transfer data from the ACE during a read operation. Therefore, tie either the DISTR input permanently low or the DISTR input permanently high, if not used.

Data Output Strobe (DOSTR, DOSTR), Pins 19 and 18: When DOSTR is high or DOSTR is low while the chip is selected, allows the CPU to write data or control words into a selected register of the ACE.

Note: Only an active DOSTR or DOSTR input is required to transfer data to the ACE during a write operation. Therefore, tie either the DOSTR input permanently low or the DOSTR input permanently high, if not used.

Address Strobe (ADS), Pin 25: When low, provides latching for the Register Select (A0, A1, A2) and Chip Select (CS0, CS1, CS2) signals.

Note: An active ADS input is required when the Register Select (A0, A1, A2) signals are not stable for the duration of a read or write operation. If not required, tie the ADS input permanently low.

Register Select (A0, A1, A2), Pins 26–28: These three inputs are used during a read or write operation to select an ACE register to read from or write into as indicated in the table below. Note that the state of the Divisor Latch Access Bit (DLAB), which is the most significant bit of the Line Control Register, affects the selection of certain INS8250A registers. The DLAB must be set high by the system software to access the Baud Generator Divisor Latches.

Functional Pin Description (Continued)

DLAB	A ₂	A ₁	A ₀	Register
0	0	0	0	Receiver Buffer (read), Transmitter Holding Register (write)
0	0	0	1	Interrupt Enable
X	0	1	0	Interrupt Identification (read only)
X	0	1	1	Line Control
X	1	0	0	MODEM Control
X	1	0	1	Line Status
X	1	1	0	MODEM Status
X	1	1	1	Scratch
1	0	0	0	Divisor Latch (least significant byte)
1	0	0	1	Divisor Latch (most significant byte)

Master Reset (MR), Pin 35: This input is buffered with a TTL-compatible Schmitt Trigger with 0.5V typical hysteresis. When high, it clears all the registers (except the Receiver Buffer, Transmitter Holding, and Divisor Latches), and the control logic of the ACE. Also, the state of various output signals (SOUT, INTRPT, OUT 1, OUT 2, RTS, DTR) are affected by an active MR input. (Refer to Table I.)

Receiver Clock (RCLK), Pin 9: This input is the 16 × baud rate clock for the receiver section of the chip.

Serial Input (SIN), Pin 10: Serial data input from the communications link (peripheral device, MODEM, or data set).

Clear to Send (CTS), Pin 36: The CTS signal is a MODEM control function input whose conditions can be tested by the CPU by reading bit 4 (CTS) of the MODEM Status Register. Bit 0 (DCTS) of the MODEM Status Register indicates whether the CTS input has changed state since the previous reading of the MODEM Status Register. CTS has no effect on the Transmitter.

Note: Whenever the CTS bit of the MODEM Status Register changes state, an interrupt is generated if the MODEM Status Interrupt is enabled.

Data Set Ready (DSR), Pin 37: When low, this indicates that the MODEM or data set is ready to establish the communications link and transfer data with the ACE. The DSR signal is a MODEM-control function input whose condition can be tested by the CPU by reading bit 5 (DSR) of the MODEM Status Register. Bit 1 (DDSR) of the MODEM Status Register indicates whether the DSR input has changed state since the previous reading of the MODEM Status Register.

Note: Whenever the DSR bit of the MODEM Status Register changes state, an interrupt is generated if the MODEM Status Interrupt is enabled.

Data Carrier Detect (DCD), Pin 38: When low, indicates that the data carrier has been detected by the MODEM or data set. The DCD signal is a MODEM-control function input whose condition can be tested by the CPU by reading bit 7 (DCD) of the MODEM Status Register. Bit 3 (DDCD) of the MODEM Status Register indicates whether the DCD input has changed state since the previous reading of the MODEM Status Register. DCD has no effect on the receiver.

Note: Whenever the DCD bit of the MODEM Status Register changes state, an interrupt is generated if the MODEM Status Interrupt is enabled.

Ring Indicator (RI), Pin 39: When low, indicates that a telephone ringing signal has been received by the MODEM or data set. The RI signal is a MODEM-control function input whose condition can be tested by the CPU by reading bit 6 (RI) of the MODEM Status Register. Bit 2 (TERI) of the MODEM Status Register indicates whether the RI input has changed from a low to a high state since the previous reading of the MODEM Status Register.

Note: Whenever the RI bit of the MODEM Status Register changes from a high to a low state, an interrupt is generated if the MODEM Status Register is enabled.

V_{CC}, Pin 40: +5V supply.

V_{SS}, Pin 20: Ground (0V) reference.

OUTPUT SIGNALS

Data Terminal Ready (DTR), Pin 33: When low, informs the MODEM or data set that the ACE is ready to communicate. The DTR output signal can be set to an active low by programming bit 0 (DTR) of the MODEM Control Register to a high level. The DTR signal is set high upon a Master Reset operation. The DTR signal is forced to its inactive state (high) during loop mode operation.

Request to Send (RTS), Pin 32: When low, informs the MODEM or data set that the ACE is ready to transmit data. The RTS output signal can be set to an active low by programming bit 1 (RTS) of the MODEM Control Register. The RTS signal is set high upon a Master Reset operation. The RTS signal is forced to its inactive state (high) during loop mode operation.

Output 1 (OUT 1), Pin 34: User-designated output that can be set to an active low by programming bit 2 (OUT 1) of the MODEM Control Register to a high level. The OUT 1 signal is set high upon a Master Reset Operation. The OUT 1 signal is forced to its inactive state (high) during loop mode operation.

Output 2 (OUT 2), Pin 31: User-designated output that can be set to an active low by programming bit 3 (OUT 2) of the MODEM Control Register to a high level. The OUT 2 signal is set high upon a Master Reset Operation. The OUT 2 signal is forced to its inactive state (high) during loop mode operation.

Chip Select Out (CSOUT), Pin 24: When high, indicates that the chip has been selected by active, CS0, CS1, and CS2 inputs. No data transfer can be initiated until the CSOUT signal is a logic 1. CSOUT goes low when chip is deselected.

Driver Disable (DDIS), Pin 23: Goes low whenever the CPU is reading data from the ACE. A high-level DDIS output can be used to disable an external transceiver (if used between the CPU and ACE on the D_y-D₀ Data Bus) at all times, except when the CPU is reading data.

Baud Out (BAUDOUT), Pin 15: 16 × clock signal for the transmitter section of the ACE. The clock rate is equal to the main reference oscillator frequency divided by the specified divisor in the Baud Generator Divisor Latches. The BAUDOUT may also be used for the receiver section by tying this output to the RCLK input of the chip.

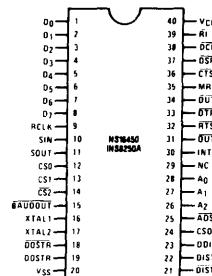
Functional Pin Description (Continued)

Interrupt (INTRPT), Pin 30: Goes high whenever any one of the following interrupt types has an active high condition and is enabled via the IER: Receiver Error Flag; Received Data Available; Transmitter Holding Register Empty; and MODEM Status. The INTRPT signal is reset low upon the appropriate interrupt service or a Master Reset operation.

Serial Output (SOUT), Pin 11: Composite serial data output to the communications link (peripheral, MODEM or data set). The SOUT signal is set to the Marking (logic 1) state upon a Master Reset operation.

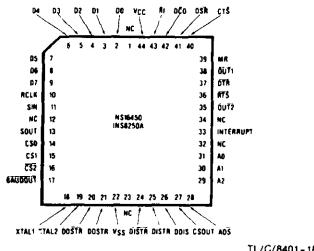
Connection Diagrams

Dual-In-Line Package



Top Menu

TABLE I. ACE Beast Functions



T1/C(8401-1)

TABLE I. ACE Reset Functions		
Register/Signal	Reset Control	Reset State
Interrupt Enable Register	Master Reset	All Bits Low (0-3 forced and 4-7 permanent)
Interrupt Identification Register	Master Reset	Bit 0 is High, Bits 1 and 2 Low Bits 3-7 are Permanently Low
Line Control Register	Master Reset	All Bits Low
MODEM Control Register	Master Reset	All Bits Low
Line Status Register	Master Reset	All Bits Low, Except Bits 5 and 6 are High
MODEM Status Register	Master Reset	Bits 0-3 Low Bits 4-7—Input Signal
SOUT	Master Reset	High
INTRPT (RCVR Errs)	Read LSR/MR	Low
INTRPT (RCVR Data Ready)	Read RBR/MR	Low
INTRPT (THR)	Read IIR/Write THR/MR	Low
INTRPT (Modem Status Changes)	Read MSR/MR	Low
OUT2	Master Reset	High
RTS	Master Reset	High
DTR	Master Reset	High
OUT1	Master Reset	High

Accessible Registers

The system programmer may access or control any of the ACE registers summarized in Table II via the CPU. These registers are used to control ACE operations and to transmit and receive data.

LINE CONTROL REGISTER

The system programmer specifies the format of the asynchronous data communications exchange via the Line Control Register. In addition to controlling the format, the programmer may retrieve the contents of the Line Control Register for inspection. This feature simplifies system programming and eliminates the need for separate storage in system memory of the line characteristics. The contents of the Line Control Register are indicated in Table II and are described below.

Bits 0 and 1: These two bits specify the number of bits in each transmitted or received serial character. The encoding of bits 0 and 1 is as follows:

Bit 1	Bit 0	Word Length
0	0	5 Bits
0	1	6 Bits
1	0	7 Bits
1	1	8 Bits

Bit 2: This bit specifies the number of Stop bits in each transmitted character. If bit 2 is a logic 0, one Stop bit is generated in the transmitted data. If bit 2 is a logic 1 when a 5-bit word length is selected via bits 0 and 1, one and a

Table II. Summary of Accessible Registers

Bit No.	Register Address											
	0 DLAB = 0	0 DLAB = 0	1 DLAB = 0	2	3	4	5	6	7	0 DLAB = 1	1 DLAB = 1	
	Receiver Buffer Register (Read Only)	Transmitter Holding Register (Write Only)	Interrupt Enable Register	Interrupt Ident. Register (Read Only)	Line Control Register	MODEM Control Register	Line Status Register	MODEM Status Register	Scratch Register	Divisor Latch (LS)	Latch (MS)	
	RBR	THR	IIR	IIR	LCR	MCR	LSR	MSR	SCR	DLL	DLM	
0	Data Bit 0*	Data Bit 0	Enable Received Data Available Interrupt (ERBFI)	"0" if Interrupt Pending	Word Length Select Bit 0 (WLS0)	Data Terminal Ready (DTR)	Data Ready (DR)	Delta Clear to Send (DCTS)	Bit 0	Bit 0	Bit 8	
1	Data Bit 1	Data Bit 1	Enable Transmitter Holding Register Empty Interrupt (ETBEI)	Interrupt ID Bit (0)	Word Length Select Bit 1 (WLS1)	Request to Send (RTS)	Overrun Error (OE)	Delta Data Set Ready (DDSR)	Bit 1	Bit 1	Bit 9	
2	Data Bit 2	Data Bit 2	Enable Receiver Line Status Interrupt (ELSI)	Interrupt ID Bit (1)	Number of Stop Bits (STB)	Out 1	Parity Error (PE)	Trailing Edge Ring Indicator (TERI)	Bit 2	Bit 2	Bit 10	
3	Data Bit 3	Data Bit 3	Enable MODEM Status Interrupt (EDSSI)	0	Parity Enable (PEN)	Out 2	Framing Error (FE)	Delta Data Carrier Detect (DDCD)	Bit 3	Bit 3	Bit 11	
4	Data Bit 4	Data Bit 4	0	0	Even Parity Select (EPS)	Loop	Break Interrupt (BI)	Clear to Send (CTS)	Bit 4	Bit 4	Bit 12	
5	Data Bit 5	Data Bit 5	0	0	Stick Parity	0	Transmitter Holding Register (THRE)	Data Set Ready (DSR)	Bit 5	Bit 5	Bit 13	
6	Data Bit 6	Data Bit 6	0	0	Set Break	0	Transmitter Empty (TEMT)	Ring Indicator (RI)	Bit 6	Bit 6	Bit 14	
7	Data Bit 7	Data Bit 7	0	0	Divisor Latch Access Bit (DLAB)	0	0	Data Carrier Detect (DCD)	Bit 7	Bit 7	Bit 15	

*Bit 0 is the least significant bit. It is the first bit serially transmitted or received.

Accessible Registers (Continued)

half Stop bits are generated. If bit 2 is a logic 1 when either a 6-, 7-, or 8-bit word length is selected, two Stop bits are generated. The Receiver checks the first Stop-bit only, regardless of the number of Stop bits selected.

Bit 3: This bit is the Parity Enable bit. When bit 3 is a logic 1, a Parity bit is generated (transmit data) or checked (receive data) between the last data word bit and Stop bit of the serial data. (The Parity bit is used to produce an even or odd number of 1s when the data word bits and the Parity bit are summed.)

Bit 4: This bit is the Even Parity Select bit. When bit 3 is a logic 1 and bit 4 is a logic 0, an odd number of logic 1s is transmitted or checked in the data word bits and Parity bit. When bit 3 is a logic 1 and bit 4 is a logic 1, an even number of logic 1s is transmitted or checked.

Bit 5: This bit is the Stick Parity bit. When bits 3, 4 and 5 are logic 1 the Parity bit is transmitted and checked by the receiver as a logic 0. If bits 3 and 5 are 1 and bit 4 is a logic 0 then the Parity bit is transmitted as a 0.

Bit 6: This bit is the Break Control bit. When it is set to a logic 1, the serial output (SOUT) is forced to the Spacing (logic 0) state. The break is disabled by setting bit 6 to a logic 0. The Break Control bit acts only on SOUT and has no effect on the transmitter logic.

Note: This feature enables the CPU to alert a terminal in a computer communications system. If the following sequence is followed, no erroneous or extraneous characters will be transmitted because of the break.

1. Load an all 0s, pad character, in response to THRE.

2. Set break after the next THRE.

3. Wait for the transmitter to be idle, (TEM1 = 1), and clear break when normal transmission has to be restored.

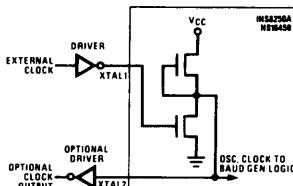
During the break, the Transmitter can be used as a character timer to accurately establish the break duration.

Bit 7: This bit is the Divisor Latch Access Bit (DLAB). It must be set high (logic 1) to access the Divisor Latches of the Baud Generator during a Read or Write operation. It must be set low (logic 0) to access the Receiver Buffer, the Transmitter Holding Register, or the Interrupt Enable Register.

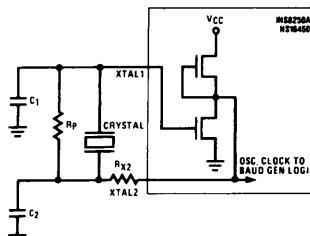
Table III. Baud Rates Using 1.8432 MHz Crystal

Desired Baud Rate	Divisor Used to Generate 16 x Clock	Percent Error Difference Between Desired and Actual
50	2304	—
75	1536	—
110	1047	0.026
134.5	857	0.058
150	768	—
300	384	—
600	192	—
1200	96	—
1800	64	—
2000	58	0.69
2400	48	—
3600	32	—
4800	24	—
7200	16	—
9600	12	—
19200	6	—
38400	3	—
56000	2	2.86

Typical Clock Circuits



TL/C/8401-12



TL/C/8401-13

Typical Crystal Oscillator Network

CRYSTAL	R _P	R _{X2}	C ₁	C ₂
3.1 MHz	1 MΩ	1.5k	10-30 pF	40-60 pF
1.8 MHz	1 MΩ	1.5k	10-30 pF	40-60 pF

Table IV. Baud Rates Using 3.072 MHz Crystal

Desired Baud Rate	Divisor Used to Generate 16 x Clock	Percent Error Difference Between Desired and Actual
50	3840	—
75	2560	—
110	1745	0.026
134.5	1428	0.034
150	1280	—
300	640	—
600	320	—
1200	160	—
1800	107	0.312
2000	96	—
2400	80	—
3600	53	0.628
4800	40	—
7200	27	1.23
9600	20	—
19200	10	—
38400	5	—

Accessible Registers (Continued)

PROGRAMMABLE BAUD GENERATOR

The ACE contains a programmable Baud Generator that is capable of taking any clock input (DC to 3.1 MHz) and dividing it by any divisor from 1 to 2¹⁶ - 1. The output frequency of the Baud Generator is $16 \times \text{the Baud [divisor} \# = (\text{frequency input}) \div (\text{baud rate} \times 16)]$. Two 8-bit latches store the divisor in a 16-bit binary format. These Divisor Latches must be loaded during initialization in order to ensure desired operation of the Baud Generator. Upon loading either of the Divisor Latches, a 16-bit Baud counter is immediately loaded. This prevents long counts on initial load.

Tables III and IV illustrate the use of the Baud Generator with crystal frequencies of 1.8432 MHz and 3.072 MHz respectively. For baud rates of 38400 and below, the error obtained is minimal. The accuracy of the desired baud rate is dependent on the crystal frequency chosen.

Note: The maximum operating frequency of the Baud Generator is 3.1 MHz. However, when using divisors of 3 and below, the maximum frequency is equal to the divisor in MHz. For example, if the divisor is 1, then the maximum frequency is 1 MHz. In no case should the date rate be greater than 56k Baud.

LINE STATUS REGISTER

This 8-bit register provides status information to the CPU concerning the data transfer. The contents of the Line Status Register are indicated in Table 2 and are described below.

Bit 0: This bit is the receiver Data Ready (DR) indicator. Bit 0 is set to a logic 1 whenever a complete incoming character has been received and transferred into the Receiver Buffer Register. Bit 0 is reset to a logic 0 by reading the data in the Receiver Buffer Register.

Bit 1: This bit is the Overrun Error (OE) indicator. Bit 1 indicates that data in the Receiver Buffer Register was not read by the CPU before the next character was transferred into the Receiver Buffer Register, thereby destroying the previous character. The OE indicator is reset whenever the CPU reads the contents of the Line Status Register.

Bit 2: This bit is the Parity Error (PE) indicator. Bit 2 indicates that the received data character does not have the correct even or odd parity, as selected by the even-parity-select bit. The PE bit is set to a logic 1 upon detection of a parity error and is reset to a logic 0 whenever the CPU reads the contents of the Line Status Register.

Bit 3: This bit is the Framing Error (FE) indicator. Bit 3 indicates that the received character did not have a valid Stop bit. Bit 3 is set to a logic 1 whenever the Stop bit following the last data bit or parity bit is detected as a zero bit (Spacing level). The FE indicator is reset whenever the CPU reads the contents of the Line Status indicator.

Bit 4: This bit is the Break Interrupt (BI) indicator. Bit 4 is set to a logic 1 whenever the received data input is held in the Spacing (logic 0) state for longer than a full word transmission time (that is, the total time of Start bit + data bits + Parity + Stop bits). The BI indicator is reset whenever the CPU reads the contents of the Line Status indicator.

Note: Bits 1 through 4 are the error conditions that produce a Receiver Line Status interrupt whenever any of the corresponding conditions are detected.

Bit 5: The bit is the Transmitter Holding Register Empty (THRE) indicator. Bit 5 indicates that the ACE is ready to accept a new character for transmission. In addition, this bit causes the ACE to issue an interrupt to the CPU when the Transmit Holding Register Empty interrupt enable is set high. The THRE bit is set to a logic 1 when a character is transferred from the Transmitter Holding Register into the Transmitter Shift Register. The bit is reset to logic 0 concurrently with the loading of the Transmitter Holding Register by the CPU.

Bit 6: This bit is the Transmitter Empty (TEM) indicator. Bit 6 is set to a logic 1 whenever the Transmitter Holding Register (THR) and the Transmitter Shift Register (TSR) are both empty. It is reset to a logic 0 whenever either the THR or TSR contains a data character.

Bit 7: This bit is permanently set to logic 0.

Note: The Line Status Register is intended for read operations only. Writing to this register is not recommended as this operation is used for factory testing.

TABLE V. Interrupt Control Functions

Interrupt Identification Register			Interrupt Set and Reset Functions			
Bit 2	Bit 1	Bit 0	Priority Level	Interrupt Type	Interrupt Source	Interrupt Reset Control
0	0	1	—	None	None	—
1	1	0	Highest	Receiver Line Status	Overrun Error or Parity Error or Framing Error or Break Interrupt	Reading the Line Status Register
1	0	0	Second	Received Data Available	Receiver Data Available	Reading the Receiver Buffer Register
0	1	0	Third	Transmitter Holding Register Empty	Transmitter Holding Register Empty	Reading the IIR Register (if source of interrupt) or Writing into the Transmitter Holding Register
0	0	0	Fourth	MODEM Status	Clear to Send or Data Set Ready or Ring Indicator or Data Carrier Detect	Reading the MODEM Status Register

Accessible Registers (Continued)

INTERRUPT IDENTIFICATION REGISTER

The ACE has an on-chip interrupt capability that allows for flexibility in interfacing popular microprocessors presently available. In order to provide minimum software overhead during data character transfers, the ACE prioritizes interrupts into four levels. The four levels of interrupt conditions are as follows: Received Line Status (priority 1); Received Data Ready (priority 2); Transmitter Holding Register Empty (priority 3); and MODEM Status (priority 4).

Information indicating that a prioritized interrupt is pending and the type of that interrupt are stored in the Interrupt Identification Register (IIR). When addressed during chip-select time, the IIR freezes the highest priority interrupt pending and no other interrupts are acknowledged until the particular interrupt is serviced by the CPU. The contents of the IIR are indicated in Table II and are described below.

Bit 0: This bit can be used in either a hardwired prioritized or polled environment to indicate whether an interrupt is pending. When bit 0 is a logic 0, an interrupt is pending and the IIR contents may be used as a pointer to the appropriate interrupt service routine. When bit 0 is a logic 1, no interrupt is pending and polling (if used) continues.

Bits 1 and 2: These two bits of the IIR are used to identify the highest priority interrupt pending as indicated in Table V.

Bits 3 through 7: These five bits of the IIR are always logic 0.

INTERRUPT ENABLE REGISTER

This 8-bit register enables the four types of interrupts of the ACE to separately activate the chip interrupt (INTRPT) output signal. It is possible to totally disable the interrupt system by resetting bits 0 through 3 of the Interrupt Enable Register. Similarly, by setting the appropriate bits of this register to a logic 1, selected interrupts can be enabled. Disabling the interrupt system inhibits the Interrupt Identification Register and the active (high) INTRPT output from the chip. All other system functions operate in their normal manner, including the setting of the Line Status and MODEM Status Registers. The contents of the Interrupt Enable Register are indicated in Table II and are described below.

Bit 0: This bit enables the Received Data Available Interrupt when set to logic 1.

Bit 1: This bit enables the Transmitter Holding Register Empty Interrupt when set to logic 1.

Bit 2: This bit enables the Receiver Line Status Interrupt when set to logic 1.

Bit 3: This bit enables the MODEM Status Interrupt when set to logic 1.

Bits 4 through 7: These four bits are always logic 0.

MODEM CONTROL REGISTER

This 8-bit register controls the interface with the MODEM or data set (or a peripheral device emulating a MODEM). The contents of the MODEM Control Register are indicated in Table II and are described below.

Bit 0: This bit controls the Data Terminal Ready (DTR) output. When bit 0 is set to a logic 1, the DTR output is forced to a logic 0. When bit 0 is reset to a logic 0, the DTR output is forced to a logic 1.

Note: The DTR output of the ACE may be applied to an EIA inverting line driver (such as the DS1488) to obtain the proper polarity input at the succeeding MODEM or data set.

Bit 1: This bit controls the Request to Send (RTS) output. Bit 1 affects the RTS output in a manner identical to that described above for bit 0.

Bit 2: This bit controls the Output 1 (OUT 1) signal, which is an auxiliary user-designated output. Bit 2 affects the OUT 1 output in a manner identical to that described above for bit 0.

Bit 3: This bit controls the Output 2 (OUT 2) signal, which is an auxiliary user-designated output. Bit 3 affects the OUT 2 output in a manner identical to that described above for bit 0.

Bit 4: This bit provides a local loopback feature for diagnostic testing of the ACE. When bit 4 is set to logic 1, the following occur: the transmitter Serial Output (SOUT) is set to the Marking (logic 1) state; the receiver Serial Input (SIN) is disconnected; the output of the Transmitter Shift Register is "looped back" into the Receiver Shift Register input; the four MODEM Control inputs (CTS, DSR, DCD, and RI) are disconnected; and the four MODEM Control outputs (DTR, RTS, OUT 1, and OUT 2) are internally connected to the four MODEM Control inputs, and the MODEM Control output pins are forced to their inactive state (high). In the diagnostic mode, data that is transmitted is immediately received. This feature allows the processor to verify the transmit-and-received-data paths of the ACE.

In the diagnostic mode, the receiver and transmitter interrupts are fully operational. The MODEM Control Interrupts are also operational, but the interrupts' sources are now the lower four bits of the MODEM Control Register instead of the four MODEM Control inputs. The interrupts are still controlled by the Interrupt Enable Register.

Bits 5 through 7: These bits are permanently set to logic 0.

MODEM STATUS REGISTER

This 8-bit register provides the current state of the control lines from the MODEM (or peripheral device) to the CPU. In addition to this current-state information, four bits of the MODEM Status Register provide change information. These bits are set to a logic 1 whenever a control input from the MODEM changes state. They are reset to logic 0 whenever the CPU reads the MODEM Status Register.

Accessible Registers (Continued)

The contents of the MODEM Status Register are indicated in Table II and are described below.

Bit 0: This bit is the Delta Clear to Send (DCTS) indicator. Bit 0 indicates that the CTS input to the chip has changed state since the last time it was read by the CPU.

Bit 1: This bit is the Delta Data Set Ready (DDSR) indicator. Bit 1 indicates that the DSR input to the chip has changed state since the last time it was read by the CPU.

Bit 2: This bit is the Trailing Edge of Ring Indicator (TERI) detector. Bit 2 indicates that the RI input to the chip has changed from a low to a high state.

Bit 3: This bit is the Delta Data Carrier Detect (DDCD) indicator. Bit 3 indicates that the DCD input to the chip has changed state.

Note: Whenever bit 0, 1, 2, or 3 is set to logic 1, a MODEM Status interrupt is generated.

Bit 4: This bit is the complement of the Clear to Send (CTS) input. If bit 4 (loop) of the MCR is set to a 1, this bit is equivalent to RTS in the MCR.

Bit 5: This bit is the complement of the Data Set Ready (DSR) input. If bit 4 of the MCR is set to a 1, this bit is equivalent to DTR in the MCR.

Bit 6: This bit is the complement of the Ring Indicator (RI) input. If bit 4 of the MCR is set to a 1, this bit is equivalent to OUT 1 in the MCR.

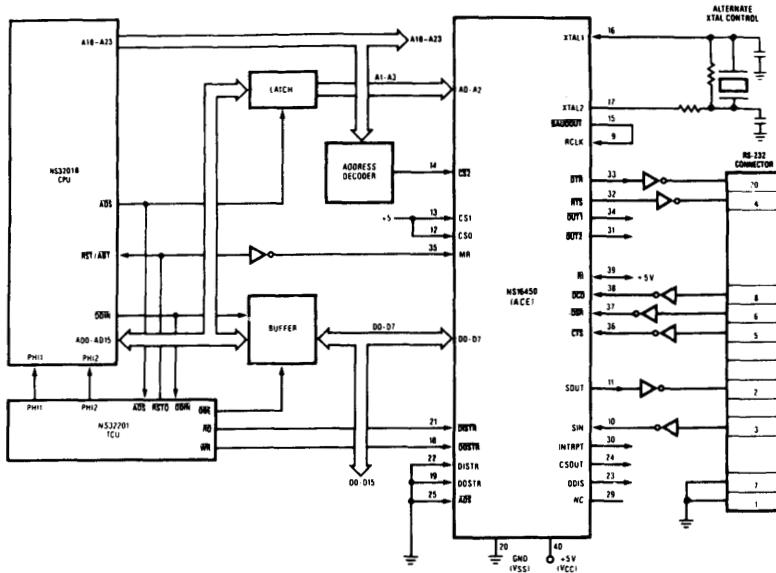
Bit 7: This bit is the complement of the Data Carrier Detect (DCD) input. If bit 4 of the MCR is set to a 1, this bit is equivalent to OUT 2 of the MCR.

SCRATCHPAD REGISTER

This 8-bit Read/Write Register does not control the ACE in any way. It is intended as a scratchpad register to be used by the programmer to hold data temporarily.

Typical Applications

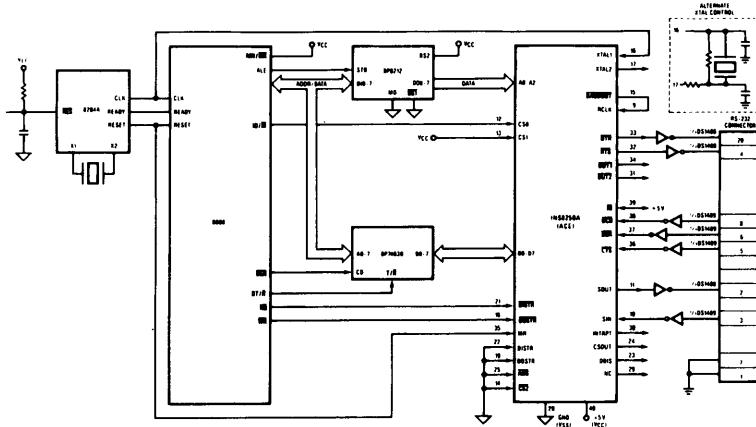
This shows the basic connections of an NS16450 to an NS32016 CPU



TL/C/8401-14

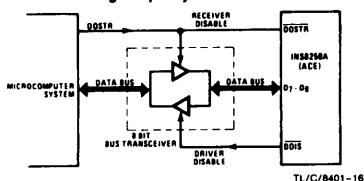
Typical Applications (Continued)

This shows the basic connections of an INS8250A to an 8088 CPU



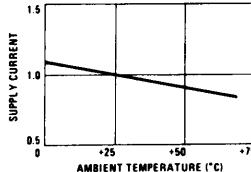
TL/C/8401-15

Typical Interface for a High-Capacity Data Bus



TL/C/8401-16

Typical Supply Current vs. Temperature, Normalized

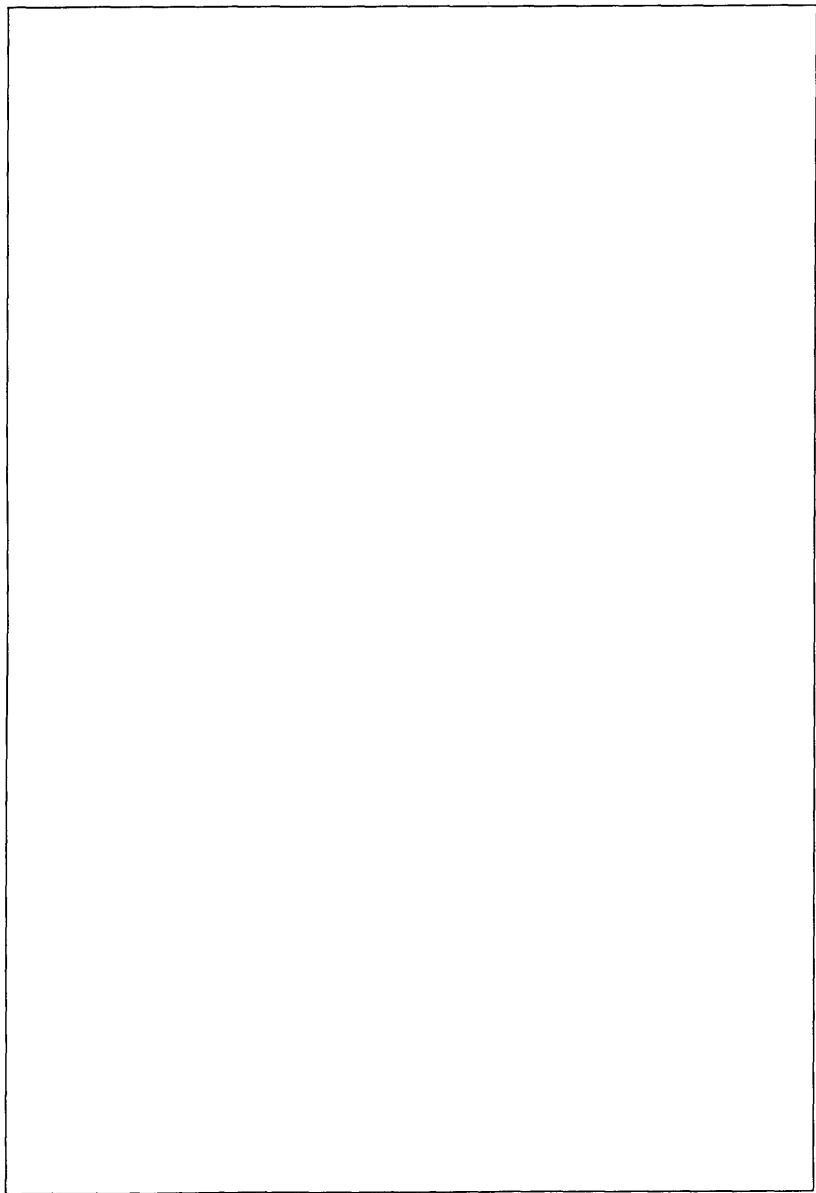


TL/C/8401-17

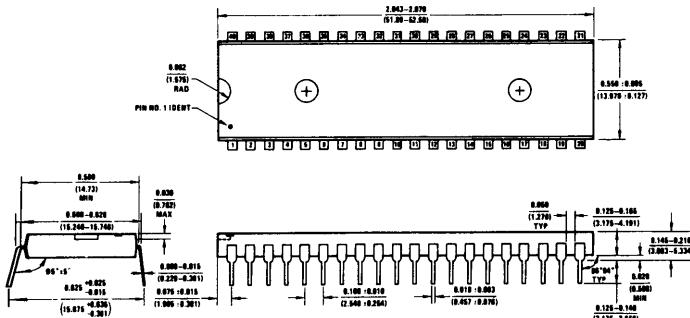
Ordering Information

Order Number	Description
Plastic Dip Package NS16450N or NS-16450N	high speed part
INS8250AN INS8250AWN NS16C450N* INS82C50AN*	$V_{CC} = 5V \pm 5\%$ $V_{CC} = 5V \pm 10\%$ CMOS high speed part CMOS $V_{CC} = 5V \pm 5\%$
Plastic Chip Carrier Package NS16450V or NS-16450V	high speed part
INS8250A NS16C450V* INS82C50AV*	$V_{CC} = 5V \pm 5\%$ CMOS high speed part CMOS $V_{CC} = 5V \pm 5\%$

*The CMOS parts will be available after 6/85.

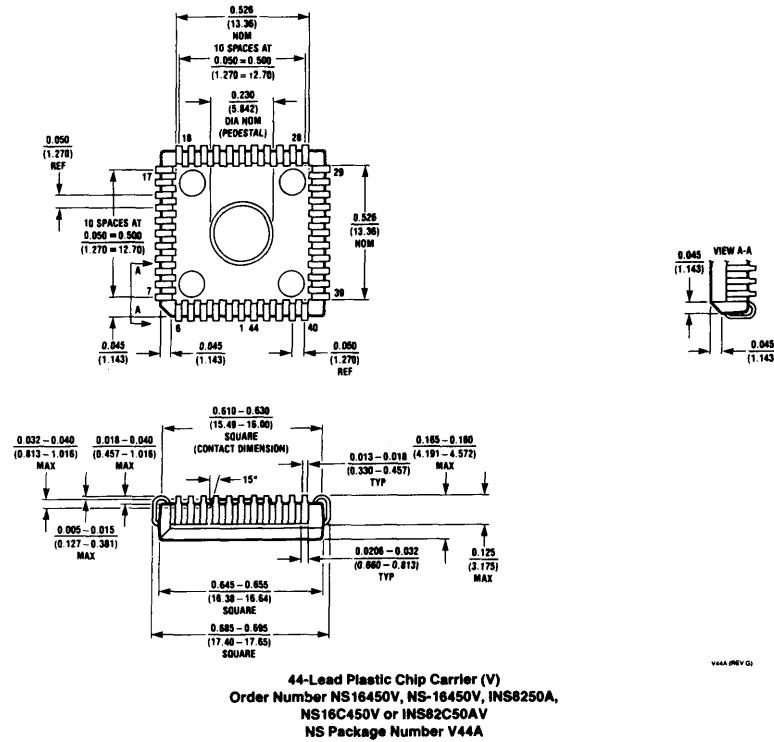


Physical Dimensions inches (millimeters)



Plastic Dual-In-Line Package (N)
Order Number NS16450N, NS-16450N, INS8250AN,
INS8250AWN, NS16C450N, or INS82C50AN
NS Package Number N40A

Physical Dimensions inches (millimeters) (Continued)



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
 Corporate Headquarters
 2905 Semiconductor Drive
 P.O. Box 5090
 Santa Clara, CA 95052-8090
 Tel: (408) 721-5000
 TWX: (916) 559-0240

National Semiconductor
 Germany
 Furtwangenstrasse Nr. 5
 D-8000 Munchen 21
 West Germany
 Tel: (089) 512-12-0
 Telex: 522772

NS Japan K.K.
 4-403 Itabukuro,
 Toshima-ku,
 Tokyo 171, Japan
 Tel: (03) 988-2131
 FAX: 011-81-3-988-1700

N.S.H.K. Ltd.
 S.E. Asia Office, N.S.H.K. Sales
 4th Floor, 49 Phra
 Athit Road, Bangkok
 10110, Thailand
 Tel: (03) 212-5066
 Telex: 551111 212-5066
 Tembahan, Kowloon,
 Hong Kong
 Tel: (03) 2521290, 3-7243645
 Cable: NSSEAMKTO
 Telex: 52998 NSSEA HK

National Semiconductors
 Do Brasil Ltda.
 Av. das Américas, 497
 01452 Sao Paulo, SP, Brazil
 Tel: (03) 729-6333
 Telex: 391-1131901 NSBR BR

National Semiconductor
 (Australia) PTY, Ltd.
 27/300 Victoria Street
 01452 Sao Paulo, SP, Brazil
 Tel: (03) 729-6333
 Telex: AA32096

NCR

MICROELECTRONICS DIVISION

NCR 8496
SOUND GENERATOR
DATA SHEET

NCR 8496 SOUND GENERATOR

TABLE OF CONTENTS

SECTION	PAGE
1 GENERAL DESCRIPTION	2
2 FUNCTIONAL DESCRIPTIONS	4
2.1 Control Registers	4
2.2 Tone Generation	4
2.3 Noise Generation	5
2.4 Data Transfer	6
2.5 CPU Interface	7
2.6 Output Circuitry	8
3 INTERFACE DEFINITIONS	9
3.1 Microprocessor Interface	9
3.2 Audio Application Interface	10
3.3 Power Interface	10
4 ELECTRICAL CHARACTERISTICS	11
4.1 Operating Conditions	11
4.2 Input Characteristics	11
4.3 Output Characteristics	12
4.4 Audio Characteristics	12
5 TIMING REQUIREMENTS	13

LIST OF ILLUSTRATIONS

1. FUNCTIONAL PIN GROUPING	2
2. FUNCTIONAL PIN OUT	2
3. FUNCTIONAL BLOCK DIAGRAM	3
4. DATA TRANSFER TIMING	13

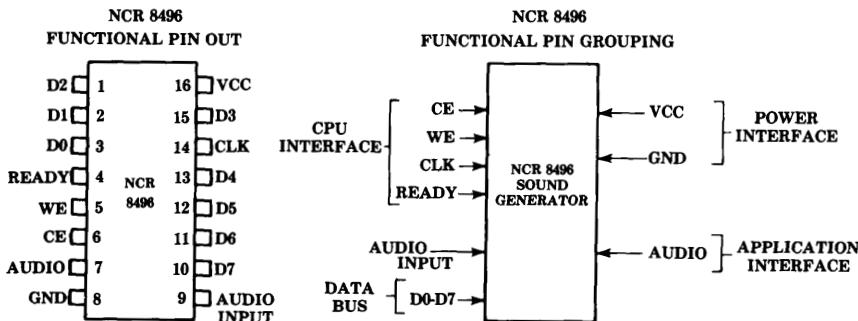
SECTION 1

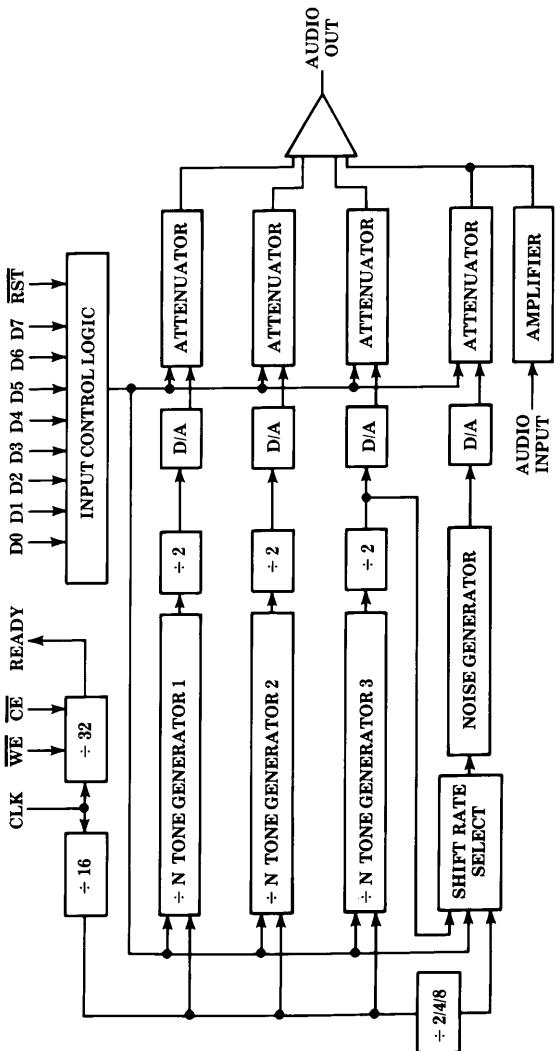
GENERAL DESCRIPTION

The NCR 8496 is an NMOS digital sound generator capable of providing applications with a low cost solution for noise and sound generation.

FEATURES

- Functionally and Pin compatible with the SNR76496
- Programmable white or periodic noise generator
- Three programmable tone generators
- Programmable attenuation values
- Simultaneous multiple sound generation
- TTL compatible
- 4 MHz maximum clock input
- External audio input added to Internal Generators





NCR 8496 FUNCTIONAL BLOCK DIAGRAM

SECTION 2

FUNCTIONAL DESCRIPTIONS

2.1 Control Registers

The NCR 8496 Sound Generator has eight (8) internal registers used to control three (3) tone generators and one (1) noise generator. A three (3) bit data word used to determine the destination control register is contained in the first byte of data for all data transfers. The internal register designations are as follows:

Address Bits			Register Destination
RO	R1	R2	Description
0	0	0	Tone 1: Frequency
0	0	1	Tone 1: Attenuation
0	1	0	Tone 2: Frequency
0	1	1	Tone 2: Attenuation
1	0	0	Tone 3: Frequency
1	0	1	Tone 3: Attenuation
1	1	0	Noise : Control
1	1	1	Noise : Attenuation

Note: RO is the most significant address bit

2.2 Tone Generation

The NCR 8496 sound generator has three (3) programmable tone generators, each with separate frequency synthesis and attenuation sections. The frequency synthesis section requires ten (10) bits of data (F0 to F9) to define half the period of the desired frequency. This data is entered into a ten (10) stage tone counter, which is decremented at a rate of $N/16$ where N is the clock input frequency. A signal is produced when this tone counter decrements to one, which toggles a divide by two counter and reloads the tone counter. Therefore, the period of the desired frequency is twice the value of the tone generator.

The frequency of each tone generation is calculated using the equation:

$$f = N/32n$$

N = the clock input frequency
n = a 10 bit binary number [2 < n < 1023]

The divide by two counter is directly connected to a four stage attenuator whose values and bit position in the data word are shown in the following table:

ATTENUATION CONTROL

Data				Value	Data				Value
A0	A1	A2	A3	dB	A0	A1	A2	A3	dB
0	0	0	0	0	1	0	0	0	-16
0	0	0	1	-2	1	0	0	1	-18
0	0	1	0	-4	1	0	1	0	-20
0	0	1	1	-6	1	0	1	1	-22
0	1	0	0	-8	1	1	0	0	-24
0	1	0	1	-10	1	1	0	1	-26
0	1	1	0	-12	1	1	1	0	-28
0	1	1	1	-14	1	1	1	1	OFF

Note: A0 is the most significant bit of data

2.3 Noise Generation

The NCR Sound Generator has two (2) noise sources (periodic and white), which share a common attenuator. These noise sources are shift registers with an exclusive NOR feedback network. One (1) of four (4) noise generator shift rates, each rate being derived from the input clock, will be controlled by the two (2) NF bits, as is shown in the following table:

NOISE GENERATOR FREQUENCY CONTROL

NF BITS		FREQUENCY CONTROL
NFO	NF1	SHIFT RATE
0	0	N/512
0	1	N/1024
1	0	N/2048
1	1	Tone Generator #3 Output

Note: NFO is the most significant bit

The choice of either periodic or white noise is controlled by the noise feedback control bit FB, as is shown in the following table:

NOISE FEEDBACK CONTROL

FB	CONFIGURATION
0	Periodic Noise
1	White Noise

2.4 Data Transfer

The NCR 8496 Sound Generator is enabled by the CPU by asserting a low logic level to CE. WE strobes the contents of the data bus to the appropriate control register. Data bus contents must be valid at this time. Data transfers cannot occur unless CE is true.

Thirty two (32) clock cycles are required by the NCR 8496 to load data into the control register. The READY output used as a handshake signal to synchronize the CPU, is asserted to a low logic level immediately following the leading edge of CE. READY assumes a true state via an external pull up register once the data transfer has been completed.

Formats for Data Transfer are as follows:

FREQUENCY UPDATE (Double Byte Transfer)

FIRST BYTE

DATA	REGISTER ADDRESS	BIT 0
F9 F8 F7 F6	R2 R1 R0	1
D7		D0

SECOND BYTE

DATA	BIT 0
F5 F4 F3 F2 F1 F0 X	0
D7	D0

NOISE SOURCE UPDATE (Single Byte Transfer)

SHIFT RATE	FEEDBACK		REGISTER ADDRESS	BIT 0
NFL NFO	FB	X	R2 R1 R0	1
D7				D0

ATTENUATOR UPDATE (Single Byte Transfer)

DATA	REGISTER ADDRESS	BIT 0
A3 A2 A1 A0	R2 R1 R0	1
D7		D0

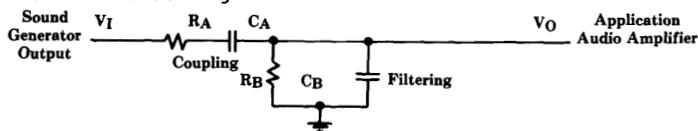
2.5 CPU INTERFACE

Eight (8) data lines (D0-D7) and three (3) control lines (WE, CE, READY) interface the NCR 8496 Sound Generator to the CPU. As indicated in Section 2.2, Tone Generation, ten (10) bits of data are required by each tone generator in selecting frequency values. Frequency updates require double byte data transfers. An additional four (4) bits of data are required to select the attenuation values. Attenuation updates require only single byte data transfers. (See Section 2.4: Data Transfer).

Tone generators can be quickly updated by initially sending both bytes of frequency and register data, followed by only the second byte of data for succeeding values only if no other control registers are accessed at the time of generator updating. This action is accomplished by latching the register address and permitting the continued transfer of data into the same register. This updating feature permits the expedited modification of the six (6) most significant bits of data needed for frequency sweeps.

2.6 OUTPUT CIRCUITRY

The NCR 8496 Sound Generator output circuitry, emulating a conventional op amp summing circuit, sums the three (3) tone and one (1) noise generator outputs, and will source/sink current to 2 mA. The 0 dB output signal per generator is nominally a 450 mV square in the negative direction from a 2V quiescent level. The output should be OR coupled into the application audio circuit via a filtering network similar to the following:



The upper and lower frequency poles for the application are determined from the following equations:

Lower Pole

$$f \cong \frac{1}{2\pi(R_A + R_B) C_A}$$

Upper Pole

$$f \cong \frac{1}{2\pi(R_A // R_B) C_B}$$

Attenuation of the output signal is:

$$\frac{V_O}{V_I} = \frac{R_B}{R_A + R_B}$$

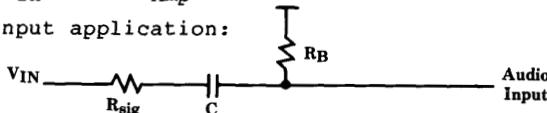
Typically $R_B \geq 10 R_A$, so that the attenuation can be small while achieving desired filtering at the same time.

2.7 AUDIO INPUT CIRCUIT

This input node can be biased on with a current to give an approximate transfer function at the output of :

$$V_O = R_{Amp} I_{in} \text{ where } R_{Amp} \text{ is on the order of } 1K \text{ Ohms}$$

Typical input application:



R_B provides the bias current to put the amplifier in the linear range.

R_{sig} controls the input current causing the signal swing.

SECTION 3
INTERFACE DEFINITION

3.1 MICROPROCESSOR INTERFACE

Signal	Pin	Description
READY	4	OUTPUT: Open collector, READY indicates that data has been read when true (high). The CPU must be placed in a wait state until READY is true.
<u>WE</u>	5	INPUT: Write Enable <u>WE</u> indicates that data is available to the NCR 8496 when true (low).
<u>CE</u>	6	INPUT: Chip Enable <u>CE</u> indicates that data may be transferred to the NCR 8496.
<u>RST</u>	9	INPUT: Master Reset <u>RST</u> is used for testing purposes only. This pin is a no connect on the SN 76489A and is internally pulled high.
D7	10	Inputs: D0-D7 is the data bust through which data is transferred. D0 is the most significant data bit. D7 is the least significant data bit.
D6	11	
D5	12	
D4	13	
D3	15	
D2	1	
D1	2	
D0	3	
CLK	14	Input Clock

3.2 AUDIO APPLICATION INTERFACE

Signal	Pin	Description
Audio	7	OUTPUT: Audio signal to application. Refer to section 2.6. Output Circuitry for recommended output connections.
Audio In	9	INPUT: Audio input signal from application. Refer to section 2.7.

3.3 POWER INTERFACE

Signal	Pin	Description
VCC	16	Supply Voltage
GND	8	Ground Reference

SECTION 4
ELECTRICAL CHARACTERISTICS

4.1 OPERATING CONDITIONS

Parameter	Symbol	Conditions	Min	Max	Units
Supply Voltage	V_{CC}		4.5	5.5	v
Supply Current	I_{CC}	Outputs Open		40	mA
Operating Temperature	T_O		0	+70	°C
Storage Temperature	T_S		-65	+150	°C
Absolute Maximum	V_{max}	To Any Pin		7.0	v

4.2 INPUT CHARACTERISTICS

Parameter	Symbol	Conditions	Min	Max	Units
Input Voltage Low	V_{IL}	$D_0-D_7, \overline{WE}, \overline{CE}, CLK$		0.8	v
Input Voltage High	V_{IH}	$D_0-D_7, \overline{WE}, \overline{CE}, CLK$	2.0		v
Input Current	I_I	$V_{in}-GND \rightarrow V_{CC}$	-10	+10	uA
Input Capacitance	C_I			15	pf

4.3 OUTPUT CHARACTERISTICS

Parameter	Symbol	Conditions	Min	Max	Units
Output Voltage Low	V_{OL}	$I_{OUT} = -2\text{mA}$ (READY)		0.4	V

4.4 AUDIO CHARACTERISTICS

Parameter	Symbol	Conditions	Typ	Max	Units
Audio Input Current		Causing half scale output swing		2.0	mA
Source Current	I_{SO}	Over Output Voltage Swing		-3	mA
Sink Current	I_{SO}	Over Output Voltage Swing		2	mA
Quiescent Output	V_{OQ}		2.1		V
Maximum Output	V_{OM}	Generators at 0dB	0.5		V
Signal Swing	V_{SW}	Generators at 0dB	450		mV
Capacitance Loading *	C_{OL}	From Pin 7 to Ground for Stability		200	pf
Audio Input Bias Voltage		$R=4.7\text{k ohms}$	1.0V	1.2V	

* Does not apply to coupling capacitors which are connected to loads >500 ohms. It is recommended that capacitors to ground be isolated by a series resistance of 500 ohms for stability.

SECTION 5
TIMING REQUIREMENTS

Parameter	Symbol	Conditions	Min	Max	Units
CE READY	t_{CER}	$CL=225\text{pf}$ $RL=2K$ to VCC		150	nS
Frequency Input	CLOCK	Transition Time	.05	4	MHz
Set up Time	t_{su2}	$\underline{\text{Data}} \rightarrow \overline{\text{WE}}$	0		nS
Hold Time	t_h	$\overline{\text{CE}} \rightarrow \overline{\text{WE}}$	0		nS
		$\text{Data} \rightarrow \text{READY}$			

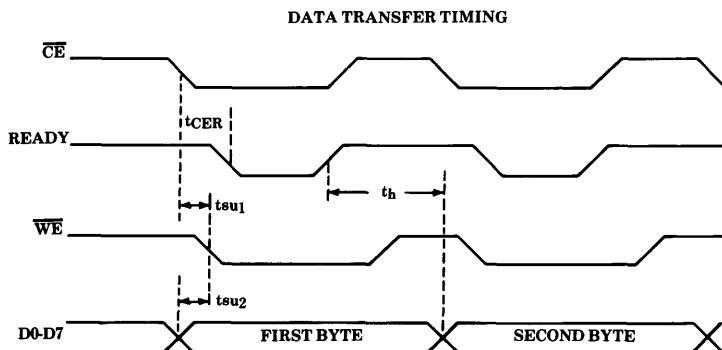


FIGURE 4



MOTOROLA

MC14529B

DUAL 4-CHANNEL ANALOG DATA SELECTOR

The MC14529B analog data selector is a dual 4-channel or single 8-channel device depending on the input coding. The device is suitable for digital as well as analog application, including various one-of-four and one-of-eight data selector functions. Since the device has bidirectional analog characteristics it can also be used as a dual binary to 1-of-4 or a binary to 1-of-8 decoder.

- Data Paths Are Bidirectional
- Quiescent Current = 1.0 nA/package typical @ 5.0 Vdc
- 10-MHz Operation (typical)
- 3-State Outputs
- Linear "On" Resistance
- "On" Resistance 120 Ohms typical @ 15 V
- Low Noise = 12 nV/ $\sqrt{\text{Cycle}}$, $f \geq 1 \text{ kHz}$ typical
- Supply Voltage Range = 3.0 Vdc to 18 Vdc
- Capable of Driving Two Low-power TTL Loads, One Low-power Schottky TTL Load or Two HTL Loads Over the Rated Temperature Range

MAXIMUM RATINGS (Voltages referenced to V_{SS})

Rating	Symbol	Value	Unit
DC Supply Voltage	V _{DD}	-0.5 to +18	Vdc
Input Voltage, All Inputs	V _{in}	-0.5 to V _{DD} + 0.5	Vdc
DC Current Drain per Pin	I	10	mAdc
Operating Temperature Range ... AL Device CL/CP Device	T _A	-55 to +125 -40 to +85	°C
Storage Temperature Range	T _{stg}	-65 to +150	°C

TRUTH TABLE

ST _X	ST _Y	B	A	Z	W
1	1	0	0	X0	Y0
1	1	0	1	X1	Y1
1	1	1	0	X2	Y2
1	1	1	1	X3	Y3
1	0	0	0	X0	
1	0	0	1	X1	
1	0	1	0	X2	
1	0	1	1	X3	
0	1	0	0	Y0	
0	1	0	1	Y1	
0	1	1	0	Y2	
0	1	1	1	Y3	
0	0	φ	φ	High impedance	

φ = Don't Care

Dual 4 Channel Mode
2 Outputs

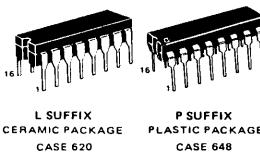
Single 8 Channel Mode
1 Output
(Z and W tied together)

This device contains circuitry to protect the control inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high-impedance circuit. A destructive high-current mode may occur if V_{in} or V_{out} is not constrained to the range V_{SS} ≤ V_{in} or V_{out} ≤ V_{DD}.

CMOS SSI

(LOW POWER COMPLEMENTARY MOS)

DUAL 4-CHANNEL ANALOG
DATA SELECTOR
OR
8-CHANNEL ANALOG
DATA SELECTOR



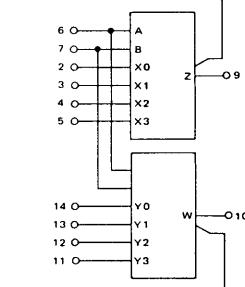
ORDERING INFORMATION

MC14XXXB	Suffix Denotes
	L Ceramic Package
	P Plastic Package
	A Extended Operating Temperature Range
	C Limited Operating Temperature Range

BLOCK DIAGRAM

3 State Output Enable

Strobe X 1 O —



Strobe Y 15 O —

V_{DD} = Pin 16

V_{SS} = Pin 8

MC14529B

ELECTRICAL CHARACTERISTICS

Characteristic	Figure	Symbol	V _{SS} Vdc	V _{DD} Vdc	T _{low} *		25°C			T _{high} *		Unit
					Min	Max	Min	Typ	Max	Min	Max	
Output Voltage "0" Level V _{in} = V _{DD} or 0	1	V _{OL}	0.0	5.0	—	0.05	—	0	0.05	—	0.05	Vdc
"1" Level V _{in} = 0 or V _{DD}		V _{OH}	0.0	5.0	4.95	—	4.95	5.0	—	4.95	—	Vdc
				10	9.95	—	9.95	10	—	9.95	—	Vdc
				15	14.95	—	14.95	15	—	14.95	—	Vdc
Input Voltage # (V _O = 4.5 or 5.0 Vdc) (V _O = 9.0 or 10 Vdc) (V _O = 13.5 or 15 Vdc) (V _O = 0.5 or 4.5 Vdc) (V _O = 1.0 or 9.0 Vdc) (V _O = 1.5 or 13.5 Vdc)	2	V _{IL}	0.0	5.0	—	1.5	—	2.25	1.5	—	1.5	Vdc
"0" Level				10	—	3.0	—	4.50	3.0	—	3.0	Vdc
"1" Level				15	—	4.0	—	6.75	4.0	—	4.0	Vdc
				5.0	3.5	—	3.5	2.75	—	3.5	—	Vdc
				10	7.0	—	7.0	5.50	—	7.0	—	Vdc
				15	11	—	11	8.25	—	11	—	Vdc
Input Current (AL Device) Control		I _{IN}	0.0	15	—	±0.1	—	+0.00001	±0.1	—	±1.0	μA/dc
Input Current (CL/CP Device) Control		I _{IN}	0.0	15	—	±0.3	—	-0.00001	±0.3	—	±1.0	μA/dc
Input Capacitance (V _{in} = 0) Control Switch Input Switch Output Feed Through		C _{IN}	0.0	—	—	—	—	5.0	7.5	—	—	pF
Quiescent Current (AL Device) (Per Package)	3	I _{DD}	—	5.0	—	1.0	—	0.001	1.0	—	60	μA/dc
				10	—	1.0	—	0.002	1.0	—	60	
				15	—	2.0	—	0.003	2.0	—	120	
Quiescent Current (CL/CP Device) (Per Package)	3	I _{DD}	—	5.0	—	5.0	—	0.001	5.0	—	70	μA/dc
				10	—	5.0	—	0.002	5.0	—	70	
				15	—	10	—	0.003	10	—	140	
"ON" Resistance (AL Device) (V _C = V _{DD} , R _L = 10 kΩ) (V _{in} = +5.0 Vdc) (V _{in} = -5.0 Vdc) (V _{in} = ±0.25 Vdc) (V _{in} = +7.5 Vdc) (V _{in} = -7.5 Vdc) (V _{in} = ±0.25 Vdc) (V _{in} = +10 Vdc) (V _{in} = ±0.25 Vdc) (V _{in} = +5.6 Vdc) (V _{in} = +15 Vdc) (V _{in} = ±0.25 Vdc) (V _{in} = +9.3 Vdc)	4,5,6	R _{ON}										Ohms
				-5.0	5.0	—	400	—	200	480	—	640
				—	—	—	400	—	200	480	—	640
				—	—	—	400	—	190	480	—	640
				-7.5	7.5	—	240	—	160	270	—	400
				—	—	—	240	—	160	270	—	400
				0	10	—	240	—	120	270	—	400
				—	—	—	400	—	180	480	—	640
				—	—	—	400	—	180	480	—	640
				0	15	—	250	—	180	270	—	400
				—	—	—	250	—	180	270	—	400
				—	—	—	250	—	215	270	—	400
"ON" Resistance (CL/CP Device) (V _C = V _{DD} , R _L = 10 kΩ) (V _{in} = +5.0 Vdc) (V _{in} = -5.0 Vdc) (V _{in} = ±0.25 Vdc) (V _{in} = +7.5 Vdc) (V _{in} = -7.5 Vdc) (V _{in} = ±0.25 Vdc) (V _{in} = +10 Vdc) (V _{in} = ±0.25 Vdc) (V _{in} = +5.6 Vdc) (V _{in} = +15 Vdc) (V _{in} = ±0.25 Vdc) (V _{in} = +9.3 Vdc)	4,5,6	R _{ON}										Ohms
				-5.0	5.0	—	410	—	200	480	—	560
				—	—	—	410	—	200	480	—	560
				-7.5	7.5	—	410	—	190	480	—	560
				—	—	—	250	—	160	270	—	350
				—	—	—	250	—	160	270	—	350
				0	10	—	250	—	120	270	—	350
				—	—	—	410	—	180	480	—	560
				—	—	—	410	—	180	480	—	560
				0	15	—	260	—	180	270	—	350
				—	—	—	250	—	270	—	—	350
				—	—	—	250	—	215	270	—	350
△"ON" Resistance Between any 2 circuits in a common package (V _{in} = ±5.0 Vdc) (V _{in} = ±7.5 Vdc)	—	△R _{ON}										Ohms
				-5.0	5.0	—	—	—	15	—	—	
				-7.5	7.5	—	—	—	10	—	—	

*T_{low} = -55°C for AL Device, -40°C for CL/CP Device.

*T_{high} = +125°C for AL Device, +85°C for CL/CP Device.

#Noise immunity specified for worst-case input combination.

Noise Margin for both "1" and "0" level = 1.0 Vdc min @ V_{DD} = 5.0 Vdc

2.0 Vdc min @ V_{DD} = 10 Vdc

2.5 Vdc min @ V_{DD} = 15 Vdc

MC14529B

SWITCHING CHARACTERISTICS ($T_A = 25^\circ\text{C}$)

Characteristic	Figure	Symbol	V_{SS}	V_{DD}	Typical All Types	Maximum		Unit
						AL Device	CL/CP Device	
V_{in} to V_{out} , Propagation Delay Time ($C_L = 50 \text{ pF}$, $R_L = 1.0 \text{ k}\Omega$)	7	t_{PLH}, t_{PHL}	0.0	5.0 10 15	20 10 8.0	40 20 15	60 30 25	ns
Propagation Delay Time, Control to Output, $V_{in} = V_{DD}$ or V_{SS} ($V_{in} < 10 \text{ Vdc}$, $C_L = 50 \text{ pF}$, $R_L = 1.0 \text{ k}\Omega$)	8	t_{PHL}, t_{PLH}	0.0	5.0 10 15	200 80 50	400 160 120	600 240 180	ns
Crosstalk, Control to Output ($C_L = 50 \text{ pF}$, $R_L = 1.0 \text{ k}\Omega$) $P_{out} = 10 \text{ mW}$	9	—	0.0	5.0 10 15	5.0 5.0 5.0	— — —	— — —	mV
Maximum Control Input Pulse Frequency ($C_L = 50 \text{ pF}$, $R_L = 1.0 \text{ k}\Omega$)	10	—	0.0	5.0 10 15 12	5.0 10 12 —	— — — —	— — — —	MHz
Noise Voltage ($f = 100 \text{ Hz}$)	11,12	—	0.0	5.0 10 15 5.0 10 15	24 25 30 12 12 15	— — — — — —	— — — — — —	nV/ $\sqrt{\text{Cycle}}$
Sine Wave (Distortion) ($V_{in} = 1.77 \text{ Vdc RMS}$ Centered @ 0.0 Vdc, $R_L = 10 \text{ k}\Omega$, $f = 1.0 \text{ kHz}$)	—	—	-5.0	5.0	0.36	—	—	%
Input/Output Leakage Current	—	—		-5.0 -5.0 -7.5 -7.5	5.0 5.0 7.5 7.5	± 0.001 ± 0.001 ± 0.0015 ± 0.0015	± 125 ± 125 ± 250 ± 250	mA
Insertion Loss ($V_{in} = 1.77 \text{ Vdc}$ RMS centered @ 0.0 Vdc, $f = 1.0 \text{ MHz}$, $I_{loss} = 20 \log_{10} \frac{V_{out}}{V_{in}}$ ($R_L = 1.0 \text{ k}\Omega$) ($R_L = 10 \text{ k}\Omega$) ($R_L = 100 \text{ k}\Omega$) ($R_L = 1.0 \text{ M}\Omega$)	—	—	-5.0	5.0		2.0 0.8 0.25 0.01	— — — —	dB
Bandwidth (-3 dB) ($V_{in} = 1.77 \text{ Vdc}$ RMS centered @ 0.0 Vdc) ($R_L = 1.0 \text{ k}\Omega$) ($R_L = 10 \text{ k}\Omega$) ($R_L = 100 \text{ k}\Omega$) ($R_L = 1.0 \text{ M}\Omega$)	—	BW	-5.0	5.0		35 28 27 26	— — — —	MHz
Feedthrough and Crosstalk ($20 \log_{10} \frac{V_{out}}{V_{in}} = -50 \text{ dB}$) ($R_L = 1.0 \text{ k}\Omega$) ($R_L = 10 \text{ k}\Omega$) ($R_L = 100 \text{ k}\Omega$) ($R_L = 1.0 \text{ M}\Omega$)	—	—	-5.0	5.0		850 100 12 1.5	— — — —	kHz

MC14529B

FIGURE 1 - OUTPUT VOLTAGE TEST CIRCUIT

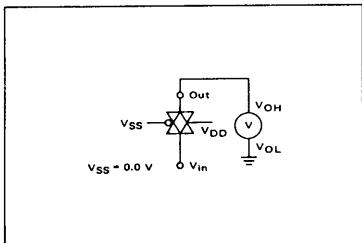


FIGURE 2 – NOISE IMMUNITY TEST CIRCUIT

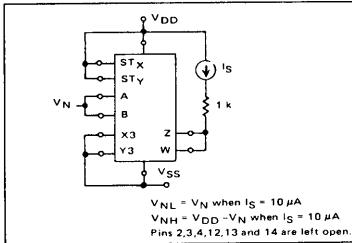


FIGURE 3 – QUIESCENT POWER DISSIPATION TEST CIRCUIT

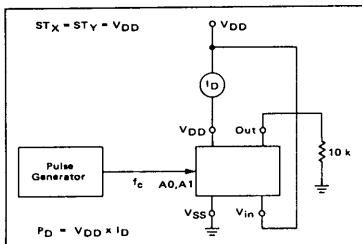
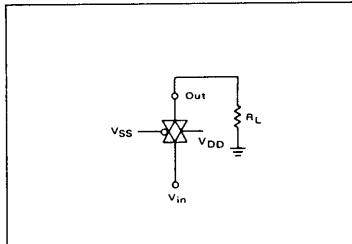


FIGURE 4 – R_{ON} CHARACTERISTICS TEST CIRCUIT



TYPICAL R_{ON} versus INPUT VOLTAGE

FIGURE 5

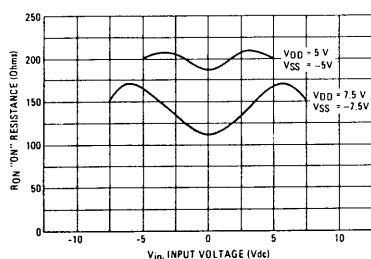
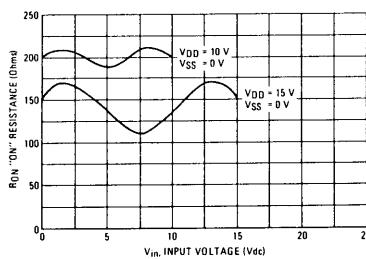


FIGURE 6



MC14529B

FIGURE 7 – PROPAGATION DELAY TEST CIRCUIT AND WAVEFORMS

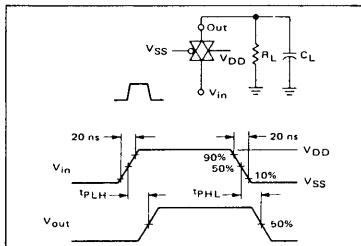


FIGURE 8 – TURN-ON DELAY TIME TEST CIRCUIT AND WAVEFORMS

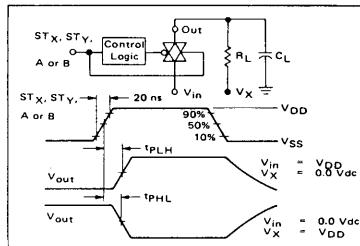


FIGURE 9 – CROSSTALK TEST CIRCUIT

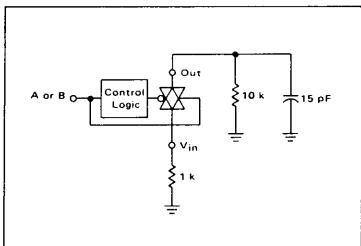


FIGURE 10 – FREQUENCY RESPONSE TEST CIRCUIT

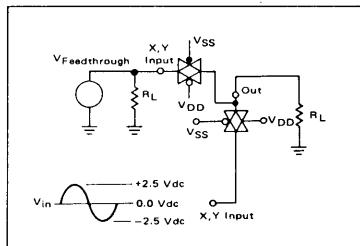


FIGURE 11 – NOISE VOLTAGE TEST CIRCUIT

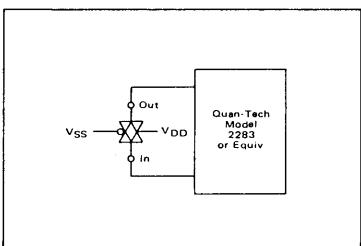
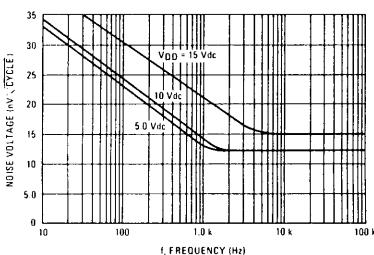
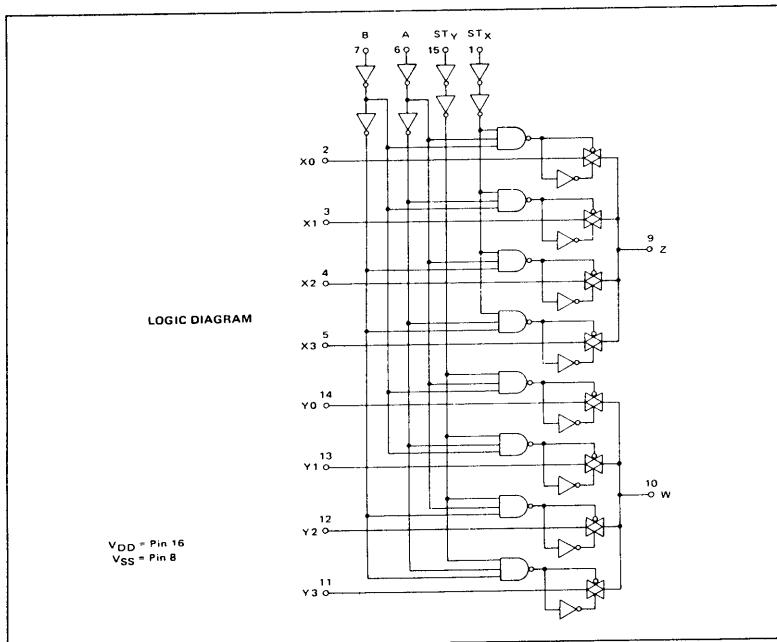
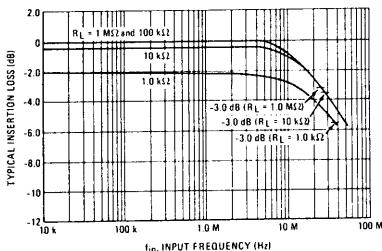


FIGURE 12 – TYPICAL NOISE CHARACTERISTICS



MC14529B

FIGURE 13 – TYPICAL INSERTION LOSS/BANDWIDTH CHARACTERISTICS





MOTOROLA SEMICONDUCTORS

PO. BOX 20912 • PHOENIX, ARIZONA 85036

MC1488

QUAD LINE DRIVER

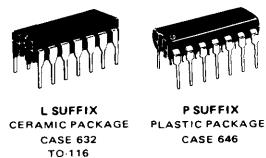
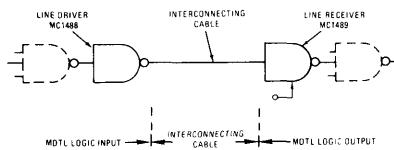
The MC1488 is a monolithic quad line driver designed to interface data terminal equipment with data communications equipment in conformance with the specifications of EIA Standard No. RS-232C.

Features:

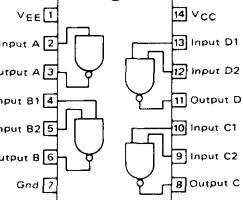
- Current Limited Output
±10 mA typ
- Power-Off Source Impedance
300 Ohms min
- Simple Slew Rate Control with External Capacitor
- Flexible Operating Supply Range
- Compatible with All Motorola MDTL and MTTL Logic Families

QUAD MDTL LINE DRIVER RS-232C SILICON MONOLITHIC INTEGRATED CIRCUIT

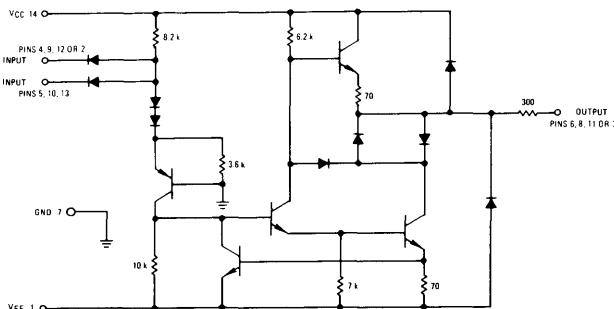
TYPICAL APPLICATION



PIN CONNECTIONS



CIRCUIT SCHEMATIC (1/4 OF CIRCUIT SHOWN)



MAXIMUM RATINGS ($T_A = +25^\circ\text{C}$ unless otherwise noted.)

Rating	Symbol	Value	Unit
Power Supply Voltage	V_{CC}	+15 -15	Vdc
Input Voltage Range	V_{IR}	$-15 \leq V_{IR} \leq 7.0$	Vdc
Output Signal Voltage	V_O	± 15	Vdc
Power Derating (Package Limitation, Ceramic and Plastic Dual-In-Line Package) Derate above $T_A = +25^\circ\text{C}$	P_D $1/R_{RJA}$	1000 6.7	mW mW/ $^\circ\text{C}$
Operating Ambient Temperature Range	T_A	0 to +75	$^\circ\text{C}$
Storage Temperature Range	T_{STG}	-65 to +175	$^\circ\text{C}$

ELECTRICAL CHARACTERISTICS ($V_{CC} = +9.0 \pm 1\%$ Vdc, $V_{EE} = -9.0 \pm 1\%$ Vdc, $T_A = 0$ to $+75^\circ\text{C}$ unless otherwise noted)

Characteristic	Figure	Symbol	Min	Typ	Max	Unit
Input Current - Low Logic State ($V_{IL} = 0$)	1	I_{IL}	-	1.0	1.6	mA
Input Current - High Logic State ($V_{IH} = 5.0$ V)	1	I_{IH}	-	-	10	μA
Output Voltage - High Logic State ($V_{IL} = 0.8$ Vdc, $R_L = 3.0 \text{ k}\Omega$, $V_{CC} = +9.0$ Vdc, $V_{EE} = -9.0$ Vdc)	2	V_{OH}	-	+6.0	+7.0	-
($V_{IL} = 0.8$ Vdc, $R_L = 3.0 \text{ k}\Omega$, $V_{CC} = +13.2$ Vdc, $V_{EE} = -13.2$ Vdc)			-	+9.0	+10.5	-
Output Voltage - Low Logic State ($V_{IH} = 1.9$ Vdc, $R_L = 3.0 \text{ k}\Omega$, $V_{CC} = +9.0$ Vdc, $V_{EE} = -9.0$ Vdc)	2	V_{OL}	-	-6.0	-7.0	-
($V_{IH} = 1.9$ Vdc, $R_L = 3.0 \text{ k}\Omega$, $V_{CC} = +13.2$ Vdc, $V_{EE} = -13.2$ Vdc)			-	-9.0	-10.5	-
Positive Output Short-Circuit Current (1)	3	I_{OS+}	-	+6.0	+10	mA
Negative Output Short-Circuit Current (1)	3	I_{OS-}	-	-6.0	-10	mA
Output Resistance ($ V_{CC} - V_{EE} = 0$, $ V_O = \pm 20$ V)	4	r_o	300	-	-	Ohms
Positive Supply Current ($R_L = \infty$) ($V_{IH} = 1.9$ Vdc, $V_{CC} = +9.0$ Vdc) ($V_{IL} = 0.8$ Vdc, $V_{CC} = +9.0$ Vdc) ($V_{IH} = 1.9$ Vdc, $V_{CC} = +12$ Vdc) ($V_{IL} = 0.8$ Vdc, $V_{CC} = +12$ Vdc) ($V_{IH} = 1.9$ Vdc, $V_{CC} = +15$ Vdc) ($V_{IL} = 0.8$ Vdc, $V_{CC} = +15$ Vdc)	5	I_{CC}	-	+15	+20	mA
($V_{IH} = 1.9$ Vdc, $V_{CC} = +9.0$ Vdc) ($V_{IL} = 0.8$ Vdc, $V_{CC} = -9.0$ Vdc) ($V_{IH} = 1.9$ Vdc, $V_{EE} = -12$ Vdc) ($V_{IL} = 0.8$ Vdc, $V_{EE} = -12$ Vdc) ($V_{IH} = 1.9$ Vdc, $V_{EE} = -15$ Vdc) ($V_{IL} = 0.8$ Vdc, $V_{EE} = -15$ Vdc)	5	I_{EE}	-	+4.5	+6.0	mA
($V_{IH} = 1.9$ Vdc, $V_{CC} = -9.0$ Vdc) ($V_{IL} = 0.8$ Vdc, $V_{CC} = -12$ Vdc) ($V_{IH} = 1.9$ Vdc, $V_{EE} = -15$ Vdc)	5	I_{EE}	-	+19	+25	mA
($V_{IL} = 0.8$ Vdc, $V_{CC} = +15$ Vdc) ($V_{IL} = 0.8$ Vdc, $V_{EE} = +15$ Vdc)	5	I_{EE}	-	+5.5	+7.0	mA
($V_{IL} = 0.8$ Vdc, $V_{CC} = -15$ Vdc) ($V_{IL} = 0.8$ Vdc, $V_{EE} = -15$ Vdc)	5	I_{EE}	-	+34	+42	mA
Power Consumption ($V_{CC} = 9.0$ Vdc, $V_{EE} = -9.0$ Vdc) ($V_{CC} = 12$ Vdc, $V_{EE} = -12$ Vdc)		P_C	-	-	333 576	mW

SWITCHING CHARACTERISTICS ($V_{CC} = +9.0 \pm 1\%$ Vdc, $V_{EE} = -9.0 \pm 1\%$ Vdc, $T_A = +25^\circ\text{C}$)

Propagation Delay Time ($z_1 = 3.0 \text{ k}$ and 15 pF)	6	t_{PLH}	-	275	350	ns
Fall Time ($z_1 = 3.0 \text{ k}$ and 15 pF)	6	t_{THL}	-	45	75	ns
Propagation Delay Time ($z_1 = 3.0 \text{ k}$ and 15 pF)	6	t_{PHL}	-	110	175	ns
Rise Time ($z_1 = 3.0 \text{ k}$ and 15 pF)	6	t_{TLH}	-	55	100	ns

(1) Maximum Package Power Dissipation may be exceeded if all outputs are shorted simultaneously.


MOTOROLA Semiconductor Products Inc.

MC1488

CHARACTERISTIC DEFINITIONS

FIGURE 1 — INPUT CURRENT

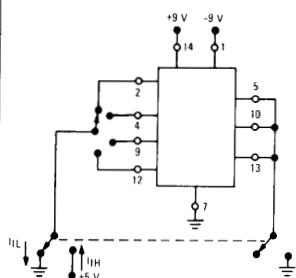


FIGURE 2 — OUTPUT VOLTAGE

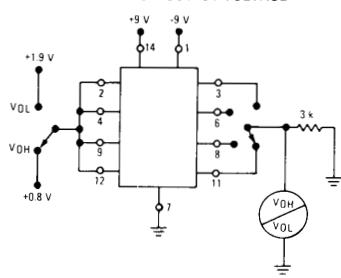


FIGURE 3 — OUTPUT SHORT-CIRCUIT CURRENT

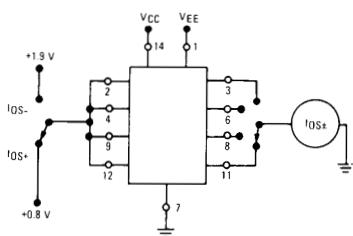


FIGURE 4 — OUTPUT RESISTANCE (POWER-OFF)

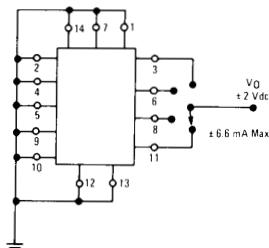


FIGURE 5 — POWER-SUPPLY CURRENTS

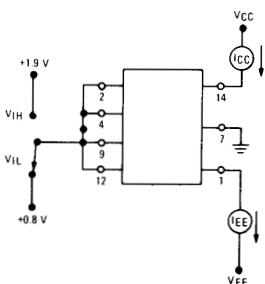
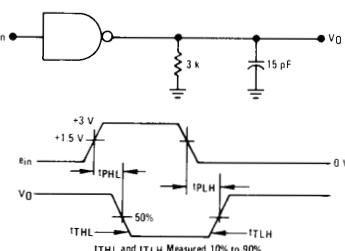
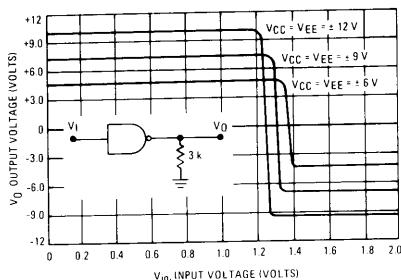


FIGURE 6 — SWITCHING RESPONSE

**MOTOROLA Semiconductor Products Inc.**

TYPICAL CHARACTERISTICS
($T_A = +25^\circ\text{C}$ unless otherwise noted.)

**FIGURE 7 – TRANSFER CHARACTERISTICS
versus POWER-SUPPLY VOLTAGE**



**FIGURE 8 – SHORT-CIRCUIT OUTPUT CURRENT
versus TEMPERATURE**

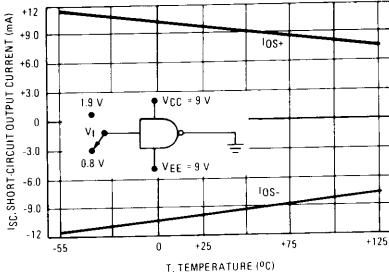
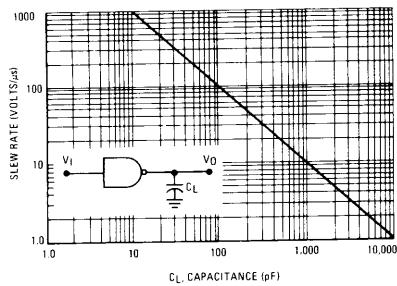
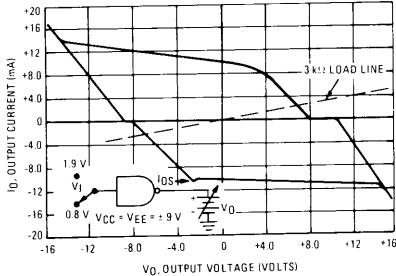


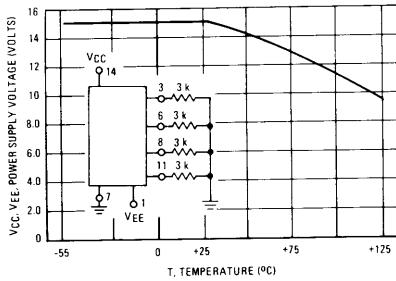
FIGURE 9 – OUTPUT SLEW RATE versus LOAD CAPACITANCE



**FIGURE 10 – OUTPUT VOLTAGE
AND CURRENT-LIMITING CHARACTERISTICS**



**FIGURE 11 – MAXIMUM OPERATING TEMPERATURE
versus POWER-SUPPLY VOLTAGE**



MOTOROLA Semiconductor Products Inc.

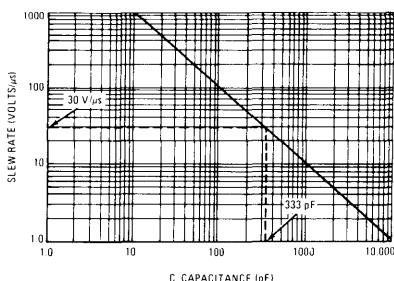
APPLICATIONS INFORMATION

The Electronic Industries Association (EIA) RS232C specification detail the requirements for the interface between data processing equipment and data communications equipment. This standard specifies not only the number and type of interface leads, but also the voltage levels to be used. The MC1488 quad driver and its companion circuit, the MC1489 quad receiver, provide a complete interface system between DTL or TTL logic levels and the RS232C defined levels. The RS232C requirements as applied to drivers are discussed herein.

The required driver voltages are defined as between 5 and 15-volts in magnitude and are positive for a logic "0" and negative for a logic "1". These voltages are so defined when the drivers are terminated with a 3000 to 7000-ohm resistor. The MC1488 meets this voltage requirement by converting a DTL/TTL logic level into RS232C levels with one stage of inversion.

The RS232C specification further requires that during transitions, the driver output slew rate must not exceed 30 volts per microsecond. The inherent slew rate of the MC1488 is much too

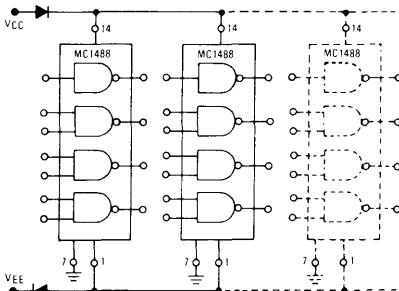
FIGURE 12 – SLEW RATE versus CAPACITANCE
FOR IS = 10 mA



fast for this requirement. The current limited output of the device can be used to control this slew rate by connecting a capacitor to each driver output. The required capacitor can be easily determined by using the relationship $C = \log_2 \frac{\Delta T}{\Delta V}$ from which Figure 12 is derived. Accordingly, a 330-pF capacitor on each output will guarantee a worst case slew rate of 30 volts per microsecond.

The interface driver is also required to withstand an accidental short to any other conductor in an interconnecting cable. The worst possible signal on any conductor would be another driver using a plus or minus 15-volt, 500-mA source. The MC1488 is designed to indefinitely withstand such a short to all four outputs in a package as long as the power-supply voltages are greater than 9.0 volts i.e., $V_{CC} > 9.0$ V, $V_{EE} < -9.0$ V. In some power-supply designs, a loss of system power causes a low impedance to ground would exist at the power inputs to the MC1488 effectively shorting the 300-ohm output resistors to ground. If all four outputs were then shorted to plus or minus 15 volts, the power dissipation in these resistors

FIGURE 13 – POWER-SUPPLY PROTECTION
TO MEET POWER-OFF FAULT CONDITIONS



would be excessive. Therefore, if the system is designed to permit low impedances to ground at the power-supplies of the drivers, a diode should be placed in each power-supply lead to prevent over heating in this fault condition. These two diodes, as shown in Figure 13, could be used to decouple all the driver packages in a system. (These same diodes will allow the MC1488 to withstand momentary shorts to the +25-volt limits specified in the earlier Standard RS232B.) The addition of the diodes also permits the MC1488 to withstand faults with power-supplies of less than the 9.0 volts stated above.

The maximum short-circuit current allowable under fault conditions is more than guaranteed by the previously mentioned 10 mA output current limiting.

Other Applications

The MC1488 is an extremely versatile line driver with a myriad of possible applications. Several features of the drivers enhance this versatility.

1. Output Current Limiting – this enables the circuit designer to define the output voltage levels independent of power supplies and can be accomplished by diode clamping of the output pins. Figure 14 shows the MC1488 used as a DTL to MOS translator where the high-level voltage output is clamped one diode above ground. The resistor divider shown is used to reduce the output voltage below the 300 mV above ground MOS input level limit.

2. Power Supply Range – as can be seen from the schematic drawing of the drivers, the positive and negative driving elements of the device are essentially independent and do not require matching power supplies. In fact, the positive supply can vary from a minimum seven volts (required for driving the negative pulldown section) to the maximum specified 15 volts. The negative supply can vary from approximately -2.5 volts to the minimum specified -15 volts. The MC1488 will drive the output to within 2 volts of the positive or negative supplies as long as the current output limits are not exceeded. The combination of the current-limiting and supply-voltage features allow a wide combination of possible outputs within the same quad package. Thus if only a portion of the four drivers are used for driving RS232C lines, the remainder could be used for DTL to MOS or even DTL to DTL translation. Figure 15 shows one such combination.



MOTOROLA Semiconductor Products Inc.

FIGURE 14 - MDTL/MTTL-TO-MOS TRANSLATOR

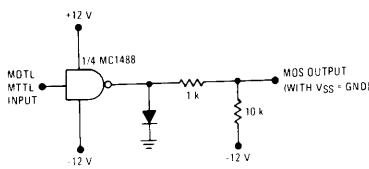
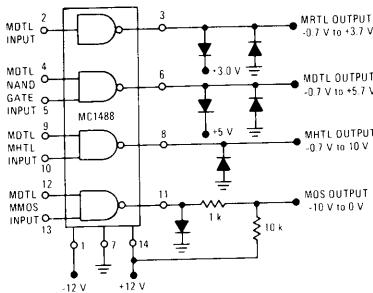


FIGURE 15 - LOGIC TRANSLATOR APPLICATIONS



THERMAL INFORMATION

The maximum power consumption an integrated circuit can tolerate at a given operating ambient temperature, can be found from the equation:

$$PD(T_A) = \frac{T_{J(max)} - T_A}{R_{\theta JA}(Typ)}$$

Where: $PD(T_A)$ = Power Dissipation allowable at a given operating ambient temperature. This must be greater than

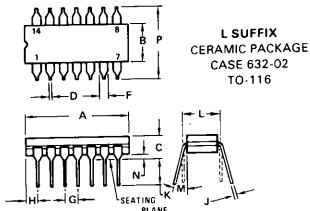
the sum of the products of the supply voltages and supply currents at the worst case operating condition.

$T_{J(max)}$ = Maximum Operating Junction Temperature as listed in the Maximum Ratings Section

T_A = Maximum Desired Operating Ambient Temperature

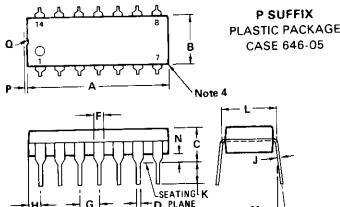
$R_{\theta JA}(Typ)$ = Typical Thermal Resistance Junction to Ambient

OUTLINE DIMENSIONS



NOTE:
1. DIMENSION "I" TO CENTER OF LEADS
WHEN FORMED PARALLEL

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	19.05	19.94	0.750	0.785
B	6.22	7.11	0.245	0.280
C	3.94	5.08	0.155	0.200
D	0.38	1.0	0.015	0.030
E	0.1	1.65	0.0035	0.065
F	—	—	—	—
G	2.54 BSC	—	0.100 BSC	—
H	1.65	2.29	0.065	0.090
J	0.20	0.30	0.008	0.012
K	3.18	4.0	0.125	0.160
L	7.37	8.13	0.290	0.320
M	—	1.55	—	1.55
N	0.51	1.27	0.020	0.050



DIM	MILLIMETERS		INCHES	
	MM	INCH	MM	INCH
A	18.16	0.715	0.715	0.770
B	6.10	0.240	0.240	0.260
C	6.10	0.240	0.160	0.200
D	0.38	0.015	0.019	0.021
E	1.02	0.040	0.040	0.070
F	2.54 BSC	0.100 BSC	—	—
G	1.32	0.052	0.052	0.055
H	0.28	0.011	0.013	0.015
J	0.28	0.011	0.015	0.015
K	2.92	0.343	0.115	0.135
L	1.69 BSC	0.300 BSC	—	—
M	0.51	0.020	0.020	0.040
N	1.02	0.040	0.040	0.050

- NOTES:
1. LEADS WITHIN 0.13 mm (0.005) RADIUS OF TRUE POSITION AT SEATING PLANE AT MAXIMUM MATERIAL CONDITION.
 2. NOTE 4 APPLIES TO CENTER OF LEADS WHEN FORMED PARALLEL.
 3. DIMENSION "B" DOES NOT INCLUDE MOLD FLASH.
 4. ROUNDED CORNERS OPTIONAL.

Motorola reserves the right to make changes to any products herein to improve reliability, function or design. Motorola does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others.



MOTOROLA Semiconductor Products Inc.

BOX 20912 • PHOENIX, ARIZONA 85036 • A SUBSIDIARY OF MOTOROLA INC.



MOTOROLA SEMICONDUCTORS

PO. BOX 20912 • PHOENIX, ARIZONA 85036

MC1489 MC1489A

QUAD LINE RECEIVERS

The MC1489 monolithic quad line receivers are designed to interface data terminal equipment with data communications equipment in conformance with the specifications of EIA Standard No. RS-232C.

- Input Resistance – 3.0 k to 7.0 kilohms
- Input Signal Range – ± 30 Volts
- Input Threshold Hysteresis Built In
- Response Control
 - a) Logic Threshold Shifting
 - b) Input Noise Filtering

QUAD MDTL LINE RECEIVERS RS-232C

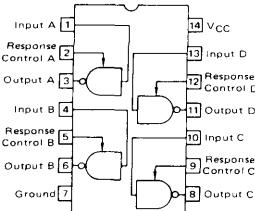
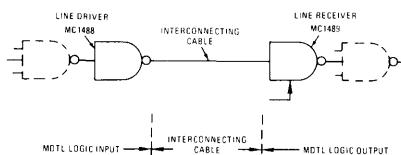
SILICON MONOLITHIC
INTEGRATED CIRCUIT



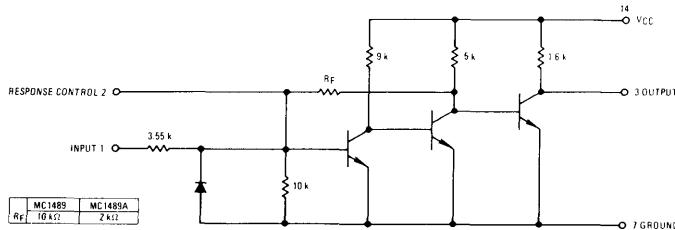
L SUFFIX
CERAMIC PACKAGE
CASE 632
TO-116

P SUFFIX
PLASTIC PACKAGE
CASE 646

TYPICAL APPLICATION



CIRCUIT SCHEMATIC (1/4 OF CIRCUIT SHOWN)



MC1489 MC1489A

R_F

10 kΩ

2 kΩ

MC1489 • MC1489AMAXIMUM RATINGS ($T_A = +25^\circ\text{C}$ unless otherwise noted)

Rating	Symbol	Value	Unit
Power Supply Voltage	V _{CC}	10	Vdc
Input Voltage Range	V _{IR}	±30	Vdc
Output Load Current	I _L	20	mA
Power Dissipation (Package Limitation, Ceramic and Plastic Dual In-Line Package) Derate above $T_A = +25^\circ\text{C}$	P _D 1/ θ_{JA}	1000 6.7	mW mW/ $^\circ\text{C}$
Operating Ambient Temperature Range	T _A	0 to +75	$^\circ\text{C}$
Storage Temperature Range	T _{stg}	-65 to +175	$^\circ\text{C}$

ELECTRICAL CHARACTERISTICS (Response control pin is open) (V_{CC} = +5.0 Vdc ± 1%, $T_A = 0$ to +75°C unless otherwise noted)

Characteristics	Figure	Symbol	Min	Typ	Max	Unit	
Positive Input Current (V _{IH} = +25 Vdc) (V _{IL} = -3.0 Vdc)	1	I _{IH}	3.6 0.43		8.3	mA	
Negative Input Current (V _{IL} = -25 Vdc) (V _{IH} = -3.0 Vdc)	1	I _{IL}	-3.6 -0.43		-8.3	mA	
Input Turn-On Threshold Voltage ($T_A = +25^\circ\text{C}$, V _{OL} ≤ 0.45 V) MC1489 MC1489A	2	V _{IHL}	1.0 1.75	1.95	1.5 2.25	Vdc	
Input Turn-Off Threshold Voltage ($T_A = +25^\circ\text{C}$, V _{OH} ≥ 2.5 V, I _L = -0.5 mA) MC1489 MC1489A	2	V _{ILH}	0.75 0.75	0.8	1.25 1.25	Vdc	
Output Voltage High (V _{IH} = 0.75 V, I _L = -0.5 mA) (Input Open Circuit, I _L = -0.5 mA)	2	V _{OH}	2.6 2.6	4.0 4.0	5.0 5.0	Vdc	
Output Voltage Low (V _{IL} = 3.0 V, I _L = 10 mA)	2	V _{OL}			0.2	0.45	Vdc
Output Short Circuit Current	3	I _{OS}			3.0		mA
Power Supply Current (V _{IH} = +5.0 Vdc)	4	I _{CC}			20	26	mA
Power Consumption (V _{IH} = +5.0 Vdc)	4	P _C			100	130	mW

SWITCHING CHARACTERISTICS (V_{CC} = 5.0 Vdc ± 1%, $T_A = +25^\circ\text{C}$)

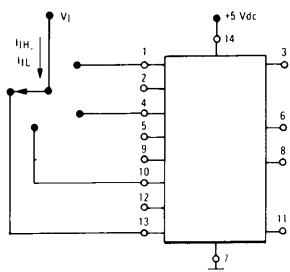
Propagation Delay Time (R _L = 3.9 kΩ)	5	t _{PLH}		25	85	ns
Rise Time (R _L = 3.9 kΩ)	5	t _{TLH}		120	175	ns
Propagation Delay Time (R _L = 390 Ω)	5	t _{PHL}		25	50	ns
Fall Time (R _L = 390 Ω)	5	t _{THL}		10	20	ns

**MOTOROLA Semiconductor Products Inc.**

MC1489 • MC1489A

TEST CIRCUITS

FIGURE 1 – INPUT CURRENT



**FIGURE 2 – OUTPUT VOLTAGE
and INPUT THRESHOLD VOLTAGE**

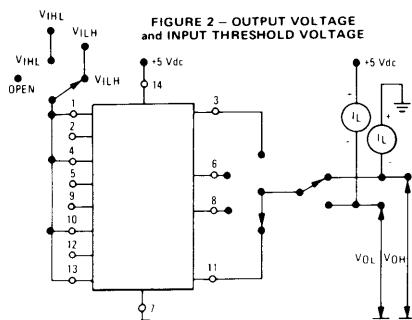


FIGURE 3 – OUTPUT SHORT CIRCUIT CURRENT

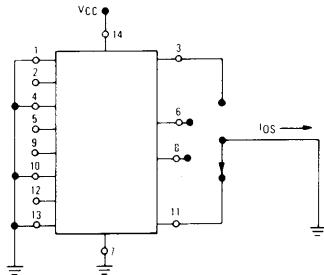


FIGURE 4 – POWER SUPPLY CURRENT

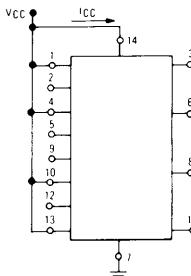


FIGURE 5 – SWITCHING RESPONSE

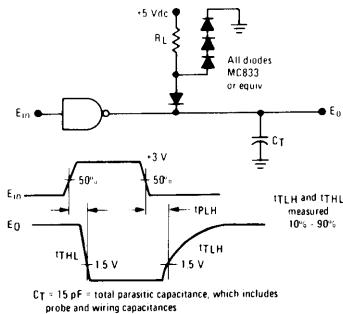
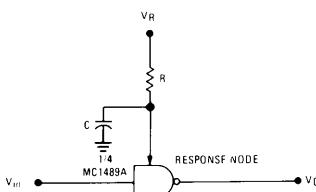


FIGURE 6 – RESPONSE CONTROL NODE



C, capacitor is for noise filtering.
R, resistor is for threshold shifting.



MOTOROLA Semiconductor Products Inc.

MC1489 • MC1489A

TYPICAL CHARACTERISTICS

($V_{CC} = 5.0$ Vdc, $T_A = +25^\circ\text{C}$ unless otherwise noted)

FIGURE 7 – INPUT CURRENT

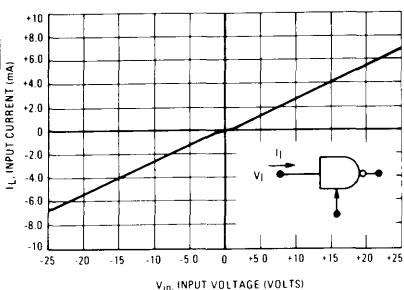


FIGURE 8 – MC1489 INPUT THRESHOLD VOLTAGE ADJUSTMENT

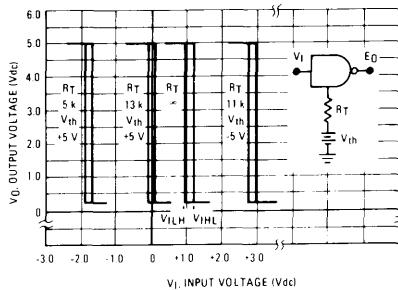


FIGURE 9 – MC1489A INPUT THRESHOLD VOLTAGE ADJUSTMENT

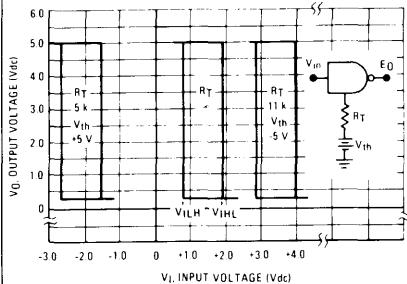


FIGURE 10 – INPUT THRESHOLD VOLTAGE versus TEMPERATURE

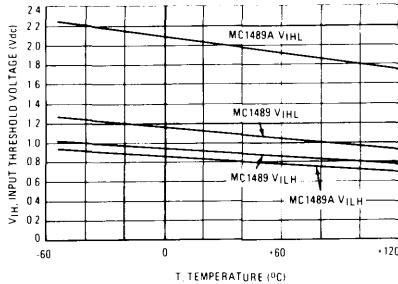
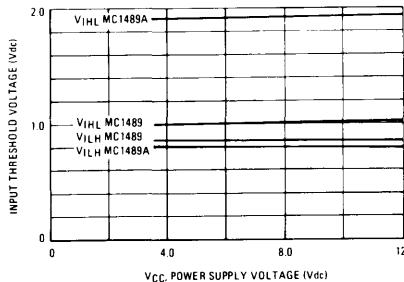


FIGURE 11 – INPUT THRESHOLD versus POWER-SUPPLY VOLTAGE



MOTOROLA Semiconductor Products Inc.

APPLICATIONS INFORMATION

General Information

The Electronic Industries Association (EIA) has released the RS-232C specification detailing the requirements for the interface between data processing equipment and data communications equipment. This standard specifies not only the number and type of interface leads, but also the voltage levels to be used. The MC1489 quad driver and its companion circuit, the MC1489 quad receiver, provide a complete interface system between DTL or TTL logic levels and the RS-232C defined levels. The RS-232C requirements as applied to receivers are discussed herein.

The required input impedance is defined as between 3000 ohms and 7000 ohms for input voltages between +3.0 and -25 volts in magnitude; and any voltage on the receiver input in an open circuit condition must be less than 2.0 volts in magnitude. The MC1489 circuits meet these requirements with a maximum open circuit voltage of one V_{GG} (Ref. Sect. 2.4).

The receiver shall detect a voltage between -3.0 and -25 volts as a logic "1" and inputs between +3.0 and +25 volts as a logic "0" (Ref. Sect. 2.3). On some interchanges leads, an open circuit or power ("OFF") condition (300 ohms or more to ground) shall be decoded as an "OFF" condition or logic "1" (Ref. Sect. 2.5). For this reason, the input hysteresis thresholds of the MC1489 circuits are all above ground. Thus an open or grounded input will cause the same output as a negative or logic "1" input.

Device Characteristics

The MC1489 interface receivers have internal feedback from the second stage to the input stage providing input hysteresis for noise

rejection. The MC1489 input has typical turn-on voltage of 1.25 volts and turn-off of 1.0 volt for a typical hysteresis of 250 mV. The MC1489A has typical turn-on of 1.95 volts and turn-off of 0.8 volt for typically 1.15 volts of hysteresis.

Each receiver section has an external response control node in addition to the input and output pins, thereby allowing the designer to vary the input threshold voltage levels. A resistor can be connected between this node and an external power-supply. Figures 6, 8 and 9 illustrate the input threshold voltage shift possible through this technique.

This response node can also be used for the filtering of high-frequency, high-energy noise pulses. Figures 12 and 13 show typical noise-pulse rejection for external capacitors of various sizes.

These two operations on the response node can be combined or used individually for many combinations of interfacing applications. The MC1489 circuits are particularly useful for interfacing between MOS circuits and MDTL/MTTL logic systems. In this application, the input threshold voltages are adjusted (with the appropriate supply and resistor values) to fall in the center of the MOS voltage logic levels. (See Figure 14.)

The response node may also be used as the receiver input as long as the designer realizes that he may not drive this node with a low impedance source to a voltage greater than one diode above ground or less than one diode below ground. This feature is demonstrated in Figure 15 where two receivers are slaved to the same line that must still meet the RS-232C impedance requirement.

FIGURE 12 – TURN-ON THRESHOLD versus CAPACITANCE FROM RESPONSE CONTROL PIN TO GND

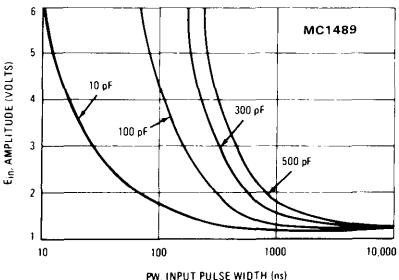


FIGURE 13 – TURN-ON THRESHOLD versus CAPACITANCE FROM RESPONSE CONTROL PIN TO GND

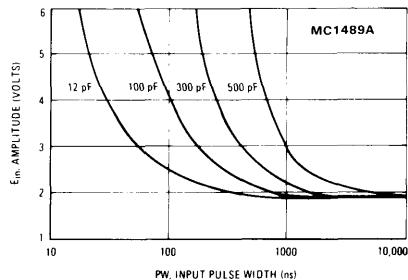
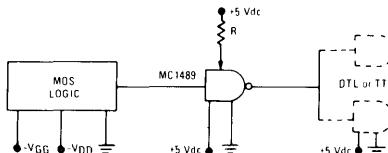


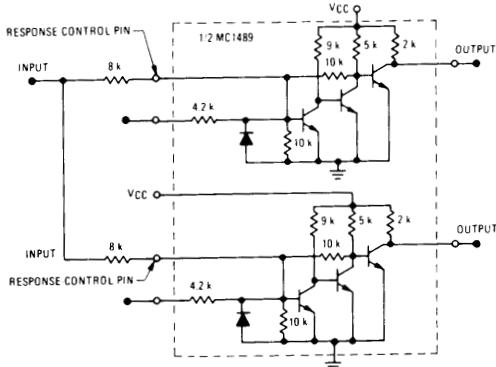
FIGURE 14 – TYPICAL TRANSLATOR APPLICATION – MOS TO DTL OR TTL



MOTOROLA Semiconductor Products Inc.

APPLICATIONS INFORMATION (continued)

FIGURE 15 – TYPICAL PARALLELING OF TWO MC1489.A RECEIVERS TO MEET RS-232C



THERMAL INFORMATION

The maximum power consumption an integrated circuit can tolerate at a given operating ambient temperature, can be found from the equation:

$$PD(T_A) = \frac{T_{J(max)} - T_A}{R_{0JA}(Typ)}$$

Where: $PD(T_A)$ = Power Dissipation allowable at a given operating ambient temperature. This must be greater than

the sum of the products of the supply voltages and supply currents at the worst case operating condition.

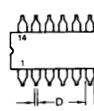
$T_{J(max)}$ = Maximum Operating Junction Temperature as listed in the Maximum Ratings Section

T_A = Maximum Desired Operating Ambient Temperature

$R_{0JA}(Typ)$ = Typical Thermal Resistance Junction to Ambient

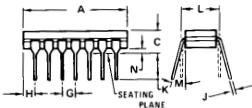
OUTLINE DIMENSIONS

L SUFFIX
CERAMIC PACKAGE
CASE 632-05
TO-116



NOTE: 1. DIMENSION "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.

	MILLIMETERS	INCHES		
DIM	MIN	MAX	MIN	MAX
A	19.05	19.94	0.750	0.785
B	8.25	8.75	0.325	0.340
C	3.94	5.08	0.155	0.200
D	0.38	0.51	0.015	0.020
E	0.89	1.65	0.035	0.065
F	2.54 BSC		0.100 BSC	
G	1.61	2.29	0.065	0.090
J	0.26	0.30	0.008	0.012
K	3.18	4.06	0.125	0.160
L	7.37	8.13	0.290	0.320
M	1.69	1.95	0.065	0.070
N	0.51	1.27	0.020	0.050

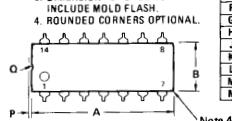


NOTES: 1. LEADS WITHIN 0.13 mm
0.005 RADIUS OF TRUE
POSITION AT SEATING
PLANE AT MAXIMUM
MATERIAL CONDITION.

2. DIMENSION "L" TO
CENTER OF LEADS
WHEN FORMED
PARALLEL.

3. DIMENSION "E" DOES NOT
INCLUDE MOLD FLASH.

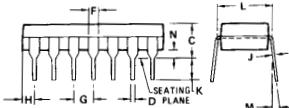
4. ROUNDED CORNERS OPTIONAL.



P SUFFIX
PLASTIC PACKAGE
CASE 646-05

	MILLIMETERS	INCHES		
DIM	MIN	MAX	MIN	MAX
A	18.16	19.56	0.715	0.770
B	6.10	6.60	0.240	0.260
C	4.06	5.08	0.160	0.200
D	0.26	0.30	0.010	0.012
F	1.02	1.78	0.040	0.070
G	2.54 BSC		0.100 BSC	
H	1.32	2.41	0.052	0.095
J	0.20	0.38	0.006	0.018
K	2.92	3.43	0.115	0.135
L	7.62 BSC		0.300 BSC	
M	0°	180°	0°	10°
N	0.51	1.02	0.020	0.040

Note 4



Motorola reserves the right to make changes to any products herein to improve reliability, function or design. Motorola does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others.



MOTOROLA Semiconductor Products Inc.

BOX 20912 • PHOENIX, ARIZONA 85036 • A SUBSIDIARY OF MOTOROLA INC.

TANDY COMPUTER PRODUCTS

I/O Address Decode Chip (U19)

```

Partno      8075173 ;
Name        iflrfi ;
Date        06/05/87 ;
Rev         01 ;
Designer    Company Tandy ;
Assembly   Location U19 ;

/*************************************************/
/*
/* This device performs I/O address decode of the FDC, Printer, */
/* DMA Page Registers, and Mode Register on Project 840. */
/* Modified version for RFI to disable Xbus */
/*************************************************/
/* Allowable Target Device Types: PLS173 */
/*************************************************/

/** Inputs **/


PIN [1..6] = [SA4..9] ;      /* Perpherial Address Lines 5-9 */
PIN 7     = !MEMIOS ;       /* Memory/IO Select */
PIN 8     = !ROMCS ;       /* Bios Rom Chip Selct */
PIN 9     = !INTA ;        /* Interrupt Acknowledge */
PIN 10    = !CPUHLDA ;     /* CPU Hold Acknowledge */
PIN 11    = !FDCTDACK ;    /* FDC DMA Acknowledge */
PIN 13    = !MEMR ;        /* Memory Read Signal */
PIN 14    = !IOR ;         /* I/O Read Signal */
PIN 15    = !FDCEN ;       /* FDC Port Enable */
PIN 16    = !SPSEL ;       /* Serial Port Select */
PIN 17    = !SPEN ;        /* Serial Port Enable */
PIN 23    = SA3 ;          /* Perpherial Address Line 3 */

/** Outputs **/


PIN 18    = !IRQ3EN ;      /* Interrupt Request 3 Enable */
PIN 19    = !IRQ4EN ;      /* Interrupt Request 4 Enable */
PIN 20    = !XBUFEN ;      /* Perpherial Bus Enable */
PIN 21    = !XBUFDIR ;    /* Perpherial Bus Direction */
PIN 22    = !SPCS ;        /* Serial Port Chip Select */

/** Declarations and Intermediate Variable Definitions **/


FIELD PORT = [SA9..3] ;

```

```

/** Logic Equations **/


SPCS      =      PORT:[2F8..2FF] & !CPUHLDA & SPEN & !SPSEL
#           PORT:[3F8..3FF] & !CPUHLDA & SPEN & SPSEL ;

XBUFDIR  =      PORT:[000..00F] & IOR & !CPUHLDA
#           PORT:[020..027] & IOR & !CPUHLDA
#           PORT:[040..047] & IOR & !CPUHLDA
#           PORT:[060..06F] & IOR & !CPUHLDA
#           PORT:[0A0..0A7] & IOR & !CPUHLDA
#           PORT:[0C0..0C7] & IOR & !CPUHLDA
#           PORT:[200..207] & IOR & !CPUHLDA
#           PORT:[2F8..2FF] & IOR & !CPUHLDA & SPEN & !SPSEL
#           PORT:[378..37F] & IOR & !CPUHLDA
#           PORT:[3D0..3DF] & IOR & !CPUHLDA
#           PORT:[3F0..3F7] & IOR & !CPUHLDA & FDCEN
#           PORT:[3F8..3FF] & IOR & !CPUHLDA & SPEN & SPSEL
#           MEMIOS & MEMR & !ROMCS
#           !MEMR & IIOR & CPUHLDA
#           FDCDACK & IOR & CPUHLDA & FDCEN
#           INTA ;

XBUFEN   =      PORT:[000..00F] & !CPUHLDA
#           PORT:[020..027] & !CPUHLDA
#           PORT:[060..06F] & !CPUHLDA
#           PORT:[080..08F] & !CPUHLDA
#           PORT:[2F8..2FF] & !CPUHLDA & SPEN & !SPSEL
#           PORT:[3F8..3FF] & !CPUHLDA & SPEN & SPSEL
#           MEMIOS & !ROMCS
#           CPUHLDA
#           INTA ;

IRQ4EN   =      SPEN & SPSEL ;
IRQ3EN   =      SPEN & !SPSEL ;

```

DRAM/DMA CONTROL CHIP

TABLE OF CONTENTS

1. Functional Descriptions
2. Memory Configuration
3. DMA Control Signal Equation
4. Block Diagram
5. Pin Configuration
6. Pin Number Assignment
7. Timing Specification
8. Timing Diagram
9. Electrical Specification

Tandy 1000TX DRAM/DMA Control IC consists of the following functional blocks:

- ROM/DRAM Decode Latch and Control signals
- Page Registers
- Memory Address Multiplexer

ROM/DRAM Decode Latch and Control signal

The ROM/DRAM Decode Latch block contains the circuitry to decode the CPU's Address bus A17 - A23 and to provide the necessary latched signals for controlling ROMS and DRAMS. CPU's Address A17 - A23 together with MCO - MC2 determine which segment (bank) of memory is being selected based on one of six possible memory configurations. (see memory map figure 1.). The memory configurations ranging from 256 kilo bytes to 1.5 mega bytes.

Additional support for memory refresh is also provided in this block. During a Refresh Cycle, the assertion of REFRESH# will cause the circuitry to ignore the current address inputs and activate RAS0#, RAS1#, RAS2#, RAS3# and LMEGCS# outputs. Also MEMCYC and PRCLK signals are used for controlling the start of a memory access and timing for all RASx#s, MA0 - MA8 and CASx#s signals. (see Timing Diagram figure 2.).

RDMCS# is decoded from A17 - A23 and latched by ALE when HLDA is active. The ROM address ranges from 0E0000h to 0FFFFFh for the low address, EE0000h to EFFFFFh and FE0000h to FFFFFFh for high address. This output signal is asserted when any one of the three address ranges is detected and REFRESH# is inactive.

The two outputs LMEGCS# and MDBEN# are intended to be used as memory buffer enable signal. LMEGCS# is active whenever any memory access is made to an address below 010000h or when REFRESH# is active. MDBEN# is memory data bus buffer and becomes active whenever CASx#s or RDMCS# is active.

The ROM/DRAM Decode's internal latch is controlled by two input signals - HLDA and ALE. During a CPU Memory Cycle, ALE will enable the RAM Decode latch and allow the input from decoder to be transferred to the output pins. When ALE goes inactive, the decoder outputs is latched for the remainder of the cycle. Asserting HLDA will enable the decoder outputs to the output pins and force RDMCS# inactive. This block has one output signal that is unlatched. This output, AF1E#, is intended to be used by external circuitry as an indication that a 16-bit memory transfer is taking place.

Page Register

At the time the 82C37A-S - DMA Controller takes control of the address bus, the first operation comes in two bytes. The first byte is a lower address bus (SA0 through SA7) that is put directly on the S address bus by DMA controller. The second byte is a upper address (SA8 through SA15) that is on its data outputs, to be latched in the 74ALS373 internally by Address Strobe (AS) signal from DMA Controller.

Two 4 by 4 registers are used to perform Page Register function. During DMA Bus Cycle, the read function is controlled by the DMA Request Acknowledge signals - DACK2# and DACK3# in conjunction with HLDA#. enables the Page Register to be output as the upper address (SA16 and A17 through A23). The write function is controlled by SA0 to SA3 in conjunction with PGREGWR# to latch data bits - XD0 to XD7 into Page Register.

Address Multiplexer

The Memory Address Multiplexer is used to provide the Row Address or Column Address and refresh counter that is required by Dynamic RAMs. Additionally, it provides the drive and buffering capability for memory address bus MA0 to MA8. MA7 is generated from SA0 or SA8 which is multiplexed by REFRESH# signal. The addresses for the memory are multiplexed as shown below.

Equation For Multiplexed Memory Addresses

Row Address (First)	Column Address (Second)
MA0 -----	SA1
MA1 -----	SA2
MA2 -----	SA3
MA3 -----	SA4
MA4 -----	SA5
MA5 -----	SA6
MA6 -----	SA7
MA7 -----	SA8/SA0
MA8 -----	SA17

2. MEMORY CONFIGURATION

OPTION	MC2	MC1	MCO	BANK0	BANK1	BANK2	CONTROL	ADDRESS RANGE
1	0	0	0		128K	128K	Ras1 Ras2	(000000-01FFFF) (020000-03FFFF)
2	0	0	1	512K			Ras0	(000000-07FFFF)
3	0	1	1	512K	128K		Ras0 Ras1	(000000-07FFFF) (080000-09FFFF)
4	0	1	0	512K	128K	128K	Ras0 Ras1 Ras2	(000000-07FFFF) (08FFFF-09FFFF) (100000-17FFFF)
5	1	1	1	512K	128K	512K	Ras0 Ras1 Ras2	(000000-07FFFF) (080000-09FFFF) (100000-17FFFF)
6	1	1	0	512K	128K	512K	Ras0 Ras1 Ras2 Ras3	(000000-07FFFF) (080000-09FFFF) (100000-17FFFF) (180000-1FFFFFF)
7	1	0	1	NOT DEFINED				
8	1	0	0	NOT DEFINED				

FIGURE 1. MEMORY CONFIGURATION.

3. DMA Control Logic Equations

```
/Bras0 = /la23 & /la22 & /la21 & /la20 & /la19 & /mc2 & mc0
      +/la23 & /la22 & /la21 & /la20 & /la19 & mc1
      +refresh;

/Bras1 = /la23 & /la22 & /la21 & /la20 & /la19 & /la18 & /la17 & /mc0
      & /mc1 & /mc2
      +/la23 & /la22 & /la21 & /la20 & la19 & /la18 & /la17 & mc1
      +refresh;

/Bras2 = /la23 & /la22 & /la21 & /la20 & /la19 & /la18 & la17 & /mc0
      & /mc1 & /mc2
      +/la23 & /la22 & /la21 & /la20 & /la19 & /la18 & /la17 & /mc0 & mc1
      & /mc2
      +/la23 & /la22 & /la21 & /la20 & /la19 & mc1 & mc2
      +refresh;

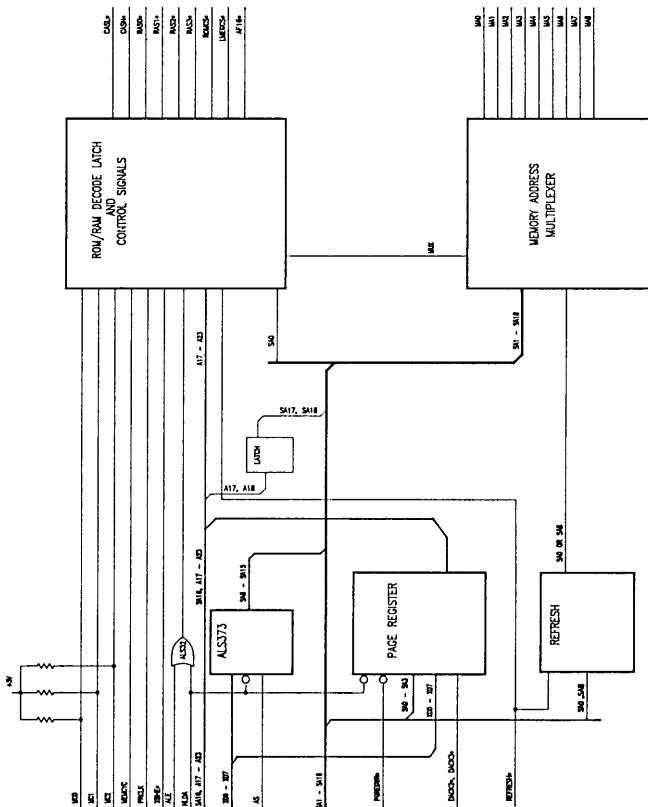
/Bras3 = /la23 & /la22 & /la21 & /la20 & la19 & /mc0 & mc1 & mc2
      +refresh;

/Bcas = /la23 & /la22 & /la21 & /la20 & /la19 & /la18 & /mc2 & /mc1
      & /mc0 & /refresh
      +/la23 & /la22 & /la21 & /la20 & /la19 & /mc2 & mc0 & /refresh
      +/la23 & /la22 & /la21 & /la20 & /la19 & mc1 & /refresh
      +/la23 & /la22 & /la21 & /la20 & la19 & /la18 & /la17 & mc1
      & /refresh
      +/la23 & /la22 & /la21 & /la20 & /la19 & /la18 & /la17 & /mc2 & mc1
      & /mc0 & /refresh
      +/la23 & /la22 & /la21 & /la20 & /la19 & mc2 & mc1 & /refresh
      +/la23 & /la22 & /la21 & /la20 & mc2 & mc1 & /mc0 & /refresh;

/B1meg = /la23 & /la22 & /la21 & /la20+refresh;

/Bromcs =/la23 & /la22 & /la21 & /la20 & la19 & la18 & la17 & /refresh
      +la23 & la22 & la21 & la20 & la19 & la18 & la17 & /refresh
      +la23 & la22 & la21 & /la20 & la19 & la18 & la17 & /refresh;
```

4. Block Diagram



5. PIN CONFIGURATION

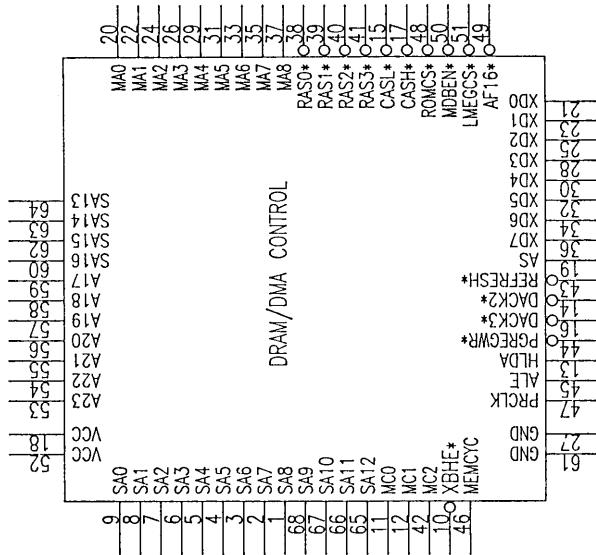


FIGURE 3. PIN CONFIGURATION

PIN	FUNCTION	STATE
52	PIN CONFIGURATION	00000000000000000000000000000000

6. INPUT/OUTPUT PINS FUNCTION AND DESCRIPTION

PIN#	PIN NAME	TYPE	DESCRIPTION
9	SA0	INPUT	CPU ADDRESS LINE
8	SA1	INPUT	CPU ADDRESS LINE
7	SA2	INPUT	CPU ADDRESS LINE
6	SA3	INPUT	CPU ADDRESS LINE
5	SA4	INPUT	CPU ADDRESS LINE
4	SA5	INPUT	CPU ADDRESS LINE
3	SA6	INPUT	CPU ADDRESS LINE
2	SA7	INPUT	CPU ADDRESS LINE
1	SA8	INPUT/OUTPUT	CPU ADDRESS LINE
68	SA9	INPUT/OUTPUT	CPU ADDRESS LINE
67	SA10	INPUT/OUTPUT	CPU ADDRESS LINE
66	SA11	INPUT/OUTPUT	CPU ADDRESS LINE
65	SA12	INPUT/OUTPUT	CPU ADDRESS LINE
64	SA13	INPUT/OUTPUT	CPU ADDRESS LINE
63	SA14	INPUT/OUTPUT	CPU ADDRESS LINE
62	SA15	INPUT/OUTPUT	CPU ADDRESS LINE
60	SA16	INPUT/OUTPUT	CPU ADDRESS LINE
59	A17	INPUT/OUTPUT	CPU ADDRESS LINE
58	A18	INPUT/OUTPUT	CPU ADDRESS LINE
57	A19	INPUT/OUTPUT	CPU ADDRESS LINE
56	A20	INPUT/OUTPUT	CPU ADDRESS LINE
55	A21	INPUT/OUTPUT	CPU ADDRESS LINE
54	A22	INPUT/OUTPUT	CPU ADDRESS LINE
53	A23	INPUT/OUTPUT	CPU ADDRESS LINE
45	ALE	INPUT	ADDRESS LATCH ENABLE Active HIGH - to latch generated RAS/CAS/LMEG.
43	REFRESH*	INPUT	REFRESH - Active LOW to initiate a refresh cycle for dynamic RAMs.
14	DACK2*	INPUT	8237 Channel 2 DMA ACKNOWLEDGE
16	DACK3*	INPUT	8237 Channel 3 DMA ACKNOWLEDGE
13	HLDA	INPUT	HOLD ACKNOWLEDGE Active HIGH - it indicates that the DMA has the system bus.
44	PGREGWR*	INPUT	PAGE REGISTER WRITE- active LOW to perform I/O WRITE cycle to the DMA Page Register.
46	MEMCYC	INPUT	MEMORY CYCLE - Active HIGH to initiate RAS/CAS/MA0-MA8 outputs.
19	AS	INPUT	ADDRESS STROBE - Active HIGH it latches the address lines SA8-SA15 D0-D7 into external latch for DMA cycle

PIN#	PIN NAME	TYPE	DESCRIPTION
10	XBHE*	INPUT	BYTE HIGH ENABLE To enable the high memory data bytes DS-D15
47	PRCLK	INPUT	16MHZ CLOCK.
11	MCO	INPUT	MEMORY CONFIGURATION SELECT LSB.
12	MC1	INPUT	MEMORY CONFIGURATION SELECT.
42	MC2	INPUT	MEMORY CONFIG. SELECT MSB.
20	MA0	OUTPUT	MUXED MEMORY ADDRESS - addressing required for DRAM memory.
22	MA1	OUTPUT	MUXED MEMORY ADDRESS - addressing required for DRAM memory.
24	MA2	OUTPUT	MUXED MEMORY ADDRESS - addressing required for DRAM memory.
26	MA3	OUTPUT	MUXED MEMORY ADDRESS - addressing required for DRAM memory.
28	MA4	OUTPUT	MUXED MEMORY ADDRESS - addressing required for DRAM memory.
31	MA5	OUTPUT	MUXED MEMORY ADDRESS - addressing required for DRAM memory.
33	MA6	OUTPUT	MUXED MEMORY ADDRESS - addressing required for DRAM memory.
35	MA7	OUTPUT	MUXED MEMORY ADDRESS - addressing required for DRAM memory.
37	MA8	OUTPUT	MUXED MEMORY ADDRESS - addressing required for DRAM memory.
17	CASH*	OUTPUT	COLUMN ADDRESS STROBE HIGH - Active LOW, it's used to select the high data byte MD8-MD15 for a DRAM access cycle.

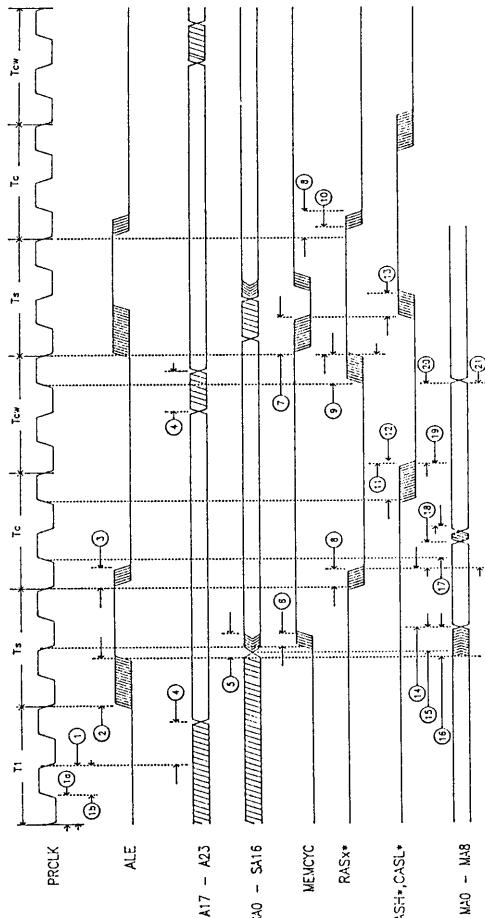
PIN#	PIN NAME	TYPE	DESCRIPTION
15	CASL*	OUTPUT	COLUMN ADDRESS STROBE LOW - Active LOW, it's used to select low data byte MDO-MD7 for a DRAM access cycle.
38	RAS0*	OUTPUT	ROW ADDRESS STROBE 0 - Active LOW, it's used for selecting DRAM Bank0.
39	RAS1*	OUTPUT	ROW ADDRESS STROBE 1 - Active LOW, it's used for selecting DRAM Bank1.
40	RAS2*	OUTPUT	ROW ADDRESS STROBE 2 - Active LOW, it's used for selecting DRAM Bank2.
41	RAS3*	OUTPUT	ROW ADDRESS STROBE 3 - Active LOW, it's used for selecting DRAM Bank2.
51	LMEGCS*	OUTPUT	LOWER MEGABYTE CHIP SELECT - Active LOW, it indicates that below 1 megabyte
50	MDBEN*	OUTPUT	MEMORY DATA BUS ENABLE Active LOW it is used to enable the data bus buffer.
49	AF16*	OUTPUT	AF16 Active LOW - it signals the control logic (CPU CNTL) that the memory cycle is a 16 bit 1 wait state cycle.
48	ROMCS*	OUTPUT	ROM CHIP SELECT Active LOW
21	XD0	INPUT	DATA BUS 0 for the peripheral bus.
23	XD1	INPUT	DATA BUS 1 for the peripheral bus.
25	XD2	INPUT	DATA BUS 2 for the peripheral bus.
28	XD3	INPUT	DATA BUS 3 for the peripheral bus.
30	XD4	INPUT	DATA BUS 4 for the peripheral bus.
32	XD5	INPUT	DATA BUS 5 for the peripheral bus.
34	XD6	INPUT	DATA BUS 6 for the peripheral bus.
36	XD7	INPUT	DATA BUS 7 for the peripheral bus.
18, 52 27, 61	VCC GND		+5V POWER SUPPLY GROUND

7. TIMING SPECIFICATIONS

SYM	PARAMETER	MIN	MAX	UNITS	REMARKS
T1	PRCLK Period	62.5	250	nsec	
T1a	PRCLK Low Time	25	125	nsec	
T1b	PRCLK High time	25	125	nsec	
T2	ALE Active Delay	6	53	nsec	
T3	ALE Inactive Delay	5	25	nsec	
T4	Address Delay	1	60	nsec	
T5	Latched Address From ALE Active	—	23	nsec	
T6	MEMCYC Active Delay From ↓ PRCLK	5	25	nsec	
T7	MEMCYC Inactive Delay From ↓ PRCLK	5	35	nsec	
T8	RASx# Active Delay From ↓ PRCLK	5	45	nsec	at 100pf
T9	RASx# Inactive Delay From ↑ PRCLK	5	30	nsec	
T10	RASx# Precharge Time	110	30	nsec	
T11	RASx# Hold Time	30	—	nsec	
T12	CASx# Active Delay Time From ↑ PRCLK	5	40	nsec	at 165pf
T13	CASx# Inactive Delay Time From ↓ MEMCYC	5	90	nsec	
T14	Row Address Setup Time	5	—	nsec	
T15	Memory Address Delay Time From SA1 to SA16	0	50	nsec	at 200pf
T16	Memory Address Delay From ALE	8	60	nsec	
T17	Memory Address Stable Delay From ↑ PRCLK	—	50	nsec	at 200pf
T18	Row Address Hold Time	20	—	nsec	
T19	Column Address Setup Time	5	—	nsec	
T20	Column Address Hold Time	30	—	nsec	
T21	Column Address Hold Time Referenced To RASx#	120	—	nsec	
T22	ALE ↑ to MDBEN ↑ A/I	—	50	nsec	
T23	ALE ↑ to LMEGCS# ↑ A/I	—	50	nsec	
T24	ALE ↑ to RDGCS# ↑ A/I	—	50	nsec	CPUHLDA HIGH

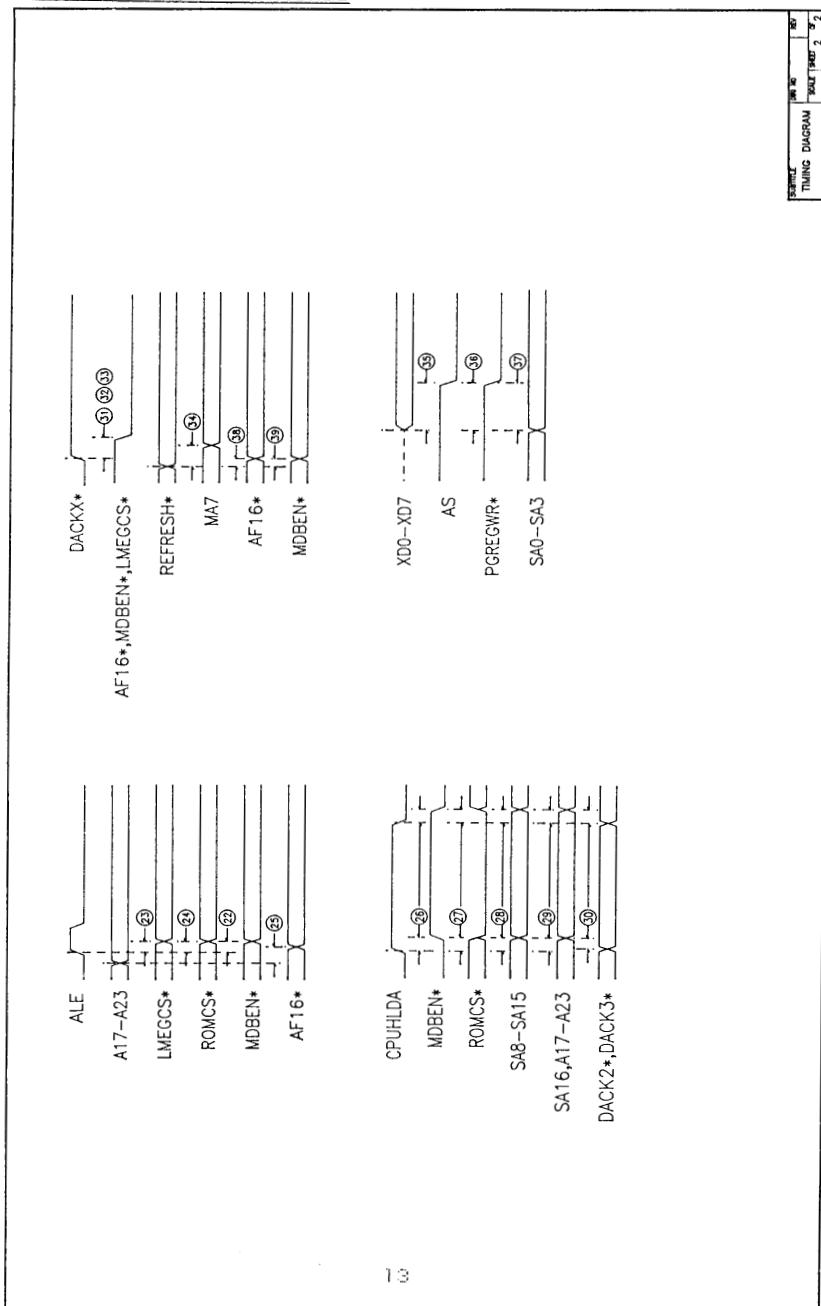
SYM	PARAMETER	MIN	MAX	UNITS	REMARKS
T25	A17-A23 ↑↓ to AF16*↑↑ A/I		50	nsec	
T26	CPUHLDAT↑↓ to MDBEN ↑I		50	nsec	
T27	CPUHLDAT↑↓ to RD MCS*↑↑ A/I		50	nsec	
T28	CPUHLDAT↑↓ to SA8-SA15 ↑↑		50	nsec	
T29	CPUHLDAT↑↓ to SA16,A17-A23↑↑		50	nsec	
T30	DACK2*, DACK3*↑↓ to SA16,↑↑ A17-A23		50	nsec	
T31					
T32}	DACKx* to AF16*, MDBEN*				
T33	LMEGCS*		80	nsec	
T34	REFRESH*↑↓ to MA7↑↑A/I		50	nsec	at 200pf
T35	XD0-XD7 Setup to AS↓	100		nsec	
T36	XD0-XD7 Setup to PGREGWR*	100		nsec	
T37	SA0-SA3 Setup to PGREGWR*	100		nsec	
T38	REFRESH*↑↓ to AF16*		50	nsec	
T39	REFRESH*↑↓ to MDBEN*		50	nsec	

8. Timing Diagram.



Sheet 2 of 2
Page 14
Date: 10/10/2014
By: [Signature]

Timing Diagram Continue



9. ELECTRICAL SPECIFICATIONS

ABSOLUTE MAX RATINGS (NON-OPERATING, VSS=0.0V)

	MIN	MAX	UNITS
STORAGE TEMPERATURE	-65	+150	Degrees C.
VOLTAGE ON ANY PIN W.R.T GROUND	-0.5	7.0	Volts

OPERATING ELECTRICAL SPECIFICATIONS:

OPERATING AMBIENT	MIN	TYP	MAX	UNITS
AIR TEMP. RANGE	0	25	70	Degrees C
POWER SUPPLIES				
VCC	4.5	5.0	5.5	Volts
VSS	0	0	0	Volts
LEAKAGE CURRENT	MIN	TYP	MAX	UNITS
Vin = 0.0 v			20	Microamps
Vin = 5.0 v	-20			Microamps

INPUT VOLTAGES

LOGIC "0" (Vil)		0.8	volts
LOGIC "1" (Vih)	2.0		volts

OUTPUT VOLTAGES CURRENT LOADING

LOGIC "0" (vol)		0.4	volts
MA[0]-MA[8]	0.8ma(min)		

SXA8-SXA16,A17-A23 @ 4ma(min)
RASx*, @ 4ma(min)
CASx*, @ 8 ma(min)
Others must be able
to SINK minimum @ 2ma

LOGIC "1" (Vol)	2.4	volts
MA[0]-MA[8],		

SXA8-SXA16,A17-A23 @ 8ma
CASx*,RASx* @ 4ma
CASx* @ 8 ma
Others must be able
to DRIVE minimum @ 2ma

INPUT CAPACITANCEAll inputs $0.0 < V_{in} < 5.0$ 10 picofarads**OUTPUT CAPACITANCE**

MA[0]-MA[8]	200	picofarads
CAS	165	picofarads
RAS	100	picofarads
SA8-SA16	150	picofarads
All other outputs	50	picofarads

TANDY COMPUTER PRODUCTS

Floppy Disk Support Chip Specification

**Floppy Disk Support Chip Specification
Contents**

Section	Page
General Description	1
Pin Description	2
Block Diagram	3
Environmental Specifications	5
DC Electrical Specifications	5
AC Characteristics	6
Timing Diagrams	9

Floppy Disk Support Logic
Tandy Part # 8xxxxxx
Jan 29, 1987
Preliminary

1.0 GENERAL DESCRIPTION

1.1 The Tandy Part # 8xxxxxx - Floppy Disk Support logic:

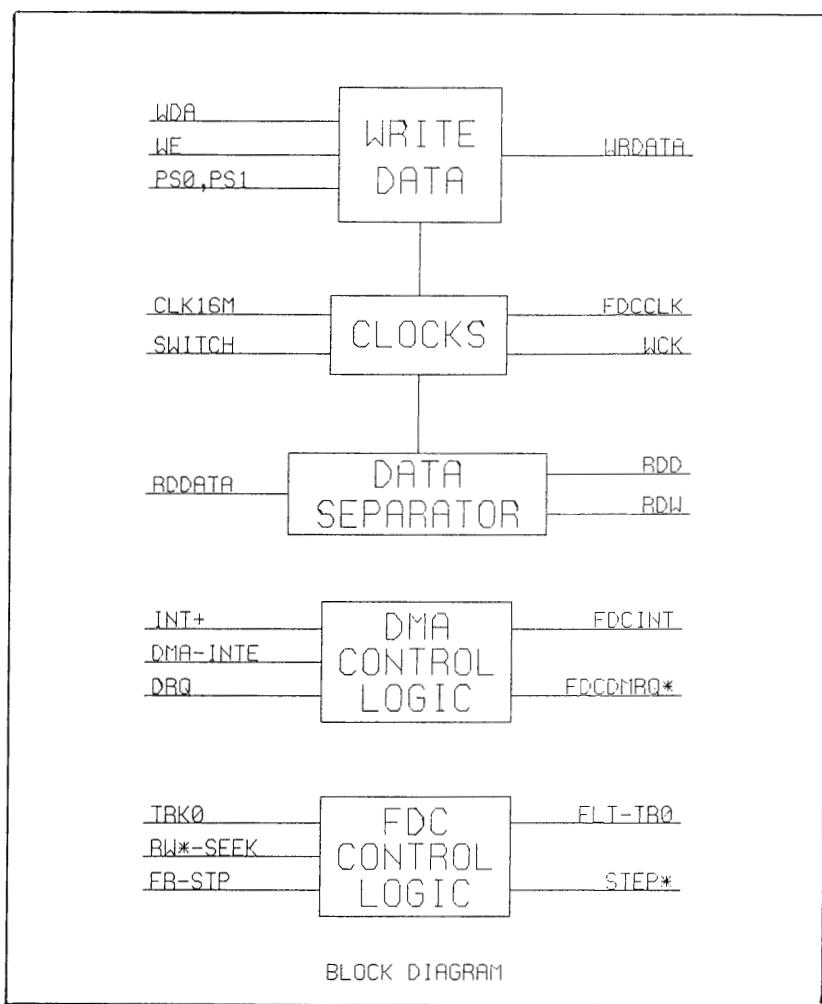
- Generates the clock to the 765 Floppy Disk Controller.
- Generates the write clock to the Floppy Disk Controller.
- Generates step pulses, track 0 indicator, DMA request, and FDC interrupt signals.
- Generates the Read Data and Read Data Window signals.
- Generates the Write Data to the Floppy Disk.

1--	CLK16M	+5V	--24
2--	WCK	SWITCH	--23
3--	FDCCLK	INT+	--22
4--	RDDATA*	DMA/INTE	--21
5--	RDD	DRQ	--20
6--	RDW	FDCINT	--19
7--	FRES/S	FDCDMRQ*	--18
8--	RW*/SEEK	PS0	--17
9--	TRK0*	PS1	--16
10--	F/TRK0	WRD	--15
11--	STEP*	WRE	--14
12--	GND	WRDATA*	--13

FIGURE 1. Pin Assignment

1.2 DESCRIPTION OF PINS:

PIN #	PIN NAME	TYPE	DESCRIPTION
1	CLK16M	INPUT	Frequency = 16.0000 Tolerance = 100ppm
2	WCK	OUTPUT	If SWITCH = 0, period = 2 us, 250 ns pulse If SWITCH = 1, period = 1 us, 250 ns pulse
3	FDCCLK	OUTPUT	If SWITCH = 0, then CLK16M/4 If SWITCH = 1, then CLK16M/2
4	RDDATA	INPUT	Serial data from FDD
5	RDD	OUTPUT	Serial data from FDC
6	RDW	OUTPUT	Read Data Window
7	FRES/S	INPUT	Step pulses to move head to another cylinder
8	RW*/SEEK	INPUT	Specifies seek mode when high
9	TRK0*	INPUT	From FDD, indicating head is on track 0
10	F/TRK0	OUTPUT	To FDC, indicating head is on track 0
11	STEP*	OUTPUT	Moves head of FDD
12	GND		Ground
13	WRDATA*	OUTPUT	Serial Data to FDD
14	WRE	INPUT	Write Enable
15	WRD	INPUT	Serial Data from FDC
16	PS1	INPUT	Write precompensation status
17	PS0	INPUT	Write precompensation status
18	FDCDMRQ*	OUTPUT	DRQ delayed by 1.0 usec.
19	FDCINT	OUTPUT	Interrupt request
20	DRQ	INPUT	FDC DMA Request
21	DMA/INTE	INPUT	DMA request and FDC interrupt enable
22	INT+	INPUT	Interrupt request generated by FDC
23	SWITCH	INPUT	0 = low density drive 1 = high density drive
24	+5V		+5 Volts



TANDY COMPUTER PRODUCTS

2.0 ENVIROMENTAL SPECIFICATIONS

2.1 Storage temperature: -65°C min., +150°C max.
2.2 Operating temperature: 0°C min., +25°C typ, +70°C max.

3.0 DC ELECTRICAL SPECIFICATIONS

3.1 Absolute Maximum Rating:

Voltage on any pin
w.r.t. Ground: -0.5 min., 7.0 max. volts

3.2 Operating Electrical Specifications:

	Min.	Typ.	Max.	Units
3.2.1 Operating Ambient: Air Temperatuue Range	0	25	70	°C
3.2.2 Power Supplies: VCC	4.5	5.0	5.5	volts
VSS	0	0	0	volts
ICC				milli-amps
Total Power				milli-watts
3.2.3 Leakage Current, All Inputs: Vin = 0.0 v			-10	micro-amps
Vin = 5.0 v			+10	micro-amps
3.2.4 Input voltages:				
3.2.4.1 Except RDDATA*, TRK*				
Logic "0"			.8	volts
Logic "1"	2.0			volts
3.2.4.2 RDDATA*, TRK*				
Positive going threshold		1.8		volts
Negative going threshold		1.2		volts
Hysteresis voltage	220			milli-volts
3.2.5 Output Voltages:				
3.2.5.1 Except WRDATA*, STEP*				
Logic "0" @ 4.0 mA load			.4	volts
Logic "1" @ 4.0 mA load	2.4			volts
3.2.5.2 WRDATA*, STEP*				
Logic "0" @ 48 mA			.5	volts
3.2.6 Input Capacitance (0.0 < Vin < 5.0)				
All inputs			10	pf
3.2.7 Output Capacitance				
All loads			50	pf

4.0 AC CHARACTERISTICS

4.1 FDCCLK Timing

Parameter	Min.	Typ.	Max.	Units
t_H	90	120	130	nSec
t_R, t_F		5	10	nSec
t_L	100	120	160	nSec
t_{CY}	245	250	255	nSec

4.2 WCK Timing

t_H	100	250	250	nSec
t_R, t_F		5	10	nSec
t_L		$t_{CY} - (t_H + t_R + t_F)$		μ Sec
t_{CY}		2.0		

4.3 WRDATA* Timing

$WCK_H - WE_H$	20		nSec
$WCK_L - WE_L$	20		nSec
PSD	20	100	nSec
WDD	20	100	nSec
WDA_W		$WCK_H - 50$	nSec
WRD_W	115	125	nSec
$WDD_H - WRD_L$ early	150	135	nSec
$WDD_H - WRD_L$ nominal	275	250	nSec
$WDD_H - WRD_L$ late	400	375	nSec
		500	nSec

4.4 DMA/INTERRUPT Timing

$I_H - FI_H$		30	nSec
$I_L - FI_L$		30	nSec
$DI_L - FI_L$		30	nSec
$WCK_H - DRQ_H$	0		nSec
$WCK_L - DRQ_H$		-20	nSec
$DRQ_H - FDRQ_H$	750	1050	nSec
$DRQ_L - FDRQ_L$		30	nSec
$DI_L - FDRQ_L$		30	nSec
$FCK_H - FDRQ_H$		30	nSec

TANDY COMPUTER PRODUCTS

4.5 CONTROL Timing

Parameter	Min.	Typ.	Max.	Units
$T_L - FT_H$			30	nSec
$T_E - FT_L$			30	nSec
$RS_L - FT_L$			30	nSec
$F_H - S_L$			30	nSec
$F_L - S_H$			30	nSec
$RS_L - S_H$			30	nSec

4.6 DATA SEPARATOR Timing

RDA_W	200	350	550	nSec
$RDA_L - RDD_H$	188		313	nSec
RDD_L^W	240	250	260	nSec
$RDD_H^W - RDW_C$	850	875	900	nSec
$RDW_C^{(ND)}_W$			2.0	μ Sec

"A"

RDA_S	3062		nSec
$RDW_C - RDD_H$	15		nSec

"B"

RDA_S	4812		nSec
$RDW_C - RDD_H$		1938	nSec

"C"

RDA_S	5062		nSec
$RDW_C - RDD_H$	15		nSec

FDSL AC TIMING

FIG. 1 FDCCLK

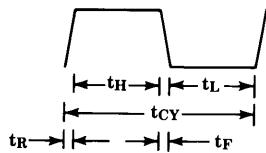


FIG. 2 WCK

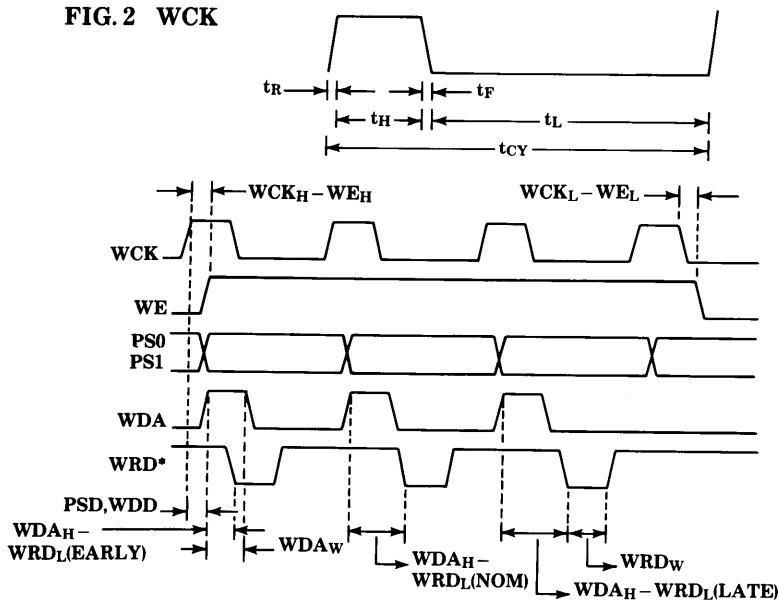


FIG. 3 WRITE DATA TIMING.

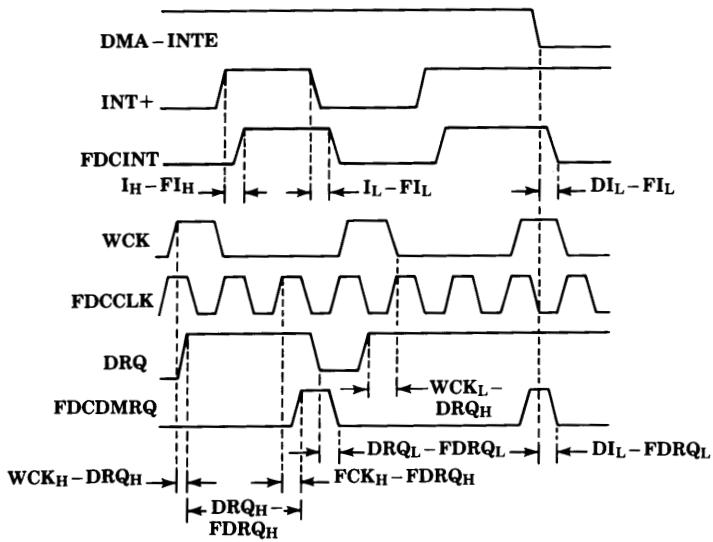


FIG. 4 DMA/INTERRUPT TIMING.

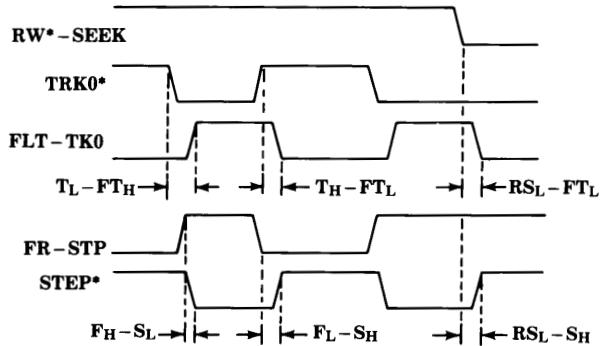


FIG. 5 CONTROL LOGIC TIMING.

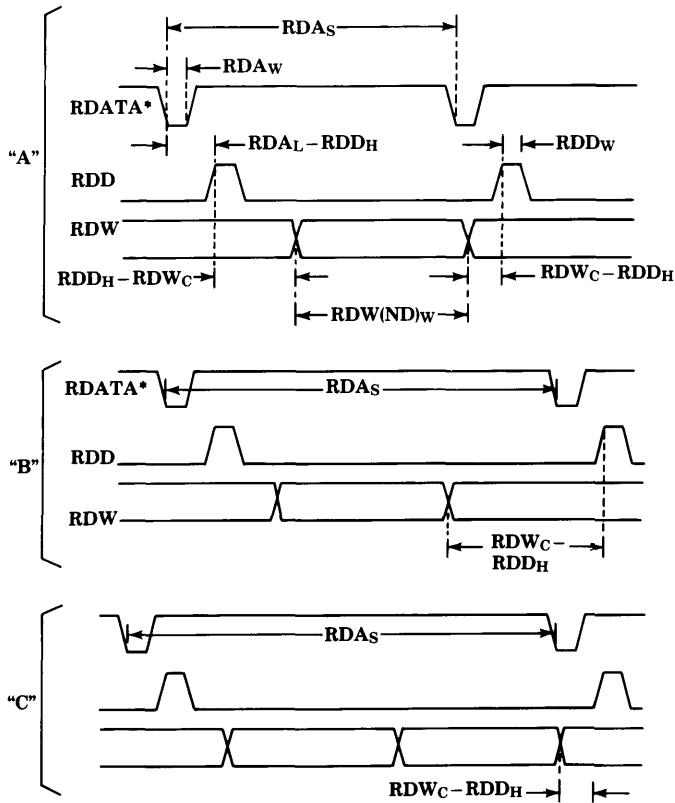


FIG. 6 DATA SEPARATOR TIMING.

TANDY COMPUTER PRODUCTS

PRINTER INTERFACE SPECIFICATION

TANDY COMPUTER PRODUCTS

**PRINTER INTERFACE SPECIFICATION
CONTENTS**

GENERAL DESCRIPTION.....	1
SPECIFICATIONS.....	3

PRINTER INTERFACE SPECIFICATION
TANDY PART # 8075068
APRIL 30, 1986

1. GENERAL DESCRIPTION

1.1 The Tandy part# 8075068 - Printer Interface I.C provides the interface between the system I/O bus and the printer. Figure 1 shows Block diagram of Printer Interface chip. Figure 2 shows pin configurations of Printer Interface chip.

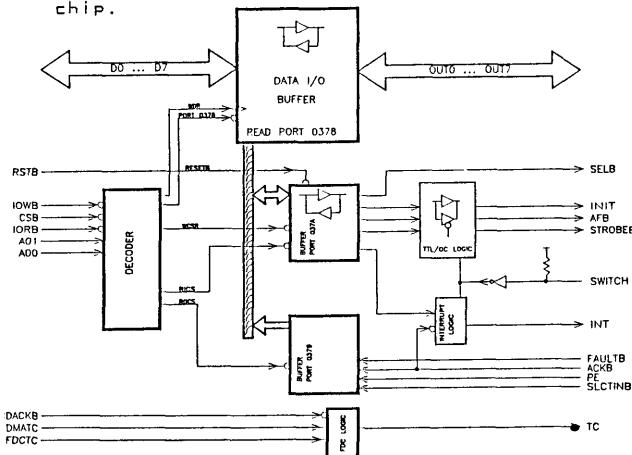


Figure 1.

	VDD
1--INT	D7--39
2--SWITCH	D5--38
3--A01	D3--37
4--A00	D1--36
5--OUT0	D0--35
6--OUT1	D2--34
7--OUT2	D4--33
8--OUT3	D6--32
9--OUT7	D8--31
10--OUT6	IOWB--30
11--OUT5	IORB--29
12--OUT4	RSTB--28
13--STROBEB	NC--27
14--AFB	SLCTINS--26
15--INIT	NC--25
16--SELB	DMATC--24
17--FAULT	FDOACKB--23
18--PE	FDCTC--22
19--BUSY	ACKB--21
20--VSS	

Figure 2.

1.2 DESCRIPTION OF EACH PINS:

Pin#	Pin Name	Type	Description
1	INT	output	Interrupt signal
2	SWITCH	input	Switch for totem pole output or open collector output on INITB, AF, STROBEB.
3	A01	input	CPU address line
4	A00	input	CPU address line
5	OUT0	input/output	Data I/O line
6	OUT1	input/output	Data I/O line
7	OUT2	input/output	Data I/O line
8	OUT3	input/output	Data I/O line
9	OUT7	input/output	Data I/O line
10	OUT6	input/output	Data I/O line
11	OUT5	input/output	Data I/O line
12	OUT4	input/output	Data I/O line
13	STROBEB	output	Printer Strobe signal
14	AFB	output	Printer Autofeed signal
15	INITB	output	Printer Initialize signal
16	SEL	output	Printer Select signal
17	FAULTB	input	Printer Fault signal
18	PE	input	Printer Paper empty signal
19	BUSY	input	Printer Busy signal
20	VSS	ground	Ground
21	ACKB	input	Printer Acknowledge signal
22	FDCTC	input	FDC Terminal Count
23	FDCDACKB	input	FDC-DMA Acknowledge signal
24	DMATC	input	DMA Terminal Count
25	TC	output	FDC Terminal Count signal
26	SLCTINB	input	Printer Select input
27	NC	--	Not used
28	RSTB	input	System Reset
29	IORB	input	CPU I/O Read strobe
30	IOWB	input	CPU I/O Write strobe
31	CSB	input	Chip select signal
32	D6	Input/output	CPU Data I/O
33	D4	Input/output	CPU Data I/O
34	D2	Input/output	CPU Data I/O
35	D0	Input/output	CPU Data I/O
36	D1	Input/output	CPU Data I/O
37	D3	Input/output	CPU Data I/O
38	D5	Input/output	CPU Data I/O
39	D7	Input/output	CPU Data I/O
40	VDD	power	+5 Volt Power Supply

2. ENVIRONMENTAL SPECIFICATIONS

2.1 Storage Temperature -65 °C to 150 °C

2.2 Operating Temperature 0 °C to 70 °C

3. ELECTRICAL SPECIFICATIONS

3.1 Absolute Maximum Ratings

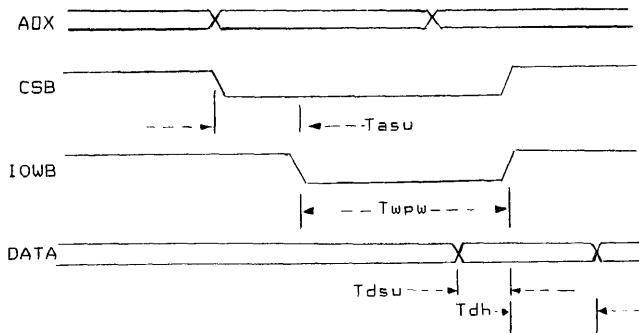
Parameter	Min.	Typ.	Max.	Units	Cond.
Voltage, any pin	-1.0		7.0	Volts	W.R.T ground
Power Dissipation			0.5	Watts	

3.2 D.C. Electrical Characteristics

Symb.	Parameter	Min.	Typ.	Max.	Units	Cond.
VDD	Supply Voltage	4.5	5.0	5.5	Volts	
Icc(q)	Quiescent current			50	uA	
Icc(o)	Operating Current			40	mA	
Vil	Input Low Voltage			0.8	Volts	TTL inputs
Vih	Input High Voltage	2.0			Volts	TTL inputs
Iin	Input Leakage	-10		10	uA	
Cin	Input Capacitance			7	pF	
Vol	Output Low Voltage			0.4	Volts	0-4 mA
Voh	Output High Voltage	2.4			Volts	0-2 mA
Ioz	High Impedance Leak	-10		10	uA	

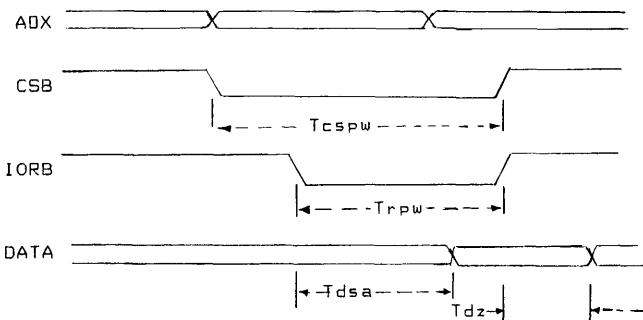
3.3 A.C Electrical Characteristics

3.3.1 Write Cycle



Symb.	Parameter	Min.	Typ.	Max.	Units	Cond.
---	---	---	---	---	---	---
Tasu	Address Setup	15			nS	
Twpw	Write Pulse Width	69			nS	
Tdsu	Data Setup	29			nS	
Tdh	Data Hold	6			nS	

3.3.2 Read Cycle



Symb.	Parameter	Min.	Typ.	Max.	Units	Cond.
Tcspw	Chip Select Width	69			nS	
Trpw	Read Pulse Width	69			nS	
Tda	Data Access			69	nS	
Tdz	Bus Hold/release	6		25	n	

TANDY COMPUTER PRODUCTS

KEYBOARD INTERFACE SPECIFICATION

TANDY COMPUTER PRODUCTS

**KEYBOARD INTERFACE SPECIFICATION
CONTENTS**

GENERAL DESCRIPTION.....	1
SPECIFICATIONS.....	3

KEYBOARD INTERFACE SPECIFICATION
 TANDY PART # 8075069
 MAY 05, 1986

1. GENERAL DESCRIPTION

- 1.1 The Tandy part# 8075069 - Keyboard Interface I.C provides two functions:
- Interface between the system I/O bus and keyboard.
 - FDC support logic that generates DRIVE SELECT SIGNAL, MOTOR ON SIGNAL, FDC TERMINAL COUNT, FDC RESET and DMA/I.

Figure 1. shows block diagram of Keyboard Interface chip
 Figure 2. shows pin configuration of Keyboard Interface chip.

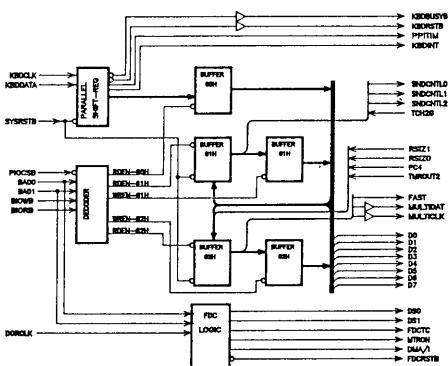


Figure 1.

1-KBDCLK	VDD	-4.0
2-KBDDATA	MULTI CLK	-39
3-KBDBUSYB	MULTI DAT	-38
4-KBDINT	FAST	-37
5-RS120	TCH2G	-36
6-RS121	PPITIM	-35
7-D0	D50	-34
8-D1	D51	-33
9-D2	FDRS1	-32
10-D3	DMA/I	-31
11-D4	TMR0	-30
12-D5	FDTCL	-29
13-D6	SNDCTL2	-28
14-D7	SNDCTL0	-27
15-PIOCSB	SNDCTL1	-26
16-BA00	TMR0UT2	-25
17-BA01	PC4	-24
18-BA0B	SYSRSTB	-23
19-BA0B	KBDRSTB	-22
20-VSS	DORCLK	-21

Figure 2.

1.2 DESCRIPTION OF EACH PINS:

Pin#	Pin Name	Type	Description
1	KBDCLK	input	Keyboard clock
2	KBDDATA	input	Keyboard data
3	KBDBUSYB	output	Keyboard busy signal
4	KBDINT	output	Keyboard interrupt signal
5	RS1Z0	input	Monochrome/color monitor mode
6	RS1Z1	input	Reserved
7	D0	input/output	Data I/O line
8	D1	input/output	Data I/O line
9	D2	input/output	Data I/O line
10	D3	input/output	Data I/O line
11	D4	input/output	Data I/O line
12	D5	input/output	Data I/O line
13	D6	input/output	Data I/O line
14	D7	input/output	Data I/O line
15	PIOCSB	input	Chip select strobe
16	BA0D	input	CPU address line
17	BA01	input	CPU address line
18	BIORB	input	CPU I/O read strobe
19	BIOWB	input	CPU I/O write strobe
20	VSS	ground	Ground
21	DORCLK	input	Decode latch clock
22	KBDRSTB	output	Keyboard reset signal
23	SYSRSTB	input	System reset signal
24	PC4	input	Video memory size mode
25	TMROUTZ	input	Timer counter from 8253 out2
26	SNDCTL1	output	Sound control 1
27	SNDCTL0	output	Sound control 0
28	SNDCTL2	output	Sound control 2
29	FDCTC	output	FDC terminal count
30	MTRON	output	Motor ON signal to disk drive
31	DMA/I	output	DMA Request & FDC Interrupt enable
32	FDCRSTB	output	FDC reset signal
33	DS1	output	Drive select 1 signal
34	DS0	output	Drive select 0 signal
35	PPITIM	output	Timer Video signal
36	TCH2G	output	Timer channel 2 gate
37	FAST	input	4.77Mhz or 7.16Mhz mode select
38	MULTIDAT	output	Multi-data
39	MULTICLK	output	Multi-clock
40	VDD	power	+5 Volt Power Supply

2. ENVIRONMENTAL SPECIFICATIONS

2.1 Storage Temperature -65 °C to 150 °C
2.2 Operating Temperature 0 °C to 70 °C

3. ELECTRICAL SPECIFICATIONS

3.1 Absolute Maximum Rating

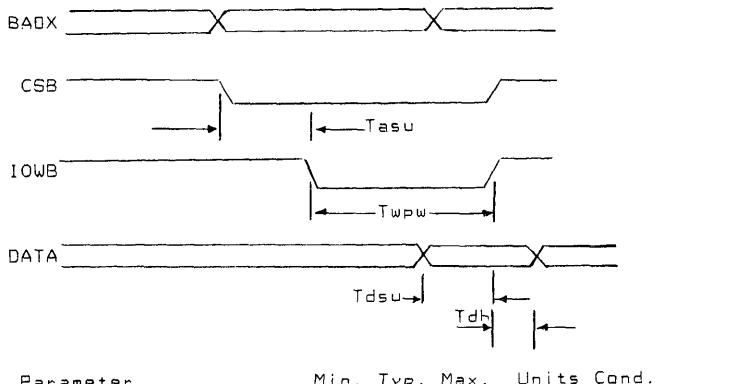
Parameter	Min.	Typ.	Max.	Units	Cond.
Voltage, any pin	-1.0		7.0	Volts	W.R.T ground
Power Dissipation			0.5	Watts	

3.2 D.C. Electrical Characteristics

Symb.	Parameter	Min.	Typ.	Max.	Units	Cond.
VDD	Supply Voltage	4.5	5.0	5.5	Volts	
Icc(q)	Quiescent current			50	uA	
Icc(o)	Operating Current			40	mA	
Vil	Input Low Voltage			0.8	Volts	TTL inputs
Vih	Input High Voltage	2.0			Volts	TTL inputs
Iin	Input Leakage	-10		10	uA	
Cin	Input Capacitance			7	pF	
Vol	Output Low Voltage			0.4	Volts	04 mA
Voh	Output High Voltage	2.4			Volts	0-2 mA
Ioz	High Impedance Leak	-10		10	uA	

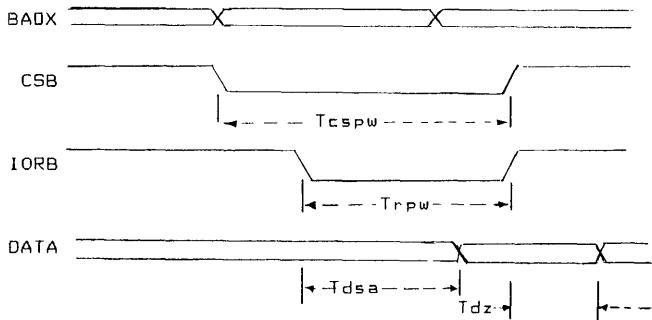
3.3 A.C Electrical Characteristics

3.3.1 Write Cycle



Symb.	Parameter	Min.	Typ.	Max.	Units	Cond.
-----	-----	-----	-----	-----	-----	-----
Tasu	Address Setup	15			nS	
Twpw	Write Pulse Width	69			nS	
Tdsu	Data Setup	29			nS	
Tdh	Data Hold	6			nS	

3.3.2 Read Cycle



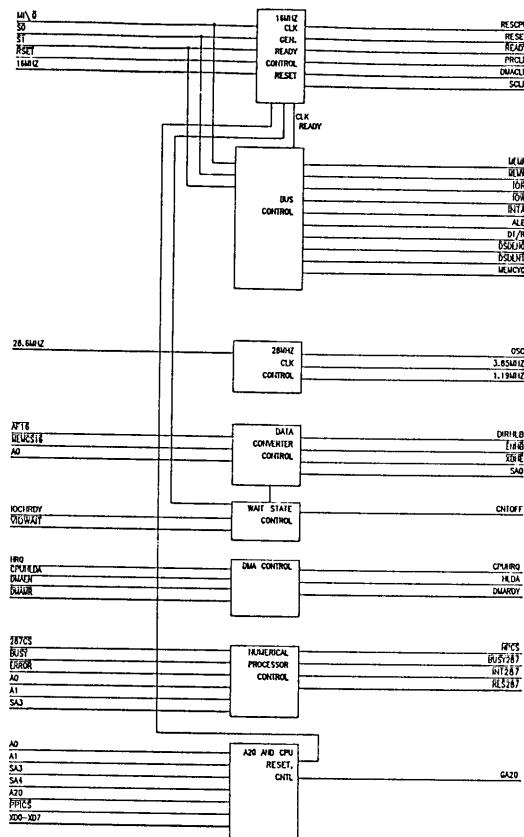
Symb.	Parameter	Min.	Typ.	Max.	Units	Cond.
---	-----	-----	-----	-----	-----	-----
Tcspw	Chip Select Width	69			nS	
Trpw	Read Pulse Width	69			nS	
Tda	Data Access			69	nS	
Tdz	Bus Hold/release	6		25	ns	

CPU Control Chip

Contents

Block Diagram	3
Functional Description	4
Clock Generation, Ready, and Reset Control Logic	4
Bus Control Logic	6
Data Conversion Logic	7
Wait State Control Logic	8
DMA Arbitration Logic	8
Numerical Co-processor Control Logic	9
A20Gate and CPU Reset Logic	10
Pin Configuration	12
Pin and Signal Descriptions	13
Electrical Specifications	19
Timing Specifications	21
Timing Diagrams	25

Block Diagram



SUBTITLE	DATE NO.	REV.
CPU CONTROL		
BLOK DIAGRAM		

Functional Description

The Tandy 1000TX CPU Control IC consists of the following function blocks:

- - Clock Generation, Ready and Reset Control Logic
- - Bus Control Logic
- - Data Conversion Logic
- - Wait State Control Logic
- - DMA Arbitration Logic
- - Numerical Co-processor Control Logic
- - A20Gate and CPU Reset Logic

Clock Generation, Ready and Reset Control Logic

The clock generation logic includes the clock inputs used to derive all of the system clocks. The 16 MHz clock input is used to generate the CPU clock (PRCLK), and is twice the CPU operation frequency of 8 MHz. The 28.6 MHz clock input is used to generate the 14.31818 MHz clock (OSC) and other clocks required by the system.

Three clocks are generated from the 28.6 MHz clock input. These are OSC, 3.58 MHz, and the 1.19 MHz clock outputs. The 28.6 MHz signal input is divided by 2 to generate the OSC output which is 14.31818 MHz. The 28.6 MHz clock is divided by 8 to generate the 3.58 MHz clock which is used by the sound generator circuit. The 28.6 MHz clock is also divided by 24 to generate the 1.19 MHz clock that is used by the Interval Timer 8254.

All other system clocks are generated from the 16 MHz clock input. PRCLK is used to drive the CPU, the Co-processor, and the DRAM/DMA Control IC. PRCLK is output as 16 MHz in the 8 MHz (FAST) mode and is divided by 2 to 8 MHz in the 4 MHz (SLOW) mode. PRCLK output buffer has sufficient drive capability to meet the 3.8 volt minimum Vih requirement of the 80286/80287 clock inputs.

SCLK and DMACLK are system clocks generated from 16 MHz. SCLK is generated by dividing PRCLK by 2 and DMACLK is generated by dividing PRCLK by 4. Both SCLK and DMACLK are held in a LOW state immediately following a reset and will not start operation until the CPU issues the first bus cycle by asserting S1* LOW. Immediately following S1* asserted LOW, SCLK and DMACLK will make their first LOW to HIGH transition at the falling edge of PRCLK during the start of Tc. This will synchronize SCLK and DMACLK with PRCLK. Refer to timing diagrams for an illustration of the startup of SCLK and DMACLK.

A ready signal (READY*) is generated by the CPU Control IC to allow the CPU to operate with slower devices such as peripherals and slow memory. READY* is synchronized with PRCLK in this control block by the control logic of READY* located in the Wait State Control Logic.

Two reset output signals are generated by the CPU Control IC to provide a reset to the main system and a separate reset to the CPU. RESET which is active HIGH is a synchronized reset and is used for resetting the main system. RESCPU is dedicated to the CPU for proper resetting of the 80286. Both RESET and RESCPU are generated when RES* input is asserted LOW to indicate a power-on reset or when RES* is asserted LOW by a reset switch. RESCPU is also generated when a Shutdown condition of the CPU is detected. When a Shutdown condition is detected, RESCPU is asserted HIGH for 16 PRCLK cycles and then negated to assure proper CPU operation. RESCPU can also be generated by doing an I/O write to Port 068 with bit 2 = 0. This will generate RESCPU to reset the CPU and can be used to jump from protected mode to real time mode of the CPU.

The RES* input is buffered internal to the CPU Control IC with a Schmitt trigger input buffer. This signal should be held in a LOW state during power-up for at least 10 msec or until all operating voltages have reached their specified range of operation.

Bus Control Logic

The Bus Control Logic of the CPU Control IC generates several Memory and I/O command and control signals that are issued by the 80286. There are two types of control signals that are generated as output signals. The command outputs are the first type of control signals generated which are decoded from the CPU status inputs MI/O*, S1*, S0*. The generated output command signals are MEMR*, MEMW*, IOR*, IOW*, INTA* which determine which type of cycle is to be performed. The second type of signals are the control signals which is ALE, DT/R*, DSDEN0*, DSDEN1*, and MEMCYC. These signals latch the address from the CPU, control the direction and enabling of the data bus buffers, and determine the start of an on board memory cycle. Table 1. contains a list of the command output signals that are generated from the CPU status line inputs.

MI/O*	S1*	S0*	Type of Cycle
0	0	0	Interrupt Acknowledge
0	0	1	I/O Read
0	1	0	I/O Write
0	1	1	None; Idle
1	0	0	Halt or Shutdown
1	0	1	Memory Read
1	1	0	Memory Write
1	1	1	None; Idle

Table 1. CPU Control Signal Generation

The state machine that controls the output timing of the command and control signals has three bus states, which are the Idle state (Ti), the Status state (Ts), and the command state (Tc). The Idle state (Ti) is generated when the CPU is not actively issuing a bus cycle. During an Idle state, all command and control output signals are in an inactive condition. The beginning of a bus cycle is detected when the CPU asserts S1* or S0*. The state machine will start a Status state (Ts) and will assert ALE to an active HIGH state until the end of Ts. At the end of the Status state, the state machine enters the command state (Tc) and will assert the decoded command issued by the CPU. The command signal may be delayed by one half bus cycle or one PRCLK clock cycle, if the conditions exist to produce a command delay. A Command delay will be generated on all I/O cycles, all Memory cycles in which both AFL16* and MEMCS16* are negated are inactive HIGH, and on all INTA* cycles.

For the control of the data bus buffer, three control signals are generated which are DT/R*, DSDEN0*, and DSDEN1*. DT/R* is used to control the direction of the data bus to and from the CPU. DSDEN0* and DSDEN1* are enable signals for the data bus buffers. DSDEN0* is qualified with A0 to control the lower 8 bits of the data bus (D0-D7). DSDEN1* is qualified with BHE* to control the upper 8 data bits (D8-D15). Control of the upper and lower data bits independently is required in the AT type architecture to control the 8 to 16 bit data conversions.

ALE is a control signal used to latch the address line from the CPU so that the address will remain stable throughout the complete command cycle. ALE is generated when either S0* or S1* is asserted from the CPU to start a bus cycle. This allows the address latches to be enabled as early as possible to provide sufficient address setup time to the on board memory array. ALE is negated at the start of the command state (Tc) to latch the address while they are guaranteed from the CPU.

The last control signal generated is MEMCYC which is used to control the start of a memory access to the on board memory array. MEMCYC is asserted HIGH at the middle of the Status state (Ts) and is held active during the complete bus cycle. MEMCYC is negated at the end of a bus cycle or the end of the command state (Tc).

Bus Conversion Logic

This block of the CPU Control IC provides the logic to control the 16 bit transfers to and from an 8 bit device or memory. The conversion logic detects when a conversion is required, asserts a wait to the CPU by asserting the READY* line to a HIGH or inactive state, and then generates the required control signals to the data buffers to perform the conversion. A conversion is required during all 16 bit transfers to all I/O devices, the Interrupt Controller, and 8 bit memory. An 8 bit memory cycle is detected by sampling the state of AF16* and/or MEMCS16* at the beginning of a memory cycle. If both are inactive HIGH then the conversion logic acknowledges it as an 8 bit memory cycle. The control signals that are generated are DIRHLB (Direction High Low Byte), ENHHLB (Enable High Low Byte), and CNTLOFF (Control Off). ENHHLB enables the conversion buffer during the conversion cycle and DIRHLB determines the direction of the buffer. CNTLOFF is used to latch the lower 8 bits (D0-D7) during a 16 bit read from an 8 bit device. SA0 is also controlled and generated by the conversion logic so the even byte is read first then SA0 is toggled to a HIGH state so the odd byte will be read.

Wait State Control Logic

The Wait State Control Logic is used to allow slow memory and peripheral devices to be used with a faster CPU. Wait states are controlled by the CPU Control IC by negating the READY* line to the CPU to inactive state. The CPU will not end the cycle in progress until the READY* line is reasserted to an active LOW state. The READY* line is negated at the middle of the Status state (T_s) to an inactive HIGH to guarantee recognition of a wait state by the CPU. The READY* line is reasserted at the middle of the Command state or Wait State to end the existing cycle.

There are several default wait state cycles that are generated by the CPU Control IC for control of all bus cycles. During a 16 bit memory cycle which is determined by the assertion of AFL6* or MEMCS16*, a default of one wait state is automatically inserted. During a 8 bit memory cycle which is determined when both AFL6* and MEMCS16* are negated, a default of four wait states are automatically inserted. All I/O cycles have a default of four wait states inserted to guarantee timing compatibility with existing I/O channel add in boards. All of the default wait state are the same in the 8 MHz (FAST) mode and the 4 MHz (SLOW) mode except for the I/O bus cycles. During an I/O cycles in the 4 MHz (SLOW) mode, a default of only two wait states are inserted. This allows only the memory cycles to be effected to slow down the program speed without degrading the performance of I/O accesses.

All bus cycles can be extended above the default number of wait states by negating the IOCHRDY (I/O Channel Ready) to inactive LOW. The CPU Control IC will continue to insert wait states to the CPU until IOCHRDY is released allowing it to be asserted. IOCHRDY should not be held LOW for more than 15 usec. because Refresh to the DRAMs will be inhibited.

DMA Arbitration Logic

The DMA Arbitration Logic in the CPU Control IC control the arbitration of the bus between the CPU and the DMA Controller. The arbitration logic detects when the DMA is requesting the bus when HRQ (Hold Request) is asserted HIGH. After HRQ is recognized, the arbitration logic will assert CPUHRQ to the CPU to request release of the bus. The CPU will respond to CPUHRQ after finishing the cycle in progress by suspending operation and asserting CPUHLDA to the CPU Control IC. The arbitration Logic will then tri-state the output command signals and generate HLDA to the DMA, allowing it to take control over the bus.

All DMA cycles have a default of one wait state automatically inserted to maintain compatibility with existing I/O channel boards and peripheral devices. The DMA cycle can be extended by and I/O device by applying a LOW state to the IOCHRDY signal. The DMA Controller will be held in a wait state until IOCHRDY is returned to a HIGH state. Refer to Timing Diagrams for Timing specifications of a DMA cycle and DMA Arbitration operation.

The CPU Control IC also controls two functions required in the 80286 architecture during a DMA cycle. In order to guarantee the integrity of an Interrupt Acknowledge (INTA) cycle, which requires two bus cycles to complete, a DMA arbitration inhibit is included in the arbitration logic. After an INTA cycle has started, a DMA cycle will be inhibited until the CPU performs a memory write cycle. The memory write cycle will normally be executed after the INTA cycle due to the stack operation of the CPU required to service an interrupt routine. The inhibit will guarantee the a DMA cycle can not be allowed until both bus cycles of an INTA cycle is complete.

Numerical Co-processor Control Logic

The Numerical Co-processor Control Logic provides an interface for the 80286 CPU to control an 80287 Co-processor. This logic controls the decoding required to select and reset the 80287, control of the BUSY* and ERROR* signals from the 80287 to the CPU, and generating the interrupt signal during an error condition.

The input signal 287CS* is an I/O decode at 0F0-0FFh I/O address. 287CS* is used by the CPU Control IC to generate the Numerical Co-processor Chip Select (NPCS*), and the reset signal (RES287*) to the 80287. Refer to the I/O decode table for details about the further internal decode done by the CPU Control IC.

When the 80287 receives a command to perform a task, it will output the BUSY* signal which is input to the CPU Control IC. The CPU Control IC will then assert BUSY287* to the CPU. During normal operation of the 80287, when it finishes the task, it will negate BUSY* which will in turn negate BUSY287* to the CPU. If the ERROR input from the 80287 is asserted during this busy time, which indicates a 80287 error, BUSY287* output is latched and INT287* is asserted LOW to generate an interrupt to the CPU. Both BUSY287* and INT287* will remain latched in an active LOW state until they are cleared by writing to the I/O address 0F0h or 0F1h. These signals are cleared after a system reset.

RES287* is generated to reset the 80287. It is generated by writing to I/O address 0F1h or when a system reset is issued.

A20 and CPU Reset Control Logic

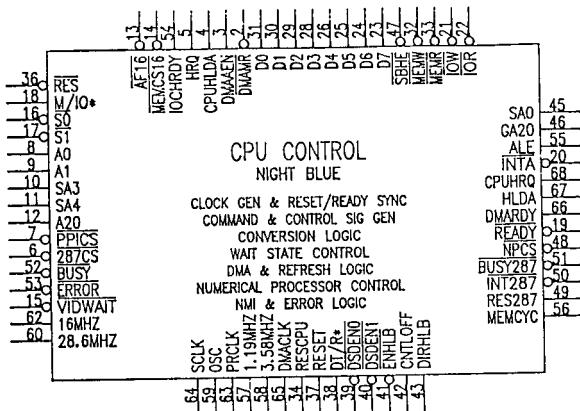
The CPU Control IC incorporates the logic required to control the Address line A20 and to reset the CPU. In the PC architecture, A20 must be held in a LOW state so the some programs that do wrapping will function correctly. After a reset, the output signal GA20 is held in a LOW state. If a program needs to address above the 1 MEG limit, OUT to I/O address 068h with data bit 1 to a HIGH or "1" state will allow A20 to be muxed to GA20.

The CPU can reset itself by doing an I/O wtrie to address 068h with bit 2 at a LOW or "0" state. After a reset has occurred, by Reading I/O address 068, it can be determined it was a power up reset or if the CPU reset itself. Refer to the I/O address map for further details.

I/O Address Map

Address	Description
062	Port C (Write Only)
Bit	
7 6 5 4 3 2 1 0	
1 1 1 1 1 1 1 1	Reserved
1 1 1 1 1 1 1 +---	Reserved
1 1 1 1 1 1 +-----	Reserved
1 1 1 1 1 +-----	Reserved
1 1 1 1 +-----	0 = 4 MHz (SLOW) Mode 1 = 8 MHz (FAST) Mode
1 1 1 1	Default after Reset
1 1 +-----	Reserved
1 +-----	Reserved
+-----	Reserved
+-----	Reserved
068	Port I (Read and Write)
Bit	
7 6 5 4 3 2 1 0	
1 1 1 1 1 1 1 1	Not Used
1 1 1 1 1 1 1 +---	0 = GA20 always 0 Default after Reset
1 1 1 1 1 1 +-----	1 = A20 Muxed to GA20
1 1 1 1 1 +-----	Write Mode 0 = Reset CPU 1 = No Effect
1 1 1 1 1 +-----	Read Mode 0 = Power On Reset 1 = CPU Reset
1 1 1 1 +-----	Not Used
1 1 +-----	Not Used
1 +-----	Not Used
+-----	Not Used
+-----	Not Used
0F0	Clear Numerical Co-processor Busy
0F1	Reset Numerical Co-processor
0F8-0FF	Numerical Co-processor Chip Select

Pin Configuration



Functional Pin Description

Pin#	Pin Name	Type	Description
62	16MHZ	I	16 MHZ Clock Input
63	PRCLK	O	80286 Processor Clock - 16:8 MHZ
64	SCLK	O	System Clock - 8:4 MHZ
65	DMACLK	O	DMA Controller Clock - 4:2 MHZ
60	28.6MHZ	I	28.63636 MHZ Clock Input
59	OSC	O	14.31818 MHZ Clock
58	3.58MHZ	O	3.5795 MHZ Clock for sound chip
57	1.19MHZ	O	1.19 MHZ Clock for Interval Timer chip 8254.
18	MI/O	I	Memory Input/Output from the CPU. Indicates a memory access when HIGH and a I/O access when LOW. Used to generate the memory and I/O command signals for the system.
17	S1*	I	Status Line 1 from the CPU
16	S0*	I	Status Line 0 from the CPU
19	READY*	O	Ready signal to the 80286 Processor. Indicates the current bus be completed when LOW.
68	CPUHRQ	O	CPU hold request signal for the CPU. Indicates DMA transfers by the DMA controller when HIGH. It is also active during refresh cycles.
4	CPUHLDA	I	Hold Acknowledge signal from CPU. Indicates the CPU granting a DMA cycle to the DMA controller when HIGH. And causes all command signals to be tri-stated provided the CNTLOFF output is LOW
33	MEMR*	O	Memory Read Command output signal instructs a memory device to place data on the bus when LOW. It is also active during refresh.

32	MEMW*	I/O	Memory Write Command Input/Output signal. Instructs a memory device to read the data on the bus when LOW.
22	IOR*	I/O	Input/Output Read signal. Indicates a Read cycle is performed with an I/O device or port when LOW.
21	IOW*	I/O	Input/Output Write signal. Indicates a Read cycle is performed with an I/O device or port when LOW.
20	INTA*	O	Interrupt Acknowledge for the Interrupt Controller. It is used by the Interrupt controller to output the interrupt vector onto the data bus when LOW.
55	ALE	O	Address Latch Enable. It is used to hold the address during bus cycle when HIGH.
56	MEMCYC	O	Memory cycle signal. It is used to generate memory control signal, RAS, MUX and CAS when HIGH.
36	RES*	I	Reset signal. It is connected to the power good and used to reset the system when LOW.
37	RESET	O	Reset output signal. It is a synchronized reset signal for general system reset when HIGH.
34	RESCPU	O	CPU Reset output signal. It is used to reset CPU when HIGH.
47	XBHE* (SBHE*)	I/O	Bus High Enable Input/Output signal. It is used to enable the high byte data bus signals when LOW
45	SA0	O	System Address 0.
8	A0	I	Address 0 input signal from CPU. It is used to generate the enable signal for the data bus.

9	A1	I	Address 1 input signal from CPU. It is used to detect the SHUT DOWN condition of the CPU.
10	SA3	I	System Address 3 input signal. It is used to generate the chip select and reset signal for 80287.
11	SA4	I	System Address 4 input signal. It is used to generate the chip select.
12	A20	I	Address 20 from the CPU. It is used to generate GA20 signal when CPUHLDA LOW.
46	GA20	O	Gated Address 20. Address 20 is being negated by XD1 and PORTI (internal). When XD1 is LOW, Gated A20 on the CPU address bus is forced LOW. When XD1 is HIGH, GA20 is transmitted as Address 20.
7	PPICS*	I	Programmable Peripheral Interface Chip Select input signal. It is used to generate Chip Select Signal signal for the peripheral interface device when is LOW.
6	287CS*	I	287 Chip Select input signal. It is used for generating the Numerical Processor Select NPCS for 80287 when is HIGH.
5	HRQ	I	Hold Request input signal from DMA controller. It is used to generate the CPU Hold Request Signal when is HIGH.
67	HLDA	O	Hold Acknowledge output signal for DMA Controller. It is used to provide Hold Acknowledge for DMA controller when is HIGH.
3	DMAAEN	I	DMA Address Enable input Signal from DMA Controller. It is used to provide enable signal for any I/O device during DMA Access to the system memory when is HIGH.

2	DMAMR*	I	DMA Memory Read signal from DMA controller. It is used to generate Memory Read - MEMR* signal when is LOW.
66	DMARDY	O	DMA Ready output signal for DMA Controller. It is used to extend memory Read and Write cycles from the DMA Controller for slower memory or I/O device when is HIGH.
52	BUSY*	I	Busy input signal from 80287. It indicates that 80287 is currently executing a command when is LOW.
53	ERROR*	I	Error input signal from 80287. It indicates an unmasked error condition exists.
51	BUSY287*	O	Busy 80287 output signal for the CPU. It indicates to the processor the operating condition of the 80287 when is LOW.
50	INT287*	O	Interrupt 80287 output signal for the Interrupt controller. It is the interrupt request from 80287
49	RES287	O	Reset 80287 output signal for the 80287. It is used to reset to the 80287 when is HIGH.
48	NPCS*	O	Numerical Processor Chip Select output signal for the 80287. It is used to select the 80287 device when is LOW.
38	DT/R*	O	Data Transmit/Received output signal. It is used to determine the data direction to and from local data bus. It is a write bus cycle when is HIGH and read bus cycle when is LOW.
39	DSDENO*	O	Data Strobe Data Enable 0 output. It is used to enable the data transceivers connected to the low byte (D0-D7) when is LOW.
40	DSDEN1*	O	Data Strobe Data Enable 1 output signal. It is used to enable the data transceivers connected to the high byte (D8-D15) data bus when is LOW.

41	ENHLB*	O	Enable High To Low byte output signal. It is used to perform the high to low conversion when is LOW.
42	CNTLOFF	O	Control Off output signal. It is used to enable the low data bus latch during byte accesses when is LOW.
43	DIRHLD	O	Direction High to Low byte output signal. It is used to perform high to low byte conversion when is HIGH.
13	AF16*	I	AF16* output signal. It is used to control the 16 bit memory accesses and to inhibit the command delays for memory accesses by I/O device when is LOW.
14	MEMCS16*	I	Memory Chip Select input signal. It is used to inhibit command delays when 16 bit memory accesses are made when is LOW.
54	IOCHRDY	I	I/O Channel Ready input signal from I/O device. It is used generate wait states in I/O or memory accesses by I/O device when is HIGH.
15	VIDWAIT*	I	Video Wait input signal from video controller. It is used to generate Video delay ready signal when is HIGH.
31	D0	I/O	Data Bus Bit 0 of the peripheral data bus.
30	D1	I/O	Data Bus Bit 1 of the peripheral data bus.
29	D2	I/O	Data Bus Bit 2 of the peripheral data bus.
28	D3	I/O	Data Bus Bit 3 of the peripheral data bus.
26	D4	I/O	Data Bus Bit 4 of the peripheral data bus.

25	D5	I/O	Data Bus Bit 5 of the peripheral data bus.
24	D6	I/O	Data Bus Bit 6 of the peripheral data bus.
23	D7	I/O	Data Bus Bit 7 of the peripheral data bus.
1,35	VCC	PWR	Power Supply.
27,44 61	VSS	GND	Ground.

ELECTRICAL SPECIFICATIONS

ELECTRICAL PARAMETERS

ABSOLUTE MAX RATINGS (NON-OPERATING, VSS=0.0V)

	MIN	MAX	UNITS
STORAGE TEMPERATURE	-65	+150	Degrees C.
VOLTAGE ON ANY PIN W.R.T GROUND	-0.5	7.0	Volts

OPERATING ELECTRICAL SPECIFICATIONS:

OPERATING AMBIENT	MIN	TYP	MAX	UNITS
AIR TEMP. RANGE	0	25	70	Degrees C
POWER SUPPLIES				
VCC	4.5	5.0	5.5	Volts
VSS	0	0	0	Volts
LEAKAGE CURRENT	MIN	TYP	MAX	UNITS
Vin = 0.0 v			20	Microamps
Vin = 5.0 v	-20			Microamps

DC ELECTRICAL CHARACTERISTICS

<u>INPUT VOLTAGES AND CURRENTS</u>	MIN	TYP	MAX	UNITS
LOGIC "0" (Vil) @ 8ma			0.8	Volts
LOGIC "1" (Vih) @ 8ma	2.0			Volts
INPUT CURRENT			+10	Microamps
All inputs 0.0 \leq Vin \leq 5.0	10			Picofarads
<u>OUTPUT VOLTAGES CURRENT LOADING</u>	MIN	TYP	MAX	UNITS
LOGIC "0" (Vol) @ 8ma			0.45	Volts
LOGIC "1" (Voh) @ 8ma	2.4			Volts

CURRENT LOADING AND CAPACITANCE OF EACH OUTPUT
 At Vol = 0.45 volts, Voh = 2.4 volts

PIN NAME	MIN	UNITS	CAPACITANCE NOTES
<u>OUTPUT</u>			
RESCPU, <u>NPCS</u> , <u>BUSY287</u> , <u>INT287</u>			
RES287, <u>CPUHRO</u> , <u>HLDA</u> , <u>DIRHLB</u>			
CNTLOFF, <u>ENHLB</u> , <u>DMARDY</u>	2	ma	20pf
RESET, <u>GA20</u>	2	ma	50pf
READY, <u>MEMCYC</u> , <u>DT/R</u> , <u>DSDENO</u>			
DSDEN1	4	ma	20pf
SCLK, <u>DMACLK</u> , <u>OSC</u> , 1.19MHZ 3.58MHZ	4	ma	50pf
PRCLK, <u>INTA</u>	8	ma	50pf
ALE	8	ma	50pf
MEMR	8	ma	80pf
<u>INPUT/OUTPUT</u>			
XD0 - XD7	2	ma	80pf
<u>XBHE</u>	4	ma	50pf
<u>MEMW</u>	8	ma	80pf
<u>IOR</u> , <u>IOW</u>	8	ma	115pf
SA0	4	ma	120pf

Timing Diagram Specifications

!SYM!	DESCRIPTION	!MIN	!MAX	!UNITS	!NOTES
1	!PRCLK Period	62	250	!ns	
2	!PRCLK Low Time	25	125	!ns	
3	!PRCLK High Time	25	125	!ns	
4	!S0, S1, M/IO Setup time to PRCLK↓	20		!ns	
5	!S0, S1, M/IO Hold time to PRCLK ↓	1		!ns	
6	!ALE active from S0, S1 ↓	5	25	!ns	
7	!ALE inactive from PRCLK↓	5	25	!ns	
7A	!OSC delay from 28.6 MHZ		30	!ns	
8	!MEMCYC active from PRCLK↓	5	25	!ns	
8A	13.85 MHZ delay from OSC		15	!ns	
9	!MEMCYC inactive delay from PRCLK ↓	8	35	!ns	
9A	1.19 MHZ delay from OSC		15	!ns	
10	!Command active delay from PRCLK ↓	5	45	!ns	
11	!Command inactive delay from PRCLK↓	5	45	!ns	
12	!DT/R active delay from PRCLK↓	5	50	!ns	!Read
13	!DT/R inactive delay from PRCLK ↓	10	40	!ns	!Read
14	!DSDEN0,1 active delay from PRCLK↓	10	50	!ns	!Read
15	!DSDEN0,1 inactive delay from DT/R	3		!ns	!Read
16	!DSDEN0,1 inactive delay from PRCLK↓	8	50	!ns	!Read
17	!DT/R inactive delay from DS DEN0,1	3		!ns	!Read
18	!DSDEN0,1 active delay from PRCLK↓	8	60	!ns	!Write
19	!DSDEN0,1 inactive delay from PRCLK↓	8	50	!ns	!Write

!SYM	DESCRIPTION	!MIN	!MAX	!UNITS	!NOTES
20	!AF16, MEMCS16 setup time to ALE ↓	35		ns	
21	!AF16, MEMCS16 hold time from ALE ↑	1		ns	
22	!PRCLK delay from 16MHZ		50	ns	
23	!RSET hold time from PRCLK ↓	10		ns	
24	!RSET setup time to PRCLK ↓	25		ns	
25	!RESET A/I delay from PRCLK ↓	5	35	ns	
26	!SCLK delay from PRCLK		25	ns	
27	!DMACLK delay from SCLK		15	ns	
28	!RESCPU inactive delay from PRCLK ↓	5	50	ns	
29	!RESCPU active delay from SCLK ↑	5	20	ns	
30	!HRQ setup to DMACLK ↑	15		ns	1
31	!HRQ holdtime from DMACLK ↑	1		ns	1
32	!CPUHRQ A/I delay from CPUHLDA ↑		30	ns	
33	!HLDA active delay from SCLK		25	ns	
34	!HLDA A/I delay from CPUHLDA		30	ns	
35	!HLDA inactive delay from DMACLK ↑		30	ns	
36	!DMAMR setup to DMACLK ↑	15		ns	
37	!CMD tristate elay from CPUHLDA		30	ns	2
38	!MEMR active delay from DMACLK ↑ !(Due to DMAMR)		25	ns	
39	!MEMR inactive delay from DMAMR ↑		25	ns	
40	!MEMR tri-state delay from DMACLK ↑ !(Due to DMAMR)		25	ns	
41	!CMD active delay from CPUHLDA		30	ns	
42	!DMARDY inactive delay !from DMAMR, IOR		30	ns	
43	!DMARDY active delay from DMACLK ↓		30	ns	

!SYM!	DESCRIPTION	!MIN	!MAX	!UNITS	!NOTES
!44	!MEMCYC active delay from <u>MEMW, MEMR</u>	!	30	!ns	!
!45	!MEMCYC inactive delay from !from <u>MEMW, MEMR</u>	!	30	!ns	!
!46	!AO setup time to PRCLK↓	0	!	!ns	!
!47	!AO hold time from PRCLK↓	5	!	!ns	!
!48	!SAO delay from PRCLK ↓	!	50	!ns	!
!49	!XBHE setup time to ALE↓	0	!	!ns	!
!50	!READY inactive delay from PRCLK↓ !(middle of Ts cycle)	!	25	!ns	!
!51	!READY active delay from PRCLK↓	!	24	!ns	!
!52	!CNTLOFF A/I delay from PRCLK	!	30	!ns	!
!53	!DIRHLB, ENHLB active delay from !from <u>IOR, IOW</u>	!	50	!ns	!
!54	!DIRHLB inactive delay !from PRCLK (<u>IOW</u>)	!	50	!ns	!
!55	!ENHLB active delay from PRCLK !(<u>IOW</u> cycle)	!	50	!ns	!
!56	!ENHLB inactive delay from <u>IOR</u>	!	50	!ns	!
!57	!ERROR holdtime from <u>BUSY</u> active	0	!	!ns	!
!58	!BUSY active pulse width	15	!	!ns	!
!59	!ERROR setup to <u>BUSY</u>	10	!	!ns	!
!60	!BUSY287 A/I delay from <u>BUSY</u> A/I	!	30	!ns	!
!61	!ERROR active pulse width	10	!	!ns	!
!62	!INT287 active delay from ! <u>BUSY, ERROR</u>	!	30	!ns	!
!63	!INT287 inactive delay from <u>ERROR</u>	!	30	!ns	!
!64	!BUSY287 inactive delay from <u>IOW</u>	!	40	!ns	!

<u>!SYM</u>	<u>DESCRIPTION</u>	<u>!MIN</u>	<u>!MAX</u>	<u>!UNITS</u>	<u>!NOTES</u>
!65	<u>!SMIO,SA0,SA3,287CS,INTA setup time!</u>	10	!	!ns	!
	<u>!to IOW</u>	!	!	!	!
!66	<u>!SMIO,SA0,SA3,287CS,INTA hold time</u>	1	!	!ns	!
	<u>!from IOW</u>	!	!	!	!
!67	<u>!RES287 active delay from IOW</u>	!	!	40	!ns
!68	<u>!RES387 inactive delay from IOW</u>	!	!	40	!ns
!69	<u>!NPSCS active delay from SMIO,SA3,</u>	!	!	35	!ns
	<u>!287CS, INTA</u>	!	!	!	!
!70	<u>!NPSCS inactive delay from SMIO,SA3,</u>	!	!	35	!ns
	<u>!287CS,INTA</u>	!	!	!	!
!71	<u>!Data Setup to IOW ↑</u>	!	30	!	!ns
!72	<u>!Data Output from IOR ↓</u>	!	!	80	!ns
!73	<u>!Data float from IOR ↑</u>	!	!	50	!ns
!74	<u>!U46 Q outputs from IOW ↑</u>	!	!	80	!ns
!75	<u>!U46 Q outputs clear from RESET ↓</u>	!	!	80	!ns
!76	<u>!A20 GATE delay from IOW ↑</u>	!	!	80	!ns
!77	<u>!A20 GATE delay from RESET ↓</u>	!	!	80	!ns
!78	<u>!GA20 delay from A20 GATE</u>	5	!	50	!ns
!79	<u>!GA20 delay from A20</u>	!	5	!	50 !ns

1. Setup and Hold times are required to guarantee recognition of signal at clock edge.
2. CMD = MEMR, MEMW, IOR and IOW.
A/I = Active and Inactive

TANDY COMPUTER PRODUCTS

1000 TX Power Supplies

TANDY COMPUTER PRODUCTS

1000 TX 67 Watt Single Input Power Supply

1000 TX 67 Watt Single Input Power Supply
Contents

Section	Page
Operating Characteristics	1
Block Diagram	2
Theory of Operation	3

OPERATING CHARACTERISTICS

	MINIMUM	TYPICAL	MAXIMUM	UNITS
Operating Voltage Range	90	120	135	VAC
Line Frequency	47	50/60	63	Hz
Output Voltages				
Vo1	4.85	5.00	5.15	V
Vo2	11.40	12.00	12.60	V
Vo3	-13.20	-12.00	-10.80	V
Output Loads				
Io1	1.25	-	7.0	A
Io2	0.15	-	2.4	A
Io3	0	-	0.25	A
Over Current Protection				
Current Limit ICL1	-	-	14.0	A
ICL2	-	-	4.8	A
ICL3	-	-	1.0	A
Over Voltage Protection				
Crowbar	5.8	-	6.8	V
Output Noise				
Vo1	-	-	50	mV P-P
Vo2	-	-	100	mV P-P
Vo3	-	-	150	mV P-P
Efficiency	63	65	-	%
Holdup Time				
Full Load at Nominal Line	16	-	-	mSec.
Insulation Resistance				
Input to Output	7	1000	-	M ohms
Input to Ground	7	1000	-	M ohms
Isolation				
Input to Ground	1.7	-	-	KVDC

5V Output Voltage Detecting Circuit

The circuit detects the change of output load current compared with the output voltage and AC line input voltage, which feeds back to the control circuit through a photo coupler PHC1 to keep the output voltage stable. The Photo coupler isolates the primary and secondary circuits.

Over-Voltage Protection

When the +5 output voltage rises, between 5.8V to 6.8V, a control signal turns on the photo coupler PHC2 (Photo Thyristor) with the current of zener diode (D11) and stops oscillation by turning on Q3, which turns off Q1 in the switching circuit.

In the case of stopped oscillation, correct the cause of the failure, and reinput the power. The overvoltage protection circuit will reset automatically under good conditions.

The Photo Thyristor isolates the primary and secondary circuits.

TANDY COMPUTER PRODUCTS

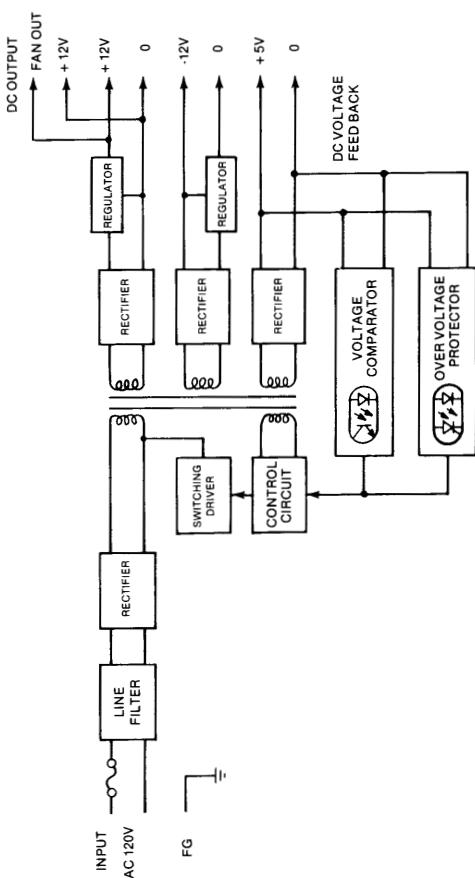
1000 TX 67 Watt Dual Input Power Supply

**1000 TX 67 Watt Dual Input Power Supply
Contents**

Section	Page
Operating Characteristics	1
Block Diagram	2
Theory of Operation	3

OPERATING CHARACTERISTICS

	MINIMUM	TYPICAL	MAXIMUM	UNITS
Operating Voltage Range	90 198	120 240	135 264	VAC
Line Frequency	47	50/60	63	Hz
Output Voltages				
Vo1		4.85	5.00	V
Vo2		11.40	12.00	V
Vo3		-13.20	-12.00	V
Output Loads				
Io1		1.25	-	A
Io2		0.15	-	A
Io3		0	-	A
Over Current Protection				
Current Limit ICL1	-	-	14.0	A
ICL2	-	-	4.8	A
ICL3	-	-	1.0	A
Over Voltage Protection				
Crowbar		5.8	-	V
Output Noise				
Vo1	-	-	50	mV P-P
Vo2	-	-	100	mV P-P
Vo3	-	-	150	mV P-P
Efficiency	63	65	-	%
Holdup Time				
Full Load at Nominal Line	16	-	-	mSec.
Insulation Resistance				
Input to Output	7	1000	-	M ohms
Input to Ground	7	1000	-	M ohms
Isolation				
Input to Ground		1.25	-	KVAC
Input to Output		3.75	-	KVAC



Power Supply Block Diagram

Theory of Operation

AC Input Circuit

This circuit is composed of an AC Power Switch, a fuse, a line filter, and an inrush current limiting circuit and rectifying smoothing circuit. The inrush current limiting circuit controls the charging current to electrolytic capacitors when power is ON. The line filter reduces noise that leaks from the power source to the AC line or that returns from the unit to the power source; it satisfies the specifications of noise regulations.

Control Circuit & Power Converter Circuit

This circuit is a self oscillation switching system, generally called an R.C.C. (Ringing Choke Converter). The R.C.C. circuit does not fix the oscillating frequency. Whenever input voltage is high or the load becomes light, the oscillating frequency will be high.

The current through R4 and R5 supplies transistor Q1's base, then Q1 turns ON. When transistor Q1 is On, the Q1 current excites the transformer T1 and voltage rises in the bias coil of T1(2-3) which leads transistor Q1 positive bias, then transistor Q1 turns ON.

When transistor Q1 turns ON, collector current charges the energy to primary inductance of transformer T1 (4-6). Increasing the collector current of transistor Q1 to the point of:

$$\begin{aligned} I_C &> I_{C(\text{sat})} \\ C &= B \end{aligned}$$

Then, transistor Q1 immediately turns OFF. In a moment, transformer T1 will have negative voltage which will be supplied to the secondary circuit through a rectifier. A Short Circuit Protector is provided to protect transistor Q1 from excess amounts of current when the secondary circuit becomes shorted. When transistor Q2 detects the voltage drop at R13, the collector of Q2 shorts the base and emitter of Q1. Then Q1 stops working so that the circuit protects Q1 from over current.

The over current protector in the -12V line is provided by the three terminal positive voltage regulators IC2, IC3 (built-in current fold back protection), which protects Q1 against excessive current from the -12V line.

5V Output Voltage Detecting Circuit

The circuit detects the change of output load current compared with the output voltage and AC line input voltage, which feeds back to the control circuit through a photo coupler PHC1 to keep the output voltage stable. The Photo coupler isolates the primary and secondary circuits.

Over-Voltage Protection

When the +5 output voltage rises, between 5.8V to 6.8V, a control signal turns on the photo coupler PHC2 (Photo Thyristor) with the current of zener diode (D11) and stops oscillation by turning on Q3, which turns off Q1 in the switching circuit.

In the case of stopped oscillation, correct the cause of the failure, and reinput the power. The overvoltage protection circuit will reset automatically under good conditions.

The Photo Thyristor isolates the primary and secondary circuits.

TANDY COMPUTER PRODUCTS

1000 TX Keyboard

**1000 TX Keyboard
Contents**

Section	Page
Specifications	1
Keyboard Timing	2
Keyboard Scan Codes	3
Keyboard Key Layout	5

KEYBOARD ASSEMBLY

The Tandy 1000 has a 90-key keyboard that includes 12 function keys, a numeric keypad, and special purpose keys for paging. The keyboard is connected to the Main Unit by a coiled cable and operated at a maximum distance of 4 feet from the main unit. Figure 1 shows the interconnecting cable connector to the keyboard assembly. The cable assembly can be disconnected from the keyboard assembly during repair.

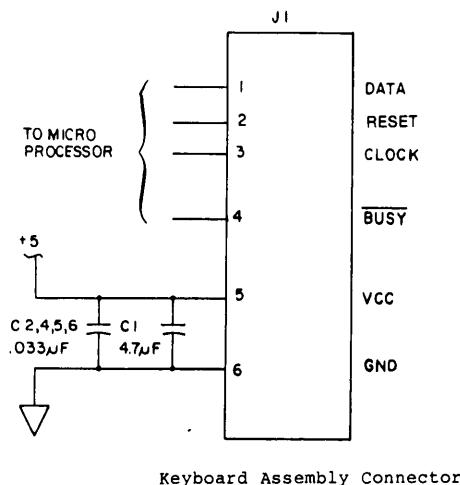


Figure 1

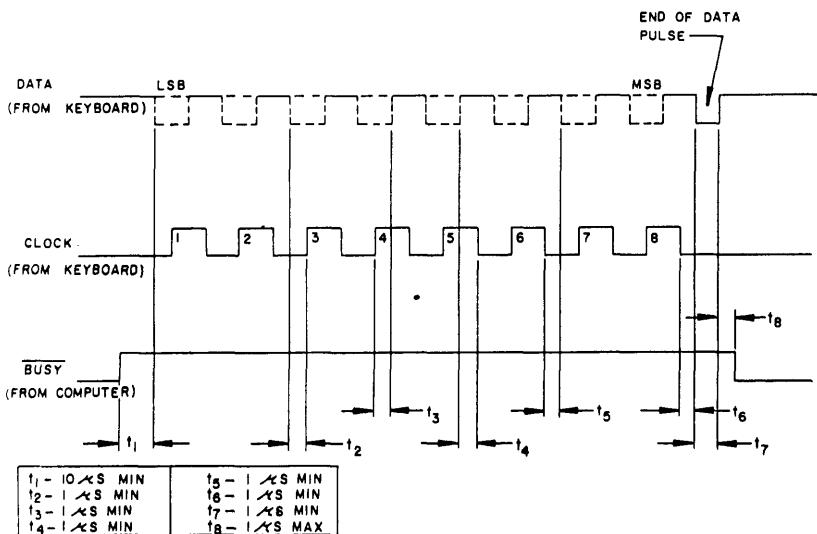
Keyboard Specifications

The keyboard is fully encoded with microprocessor control, and requires +5 VDC supplied from the Main Unit.

1. Key Type — all keys generate "make" and "break" codes. See the Key Code Chart. Break codes are formed by adding 80H to the make code. Keys 49 and 71 have alternate action that "makes" on one actuation of the key and "breaks" on succeeding actuation. No code is generated for these two keys when the key is released.
2. Number of Keys — 90
3. Repeat Strobe — there is a repeat strobe of 66 to 111 mSec when any key is depressed for more than 1 second, with the exception of SHIFT, CTRL, CAPS, ENTER, and NUMBER LOCK.

Keyboard Timing

Figure 2 is the timing chart for the Tandy 1000 Keyboard Assembly.



Keyboard Assembly Timing Chart

Figure 2

Key Code Chart

Key Number	Legend	Scan Code
1	F1	3B
2	F2	3C
3	F3	3D
4	F4	3E
5	F5	3F
6	F6	40
7	F7	41
8	F8	42
9	F9	43
10	F10	44
11	F11	59
12	F12	5A
13	INSERT +	55
14	DELETE -	53
15	BREAK	54
16	ESC	01
17	1 !	02
18	2 @	03
19	3 #	04
20	4 \$	05
21	5 %	06
22	6 ^	07
23	7 &	08
24	8 *	09
25	9 (0A
26	0)	0B
27	- _	0C
28	= +	0D
29	BACKSPACE	0E
30	ALT	38
31	PRINT	37
32	7 (backslash)	47
33	8 (Tilde)	48
34	9 PG UP	49
35	TAB	0F
36	Q	10
37	W	11
38	E	12
39	R	13
40	T	14
41	Y	15
42	U	16
43	I	17
45	P	19

Key Number	Legend	Scan Code
46	{ [1A
47	}]	1B
48	HOLD	46
49	NUM LOCK	45
50	4 :	4B
51	5	4C
52	6	4D
53	CTRL	1D
54	A	1E
55	S	1F
56	D	20
57	F	21
58	G	22
59	H	23
60	J	24
61	K	25
62	L	26
63	;	27
64	;	28
65	ENTER	1C
66		29
67	HOME	58
68	1 END	4F
69	2 (Grave)	50
70	3 PG DN	51
71	CAPS	3A
72	SHIFT	2A
73	Z	2C
74	X	2D
75	C	2E
76	V	2F
77	B	30
78	N	31
79	M	32
80	, <	33
81	. ‡	34
82	/ ?	35
83	SHIFT	36
84		2B
85		4A
86		4E
87	0	52
88		56
89	ENTER	57
90 (Space Key)		

91 thru 95 — reserved for International

Keyboard Layout

Shown below is the keyboard layout and number designation of the keys on the Tandy 1000 keyboard. They should be used with the Key Code Chart for determining data value transmitted by the keyboard.

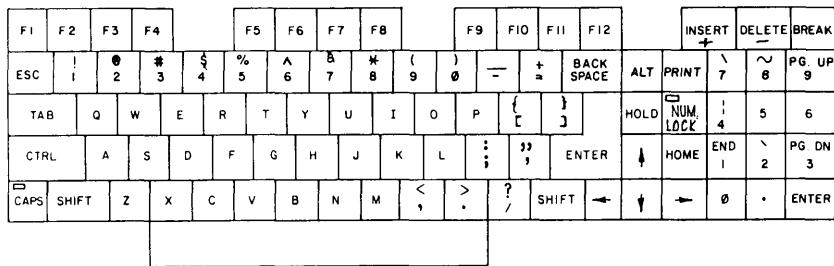


Figure 3 Keyboard Identification

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
35	36	37	38	39	40	41	42	43	44	45	46	47	48	49
53	54	55	56	57	58	59	60	61	62	63	64	65	66	67
71	72	73	74	75	76	77	78	79	80	81	82	83	84	85
91	92			90						93	94	95		

NOTE: KEYS 91 THRU 95 NOT USED ON U.S. VERSION, USED ON INTERNATIONAL VERSION ONLY

Figure 4 Key Number Identification

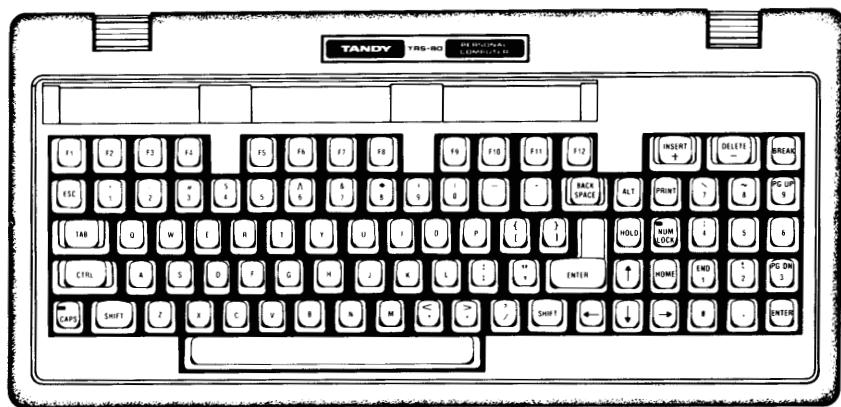


Figure 5

TANDY COMPUTER PRODUCTS

1000 TX Disk Drive

**Specifications
of
MP-F63W-72D**

Double Sided 80 Tracks
Recording Capacity 1MBytes
Transfer Rate 250 Kbits/sec
[TTL interface, without Ready signal]

VALID for MP-F63W-72D,
with the following serial numbers :

SONY CORPORATION

***** Contents *****

1. Description
2. Specifications
 - 2.1 Configuration
 - 2.2 Physical Dimensions
 - 2.3 Performance
 - 2.3.1 Capacity
 - 2.3.2 Transfer Rate
 - 2.3.3 Access Time
 - 2.3.4 Functional
 - 2.3.5 Reliability
 - 2.4 Input Power Requirements
 - 2.4.1 Power Consumption
 - 2.4.2 Supply Voltages
 - 2.5 Environmental Limits and Orientation
 - 2.5.1 Temperature
 - 2.5.2 Humidity
 - 2.5.3 Vibration
 - 2.5.4 Shock
 - 2.5.5 Orientation
3. Interface
 - 3.1 Pin Assignment
 - 3.1.1 Signal Connector
 - 3.1.2 Signal Connector Pin Assignment
 - 3.1.3 Power Supply Connector
 - 3.1.4 Power Supply Connector Pin Assignment
 - 3.2 DC Characteristics of Interface Signals
 - 3.2.1 Output Signals from Drive
 - 3.2.2 Inputs Signals to Drive

3.3 Signal Definitions

- 3.3.1 DRIVE SELECT 0,1,2,3**
- 3.3.2 MOTOR ON**
- 3.3.3 STEP**
- 3.3.4 DIRECTION**
- 3.3.5 HEAD SELECT**
- 3.3.6 WRITE GATE**
- 3.3.7 WRITE DATA**
- 3.3.8 INDEX**
- 3.3.9 TRACK 00**
- 3.3.10 WRITE PROTECT**
- 3.3.11 READ DATA**
- 3.3.12 DISK CHANGE**

3.4 Timing Requirements

- 3.4.1 Head Access**
- 3.4.2 TRACK 00 Signal**
- 3.4.3 Write Data Timing**
- 3.4.4 Read Data Timing**
- 3.4.5 Index Pulse**
- 3.4.6 Disk Change**

3.5 Power on and Power off Requirements

- 3.5.1 Data Protection**
- 3.5.2 Power Supply Sequencing**

3.6 Disk Motor Rotation and Disk Insertion.

3.7 Power-On Reset Timing

4. Safety

5. Power On Initialization

6. Test Points

1. Description

This document describes the specifications for the MP-F63W-72D of which the recording capacity is 1MB, the maximum track-to-track access time is 3 msec and a TTL compatible signal interface.

The main features of the MP-F63W-72D are low power consumption, low height, and high reliability with a simple mechanism and electric circuit.

NB: The specifications defined in this booklet are valid only if the drive is used with Sony media or any other ANSI specification media agreed upon by Sony and the drive customer.

2. Specifications

2.1 Configuration

The drive consists of Read/Write heads, head positioning mechanism, disk motor, interface logic circuit and Read/Write circuit.

2.2 Physical Dimensions

The detailed physical dimensions are shown in Figure 2.1.
The main dimensions are:

- 1) Height : 30 mm
- 2) Width : 101.6 mm
- 3) Depth : 150 mm
- 4) Weight : 480 g max.

2.3 Performance

2.3.1 Recording Capacity

Unformatted capacity : 1.0 Mbyte/disk for MFM
0.5 Mbyte/surface for MFM
6.25 Kbyte/track for MFM

2.3.2 Transfer Rate

Burst transfer rate : 250 Kbits/sec for MFM

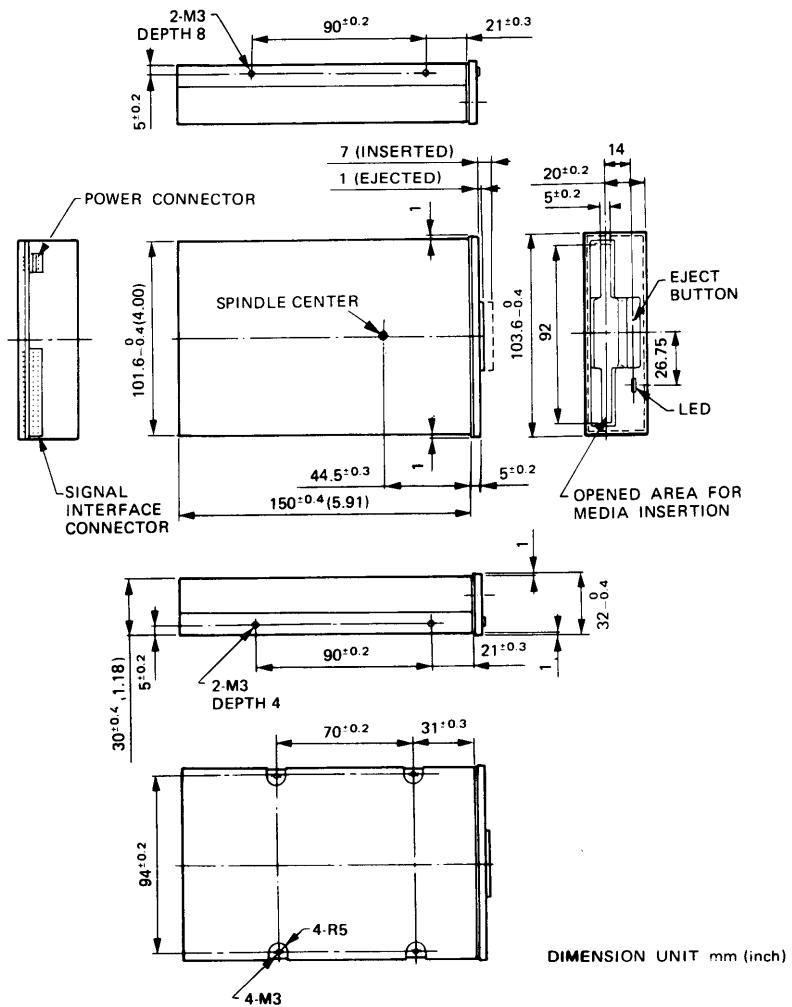


Figure 2-1. PHYSICAL DIMENSIONS

2.3.3 Access Time

- a. Track to Track Slew Rate : 3 msec max.
- b. Head Settling Time : 15 msec max.
- c. Motor Start Time : 500 msec max.
(700 msec max.*)

Motor start time is defined as the time period necessary to stabilise the Motor Rotational Speed variance to less than +/-1.5% after turning the MOTOR ON signal on.

NB. When a disk is inserted in the drive, the Motor Start Time will be 700 msec at maximum, but, after that it will be 500 msec max. as the disk is kept inserted.

2.3.4 Functional

- a. Rotation Speed : 300 rpm
The continuous speed variation is within +/-1.5%.
The instantanuous speed variation is within +/-1.0%.
- b. Recording Density : 8717 BPI (Side 1, Track 79) in a 1MB mode
- c. Track Density : 135 TPI
- d. Cylinders : 80
- e. Tracks : 160
- f. R/W Heads : 2

2.3.5 Reliability

- a. Mean Time Between Failures (MTBF) : 10,000 POH
- b. Mean Time to Repair (MTTR) : 30 minutes
- c. Preventive Maintenance (PM) : Not Required
- d. Components life : 5 years or 15000 POH
- e. Error Rate :
 - 1. Soft Read Error : Less than 1 per 10^9 bits read
 - 2. Hard Read Error : Less than 1 per 10^{12} bits read
 - 3. Seek Error : Less than 1 per 10^6 seeks

2.4 Input Power Requirements

2.4.1 Power Consumption

TTL Interface

Standby	250 mW
Operation (read/write mode)	2.8 W

2.4.2 Supply Voltages

<u>Voltage</u>	<u>Max. Ripple</u>	<u>Current</u>
+12.0V +/-5%	0.1Vpp	Standby 0.3 mA Average 130 mA (Read) Peak 500 mA (Motor Start) Peak 450 mA (stepping during Motor On)
+5.0V +/-5%	0.1Vpp	Standby 50 mA Operating 240 mA

2.5 Environmental Limits

2.5.1 Temperature Range

Operating	: 5°C to 50°C ambient (40°F to 122°F)
Transportation	: -40°C to 60°C (-40°F to 140°F)
Storage	: -20°C to 60°C (-20°F to 140°F)

2.5.2 Humidity Range

Operating	: 8% to 80% relative humidity with a wet bulb temperature of 29°C (85F) and no condensation.
Transportation and Storage	: 5% to 95% relative humidity and no condensation

2.5.3 Vibration

Operating	: The unit can perform Read/Write operations without errors at continuous vibration from 10 to 500 Hz at a maximum of 0.5G along each of the three mutually perpendicular axes.
Transportation and Storage	: The unit can withstand continuous vibration from 10 to 300 Hz with a maximum level of 2.0G along each of the three mutually perpendicular axes without any degradation of any characteristics below the performance specifications.

2.5.4 Shock

Operating : The unit can withstand a shock of 5.0G shock for 11 msec with a 1/2 sine wave shape in each of the three mutually perpendicular axis while performing normal read/write functions without damage or any loss of data.

Transportation and Storage : The unit when unpacked can withstand an 11 msec with a 1/2 sine wave shock of 60G on any of the three mutually perpendicular axes.

2.5.5 Orientation

The drive does not necessarily need to be horizontally positioned. In fact, as seen in figure 2-3, there are many other possible orientations.

3. Signal Interface

3.1 Connector and Pin Assignments

3.1.1 Signal connector

Receptacle : 3M 3414-6500xx or Equivalent

Cable : 3M 3365/34 or Equivalent

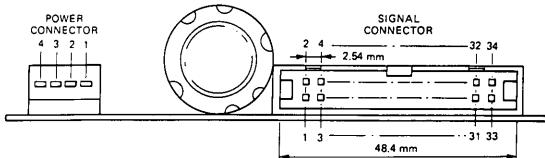


Figure 3-1. PIN ASSIGNMENT (REAR VIEW OF DRIVE)

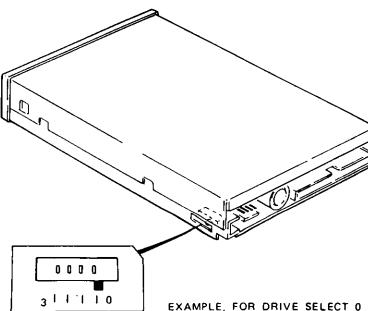


Figure 3-2. DRIVE SELECT SWITCH

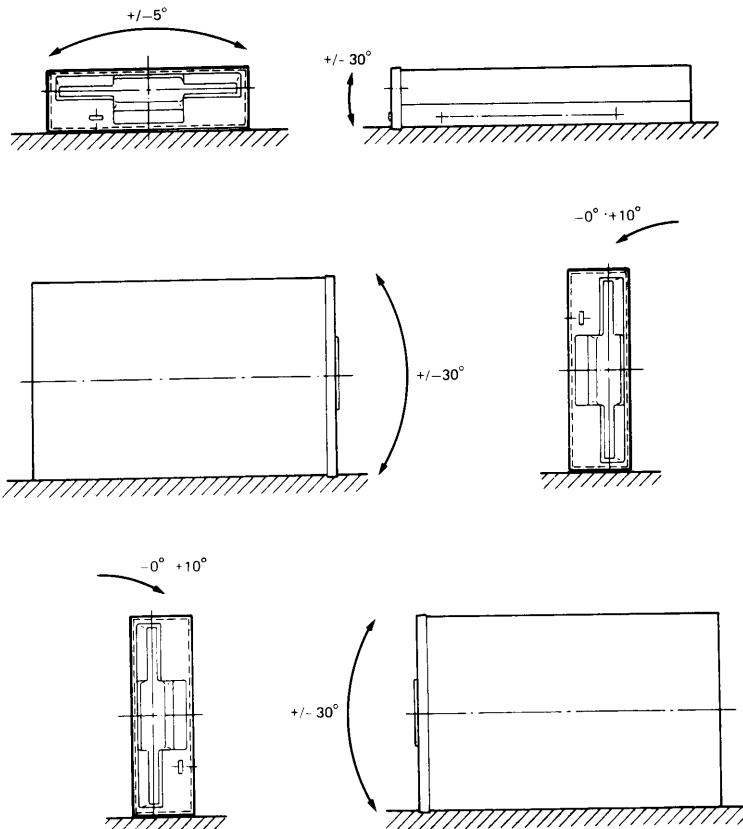


Figure 2-3. ORIENTATIONS

3.1.2 Signal Connector Pin Assignment

PIN	SIGNAL DESCRIPTION	PIN	SIGNAL DESCRIPTION
1	CN	2	N.C.
3	5V	4	N.C.
5	5V	6	DRIVE SELECT 3
7	5V	8	INDEX
9	5V	10	DRIVE SELECT 0
11	5V	12	DRIVE SELECT 1
13	RETURN	14	DRIVE SELECT 2
15	RETURN	16	MOTOR ON
17	RETURN	18	DIRECTION
19	RETURN	20	STEP
21	RETURN	22	WRITE DATA
23	RETURN	24	WRITE GATE
25	RETURN	26	TRACK 00
27	RETURN	28	WRITE PROTECT
29	12V	30	READ DATA
31	12V	32	HEAD SELECT
33	12V	34	DISK CHANGE

3.1.3 Power Supply Connector

Receptacle : AMP 171822-4 or Equivalent
Contact : AMP 170262-1 or Equivalent
Wire : AWG 20

3.1.4 Power Supply Connector Pin Assignment

PIN	SIGNAL DESCRIPTION
1	+5V
2	GND (+5V Return)
3	GND (+12V Return)
4	+12V

3.2 DC Characteristics of Interface Signals

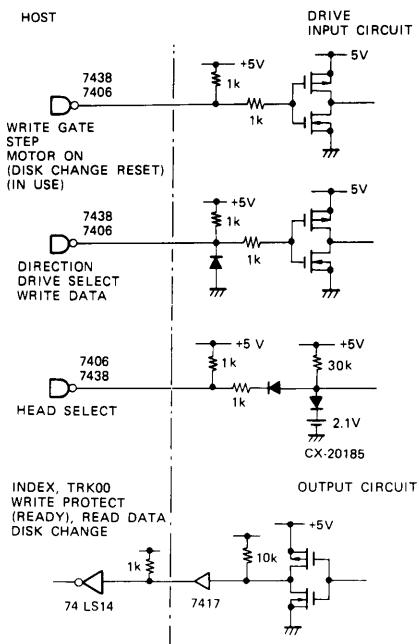
3.2.1 Output Signal from Drive

Name	Output <u>I_{OH}(mA)</u>	Current <u>I_{OL}(mA)</u>	Output <u>V_{OH}(V)</u>	Voltage <u>V_{OL}(V)</u>
TTL interface	0.25	40		0.7
All outputs				

3.2.2 Input Signal to Drive

Name	Input Current <u>V_{IN}=0.4V</u> <u>I_{IL}(mA)</u>	Input Voltage Threshold	
		<u>V_{IH}(V)</u>	<u>V_{IL}(V)</u>
TTL interface	-5.0 (at +0.4V)	2.2	0.8
All inputs			

3.2.3 Recommended Circuit for Signal Interface



The Interface signals in parenthesis are only for MP-F63W-72D
The line from the drive to the controller should be pulled up by a resistor of 1K ohm.

The cable length must be less than 1.5m. (4.92ft.)

Recommended driver IC : 7406, 7438

3.3 Signal Definitions

3.3.1 DRIVE SELECT 0,1,2,3

The select lines are used to enable or disable all other interface lines except a MOTOR ON line. When the SELECT line is true (low), the drive is enabled and is considered active. When the SELECT line is false (high), all controller inputs except the MOTOR ON line are ignored and all output lines are disabled.

NB. IN USE (LED) lamp:

When a drive is selected, the IN USE lamp on the selected drive is turned on, and when a drive is not selected, it is turned off.

3.3.2 MOTOR ON

When this input is true (low) and a disk is inserted, the spindle motor will start to run. When this line is made false (high) or a disk is ejected, the spindle motor will decelerate and stop.

However, if the MOTOR ON signal becomes false (high) during either a write or erase operation, the disk motor will not stop rotating until both the ERASE GATE signal and the WRITE GATE signal become false (high).

3.3.3 STEP

When a drive is selected, a true (low) pulse on this line will cause the Read/Write head to move to the adjacent track. The direction of the head movement is determined by the DIRECTION input at the trailing edge of the pulse. The step operation can be performed even if there is no disk inserted in the drive.

3.3.4 DIRECTION

When a drive is selected, a false (high) level on this input will cause a STEP pulse input to move the Read/Write head away from the disk spindle. A true (low) level will cause a STEP input to move the Read/Write head toward the drive spindle.

3.3.5 HEAD SELECT

When a drive is selected, a true (low) level on this input will cause Head 1 (upper) to be selected. A false (high) level on this input will cause Head 0 (lower) to be selected.

If the HEAD SELECT signal changes during either write or erase operation, the head is not be changed until both ERASE GATE and WRITE GATE signal becomes high (false).

3.3.6 WRITE GATE

When this line is made true (low) while a drive is selected, the write current circuits are enabled and information may be written under control of the WRITE DATA input.

3.3.7 WRITE DATA

If the WRITE GATE is true (low), a true pulse (low) on the WRITE DATA line causes a bit to be written on the disk. Pulses on this line is neglected when WRITE GATE signal is false (high).
No pre-compensation is required.

3.3.8 INDEX

When the drive is selected, a true (low) pulse is generated on this line by each revolution of the spindle.

3.3.9 TRACK 00

This line is true (low) when the drive is selected and the Read/Write head is positioned on track 00.

3.3.10 WRITE PROTECT

If a write-protect disk is inserted while a drive is selected, this line will be true (low) and the drive will not be able to write data. At all other times, except when a disk is ejected while the drive is selected, this line will be false (high).

3.3.11 READ DATA

When the drive is selected, a true (low) pulse is generated on this line every time a bit is detected.

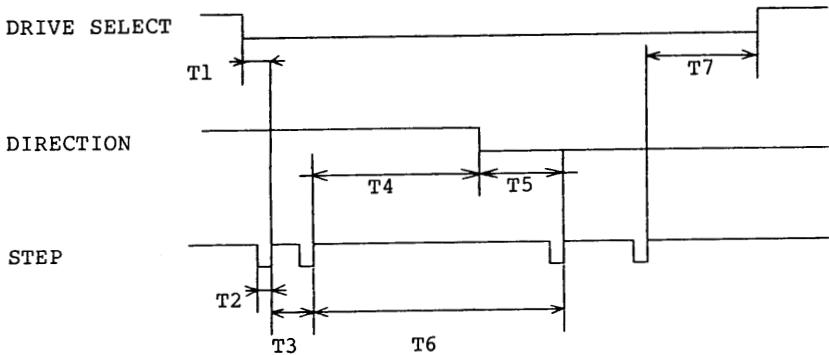
3.3.12 DISK CHANGE

This line is true (low) whenever a disk is removed from the drive. The line remains true (low) until both the following conditions have been met:

1. A disk is inserted,
and
2. A STEP pulse has been received when the drive is selected.

3.4 Timing Requirements

3.4.1 Head Access



T_1 : 0.5 us min.

T_2 : 1.3 us min.

T_3 : 3.0 ms min.

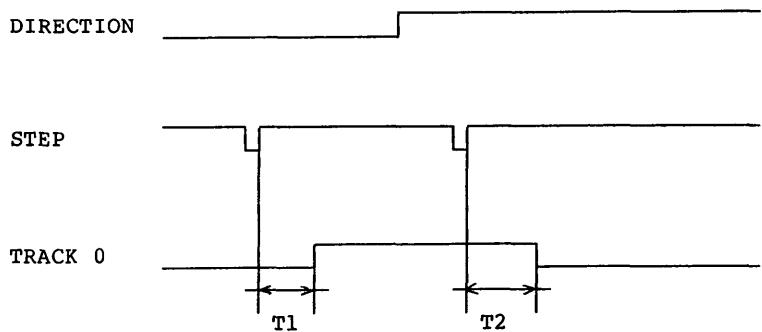
T_4 : 2.4 us min.

T_5 : 0.5 us min.

T_6 : 18 ms min.

T_7 : 2.5 us min.

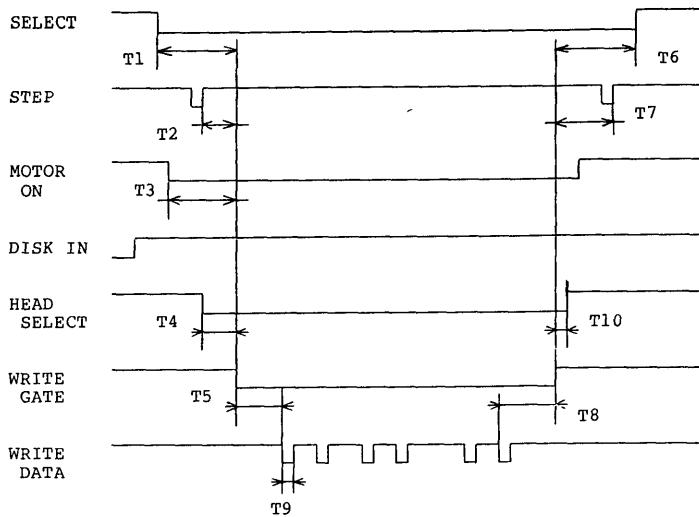
3.4.2 TRACK 00 Signal



T1 : 2.9 msec max.

T2 : 2.9 msec max.

3.4.3 Write Data Timing



T1 : 0.5 us min. T6 : 0.5 us min.

T2 : 18 ms min. T7 : 1020 us min.

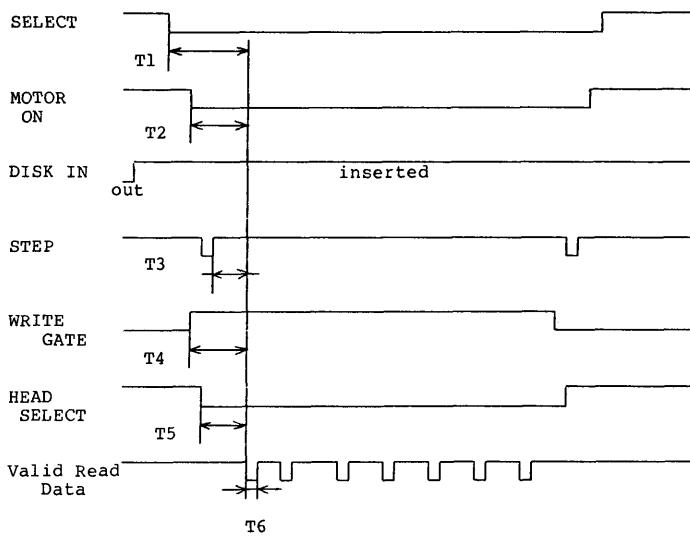
T3 : 500 ms max.* T8 : 8.0 us min.

T4 : 100 us min. T9 : 150 ns min., 2000 ns max.

T5 : 8 us max. T10 : See 3.3.5

*NB. When a disk is inserted in the drive, the Motor Start Time is 700 msec at maximum, but, after that, it is 500 msec max. as the disk is kept inserted.

3.4.4 Read Data Timing



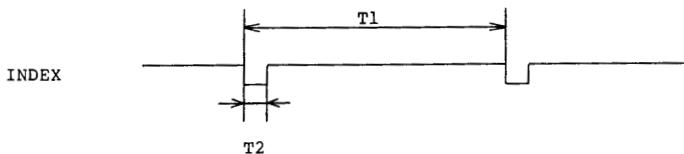
T1 : 0.5 us max. T4 : 1050 us max.

T2 : 500 ms max.* T5 : 100 us max.

T3 : 18 ms max. T6 : 550 ns min., 1200 ns max.

NB. When a disk is inserted in the drive, the T2 is 700 msec at maximum, but, after that it is 500 msec max. as the disk is kept inserted.

3.4.5 Index Pulse

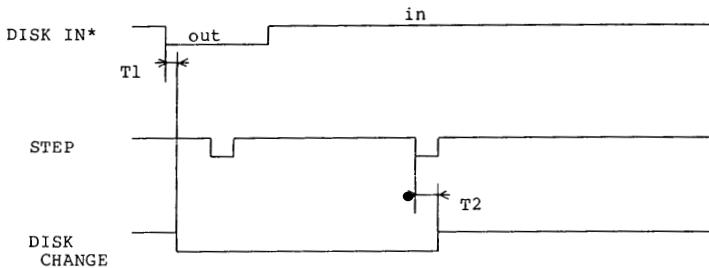


T_1^* : 197 ms min., 203 ms max.

T_2 : 1.25 ms min., 1.45 ms max.

*When the disk motor rotation is at the stationary state.

3.4.6 Disk Change



T_1 : 0.5 us max. T_2 : 1.6 us max.

*DISK IN, the disk-in sensor signal inside the drive, is high when a disk is inserted in the drive.

3.5 Power On and Power off Requirements

3.5.1 Data Protection .

Turning power on or off does not cause any damage to recorded data on the disk as the drive is not in the midst of writing when the power is shut off or supplied.

3.5.2 Power Supply Sequencing

No special supply sequencing is required by the disk drive as long as both the 5V and 12V power supplies have a monotonic rise time of less than 100msec. Then the power is turned off, although there are no sequencing or timing requirements, both power supplies must fall monotonically to 0V.

3.6 Disk motor rotation and Disk Insertion.

Even if the MOTOR ON signal is true (low), the disk motor does not rotate until a disk is inserted.

4. Safety

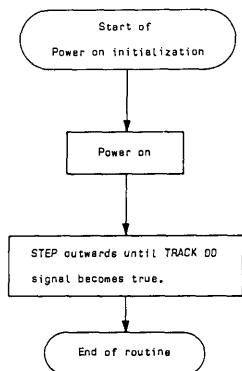
MP-F63W-00D will meet the following product safety regulations:

U.L. 478
C.S.A. C.22.2, No.154
U.L. 94V-0 for Front Bezel

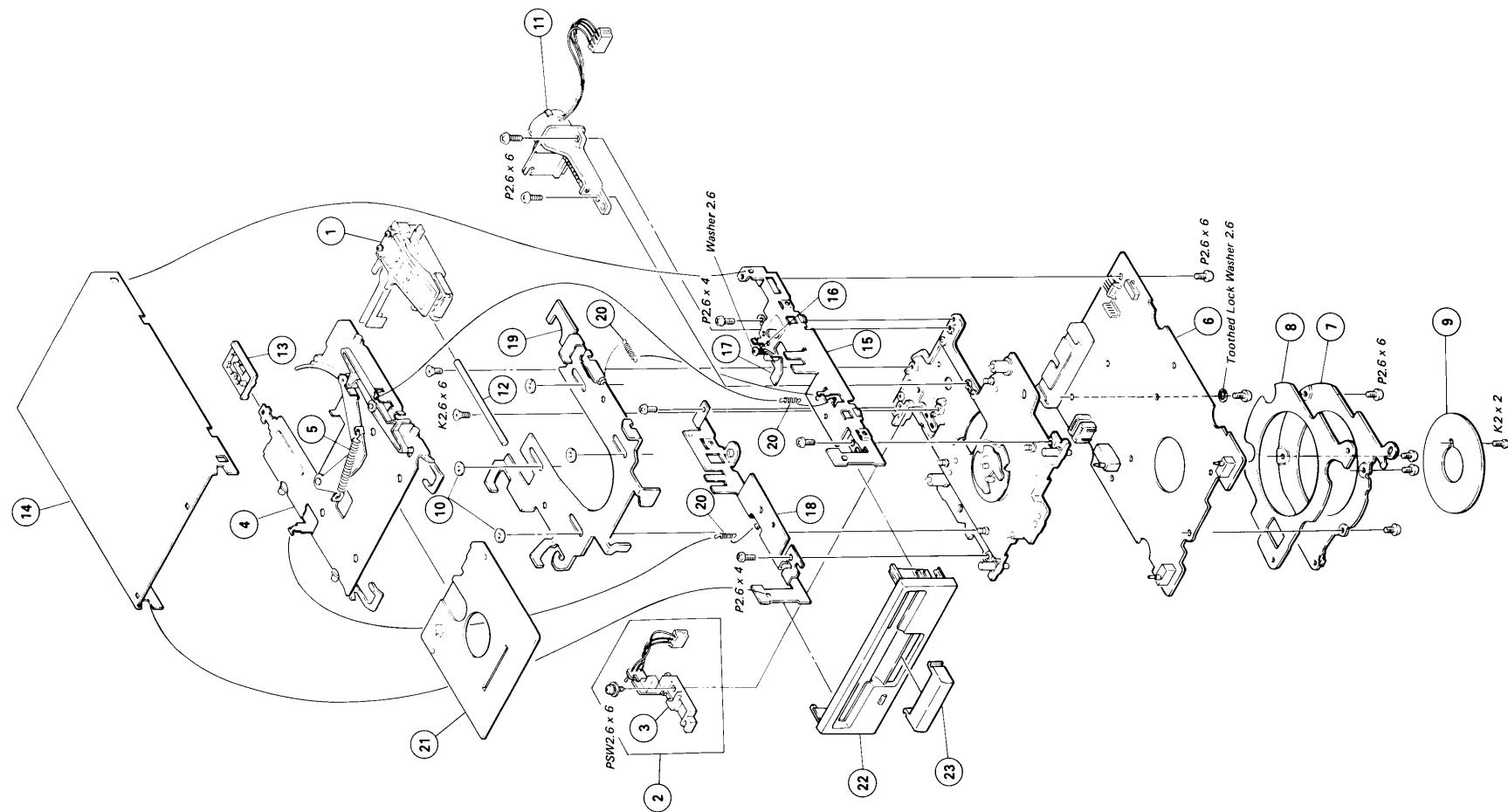
5. Power On Initialization

In order to reduce the peak current requirement when used in a daisy chain, the MP-F63W-72D has been designed not to seek track 00 automatically. If all the drives connected in the daisy chain sought track 00 simultaneously, this would place a significant power drain on the host system. Thus, the host system must perform the following routine just after power on in order to reset the track counter inside the drive.

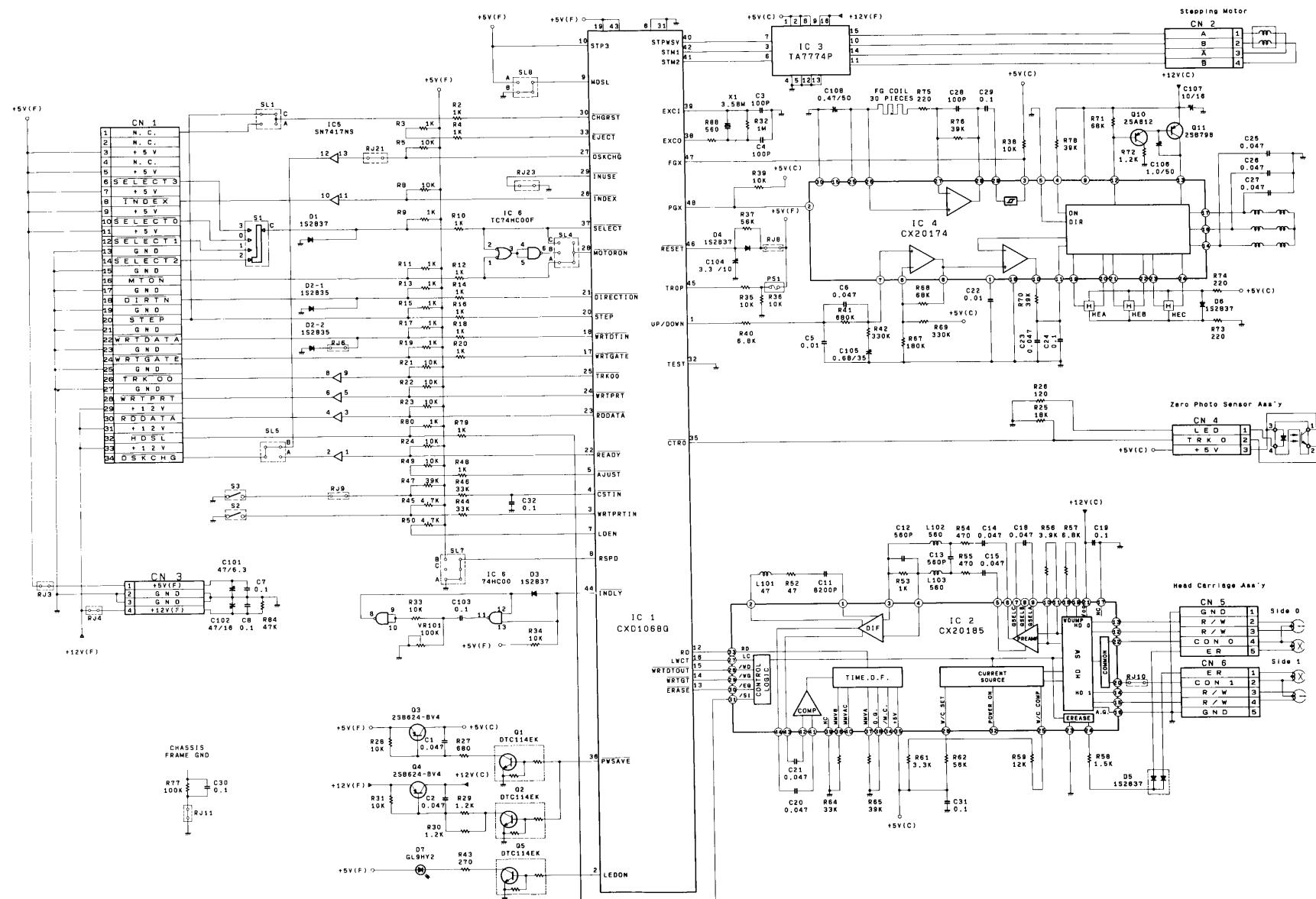
Power On Initialization



PARTS ASS'Y LOCATION



8790142



8790142

TANDY COMPUTER PRODUCTS

1000 TX Options

SOFTWARE

Software Contents

BIOS Services

Device I/O Services	1
Keyboard	3
Video Display	5
Serial Communications	13
Line Printer	16
System Clock	17
Floppy Disk I/O Support	19
Equipment	22
Memory Size	22
EEROM	23

Keyboard ASCII and Scan Codes

Keyboard ASCII and Scan Codes	24
-------------------------------------	----

MS-DOS Memory Map

ROM BIOS Data Area	29
Additional Data Area	33

BIOS Services

Device I/O Services

Introduction

The BIOS (Basic Input/Output System) is the lowest-level interface between other software (application programs and the operating system itself) and the hardware. The BIOS routines provide various device input/output services, as well as boot strap and print screen and other services. Some of the services that BIOS provides are not available through the operating system, such as the graphics routines.

All calls to the BIOS are made through software interrupts (that is, by means of assembly language "INT x" instructions). Each I/O device is provided with a software interrupt, which transfers execution to the routine.

Entry parameters to BIOS routines are normally passed in CPU registers. Similarly, exit parameters are generally returned from these routines to the caller in CPU registers. To insure BIOS compatibility with other machines, the register usage and conventions are, for the most part, identical.

The following pages describe the entry and exit requirements for each BIOS routine. To execute a BIOS call, load the registers as indicated under the "Entry Conditions." (Register AH will contain the function number in cases where a single interrupt can perform more than one operation.) Then issue the interrupt given for the call. The example, the following can be used to read a character from the keyboard:

```
MOV AH, 0  
INT 16H
```

Upon return, AL contains the ASCII character and AH the keyboard scan code.

Note: All registers except those used to return parameters to the caller are saved and restored by the BIOS routines.

Below is a quick reference list of software interrupts for all device I/O and system status services.

Service	Software Interrupts
Keyboard	16 hex (22 dec)
Video Display	10 hex (16 dec)
Serial Communications	14 hex (20 dec)
Line Printer	17 hex (23 dec)
System Clock	1A hex (26 dec)
Floppy Disk	13 hex (19 dec)
Equipment	11 hex (17 dec)
Memory Size	12 hex (18 dec)

Keyboard

16 hex (22 dec)

Function Summary:

AH = 0: Read Keyboard (destructive with wait)

AH = 1: Scan Keyboard (nondestructive, no wait)

AH = 2: Get Current Shift Status

Function Descriptions:

Read Keyboard

Read the next character typed at the keyboard. Return the ASCII value of the character and the keyboard scan code, removing the entry from the keyboard buffer (destructive read).

Entry Conditions:

AH = 0

Exit Conditions:

AL = *ASCII value of character*

AH = *keyboard scan code*

Scan Keyboard

Set up the zero flag (Z flag) to indicate whether a character is available to be read from the keyboard or not. If a character is available, return the ASCII value of the character and the keyboard scan code. The entry remains in the keyboard buffer (non-destructive read).

Entry Conditions:

AH = 1

Exit Conditions:

Z = no character is available

NZ = a character is available, in which case:

 AL = *ASCII value of character*

 AH = *keyboard scan code*

Get Shift Status

Return the current shift status.

Entry Conditions:

AH = 2

Exit Conditions:

AL = *current shift status (bit settings: set = true, reset = false)*

- bit 0 = RIGHT SHIFT key depressed
- bit 1 = LEFT SHIFT key depressed
- bit 2 = CTRL (control) key depressed
- bit 3 = ALT (alternate mode) key depressed
- bit 4 = SCROLL state active
- bit 5 = NUMBER lock engaged
- bit 6 = CAPS lock engaged
- bit 7 = INSERT state active

Video Display

These routines provide an interface to the video display, which is the output half of the console (CON) device. MS-DOS considers the video display to be the default standard output (STDOOUT) device.

Software Interrupts:

10 hex (16 dec)

Function Summary:

Control Routines:

- AH = 0: Set CRT Mode
- AH = 1: Set Cursor Type
- AH = 2: Set Cursor Position
- AH = 3: Get Cursor Position
- AH = 4: Read Light Pen Position
- AH = 5: Select Active Page
- AH = 6: Scroll Active Page Up
- AH = 7: Scroll Active Page Down

Text Routines:

- AH = 8: Read Attribute/Character
- AH = 9: Write Attribute/Character
- AH = 10: Write Character Only

Graphics Routines:

- AH = 11: Set Color Palette
- AH = 12: Write Dot
- AH = 13: Read Dot

Other Routines:

- AH = 14: Write TTY* to active page
- AH = 15: Get CRT Mode
- AH = 16: Set Palette Registers

*Screen width is determined by the mode previously set. Some "control" characters (ASCII 00H-1FH) perform the usual special terminal function. These include (but are not limited to) BEL (07H), BS (08H), LF (0AH), and CR (0DH).

Function Descriptions:

Set CRT Mode

Entry Conditions:

AH = 0

AL = *mode value*, as follows:

Alpha Modes

AL = 0: 40x25 black and white

AL = 1: 40x25 color

AL = 2: 80x25 black and white

AL = 3: 80x25 color

Graphics Modes

AL = 4: 320x200 color graphics

AL = 5: 320x200 black and white graphics with 4 shades

AL = 6: 640x200 black and white graphics with 2 shades

AL = 7: Reserved

Additional Modes

AL = 8: 160x200 color graphics with 16 colors

AL = 9: 320x200 color graphics with 16 colors

AH = A: 640x200 color graphics with 4 colors

Note: If the high order bit of the AL register is 1 then the video buffer is not cleared.

Set Cursor Type

Set the cursor type and attribute.

Entry Conditions:

AH = 1

CH = *bit values*:

bits 5-6 = cause an invisible or erratically blinking cursor

bits 5-6 = 00 produces a visible, blinking cursor.

bits 4-0 = start line for cursor within character cell

CL = *bit values*:

bits 4-0 = end line for cursor within character cell

Set Cursor Position

Write (set) cursor position.

Entry Conditions:

AH = 2

BH = *page number (must be 0 for graphics modes)*

DH = *row (0 = top row)*

DL = *column (0 = leftmost column)*

Get Cursor Position

Read (get) cursor position.

Entry Conditions:

AH = 3

BH = *page number (must be 0 for graphics modes)*

Exit Conditions:

DH = *row of current cursor position (0 = top row)*

DL = *column of current cursor position (0 = leftmost column)*

CX = *cursor type currently set [1]:*

See Set Cursor Type (AH = 1) above.

Read Light Pen Position

Reads light pen position.

Entry Conditions:

AH = 4

Exit Conditions:

AH = 0: light pen switch not activated

AH = 1: light pen values in registers

DH = *row of current light pen position*

DL = *column of current light pen position*

CH = *raster line (0-199)*

BX = *pixel column (0-319 or 0-639)*

Select Active Page

Select active display page (valid in alpha mode only).

Entry Conditions:

AH = 5

AL = 0 through 7: *new page value for modes 0, 1*

AL = 0 through 3: *new page value for modes 2, 3*

AL = 80H: *read CRT/CPU page registers*

AL = 81H: *set CPU page register to value in BL*

AL = 82H: *set CRT page register to value in BH*

AL = 83H: *set both CRT and CPU page registers in BH and BL*

Exit Conditions:

If bit 7 of AL = 1 upon entry, BH = *contents of CRT page register*
BL = *contents of CPU page register*

Scroll Up

Scroll active page up.

Entry Conditions:

AH = 6

AL = *numbers of lines to scroll. The number of lines that will be left blank at the bottom of the window. (0 means blank entire window)*

CH = *row of upper left corner of scroll window*

CL = *column of upper left corner of scroll window*

DH = *row of lower right corner of scroll window*

DL = *column of lower right corner of scroll window*

BH = *attribute (alpha modes) or color (graphics modes) to be used on blank line*

Attributes:

Color modes:

foreground color:

bit 0 = blue

bit 1 = green

bit 2 = red

bit 3 = intensity

All bits off = black

background color:

bit 4 = blue

bit 5 = green

bit 6 = red

bit 7 = blink

All bits off = white

Scroll Down

Scroll active page down.

Entry Conditions:

AH = 7

AL = *number of lines to scroll (0 means blank entire window)*

CH = *row of upper left corner of scroll window*

CL = *column of upper left corner of scroll window*

DH = *row of lower right corner of scroll window*

DL = *column of lower right corner of scroll window*

BH = *attribute (alpha modes) or color (graphics modes) to be used on blank line. See Scroll Up (AH = 6) for attribute values and Set Color Palette (AH = 11) for color values.*

Read Attribute or Color/Character

Read a character and its attribute or color at the current cursor position.

Entry Conditions:

AH = 8

BH = *display page number (not used in graphics modes)*

Exit Conditions:

AL = *character read*

AH = *attribute of character (alpha modes only)*

Write Attribute or Color/Character

Write a character and its attribute or color at the current cursor position.

Entry Conditions:

AH = 9

BH = *display page number (not used in graphics modes)*

CX = *number of characters to write*

AL = *character to write*

BL = *attribute of character (for alpha modes) or color of character (for graphics modes; if bit 7 of BL is set, the color of the character is XOR'ed with the color value). See Scroll Up (AH = 6) for attribute values and Set Color Palette (AH = 11) for color values.*

Write Character Only

Write character only at current cursor position.

Entry Conditions:

AH = 10

BH = *display page number (valid for alpha modes only)*

CX = *number of characters to write*

AL = *character to write*

BL = *color of character (graphics mode)*

Set Color Palette [3]

Select the color palette.

Entry Conditions:

AH = 11

BH = 0 Set background color (0-15) to color value in BL.

BL = *color value (0 = black / 1 = blue / 2 = green / 3 = cyan / 4 = red / 5 = magenta / 6 = yellow / 7 = gray / 8 = dark gray / 9 = light blue / 10 = light green / 11 = light cyan / 12 = light red / 13 = light magenta / 14 = light yellow / 15 = white)*

or

BH = 1 Set default palette to the number (0 or 1) in BL.

In black and white modes:

BL = 0: 1 for white

BL = 1: 1 for black

In 4 color graphics modes:

BL = 0 (1 = green / 2 = red / 3 = yellow)

BL = 1 (1 = cyan / 2 = magenta / 3 = white)

In 16 color graphics modes:

(1 = blue / 2 = green / 3 = cyan / 4 = red / 5 = magenta
/ 6 = yellow / 7 = light gray / 8 = dark gray / 9 = light
blue / 10 = light green / 11 = light cyan / 12 = light red /
13 = light magenta / 14 = yellow / 15 = white)

Note: For alpha modes palette entry 0 indicates the border color.
For graphics mode palette entry 0 indicates the border and the
background color.

Write Dot

Write a pixel (dot).

Entry Conditions:

AH = 12

DX = *row number*

CX = *column number*

AL = *color value* (When bit 7 of AL is set, the resultant color
value of the dot is the exclusive OR of the current dot color
value and the value in AL.)

Read Dot

Read a pixel (dot).

Entry Conditions:

AH = 13

DX = *row number*

CX = *column number*

Exit Conditions:

AL = *color value of dot read*

Write TTY

Write a character in teletype fashion. (Control characters are interpreted in the normal manner.)

Entry Conditions:

AH = 14

AL = *character to write*

BL = *foreground color (graphics mode)*

Get CRT Mode

Get the current video mode.

Entry Conditions:

AH = 15

Exit Conditions:

AL = *current video mode; see Set CRT Mode (AH = 0) above for values*

AH = *number of columns on screen*

BH = *current active display page*

Set Palette Registers

Sets palette registers.

Entry Conditions:

AH = 16

AL = 0 Set Palette register

 BL = *number of the palette register (0-15) to set*

 BH = *color value to store*

AL = 1 Set border color register

 BH = *color value to store*

AL = 2 Set palette color value to store and border registers

 ES:DX :points to a 17 byte list.

 bytes 0-15 = *values for palette registers 0-15*

 byte 16 = *value for the border register*

Note: CS,SS,DS,ES,BX,CX,D_X are preserved.

Serial Communications

These routines provide asynchronous byte stream I/O from and to the RS-232C serial communications port. This device is labeled the auxiliary (AUX) I/O device in the device list maintained by MS-DOS.

Software Interrupts:

14 hex (20 dec)

Function Summary:

AH = 0: Reset Comm Port

AH = 1: Transmit Character

AH = 2: Receive Character

AH = 3: Get Current Comm Status

DX = communication port number (0 or 1).

Function Descriptions:

Reset Comm Port

Reset (or initialize) the communication port according to the parameters in AL, DL, and DH.

Entry Conditions:

AH = 0

AL = RS-232C parameters, as follows:

DX = port number (0 or 1)

7	6	5	4	3	2	1	0
Baud Rate			Parity		Stop Bits		Word Length

000 = 110 baud x0 = none 0 = 1 bit 10 = 7 bits

001 = 150 baud 01 = odd 1 = 2 bits 11 = 8 bits

010 = 300 baud 11 = even

011 = 600 baud

100 = 1200 baud

101 = 2400 baud

110 = 4800 baud

111 = 9600 baud

Exit Conditions:

AX = RS-232 status; see Get Current Comm Status (AH = 3) following

Transmit Character

Transmit (output) the character in AL (which is preserved).

Entry Conditions:

AH = 1

AL = *character to transmit*

DX = *port number (0 or 1)*

Exit Conditions:

AH = *RS-232 status; see Get Current Comm Status (AH = 3) below* (If bit 7 is set, the routine was unable to transmit the character because of a timeout error.)

AL is preserved.

Receive Character

Receive (input) a character in AL (wait for a character, if necessary). On exit, AH will contain the RS-232 status, except that only the error bits (1,2,3,4,7) may be set; the timeout bit (7), if set, indicates that data set ready was not received and the bits in AH are not meaningful. Thus, AH is non-zero only when an error occurred.

Entry Conditions:

AH = 2

DX = *port number (0 or 1)*

Exit Conditions:

AL = *character received*

AH = *RS-232 status; see Get Current Comm Status (AH = 3) below*

Get Current Comm Status

Read the communication status into AX.

Entry Conditions:

AH = 3

DX = *port number (0 or 1)*

Exit Conditions:

AH = *RS-232 status*, as follows (*set = true*):

- bit 0 = data ready
- bit 1 = overrun error
- bit 2 = parity error
- bit 3 = framing error
- bit 4 = break detect
- bit 5 = transmitter holding register empty
- bit 6 = transmitter shift register empty
- bit 7 = timeout occurred

AL = *modem status*, as follows (*set = true*):

- bit 0 = delta clear to send
- bit 1 = delta data set ready
- bit 2 = trailing edge ring detector
- bit 3 = delta receive line signal detect
- bit 4 = clear to send
- bit 5 = data set ready
- bit 6 = ring indicator
- bit 7 = receive line signal detect

Line Printer

These routines provide an interface to the parallel line printer. This device is labeled "PRN" in the device list maintained by the operating system.

Software Interrupts:

17 hex (23 dec)

Function Summary:

AH = 0: Print Character

AH = 1: Reset Printer Port

AH = 2: Get Current Printer Status

Function Descriptions:

Print Character

Print a character.

Entry Conditions:

AH = 0

AL = *character to print*

DX = *printer to be used (0-2)*

Exit Conditions:

0AH = *printer status; see Get Current Printer Status (AH = 2) below*

(If bit 0 is set, the character could not be printed because of a timeout error.)

Reset Printer Port

Reset (or initialize) the printer port.

Entry Conditions:

AH = 1

DX = *printer to be used (0-2)*

Exit Conditions:

AH = *printer status; see Get Current Printer Status (AH = 2) below*

Get Current Printer Status

Read the printer status into AH.

Entry Conditions:

AH = 2

Exit Conditions:

DX = *printer to be used (0-2)*

AH = *printer status, as follows (set = true):*

- bit 0 = timeout occurred
- bit 1 = [unused]
- bit 2 = [unused]
- bit 3 = I/O error
- bit 4 = selected
- bit 5 = out of paper
- bit 6 = acknowledge
- bit 7 = not busy

System Clock

These routines provide methods of reading and setting the clock maintained by the system. This device is labeled CLOCK in the device list of the operating system. An interface for setting the multiplexer for audio source is also provided.

Software Interrupts:

1A hex (26 dec)

Function Summary:

AH = 0: Get Time of Day

AH = 1: Set Time Of Day

AH = 80H: Set Up Sound Multiplexer

The clock runs at the rate of 1,193,180/65,536 per second (about 18.2 times per second).

Function Descriptions:

Get Time Of Day

Get (read) the time of day in binary format.

Entry Conditions:

AH = 0

Exit Conditions:

CX = *high (most significant) portion of clock count*

DX = *low (least significant) portion of clock count*

AL = 0 if the clock was read or written (via AH = 0,1) within the current 24-hour period; otherwise, AL > 0

Set Time Of Day

Set (write) the time of day using binary format.

Entry Conditions:

AH = 1

CX = *high (most significant) portion of clock count*

DX = *low (least significant) portion of clock count*

Sound Multiplexer

Sets the multiplexer for audio source.

Entry Conditions:

AH = 80

AL = *source of sound*

00 = 8253 channel 2

02 = audio in

03 = complex sound generator chip

Disk I/O Support for the Floppy Only System Configuration

Software Interrupt:

13 hex (19 dec)

Function Summary:

AH = 0: Reset Floppy Disk

AH = 1: Return Status of Last Floppy Disk Operation

AH = 2: Read Sector(s) from Floppy Disk

AH = 3: Write Sector(s) to Floppy Disk

AH = 4: Verify Sector(s) on Floppy Disk

AH = 5: Format Track on Floppy Disk

Function Descriptions:

Reset Floppy Disk

Reset the diskette system. Resets associated hardware and recalibrates all diskette drives.

Entry Conditions:

AH = 0

Exit Conditions:

See "Exits From All Calls" below.

Return Status of Last Floppy Disk Operation

Return the diskette status of the last operation in AH.

Entry Conditions

AH = 1

Exit Conditions:

AL = *status of the last operation; see "Exits From All Calls" below for values*

Read Sector(s) from Floppy Disk

Read the desired sector(s) from disk into RAM.

Entry Conditions:

AH = 2
DL = *drive number (0-1)*
DH = *head number (0-1)*
CH = *track number (0-79)*
CL = *sector number (1 to 9)*
AL = *sector count (1 to 9)*
ES:BX = *pointer to disk buffer*

Exit Conditions:

See “Exits From All Calls” below.
AL = *number of sectors read*

Write Sector(s) to Floppy Disk

Write the desired sector(s) from RAM to disk.

Entry Conditions:

AH = 3
DL = *drive number (0-1)*
DH = *head number (0-1)*
CH = *track number (0-79)*
CL = *sector number (1 to 9)*
AL = *sector count (1 to 9)*
ES:BX = *pointer to disk buffer*

Exit Conditions:

See “Exits From All Calls” below.
AL = *number of sectors written*

Verify Sector(s) on Floppy Disk

Verify the desired sector(s).

Entry Conditions:

AH = 4
DL = *drive number (0-1)*
DH = *head number (0-1)*
CH = *track number (0-79)*
CL = *sector number (1 to 9)*
AL = *sector count (1 to 9)*

Exit Conditions:

See “Exits From All Calls” below.
AL = *number of sectors verified*

Format on Floppy Disk

Format the desired track.

Entry Conditions:

AH = 5

DL = *drive number (0-1)*

DH = *head number (0-1)*

CH = *track number (0-79)*

CL = *sector number (1-9)*

ES:BX = *pointer to a group of address fields for each track.* Each address field is made up of 4 bytes. These are C, H, R, and N, where:

C = track number

H = head number

R = sector number

N = the number of bytes per sector (00 = 128, 01 = 256, 02 = 512, 03 = 1024)

There is one entry for every sector on a given track.

Exit Conditions:

See “Exits From All Calls” below.

Exits From All Calls:

AH = *Status of operation, where set = true:*

Error Code	Condition
01H	Illegal Function
02H	Address Mark Not Found
03H	Write Protect Error
04H	Sector Not Found
08H	DMA Overrun
09H	Attempt To DMA Across A 64K Boundary
10H	Bad CRC on Disk Read
20H	Controller Failure
40H	Seek Failure
80H	Device Timeout, Device Failed To Respond

[NC] = operation successful (AH = 0)

[C] = operation failed (AH = error status)

Equipment

This service returns the “equipment flag” (hardware configuration of the computer system) in the AX register.

Software Interrupts:

11 hex (17 dec)

The “equipment flag” returned in the AX register has the meanings listed below for each bit:

Reset = the indicated equipment is not in the system

Set = the indicated equipment is in the system

bit 0	diskette installed
bit 1	math coprocessor
bit 2,3	always = 11
bit 4,5	initial video mode 01 = 40x25 BW 10 = 80x25 BW
bit 6,7	number of diskette drives (only if bit 0 = 1) 00 = 1 01 = 2
bit 8	0 = dma present 1 = no dma on system
bit 9, 10, 11	number of RS 232 cards
bit 12	game I/O attached
bit 13	not used
bit 14, 15	number of printers

Memory Size

This service returns the total number of kilobytes of RAM in the computer system (contiguous starting from address 0) in the AX register. The maximum value returned is 640.

Software Interrupts:

12 hex (18 dec)

EEROM (Tandy 1000 HX only)

15 hex (21 dec)

Function Summary

AH = 70H, AL = 0: Read a 16 bit word from EEROM

AH = 70H, AL = 1: Write a 16 bit word to EEROM

Function Descriptions

Read From EEROM

Read the 16 bit value from the indicated EEROM word.

Entry Conditions:

AH = 70H

AL = 0

BL = word number to read (0 -15)

Exit Conditions:

DX = word value

Carry Flag set indicates EEROM call not supported, system is not a 1000HX

Write To EEROM

Write a 16 bit value to the indicated EEROM word

Entry Conditions:

AH = 70H

AL = 1

BL = word number to write (0-15)

DX = word value to write

Exit Conditions:

Carry Flag set indicates EEROM call not supported, system is not a 1000HX

KEYBOARD ASCII AND SCAN CODES

The table in this appendix lists the keys on the Tandy 1000 keyboard in scan code order, along with the ASCII codes they generate. For each key, the following entries are given:

Scan Code — A value in the range 01H-5AH which uniquely identifies the physical key on the keyboard that is pressed.

Keyboard Legend — The physical marking(s) on the key. If there is more than one marking, the upper one is listed first.

ASCII Code — The ASCII codes associated with the key. The four modes are:

Normal — The normal ASCII value (returned when only the indicated key is depressed).

SHIFT — The shifted ASCII value (returned when SHIFT is also depressed).

CTRL — The control ASCII value (returned when CTRL is also depressed).

ALT — The alternate ASCII value (returned when ALT is also depressed).

Remarks — Any remarks or special functions.

The following special symbols appear in the table:

- x Values preceded by "x" are extended ASCII codes (codes preceded by an ASCII NUL, 00).
- No ASCII code is generated.
- * No ASCII code is generated, but the special function described in the Remarks column is performed. If no comment is included, the key does not generate a code and no function is performed.

Note: All numeric values in the table are expressed in hexadecimal.

QWERTY (USA) — MODEL 1000

Scan Code	Keyboard Legend	Normal	ASCII Codes			ALT	Remarks
			SHIFT	CTRL			
01	ESC	1B	1B	1B	x8B		
02	1	31	21	xE1	x78		
03	2	32	40	x03	x79		
04	3	33	23	xE3	x7A		
05	4	34	24	xE4	x7B		
06	5	35	25	xE5	x7C		
07	5	36	5E	1E	x7D		
08	7	37	26	xE7	x7E		
09	8	38	2A	xE8	x7F		
0A	9	39	28	xE9	x80		
0B	0	30	29	xE0	x81		
0C	-	2D	5F	1F	x82		
0D	=	3D	2B	xF5	x83		
0E	BACK SPACE	08	08	7F	x8C		
0F	TAB	09	x0F	x8D	x8E		
10	Q	71	51	11	x10		
11	W	77	57	17	x11		
12	E	65	45	05	x12		
13	R	72	52	12	x13		
14	T	74	54	14	x14		
15	Y	79	59	19	x15		
16	U	75	55	15	x16		
17	I	69	49	09	x17		
18	O	6F	4F	0F	x18		
19	P	70	50	10	x19		
1A	[5B	7B	B	xE8		
1B]	5D	7D	1D	xF0		
1C	ENTER	0D	0D	0A	x8F	Main Keyboard	
1D	CTRL	*	*	*	*	Control Mode	
1E	A	61	41	01	x1E		
1F	S	73	53	13	x1F		
20	D	64	44	04	x20		
21	F	66	46	06	x21		
22	G	67	47	07	x22		
23	H	68	48	08	x23		
24	J	6A	4A	0A	x24		
25	K	6B	4B	0B	x25		
26	L	6C	4C	0C	x26		
27	:	3B	3A	xF6	xF8		
28	"	27	22	xF7	xF1		
29	UP ARROW	x48	x85	x90	x91		
2A	SHIFT	*	*	*	*	Left SHIFT	
2B	LEFT ARROW	x4B	x87	x73	x92		
2C	Z	7A	5A	1A	x2C		
2D	X	78	58	18	X2D		
2E	C	63	43	03	x2E		
2F	V	76	56	16	xF		

Scan Code	Keyboard Legend	Normal	ASCII Code			ALT	Remarks
			SHIFT	CTRL			
30	B	62	42	02	x30		
31	N	6E	4E	0E	x31		
32	M	6D	4D	0D	x32		
33	,	<	2C	3C	xF9	x89	
34	.	>	2E	3E	xA	X8A	
35	/	?	2F	3F	xFB	xF2	
36	SHIFT	*	*	*	*	*	Right SHIFT
37	PRINT	10	*	x72	x46	SCR Print Toggle	
38	ALT	*	*	*	*	*	Alternate Mode
39	space bar	20	20	20	20		
3A	CAPS	*	*	*	*	*	Caps lock
3B	F1	x3B	x54	x5E	x68		
3C	F2	x3C	x55	x5F	x69		
3D	F3	x3D	x56	x60	x6A		
3E	F4	x3E	x57	x61	x6B		
3F	F5	x3F	x58	x62	x6C		
40	F6	x40	x59	x63	x6D		
41	F7	x41	x5A	x64	x6E		
42	F8	x42	x5B	x65	x6F		
43	F9	x43	x5C	x66	x70		
44	F10	x44	x50	x67	x71		
45	NUM LOCK	*	*	*	*	*	number lock
46	HOLD	*	*	*	*	*	Freeze display
47	7	\	37	5C	x93	*	
48	8	-	38	7E	x94	*	
49	9	PG UP	39	x49	x84	*	
4A	DOWN ARROW	x50	x86	x96	x97		
4B	4		34	7C	x95	*	
4C	5		35	xF3	xF	*	
4D	6		36	xF4	xF	*	
4E	RIGHT ARROW	x4D	x88	x74	xEA		
4F	1	END	31	x4F	x75	*	
50	2	'	32	60	x9A	*	
51	3	PG DN	33	x51	x76	*	
52	0		30	x9B	x9C	*	
53	-	DELETE	2D	x53	x9D	x9E	
54	BREAK	x00	x00	*	*	*	scroll lock bit toggle control brk routine (INT 1BH)
55	+	INSERT	2B	x52	x9F	xA0	
56			2E	xA1	xA4	xA5	Numeric keypad
57	ENTER		0D	0D	0A	x8F	Numeric keypad
58	HOME		x47	x4A	x77	xA6	
59	F11		x98	xA2	xA	xB6	
5A	F12		x99	xA3	xAD	xB7	

-
- * Indicates special functions performed
 - means this key combination is suppressed in the keyboard driver
 - X values preceded by "X" are extended ASCII codes (codes preceded by an ASCII NUL)
 - † The **ALT** key provides a way to generate the ASCII codes of decimal numbers between 1 and 255. Hold down the **ALT** key while you type *on the numeric keypad* any decimal number between 1 and 255. When you release ALT, the ASCII code of the number typed is generated and displayed.

Note: When the NUM LOCK light is off, the Normal and SHIFT columns for these keys should be reversed.

MS-DOS Memory Map

HEXADECIMAL STARTING ADDRESS (SEGMENT:OFFSET)	DESCRIPTION
000:00	BIOS Interrupt Vectors
000:80	Available Interrupt Vectors
0040:00 ¹	ROM BIOS Data Area
0050:00	MSDOS and BASIC Data Area
0070:00	I/O.SYS Drivers
0190.00 ²	MS-DOS
05B0:00 ²	Available to user
X800:00 ³	Video RAM in 32K video modes
XC00:00 ³	Video RAM in 16K video modes
B800:00 ⁴	Video RAM Window (32K)
F000:00	Reserved for system ROM
FC00:00	System BIOS ROM

Notes:

1. Detailed description in following pages.
2. Approximate address; subject to change.
3. "X" is defined as follows:

Memory Size	X Value
128K	1
256K	3
384K	5
512K	7
640K	9
768K	B

4. Video memory accessed through the B800:0 window for all video modes.

ROM BIOS Data Area

The following table gives the starting offset, and length of each BIOS device driver. This area is located at segment 40:00.

Comm card addresses	0000	8 (1 word per card)
Printer addresses	0008	8 (1 word per printer)
Devices installed	0010	2 (16 bits)
Not used	0012	1
Memory size	0013	2 (1 word)
I/O channel RAM size	0015	2 (1 word)
KBD data area	0017	39
Disk data area	003E	11
Video data area	0049	30
Not used	0067	5
Clock data area	006C	5
KBD Break & Reset flags	0071	3
Not used	0074	4
Printer Timeout counter	0078	4 (1 byte per printer)
Comm Timeout counter	007C	4 (1 byte per card)
KBD extra data area	0080	4 (2 words)

The structure and usage of the Video driver RAM data area is as follows:

HEX Offset From Segment 0040:000	Length and Intended Use
49H	1 byte - current CRT mode (0-7)
4AH	1 word - screen column width
4CH	1 word - byte length of screen
4EH	1 word - address/offset of beginning of current display page
50H	8 words - row/col coordinates of the cursor for each of up to 8 display pages
60H	1 word - current cursor type (See "set cursor type" for correct encoding)
62H	1 byte - current display page
63H	1 word - base address + 4 of the CRT controller card
65H	1 byte - copy of value written to the Mode Select Register
66H	1 byte - current color palette setting

The equipment check BIOS call (INT 11H) and memory size BIOS call (INT 12H) return information from the following data areas:

HEX Offset From Segment 0040:000	Length and Intended Use
10H	Devices installed word
13H	Memory installed word

The structure and usage of the floppy disk driver RAM data area is as follows:

HEX Offset From Segment 0040:0000	Length and Intended Use
3EH	1 byte - drive recalibration status - bit 3-0, if 0 then drive 3-0 needs recal before next seek bit 7 indicates interrupt occurrence
3FH	1 byte - motor status - bit 3-0 drive 3-0 motor is on/off, bit 7 - current operation is write, requires delay
40H	1 byte - motor turn off time out counter (see Timer ISR)
41H	1 byte - disk status - codes defined below
42H	7 bytes - 7 bytes of status returned by the controller during result phase of operation

Value	Error Condition
01H	Illegal Function
02H	Address Mark Not Found
03H	Write Protect Error
04H	Sector Not Found
08H	DMA Overrun
09H	Attempt to DMA Across a 64K Boundary
10H	Bad CRC on Disk Read
20H	Controller Failure
40H	Seek Failure
80H	Device Timeout, Device Failed to Respond

The structure and usage of the RS232 driver RAM data area is as follows:

HEX Offset From Segment 0040:00	Length and Intended Use
00H	4 words - Base address of each one of 4 possible comm cards
7CH	4 words - 1 word timeout count for each of 4 possible comm cards

The structure and usage of the Keyboard driver RAM data area is as follows:

HEX Offset From Segment 0040:0010	Length and Intended Use
17	1 byte bits 7 - Keyboard shift state flag returned by function 02 0 - Right SHIFT key depressed, 1 - Left SHIFT key depressed,
18	1 byte bits 7 - Secondary shift state flag 0 - not used bits 6 - CAPS LOCK on/off, 1 - NUM LOCK on/off, 2 - SCROLL LOCK on/off, 3 - ALT key depressed, 4 - CTRL key depressed, 5 - Caps Lock depressed, 6 - Num Lock depressed, 7 - Scroll Lock depressed, 8 - Pause on/off, 9 - not used
19	1 byte - used to store ALT keypad entry
1A	1 word - pointer to beginning of the keyboard buffer
1C	1 word - pointer to end of the keyboard buffer
1E	16 - keyboard buffer (enough for words) 15 - typeahead entries

The structure and usage of the clock service routine is as follows:

HEX Offset From Segment 0040:0000	Length and Intended Use
6CH	1 word - Least significant 16 bits of clock count
6EH	1 word - Most significant 16 bits of clock count
70H	1 byte - Twenty four hour rollover flag

ADDITIONAL DATA AREA

HEX Offset From Segment 0040:000

B0H	2 words	international support
B4H	1 byte	0 = No monochrome monitor FFH = monochrome monitor
B5H	1 byte	Bit 0: 0 = drive A is 5-1/4 1 = drive A is 3-1/2 Bit 1: 0 = drive B is 5-1/4 1 = drive B is 3-1/2 Bit 2: 0 = Tandy 1000 keyboard layout 1 = IBM keyboard layout Bit 3: 0 = Slow CPU speed mode 1 = Fast CPU speed mode Bit 4: 0 = Internal color video support enabled 1 = Internal color video support disabled, external color video enabled Bit 5: 0 = No external monochrome video installed 1 = External monochrome video installed
B6H	1 byte	Bit 0: 0 = drive C is 5-1/4 1 = drive C is 3-1/2

RADIO SHACK
A Division of Tandy Corporation
Fort Worth, Texas 76102