Z8000® Z8016 Z-DTC Direct Memory Access Transfer Controller

Zilog

Product Specification

April 1985

FEATURES

- Memory-to-peripheral transfers up to 2.66M bytes per second at 4 MHz.
- Memory-to-memory transfers up to 1.33M bytes per second at 4 MHz.
- Two fully independent, multi-function channels.
- Masked data pattern matching for Search and Transfer-and-Search operations.
- Funneling option that permits mixing of byte and word data during transfer operations.
- Can operate in logical address space with Zilog Memory Management Units, providing an 8M byte logical addressing range and 16M byte physical addressing range.
- Programmable chaining operation provides automatic loading of control parameters from memory into each channel.
- Software- or hardware-controlled Wait state insertion.
- Z-BUS™ daisy-chain interrupt hierarchy and bus-request structure.

GENERAL DESCRIPTION

The Z8016 DMA Transfer Controller (DTC) is a high performance data transfer device designed to match the power and addressing capability of the Z8000 CPUs. In addition to providing block data transfer capability between memory and peripherals, each of the two DTC channels can perform peripheral-to-peripheral and memory-to-memory transfers. A special Search mode of operation compares data read from memory or peripherals with the contents of a pattern register. A search can be performed concurrently with transfers or as an operation in itself.

In all operations (Search, Transfer, and Transfer-and-Search), the DTC can operate in either Flowthrough or Flyby transfer mode. In the Flowthrough mode, data is stored temporarily within the DTC on its way from source to destination. In this mode transfers can be made between a word-oriented memory and a byte-oriented peripheral through the bidirectional byte/word funneling option. In Flyby mode, data is transferred in a single step (from source to destination), thus providing twice the throughput.

The Z8016 DTC takes full advantage of the Z8000 memory management scheme by interfacing directly to the Z8010 Memory Management Unit (MMU) or the Z8015 Paged Memory Management Unit (PMMU). In this configuration, 8M bytes of logical address range are provided for each CPU address space. Alternatively, the Z8016 DTC can operate independently of an MMU, directly addressing up to 16M bytes of physical address space.

In addition to providing a hardware $\overline{WAIT}$ input to accommodate different memory or peripheral speeds, the Z8016 DTC allows the user to program the automatic insertion of either zero, one, two, or four Wait states for either source or destination addresses. Alternatively, the $\overline{WAIT}$ input pin function can be disabled and these software-programmed Wait states used exclusively.

The Z8016 DTC minimizes CPU involvement by allowing each channel to load its control registers from memory automatically when a DMA operation is complete. By loading the address of the next block of control parameters as part of this operation, command chaining is accomplished. The only action required of the CPU is to load the address of the control parameter table into the channel's Chain Address register and then issue a Start Chain command.

In some DMA applications, data is transferred continuously between the same two locations. To service these repetitive DMA operations, base registers are provided on each channel to reinitialize the current source and destination address registers. This re-initialization eliminates the need for reloading registers from memory tables.

The Z8016 DTC is directly Z-BUS compatible, and operates within the Z8000 daisy-chain vectored-priority interrupt scheme. The Demand Interleave operation allows the DTC to surrender the bus to the external system, or to alternate between internal channels. This capability allows for parallel operations between dual channels or between a DTC channel and the CPU.

The DTC can be used to provide a central DMA function for the CPU or to provide dispersed DMA operations in conjunction with a wide variety of Z8000 Family peripheral controllers.

The Z8016 DTC is packaged in a 48-pin DIP and uses a single +5 V power supply.

The Z8016 DTC pin functions and assignments are shown in Figures 1 and 2, respectively.

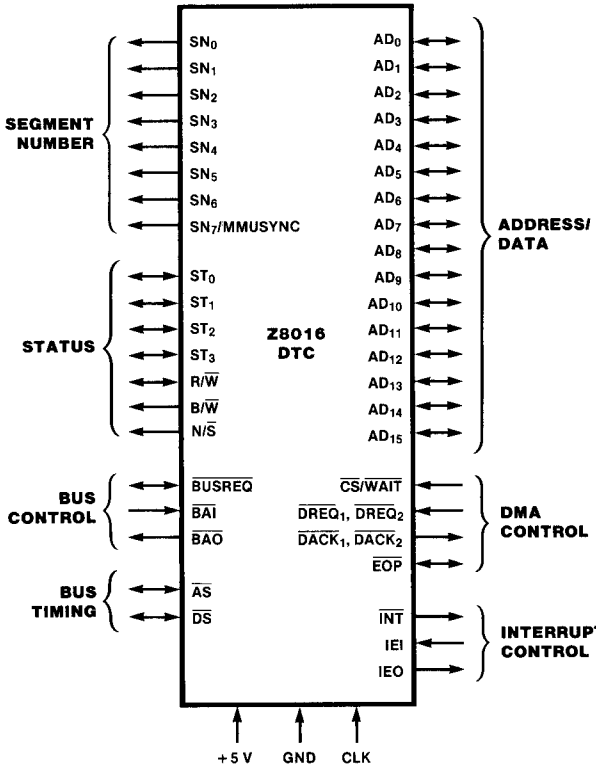


Figure 1. Pin Functions

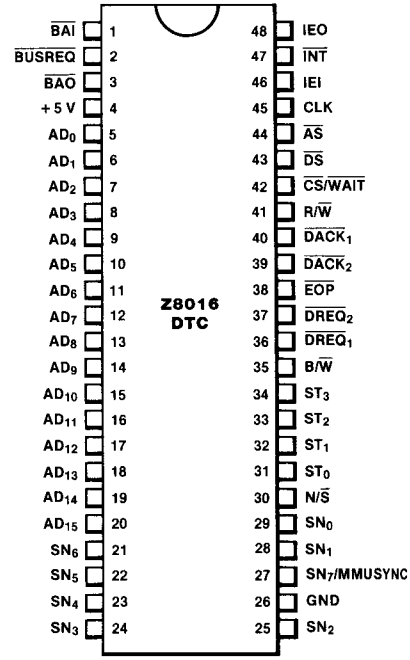


Figure 2. 48-pin Dual-In-Line Package (DIP), Pin Assignments

SIGNAL DESCRIPTIONS

AD_0-AD_{15}. *Address/Data Bus* (bidirectional, active High, 3-state) pins 5-20. These multiplexed Address/Data lines are used for all I/O and memory transactions.

$\overline{AS}$. *Address Strobe* (bidirectional, active Low, 3-state) pin 44. When the DTC is bus master the rising edge of $\overline{AS}$ (while $\overline{DS}$ is High) indicates that addresses are valid. When the DTC is not bus master, the address lines are sampled on the rising edge of $\overline{AS}$. There are no timing requirements between $\overline{AS}$ as an input and the DTC clock, because the Z-BUS does not use a bused clock. If $\overline{AS}$ and $\overline{DS}$ are simultaneously Low, the DTC will be reset.

$\overline{BAI}$. *Bus Acknowledge In* (input, active Low) pin 1. Signals that the bus has been released for DTC control. In multiple-DTC configurations, the $\overline{BAI}$ pin of the highest-priority DTC is normally connected to the Bus Acknowledge pin of the CPU. Each lower-priority DTC has its $\overline{BAI}$ connected to the $\overline{BAO}$ of the next higher-priority DTC.

$\overline{BAO}$. *Bus Acknowledge Out* (output, active Low) pin 3. In a multiple-DMA configuration, this pin signals that no higher-priority DTC has requested the bus. $\overline{BAI}$ and $\overline{BAO}$ form a daisy chain for multiple-DTC priority resolution.

2129-001, 002

$\overline{\text{BUSREQ}}$. *Bus Request* (bidirectional, active Low, open-drain) pin 2. $\overline{\text{BUSREQ}}$ is used by the DTC to obtain control of the bus from the CPU. Before driving $\overline{\text{BUSREQ}}$ active, the DTC samples this line to ensure that another request is not already being made by another device. Since the DTC internally synchronizes the sampled $\overline{\text{BUSREQ}}$ signal, transitions on $\overline{\text{BUSREQ}}$ can be asynchronous with respect to the DTC clock.

B/$\overline{\text{W}}$. *Byte/Word* (output, 3-state) pin 35. This output indicates the type of data transferred on the Address/Data (A/D) bus. A High on this line indicates a byte (8-bit) transfer and a Low indicates a word (16-bit) transfer. This signal is activated when $\overline{\text{AS}}$ goes Low and remains valid for the duration of the transaction.

CLK. *DTC Clock* (input) pin 45. The Clock signal controls internal operations and the rates of data transfer. It is usually derived from a master system clock or an associated CPU clock. When the DTC is used with an MMU, both must be driven from the same clock signal. While many DTC input signals are asynchronous, transitions for other signals (such as $\overline{\text{WAIT}}$ inputs) must meet setup and hold requirements relative to the DTC clock. (See the timing diagrams for details.)

$\overline{\text{CS}}$/$\overline{\text{WAIT}}$. *Chip Select/Wait* (input, active Low) pin 42. When the DTC is not in control of the system bus, this pin serves as a Chip Select ($\overline{\text{CS}}$) input. A CPU or other external device uses $\overline{\text{CS}}$ to activate the DTC for reading and writing the DTC's internal registers. ($\overline{\text{CS}}$ can be held Low for multiple transfers to and from the DTC, provided that $\overline{\text{AS}}$ and $\overline{\text{DS}}$ are enabled for each transfer.) There are no timing requirements between the $\overline{\text{CS}}$ input and the DTC clock; the $\overline{\text{CS}}$ input timing requirements are only defined relative to $\overline{\text{AS}}$.

When the DTC is in control of the system bus, this pin serves as the $\overline{\text{WAIT}}$ input. Slow memories and peripheral devices can use $\overline{\text{WAIT}}$ to extend $\overline{\text{DS}}$ during bus transfers. Unlike the $\overline{\text{CS}}$ input, transitions on the $\overline{\text{WAIT}}$ input must meet certain timing requirements relative to the DTC clock (see the Active State timing diagram for details). The $\overline{\text{WAIT}}$ function can be disabled using a control bit in the Master Mode register, in which case this input is treated as a Chip Select only and is ignored when the DTC is in control of the system bus.

$\overline{\text{DACK}}_1$, $\overline{\text{DACK}}_2$. *DMA Acknowledge* (output, active Low) pins 39 and 40. There is one DMA Acknowledge line associated with each channel. The $\overline{\text{DACK}}$ lines are programmed in the Channel Mode register to be pulsed, held active, or held inactive during DMA transfers. During Flyby operations the $\overline{\text{DACK}}$ line is used for two purposes. It selects the peripheral involved in the transfer, and it provides timing information on when to access the bus. During flowthrough operations the $\overline{\text{DACK}}$ line can be programmed to be active or inactive during a DMA transfer. $\overline{\text{DACK}}$ is not output during chaining operations.

$\overline{\text{DREQ}}_1$, $\overline{\text{DREQ}}_2$. *DMA Request* (input, active Low) pins 36 and 37. There is a DMA Request line associated with each channel. These lines can make transitions independent of the DTC clock. They are used by external logic to initiate and control DMA operations performed by the DTC.

$\overline{\text{DS}}$. *Data Strobe* (bidirectional, active Low, 3-state) pin 43. A Low on this signal while $\overline{\text{AS}}$ is High indicates that the A/D bus is being used to transfer data. When the CPU is bus master and is transferring information to or from the DTC, $\overline{\text{DS}}$ is a timing input used by the DTC to move data to or from the A/D bus.

$\overline{\text{EOP}}$. *End of Process* (bidirectional, active Low, open-drain, asynchronous) pin 38. This line is output when a Terminal Count (TC) or Match Condition (MC) termination occurs (see Termination section). An external source can terminate a DMA operation in progress by driving $\overline{\text{EOP}}$ Low. $\overline{\text{EOP}}$ always applies to the active channel; if no channel is active, $\overline{\text{EOP}}$ is ignored. The Suppress output of the MMU can be connected to $\overline{\text{EOP}}$ to terminate DMA accesses that violate the MMU protection settings. To provide full access protection, an external $\overline{\text{EOP}}$ is accepted even during chaining.

IEI. *Interrupt Enable In* (input, active High) pin 46. IEI is used with IEO to form an interrupt daisy chain when there is more than one interrupt-driven device. A High IEI indicates that no other higher-priority device has an interrupt under service or is requesting an interrupt.

IEO. *Interrupt Enable Out* (output, active High) pin 48. IEO is High only if IEI is High and the CPU is not servicing an interrupt from the requesting DTC. IEO is connected to the next lower-priority device's IEI input and thus inhibits interrupts from lower-priority devices.

$\overline{\text{INT}}$. *Interrupt Request* (output, open-drain, active Low) pin 47. This signal is pulled Low when the DTC requests an interrupt.

N/$\overline{\text{S}}$. *Normal/System* (output, 3-state) pin 30. The N/$\overline{\text{S}}$ signal is activated when the DTC is bus master. The N/$\overline{\text{S}}$ signal indicates which memory space is being accessed by going High for normal memory and Low for system memory.

R/$\overline{\text{W}}$. *Read/Write* (bidirectional, 3-state, Low = write) pin 41. When the DTC is not bus master, R/$\overline{\text{W}}$ is a status input used to indicate whether data is being read from (High) or written to (Low) the DTC. When the DTC is bus master, R/$\overline{\text{W}}$ is an output used to indicate whether the DTC is reading or writing the addressed location. During Flyby DMA operations, the "Flyby peripheral" (Figure 3) inverts the R/$\overline{\text{W}}$ signal to determine whether it must read or write.

SN_0-SN_6. *Segment Number* (output, 3-state) pins 21-25 and 28-29. In logical address configuration, these lines provide the segment number field of a 23-bit segmented address. The SN_0–SN_6 I/O address information can be used to increase the DTC's logical I/O address space beyond that of the CPU. In physical address configuration, these lines provide bits 23 through 17 of a 24-bit linear address. The 24th bit (MSB) is output on SN_7/MMU Sync.

SN_7 or MMU Sync. *Segment Number 7 or MMU Sync* (output, 3-state) pin 27. In a logical address space configuration (with MMU), this line outputs an active High pulse prior to each machine cycle. The MMU uses this signal to synchronize access to its translation table and to differentiate between CPU and DTC control. The MMU ignores MMUSYNC if the status lines (ST_0–ST_3) indicate that an I/O transaction is being performed. This output is Low when the DTC is not bus master and the MM1 bit in the Master Mode register is set.

In a physical address space configuration (without MMU), this line outputs SN_7, which becomes the 24th address bit in a linear address space. The 24-bit linear address configuration allows the DTC to access 16M bytes of memory. This pin floats to the high impedance state when the DTC is not bus master and the MM1 bit is cleared.

ST_0-ST_3. *Status* (bidirectional, 3-state) pins 31-34. When the DTC is bus master, these lines are outputs indicating the type of memory or I/O transaction being performed. When the DTC is not bus master, the status lines are inputs used to detect Interrupt and Segment Trap Acknowledge cycles (Table 1).

Table 1. Status Codes

ST_3	ST_2	ST_1	ST_0	Transaction/Operation	Status Code Generated/Decoded
0	0	0	0	Internal Operation	
0	0	0	1	Memory Refresh	
0	0	1	0	I/O Transaction	Generated
0	0	1	1	Special I/O Transaction	Generated
0	1	0	0	Segment Trap Acknowledge	Decoded
0	1	0	1	Nonmaskable Interrupt Acknowledge	Decoded
0	1	1	0	Nonvectored Interrupt Acknowledge	Decoded
0	1	1	1	Vectored Interrupt Acknowledge	Decoded
1	0	0	0	Memory Transaction for Data/DTC Chaining	Generated
1	0	0	1	Memory Transaction for Stack	Generated
1	0	1	0	Reserved	
1	0	1	1	Reserved	
1	1	0	0	Memory Transaction for Program Fetch (Subsequent Word)	Generated
1	1	0	1	Memory Transaction for Program Fetch (First Word)	
1	1	1	0	Reserved	
1	1	1	1	Reserved	

FUNCTIONAL DESCRIPTION

Channel Initialization

The Z8016 DTC operates with a minimum of interaction with the host CPU. Each channel's operation is determined by the settings of its own set of control registers. Each channel is initialized when the DTC loads its control parameters from memory into its control registers during the chaining operation. To initiate the chaining operation, the CPU is required to program the Master Mode register and each channel's Chain Address register. Then each channel's control registers are automatically loaded by the DTC with control parameters stored in a chain control table in memory, located at the address pointed to by that channel's Chain Address register. Once the channel registers are loaded, the DTC is ready to perform DMA operations.

Initiating DMA Operations. DMA operations can be initiated in three ways:

- Software Request. The CPU can issue Software Request commands to start DMA operations on a specific channel. This channel must then request control of the bus and perform transfers.
- Hardware Request. DMA operations can be started by forcing a channel's $\overline{DREQ}$ input Low, as described in the Transfer Modes section.
- Starting After Chaining. If the Software Request bit of the Channel Mode register is loaded with a 1 during chaining, the channel will perform the programmed DMA operation at the end of chaining. If the channel is

programmed for Single Operation or Demand mode, it will perform the operation immediately. The channel will give up the bus after chaining and before the operation if the CPU Interleave bit in the Master Mode register is set. Note that once a channel starts a chaining operation by fetching a reload word, it retains bus control at least until all of the registers specified in the reload word have been loaded from memory.

Transfers

The Z8016 DTC uses three basic types of operation: Transfer, Search, and Transfer-and-Search.

During a Transfer operation, the DTC obtains control of the system A/D bus from the CPU. Data is read from one addressable port (source) and is written to another addressable port (destination) in words or bytes. This applies to both Flyby and Flowthrough transfers.

Flyby transfers use a single addressing/transfer cycle, in which data is transferred directly from the source to the destination with no intermediate storage (Figure 3). This method of transfer provides higher throughput than Flowthrough transfers but cannot be used for memory-to-memory transfer.

Flowthrough transfers are used for all combinations of addressable memory and I/O spaces. These transfers use independent double Addressing/Transfer cycles, in which data is stored temporarily in the DTC while being transferred from source to destination (Figure 4). Flowthrough transfers can use the funneling option, which allows mixing of data sizes between source and destination. For example, a byte-oriented peripheral can conveniently supply data to a word-oriented memory. This option requires no added circuitry for either memory or peripherals.

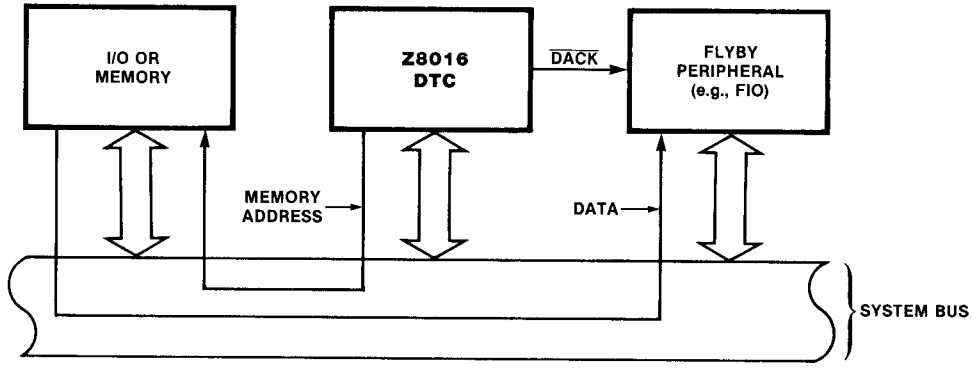


Figure 3. Configuration of a Flyby Transaction

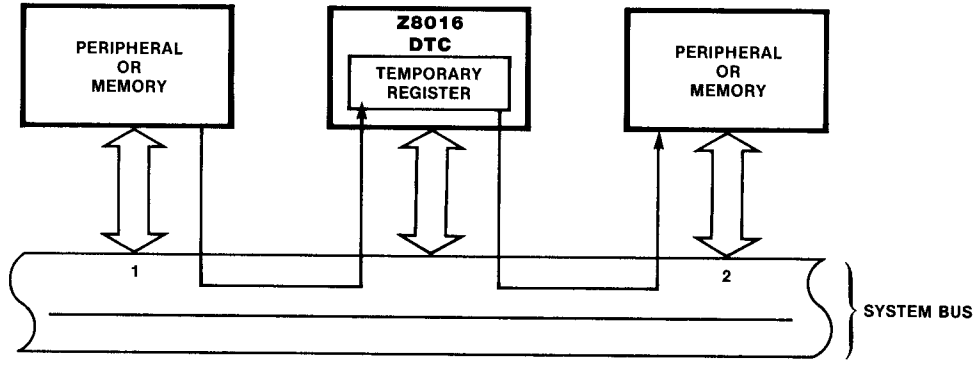


Figure 4. Configuration of a Flowthrough Transaction

During a Search operation, data is read from the source port and compared byte-by-byte with a pattern register containing a programmable match byte. The Search operation can be programmed to stop either when the read data matches (Stop-on-Match) or when it fails to match the masked pattern (Stop-on-No-Match). For word reads, the Channel Mode register can be used to select either 8- or 16-bit compares.

Transfer-and-Search operations combine the transfer and search functions to facilitate the transfer of variable-length data blocks. While data is being transferred between two ports, a simultaneous search is made for a bit-maskable byte match. Transfer-and-Search can be performed in either Flowthrough or Flyby mode. A Flyby Transfer-and-Search can be used to increase throughput for transfers between peripherals or between memory and a peripheral; it cannot be used for memory-to-memory transfers.

Transfer Modes. The Z8016 DTC operates in either of two transfer modes: Single or Demand. The Demand mode is further divided into the Demand Dedicated with Bus Hold, Demand Dedicated with Bus Release, and Demand Interleave modes.

The Single mode is used with peripherals that transfer single bytes or words at irregular intervals. Each Software Request command causes the channel to perform a single DMA operation and each application of a High-to-Low transition on the $\overline{DREQ}$ input also initiates a DMA operation. Each time a Single mode DMA operation ends, the channel relinquishes the bus unless a new transition has occurred on $\overline{DREQ}$.

In the Demand mode, when the $\overline{DREQ}$ input is active, transfer cycles are executed repeatedly until the transfer is completed. In the Demand Dedicated with Bus Hold mode, the active channel retains control of the bus until the transfer is complete, even after the $\overline{DREQ}$ input has gone inactive. In the Demand Dedicated with Bus Release mode, the active channel releases control of the bus when the $\overline{DREQ}$ input goes inactive. When the $\overline{DREQ}$ input becomes active again, control of the bus is re-acquired and the transfer operation continues.

The Demand Interleave mode has two options, programmable in the Master Mode register bit MM2. If MM2 is set, the DTC relinquishes and re-requests bus control after every DMA operation.

This permits the CPU and other devices to gain bus control. If both channels receive active $\overline{DREQ}$ inputs, each channel relinquishes control to the CPU after each operation. In the second option (MM2 is 0), control can pass from one channel to the other without requiring the DTC to release bus control. If both channels receive active $\overline{DREQ}$ inputs, control alternates between channels and the DTC retains bus control until all channel operations are complete.

Wait States. The Z8016 DTC can insert Wait cycles into the DMA Transaction cycle under hardware or software control. The $\overline{CS}/\overline{WAIT}$ input can be multiplexed to function as a Chip Select for the DTC when it does not have control of the bus, and as a $\overline{WAIT}$ input when the DTC is the bus controller. Multiplexing $\overline{CS}$ and $\overline{WAIT}$ requires external logic, but the DTC can be programmed to insert Wait states automatically without external logic when accessing either I/O or memory addresses. Either zero, one, two, or four Wait states can be added. Wait states can be programmed separately for the Current Address registers and for the Chain Address register. Programmable Wait cycle insertion allows memories and peripherals of different speeds to be associated with I/O and memory addresses.

Interrupts. On the Z8016 DTC, each channel is an interrupt source and has its own vector register for identifying the source of the interrupt during a CPU/DTC Interrupt Acknowledge transaction. An interrupt can result from a Match Condition (MC), End-Of-Process (EOP), or Terminal Count (TC) on either channel. The user selects the action to be performed by setting bits in the Channel Mode register.

Three bits in each channel's Status register control interrupts. These are the Channel Interrupt Enable (CIE) bit, the Interrupt Pending (IP) bit, and the Interrupt Under Service (IUS) bit.

Devices connected to any of the CPU's three interrupt inputs resolve priority conflicts with an interrupt daisy chain, as shown in Figure 5. The daisy chain has two functions. During an Interrupt Acknowledge transaction, it determines which interrupt source is being acknowledged. At all other times, it determines which interrupt sources can initiate an interrupt request.

The Z8016 DTC has an interrupt queuing capability, which includes a two-deep interrupt queue on each channel. This allows the DTC to continue normal operation between the time an interrupt is issued and the time the Interrupt Acknowledge is received.

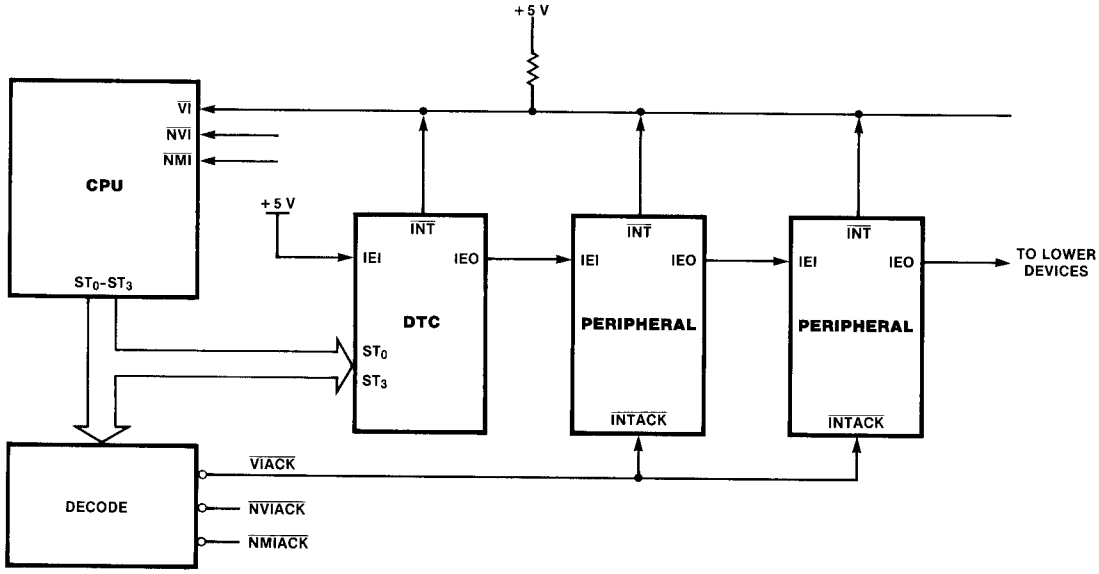


Figure 5. Interrupt Daisy Chain

Termination

DMA operations can end in one of the following three ways:

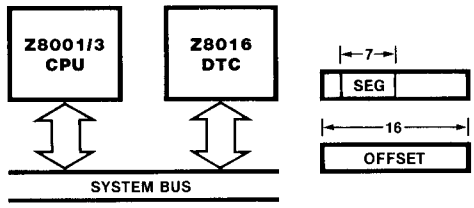
- A Terminal Count (TC) termination occurs when a channel's Current Operation Count register goes to 0.
- An End-of-Process (EOP) termination occurs when the DTC'S EOP pin is driven Low by external logic.
- A Match Condition (MC) termination occurs when data being Searched or Transferred-and-Searched meets the match condition programmed in the Channel Mode register.

MEMORY MANAGEMENT

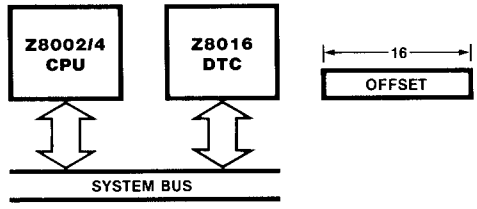
The DTC can be configured to operate in physical address space or logical address space. When the DTC is operated in logical address space, the segment and offset portions of the address registers combine to form 23-bit logical addresses. In conjunction with a CPU, DMA operations can be handled through the Z8010 MMU or the Z8015 PMMU. MMUs offer dynamic segment relocation, segment protection, and other memory management features.

In the physical address space configuration, the segment and offset portions of the DTC's address registers are combined with the SN_7 output to form a single 24-bit linear address. The extended I/O addressing capability of the DTC can be used to increase the DTC's physical I/O address space beyond that of the CPU. Figure 6 illustrates various DTC configuration options with the Z8000 CPUs and MMUs.

DTC WITH Z8001/3 (SEGMENTED) CPU



DTC WITH Z8002/4 (NONSEGMENTED) CPU



DTC WITH Z8001/3 (SEGMENTED) CPU AND MMU

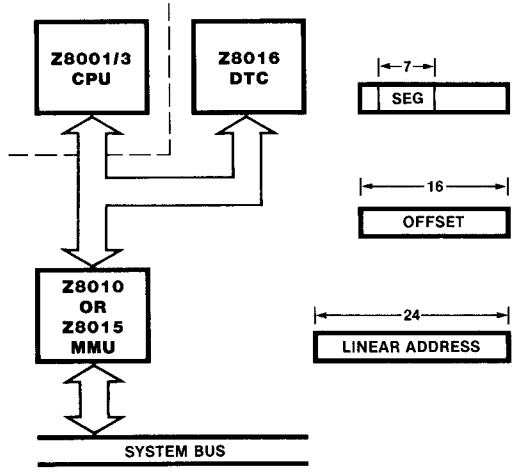


Figure 6. DTC Configurations

INTERNAL STRUCTURE

The internal structure of the Z8016 DTC includes driver and receiver circuitry for interfacing with Zilog's Z-BUS. The DTC's internal bus interfaces with the Z-BUS and services all internal logic and registers, as illustrated in the DTC block diagram (Figure 7).

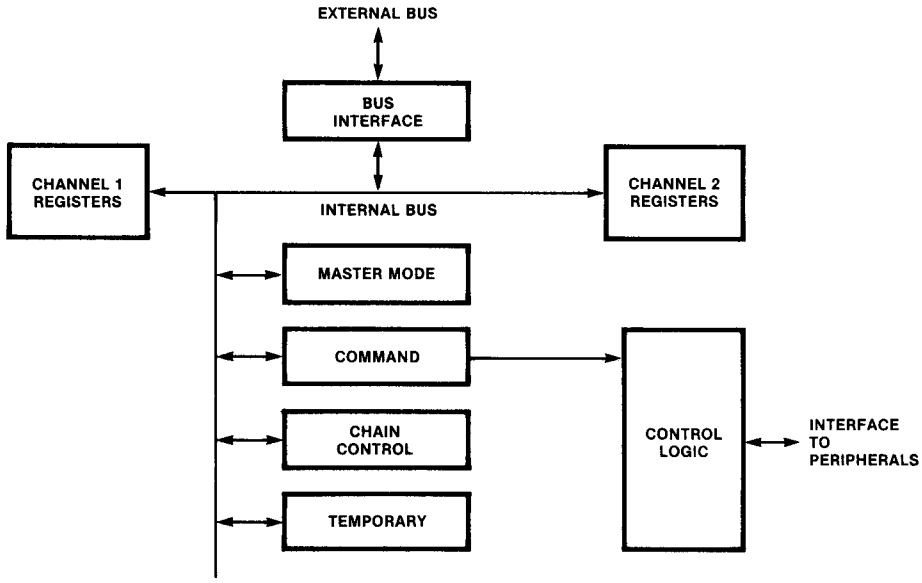


Figure 7. DTC Block Diagram

REGISTER DESCRIPTION

The DTC contains chip-level control registers as well as channel-level registers that are duplicated for each channel. Registers on the DTC that can be read by the CPU are either fast- or slow-readable. CPU I/O instructions can read fast-readable registers without Wait states. Slow-readable registers can be read by the CPU only if Wait states are inserted. This requires external logic to generate and time the application of Low signals on the CPUs $\overline{WAIT}$ input if the slow-readable registers are to be read.

Control Registers

The four control registers direct the functioning of the DTC. (Figure 8.)

Master Mode Register. This register selects the way in which the DTC interfaces to the system. The following descriptions indicate how the individual bits in the Master Mode register are used. The Master Mode register is fast-readable.

Chip Enable (CE). The setting of this bit enables the DTC to request the bus, perform DMA operations and reload registers.

Logical/Physical Address Space (LPA). The setting of this bit determines how the system will view the segment and offset portions of the Current ARA and ARB registers. When LPA is set to 1 (Logical Address Space), the segment and offset portions of the Current ARA and ARB registers are treated as separate portions of the address. The 16-bit offset portion of the address will appear on pins AD_0–AD_{15} when $\overline{AS}$ is Low. The 7-bit segment number appears on pins SN_0–SN_6 for the duration of the transaction.

When this bit is set to 0 (Physical Address Space), the segment and offset portions of the Current ARA and ARB registers are treated as a single address and all eight segment bits in the register are used. Both the I/O and the memory addresses in Physical Memory Space are generated by loading the offset portion of the Current Address register onto the AD_0–AD_{15} bus and the segment portion of that register onto the SN_0–SN_7 bus. (In conjunction with the nonsegmented Z8000 CPUs, either Logical or Physical Address Space setting may be used.)

Wait Line Enable (WLE). This bit is set to enable sampling of the $\overline{CS/WAIT}$ line during memory and I/O transactions.

Disable Lower Chain (DLC). This bit is set to inhibit all lower priority devices on the interrupt daisy chain. While DLC is 0, the DTC generates Low and High signals on the IEO output in response to IEI.

No Vector on Interrupt (NVI). This bit determines whether the DTC channel or a peripheral returns a vector during Interrupt Acknowledge cycles. While the bit is cleared, a channel receiving an Interrupt Acknowledge will drive the contents of its Interrupt Save register onto the A/D bus while $\overline{DS}$ is Low. While this bit is set, interrupts are serviced in an identical manner, but the A/D bus remains in a high impedance state throughout the Acknowledge cycle.

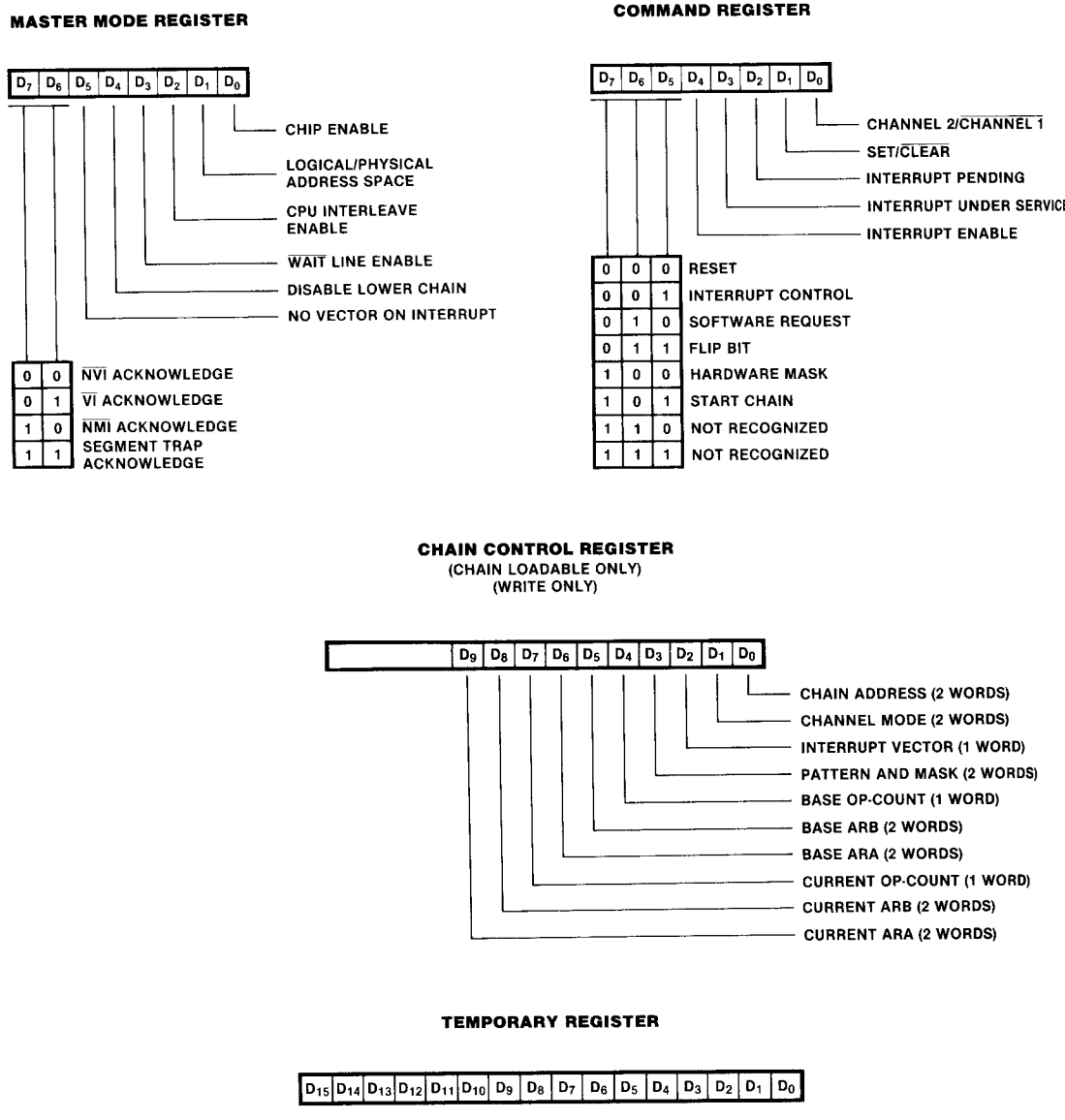


Figure 8. Control Registers

2129-008

Interrupt Acknowledge Field (two bits). This field is used to select the type of Interrupt Acknowledge cycle the DTC is to respond to. The setting of this field must correspond to the IEI/IEO daisy chain on which the DTC is located. The DTC can respond to Nonmaskable Interrupt (NMI), Nonvectored Interrupt (NVI), or Segment Trap Acknowledge cycles.

CPU Interleave Enable. When this bit is set, interleaving of bus use between the CPU and the DTC is enabled.

Chain Control Register. This 16-bit register specifies which registers are to be loaded from memory during a chaining operation. The Chain Control register is loaded from the memory location pointed to by the Chain Address register. The Chain Control register is chain loadable only and cannot be accessed by the CPU.

Command Register. The Command register is an 8-bit write-only register written to by the host CPU to execute commands. The Command register is loaded from the data on AD_7–AD_0; the data on AD_{15}–AD_8 is disregarded.

Temporary Register. This 16-bit register is used to hold data during Flowthrough transfers, Search operations, and Transfer-and-Search operations. The Temporary register cannot be written or read by the CPU.

Channel-Level Registers

Each of the DTC's two channels has a complete set of channel-level registers. This set consists of both General-Purpose and Special-Purpose registers, as illustrated in Figure 9. The General-Purpose registers are commonly found on DMA devices and can be read or written by the CPU. The Special-Purpose registers provide additional features specific to the Z8016 DTC.

General-Purpose Registers. The General-Purpose register set on each channel consists of the Current Address registers A and B, the Base Address registers A and B, the Base and Current Operation Count registers, and the Channel Mode register (Figure 10).

Current and Base Address Registers A and B. The Current Address registers A and B are used to point to the source and destination for DMA operations. The contents of the Base Address registers A and B are transferred into the Current Address registers A and B at the end of a DMA operation if the user enables base-to-current reloading in the Completion field of the Channel Mode register. The base-to-current reload operation facilitates repetitive DMA operations without the multiple memory accesses required by chaining.

Each of the Base and Current Address registers A and B consist of two words. The first word contains a 7-bit Tag field and an 8-bit Segment Number field. The second word contains a 16-bit offset. The use of the Tag field is described below. The use of the Segment Number field depends upon the setting of the LPA bit in the Master Mode register. The Base and Current Address registers are fast-readable and can be loaded by chaining.

Programmable Wait Field. This field allows the insertion of zero, one, two, or four Wait states into memory or I/O accesses addressed by the offset and segment fields.

Address Control Field. At the end of each iteration of a DMA operation, the address can be incremented, decremented, or left unchanged. Memory addresses are changed by one if the address points to a byte operand or by two if the address points to a word operand.

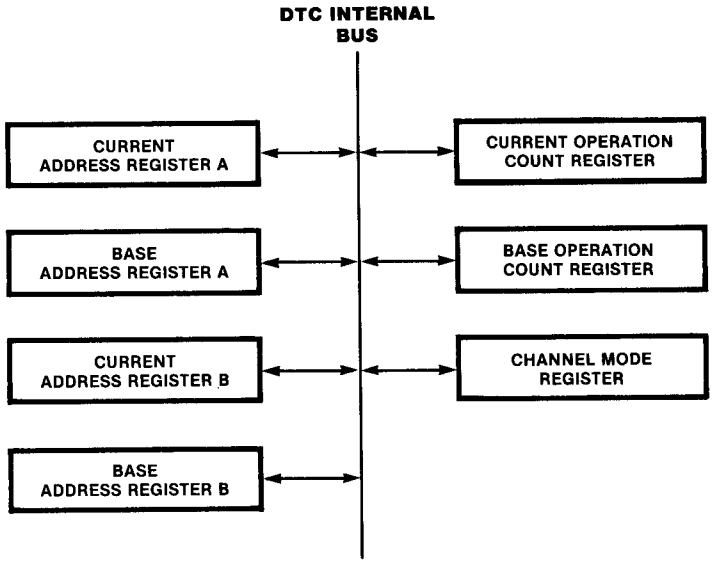
Address Reference Field. This portion of the Tag field is used to select whether the address pertains to memory space or I/O space. The N/$\overline{S}$ output line is always Low (indicating System) for I/O space but can be either High (Normal) or Low (System) for memory space.

Current and Base Operation Count Registers. The 16-bit Current Operation Count register specifies the number of words or bytes to be transferred, searched, or transferred-and-searched. For word-to-word operations and byte-word funneling, this register must be programmed with the number of words to be transferred or searched.

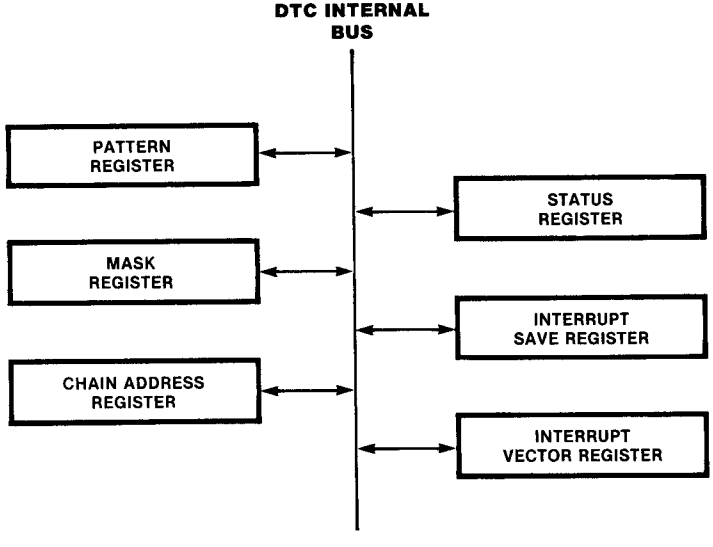
The Base Operation Count register reinitializes the current source and destination in the Current Operation Count register. Each time data is transferred or searched, the Current Operation Count register is decremented by one. Once all of the data is transferred or searched, the Current Operation Count register will contain zero. If the transfer on search stops before the Current Operation Count register reaches zero, the contents of the register indicate the number of bytes or words remaining to be transferred or searched. This allows a channel to be restarted from where it left off without requiring reloading of the Current Operation Count register. The Current and Base Operation Count registers are slow-readable and can be loaded by chaining.

Channel Mode Register. This register selects the type of DMA operation the channel is to perform, how the operation is to be executed, and what action is to be taken when the operation finishes. The Channel Mode register is slow-readable and can be loaded by chaining.

Data Operation and Transfer Type Field. These fields are used to select the type of operation the channel is to perform along with the operand size. The specific codes are listed in Tables 2 and 3. The Flip bit is used to select which of the Current Address Registers A (ARA), or B (ARB), points to the source and which points to the destination address.



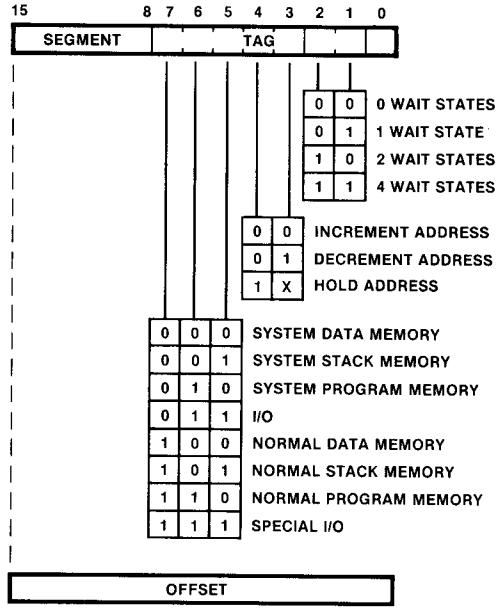
GENERAL-PURPOSE CHANNEL REGISTERS



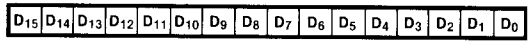
SPECIAL-PURPOSE CHANNEL REGISTERS

Figure 9. Channel-Level Registers

BASE AND CURRENT ADDRESS REGISTERS A AND B



BASE AND CURRENT OPERATION COUNT REGISTERS



CHANNEL MODE REGISTER

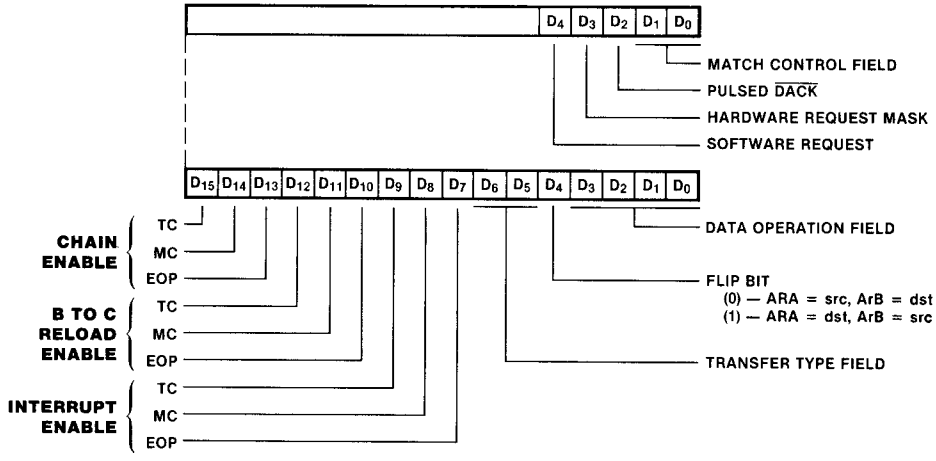


Figure 10. General-Purpose Channel Registers

Table 2. Data Operation Field

Code/Operation	Operand Size ARA	Operand Size ARB	Transaction Type
Transfer			
0001	Byte	Byte	Flowthrough
100X	Byte	Word	Flowthrough
0000	Word	Word	Flowthrough
0011	Byte	Byte	Flyby
0010	Word	Word	Flyby
Transfer-and-Search			
0101	Byte	Byte	Flowthrough
110X	Byte	Word	Flowthrough
0100	Word	Word	Flowthrough
0111	Byte	Byte	Flyby
0110	Word	Word	Flyby
Search			
1111	Byte	Byte	N/A
1110	Word	Word	N/A
101X	Illegal		

Completion Field. This field is used to program the action taken by the channel at the end of a DMA operation. When a DMA operation ends, the channel can perform any combination of the following options:

- Interrupt the CPU (Interrupt Enable field)
- Base-to-Current reload (B to C Reload field)
- Chain reload the next DMA operation (Chain Enable field)

The options are performed according to the bits set in the Interrupt Enable, B to C Reload, and Chain Enable fields for each type of termination that occurs; the NAC bit in the Status register is automatically set on completion of a DMA operation.

Match Control Field. This 2-bit field determines whether matches use an 8-bit or 16-bit pattern and whether the channel is to Stop-On-Match or Stop-On-No-Match. The specific codes for the Match Control field are listed in Table 3.

Table 3. Transfer Type Field and Match Control Field

Code	Transfer Type	Match Control
00	Single Transfer	Stop on No Match
01	Demand Dedicated/Bus Hold	Stop on No Match
10	Demand Dedicated/Bus Release	Stop on Word Match
11	Demand Interleave	Stop on Byte Match

Pulse $\overline{\text{DACK}}$ (PD). This bit determines when the $\overline{\text{DACK}}$ line is active. While cleared, the channel's $\overline{\text{DACK}}$ line is active whenever the channel is performing a DMA operation, regardless of the type of transaction. While the PD bit is set, the $\overline{\text{DACK}}$ pin is inactive during chaining, Flowthrough Transfers, Flowthrough Transfer-and-Searches, and Searches. $\overline{\text{DACK}}$ is pulsed active during Flyby Transfers and Flyby Transfer-and-Searches at the time necessary to strobe data into, or out of, the Flyby peripheral.

Hardware Request Mask (HRM). If this bit is set, a DMA operation can be started by applying a Low on the channel's $\overline{\text{DREQ}}$ input.

Software Request (SR). If this bit is set during chaining, the channel performs the programmed DMA operation at the end of the chaining operation.

Special Purpose Registers. The Special-Purpose registers on each channel are the Pattern and Mask registers, the Status register, the Interrupt Vector register, the Interrupt Save registers, and the Chain Address register (Figure 11).

Pattern and Mask Registers. These registers are used in Search and Transfer-and-Search operations. The Pattern register contains the pattern that the read data is compared to. The Mask register allows the user to exclude or mask selected Temporary register bits from comparison by setting the corresponding Mask register bit to 1. The Pattern and Mask registers are slow-readable and can be loaded by chaining.

Status Register. The Status register on each channel reports the status of that channel. The functions of the individual bits are indicated in the following field descriptions. The Status register is fast-readable.

Completion Status Field. Three bits indicate whether the DMA operation ended as a result of TC, MC, or EOP. The TC bit is set if the Operation Count (reaching zero) ends the DMA operation. The MC bit is set if a pattern match termination occurs. The EOP bit is set when an EOP termination ends a DMA transfer. The appropriate combination of the TC, MC, and EOP bits is set if multiple reasons exist for ending a DMA operation. The Match Condition High byte (MCH) and Match Condition Low byte (MCL) bits report the match states of the upper and lower comparator bytes of the last word transferred. The MCH and MCL bits are updated with each transfer.

These bits are set when the associated comparator bytes are matched, regardless of whether Stop-on-Match or Stop-on-no-Match is programmed.

Hardware Interface Status Field. The Hardware Request (HRQ) bit provides a means of monitoring the channel's $\overline{\text{DREQ}}$ input line. While $\overline{\text{DREQ}}$ is Low, the HRQ bit is set. While the Hardware Mask (HM) bit is set, the DTC is prevented from responding to a Low on the $\overline{\text{DREQ}}$ line. However, the HRQ bit always reports the status of $\overline{\text{DREQ}}$ regardless of the status of the HM bit.

DTC Status Field. This field reports the current channel status to the CPU. The "channel initialized and waiting for request" status is implicitly indicated if bits ST_{12} through ST_9 are clear.

Second Interrupt Pending (SIP). When a second interrupt is to be issued before the first interrupt is acknowledged, this bit is set and the channel relinquishes the bus until an Acknowledge occurs.

Waiting for Bus (WFB). This bit is set when the channel is waiting for bus control to perform a DMA operation.

No Auto-Reload or Chaining (NAC). This bit is set under the following conditions:

- A channel completes a DMA operation and neither Base-to-Current reloading nor auto-chaining is enabled.
- A channel is issued an $\overline{EOP}$ during chaining.
- A Reset is issued to the DTC.

Chaining Abort (CA). This bit is set when a channel is issued an $\overline{EOP}$ during chaining or a Reset is issued to the DTC. The Chain Abort (CA) bit holds the No Auto-Reload or Chaining (NAC) bit in the set state until the $\overline{EOP}$ bit is cleared. The CA bit is cleared when a new Chain Address Segment and Tag word or Offset word is loaded into the channel.

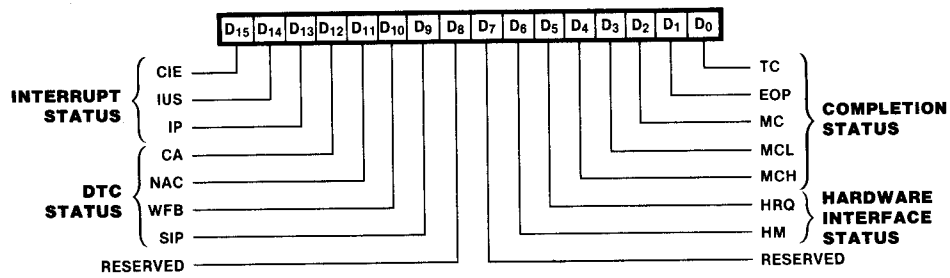
Interrupt Status Field. The Channel Interrupt Enable (CIE), Interrupt Pending (IP), and Interrupt Under Service (IUS) bits are used to control the way a channel generates an interrupt. An interrupt source with its IP bit set makes an interrupt request if all of the following conditions are met: Interrupts are enabled, (CIE bit = 1), there is no Interrupt Under Service (IUS bit = 0), no higher priority interrupt is being serviced, and no Interrupt Acknowledge transaction is in progress. When an interrupt source has an Interrupt Under Service (IUS = 1), all lower priority interrupt sources are prevented from requesting interrupts.

Interrupt Vector and Interrupt Save Registers. The 8-bit Interrupt Vector register contains the vector or identifier to be output during an Interrupt Acknowledge cycle. When an interrupt occurs, the contents of the Interrupt Vector register and bits ST_9–ST_{15} of the Status register are stored in the 16-bit Interrupt Save register. Because the vector and status are stored, a new vector can be loaded during chaining and a new DMA operation can be performed before an Interrupt Acknowledge cycle occurs. If another interrupt occurs on the channel before the first is acknowledged, further channel activity is suspended. When a clear IP command is issued, the status and vector for the second interrupt are loaded into the Interrupt Save register and channel operation resumes. The DTC can retain only two interrupts for each channel. The Interrupt Save register is fast-readable.

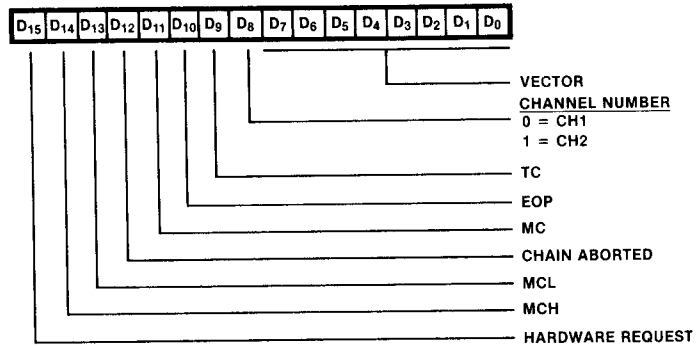
Chain Address Register. This register points to the chain control table in memory containing data to be loaded into the channel's registers. The Chain Address register consists of two words (Figure 11). The first word consists of a Segment and Tag field. The second word contains the 16-bit offset portion of the memory address. Bit 15 in the Segment field is ignored when the DTC is configured for logical address space (LPA = 1). The Tag field contains two bits used to designate the number of Wait states to be inserted during accesses to the Chain Control table. The Chain Address register is fast-readable and is loadable by chaining.

Table 4 provides a list of register addresses.

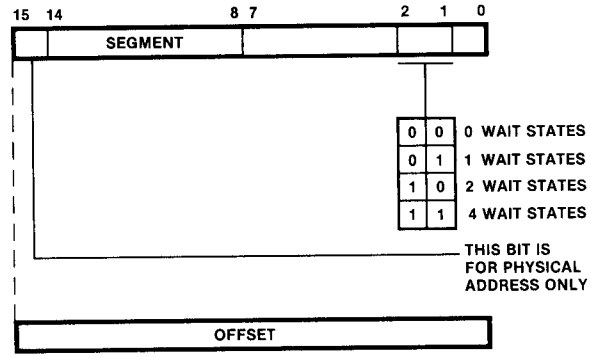
STATUS REGISTER



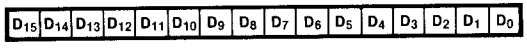
INTERRUPT SAVE REGISTER



CHAIN ADDRESS REGISTER



PATTERN AND MASK REGISTERS



INTERRUPT VECTOR REGISTER

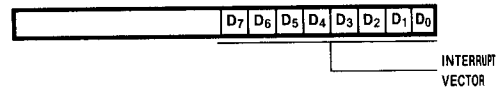


Figure 11. Special-Purpose Channel Registers

2129-011

Table 4. Register Address Summary

Address (AD_7-AD_0)	(Hex)	Control Registers
X0111100X	38	Master Mode
X0101111X	2E	Command Channel 1
X0101110X	2C	Command Channel 2
		General-Purpose Channel Registers
X0011101X	1A	Current Address Register A-Channel 1, Segment/Tag
X0001101X	0A	Current Address Register A-Channel 1, Offset
X0011100X	18	Current Address Register A-Channel 2, Segment/Tag
X0001100X	08	Current Address Register A-Channel 2, Offset
X0010001X	12	Current Address Register 8-Channel 1, Segment/Tag
X0000001X	02	Current Address Register B-Channel 1, Offset
X0010000X	10	Current Address Register B-Channel 2, Segment/Tag
X0000000X	00	Current Address Register B-Channel 2, Offset
X0011111X	1E	Base Address Register A-Channel 1, Segment/Tag
X0001111X	0E	Base Address Register A-Channel 1, Offset
X0011110X	1C	Base Address Register A-Channel 2, Segment/Tag
X0001110X	0C	Base Address Register A-Channel 2, Offset
X0010011X	16	Base Address Register B-Channel 1, Segment/Tag
X0000011X	06	Base Address Register B-Channel 1, Offset
X0010010X	14	Base Address Register B-Channel 2, Segment/Tag
X0000010X	04	Base Address Register B-Channel 2, Offset
X0110001X	32	Current Operation Count Channel 1
X0110000X	30	Current Operation Count Channel 2
X0110011X	36	Base Operation Count Channel 1
X0110010X	34	Base Operation Count Channel 2
		Special-Purpose Channel Registers
X1001101X	4A	Pattern Channel 1
X1001100X	48	Pattern Channel 2
X1001111X	4E	Mask Channel 1
X1001110X	4C	Mask Channel 2
X0101111X	2E	Status Channel 1
X0101110X	2C	Status Channel 2
X0101101X	2A	Interrupt Save Channel 1
X0101100X	28	Interrupt Save Channel 2
X1011101X	5A	Interrupt Vector Channel 1
X1011100X	58	Interrupt Vector Channel 2
X0100011X	26	Chain Address, Channel 1 Segment/Tag
X0100001X	22	Chain Address, Channel 4 Offset
X0100010X	24	Chain Address, Channel 2 Segment/Tag
X0100000X	20	Chain Address, Channel 2 Offset
X1010111X	56	Channel Mode Channel 1 High
X1010001X	52	Channel Mode Channel 1 Low
X1010101X	54	Channel Mode Channel 2 High
X1010000X	50	Channel Mode Channel 2 Low

NOTE: X = ignored.

ADDRESSING

The address generated by the DTC is always a byte address, even though the memory is organized as 16-bit words. All word-sized data is word-aligned and must be addressed by even addresses ($A_0 = 0$). With byte transfers, the least significant address bit determines which half of the A/D bus is used for the transfer. An even address specifies the most significant byte (AD_8–AD_{15}), and an odd address specifies the least significant byte (AD_0–AD_7). This addressing mechanism applies to memory accesses as well as to I/O and Special I/O accesses.

COMMANDS

The Z8016 DTC responds to several commands that give the CPU direct control over operating parameters. The commands described below are executed immediately after being written by the CPU into the DTC's Command register. A summary of the DTC commands is given in Table 5.

Reset

The Reset command forces the DTC into an idle state, in which it waits for a Start Chain command. The Start Chain command initiates a chain operation on either channel.

Software Request

A channel's Software Request command initiates a previously programmed transfer. If both channels are active, Channel 1 has priority.

Set/Clear Hardware Mask

The Set/Clear Hardware Mask command sets or clears the Hardware Mask bit in the selected channel's Mode register.

Table 5. DTC Command Summary

Command	Opcode Bits 7654	Opcode Bits 3210	Example Code (HEX)
Reset	000X	XXXX	00
Start Chain Channel 1	101X	XXX0	A0
Start Chain Channel 2	101X	XXX1	A1
Clear Software Request Channel 1	010X	XX00	40
Clear Software Request Channel 2	010X	XX01	41
Set Software Request Channel 1	010X	XX10	42
Set Software Request Channel 2	010X	XX11	43
Clear Hardware Mask Channel 1	100X	XX00	80
Clear Hardware Mask Channel 2	100X	XX01	81
Set Hardware Mask Channel 1	100X	XX10	82
Set Hardware Mask Channel 2	100X	XX11	83
Clear CIE, IUS, IP Channel 1	001E	SP00	*
Clear CIE, IUS, IP Channel 2	001E	SP01	*
Set CIE, IUS, IP Channel 1	001E	SP10	*
Set CIE, IUS, IP Channel 2	001E	SP11	*
Clear Flip Bit Channel 1	011X	XX00	60
Clear Flip Bit Channel 2	011X	XX01	61
Set Flip Bit Channel 1	011X	XX10	62
Set Flip Bit Channel 2	011X	XX11	63

*NOTES:
1. E = Set to 1 to perform set/clear on CIE, Clear to 0 for no effect on CIE.
2. S = Set to 1 to perform set/clear on IUS, Clear to 0 for no effect on IUS.
3. P = Set to 1 to perform set/clear on IP, Clear to 0 for no effect on IP.
4. X = "don't care" bit. This bit is not decoded and may be 0 or 1.
5. Flip bit = reset to 0 for ARA = src, ARB = dst. Set to 1 for ARA = dst, ARB = src.

Set/Clear IP, IUS, and CIE

The Set/Clear IP, IUS, and CIE commands manipulate the Interrupt Control bits located in each channel's Status register. These bits implement the interrupt daisy-chain control. The IP, IUS, and CIE bits for each channel can be set and cleared individually or in combination.

Set/Clear Flip Bit

The Set/Clear Flip Bit command reverses the source and destination, thereby reversing the direction of data transfer without reprogramming the channel.

TIMING

The following descriptions and timing diagrams refer to the relative timing relationships of DTC signals during basic operations. For exact timing information, refer to the composite timing diagrams.

Bus Request And Acknowledge

Before the DTC can perform a DMA operation, it must gain control of the system bus. The $\overline{BUSREQ}$, $\overline{BAI}$, and $\overline{BAO}$ interface pins provide connections between the DTC and the host CPU and other DMA devices to arbitrate which device has control of the system bus. When the DTC wants to gain bus control, it drives $\overline{BUSREQ}$ Low. Bus Request and Acknowledge timing is shown in Figure 12.

Flowthrough Transactions

Timing for Flowthrough I/O and Flowthrough Memory transactions (Figures 13 and 14, respectively) is identical. There are two types of I/O space on the Z8016: I/O and Special I/O. Status lines ST_0–ST_3 specify when an I/O operation is being performed and which of the two I/O spaces is being accessed. During an I/O transaction, status signal N/$\overline{S}$ will be Low to indicate a System Level operation.

The timing for I/O operations is identical to the timing of Flowthrough memory transactions. An I/O cycle consists of three states: T_1, T_2, and T_3. The TWA state is a Wait state that can be inserted into the transaction cycle. The $\overline{AS}$ output is pulsed Low to mark the beginning of a T-cycle. The N/$\overline{S}$ line is set Low (System) and the R/$\overline{W}$ and B/$\overline{W}$ lines select Read or Write operations for bytes or words. The N/$\overline{S}$, R/$\overline{W}$ and B/$\overline{W}$ lines become stable during T_1 and remain stable until the end of T_3.

I/O address space is byte-addressed but both 8- and 16-bit data sizes are supported. During I/O transactions, the B/$\overline{W}$ output is High for byte transactions and Low for word transactions.

The R/$\overline{W}$ output is High during Read operations and Low during Write operations. $\overline{DS}$ is driven Low to signal the peripherals that data can be gated onto, or received from, the bus. $\overline{DS}$ is driven High to signal the end of the I/O transaction.

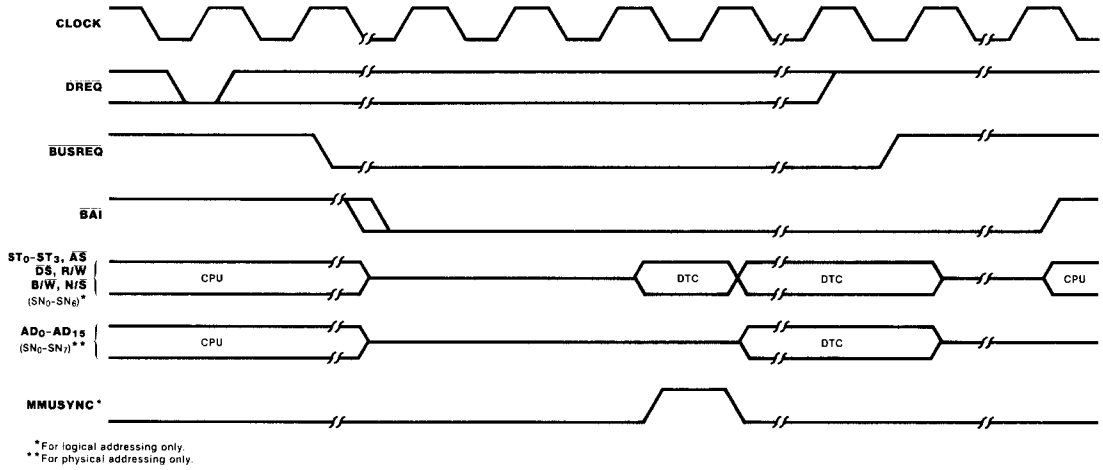
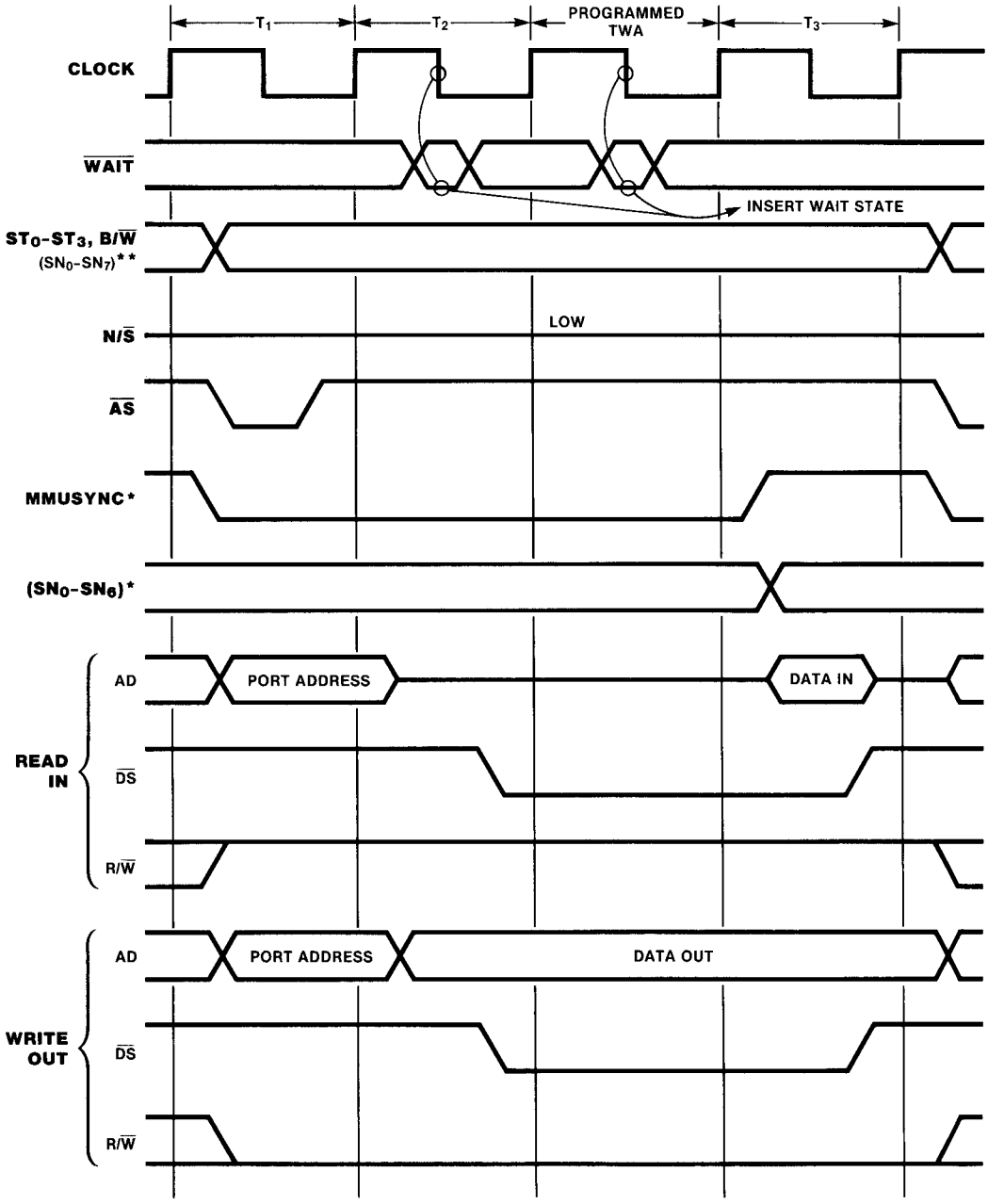
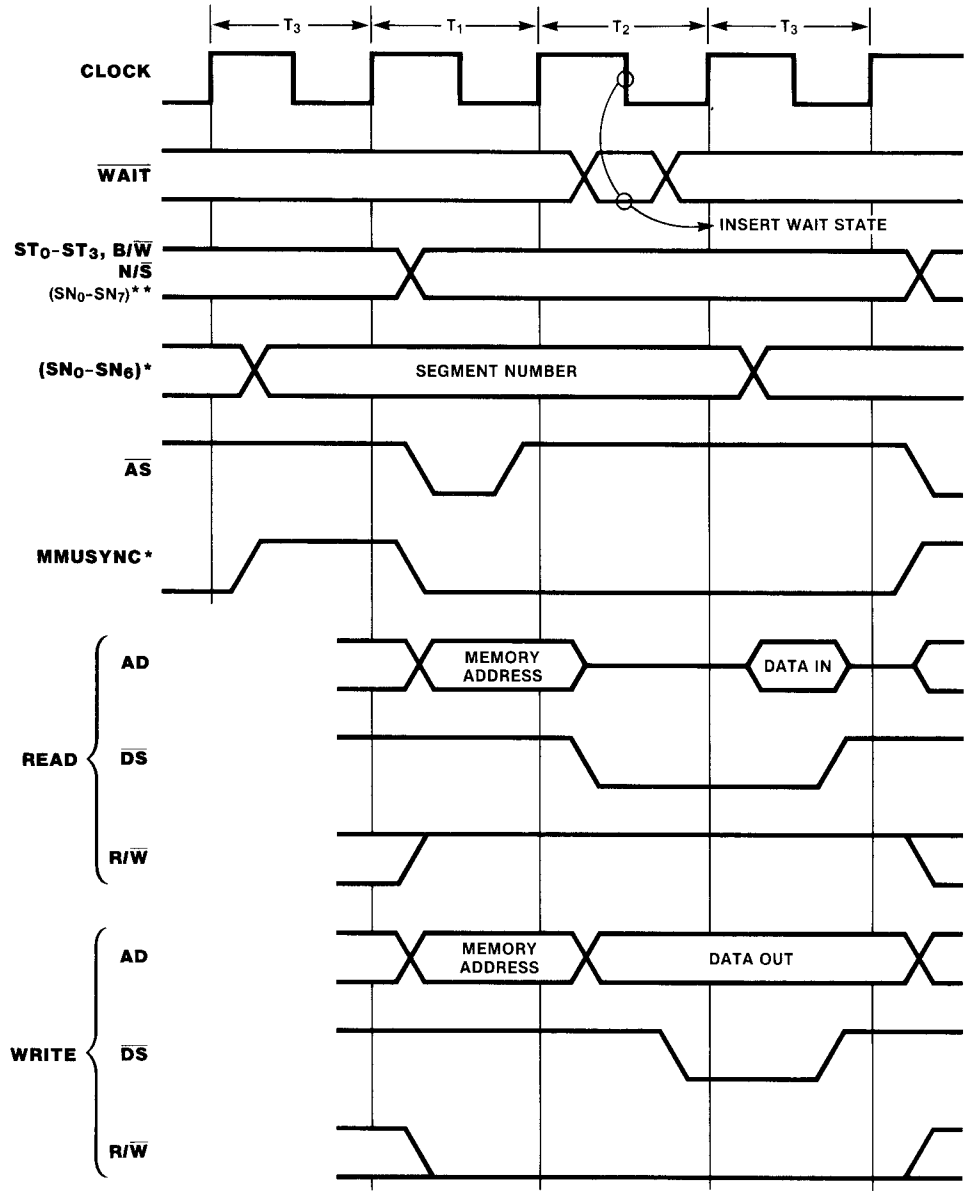


Figure 12. Bus Request and Acknowledge Timing



*For logical addressing only.
**For physical addressing only.

Figure 13. Flowthrough I/O Transaction Timing

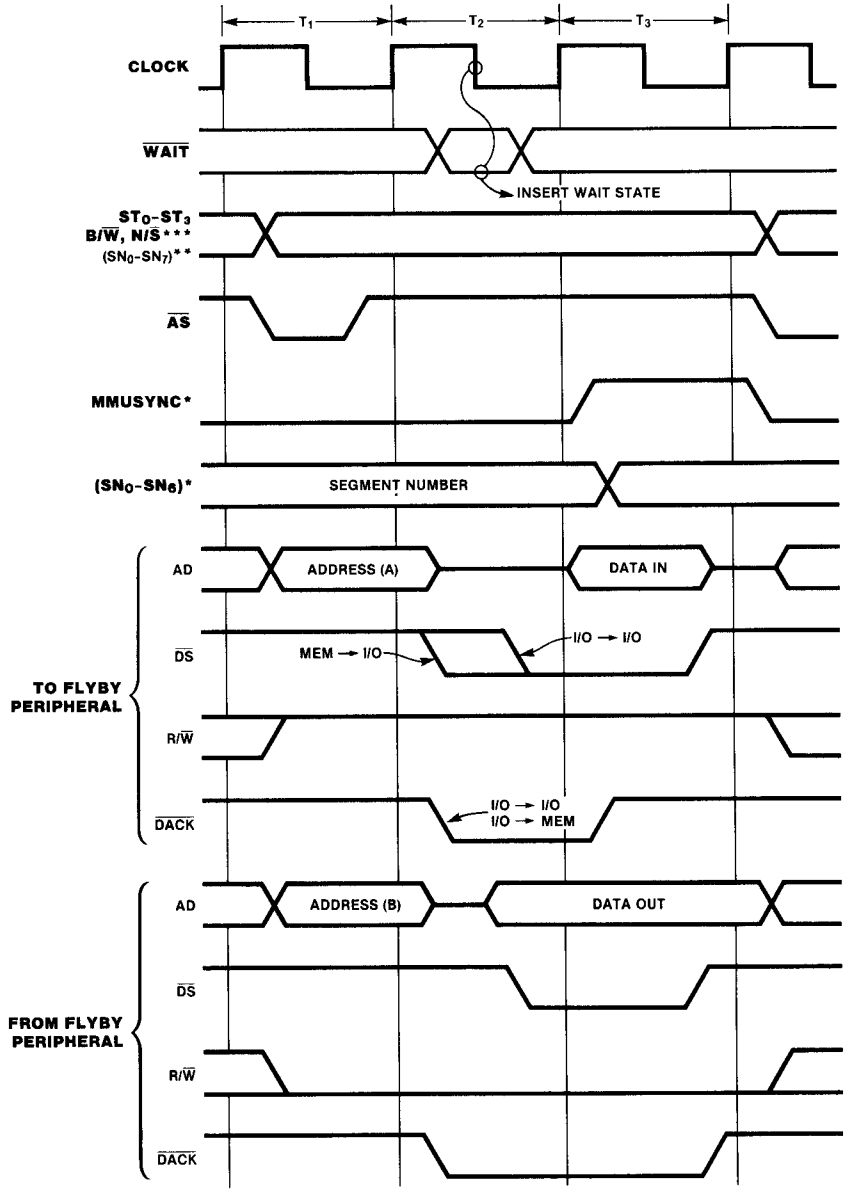


*For logical addressing only.
**For physical addressing only.

Figure 14. Flowthrough Memory Transaction Timing

Flyby Transactions

A Flyby operation is performed during three T-states. $\overline{AS}$ is pulsed during T_1 to signal the output of address information. R/$\overline{W}$ is High if the current ARA specifies source, and Low if the current ARB specifies destination. $\overline{DS}$ and $\overline{DACK}$ are driven active during T_2 to initiate the transfer, and driven inactive during T_3 to conclude the transfer. Wait states can be inserted between T_2 and T_3 to extend the active time to $\overline{DS}$ and $\overline{DACK}$. Flyby transaction timing is shown in Figure 15.



*Toggles for memory access in logical address space only.
**For physical addressing only.
***N/S will be low for I/O transactions.

(A) Address is current ARA
(B) Address is current ARB

Figure 15. Flyby Transaction Timing

2129-015

DREQ Timing

The following section describes DREQ timing for various operations.

A High-to-Low transition of DREQ causes a single iteration of a DMA operation. A new transition can occur after the Low-to-High AS transition on the first memory or I/O access of the DMA iteration. Figure 16 shows the timing for a new transition to be applied and recognized to avoid giving up the bus at the end of the current iteration.

In Bus Hold mode, DREQ is sampled when a channel gains bus control. If DREQ is Low, an iteration of a DMA operation is performed. If DREQ is High, the channel retains bus control and continues to drive all bus control signals active or inactive, but performs no DMA operation.

In Demand mode during DMA operation, DREQ is sampled to determine whether the channel should perform another cycle or release the bus (Figure 17).

DREQ is sampled after each End of Chaining or Base-to-Current Reloading operation. If DREQ is active, the channel begins performing DMA operations immediately, without releasing the bus.

DACK Timing

During I/O and memory transactions, WAIT is sampled in the middle of T_2. If WAIT is High, and no programmable Wait states are selected, the DTC proceeds to T_3. Otherwise, one or more Wait states are inserted. WAIT is also sampled during T_{WA}. If WAIT is High the DTC proceeds to T_3, otherwise, additional Wait states are inserted. When both hardware and software Wait states are inserted, each WAIT time is sampled. A Low causes a hardware Wait state to be inserted in the next cycle. Software Wait state insertion is suspended until WAIT is High. Hardware Wait states can be inserted any time during the software Wait state sequence. DACK timing is shown in Figure 18.

EOP Timing

EOP is driven Low when a TC, MC, or EOP termination occurs. When a DMA operation has terminated, EOP is sampled on the falling edge of T_3 to determine if EOP has been driven Low. The generation of internal EOPs and sampling of external EOPs for Transfers-and-Searches follows the same timing used for Transfers. EOP timing is shown in Figure 19.

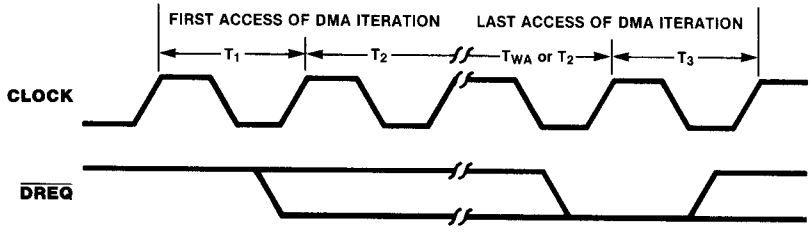
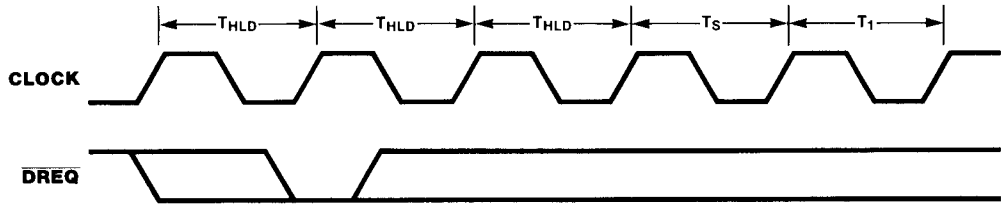
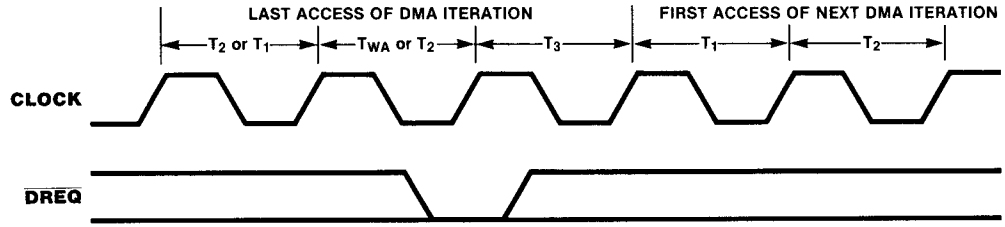


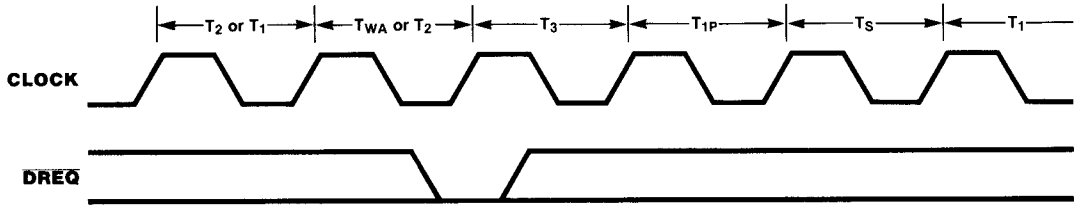
Figure 16. Sample DREQ During Single Transfer DMA Operations



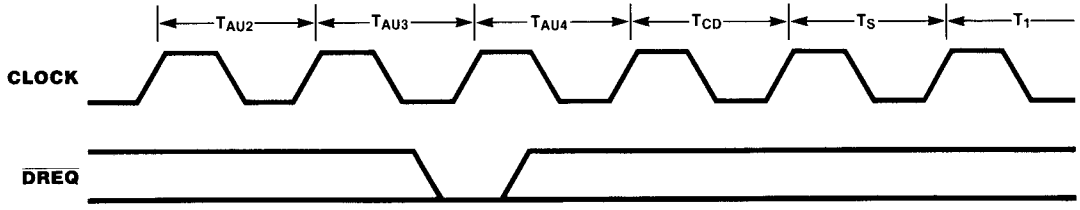
(A) Sampling of $\overline{DREQ}$ While in Bus Hold Mode



(B) $\overline{DREQ}$ Sampling in Demand Mode During DMA Operations



(C) Sampling $\overline{DREQ}$ at the End of Chaining



(D) Sampling $\overline{DREQ}$ at End of Base-to-Current Reloading

Figure 17. $\overline{DREQ}$ Sampling in Demand Mode

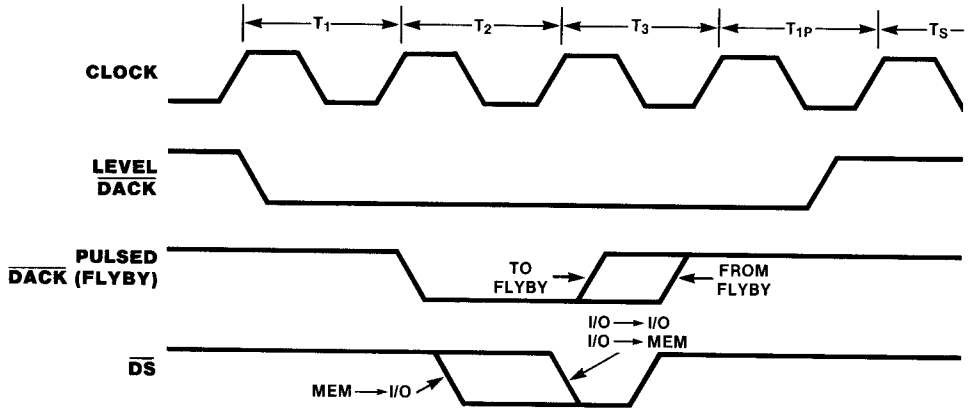


Figure 18. $\overline{\text{DACK}}$ Timing

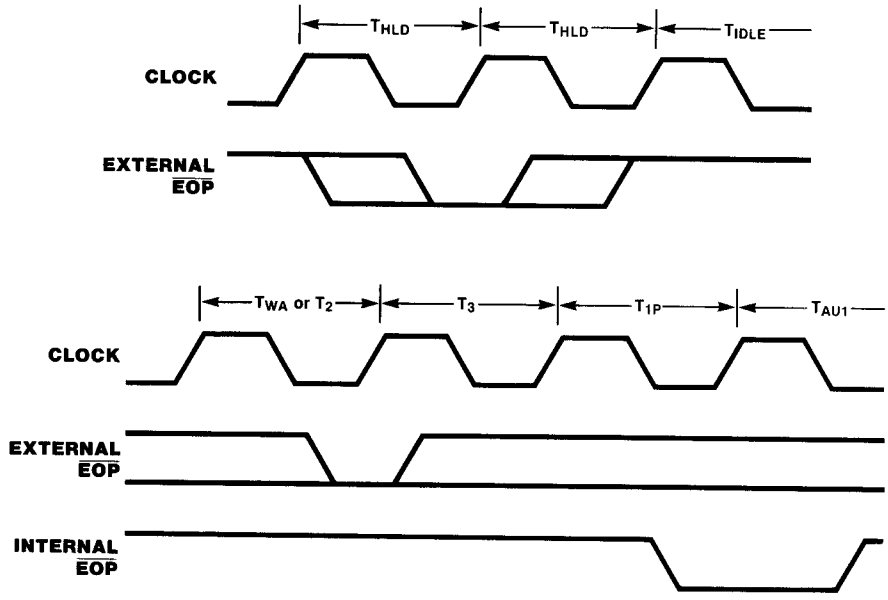
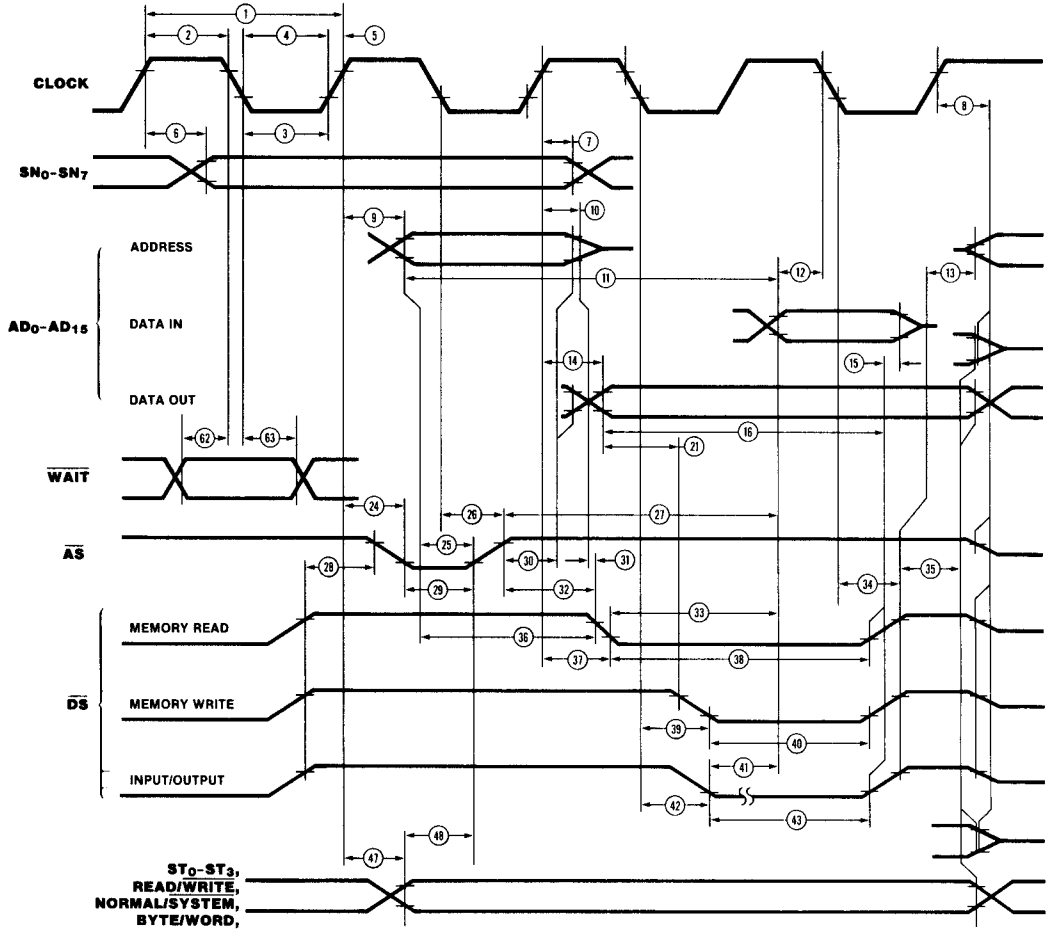


Figure 19. $\overline{\text{EOP}}$ Timing

ACTIVE STATE TIMING



AC CHARACTERISTICS†

Timing for DTC as Bus Master

Number	Symbol	Parameters	4 MHz Min	4 MHz Max	6 MHz Min	6 MHz Max
1	TcC	Clock Cycle Time	250	2000	165	
2	TwCh	Clock Width (High)	105		70	
3	TwCl	Clock Width (Low)	105		70	
4	TfC	Clock Fall Time		20		10
5	TrC	Clock Rise Time		20		15
6	TdC(SNv)	Clock ↑ to Segment Number Valid (50pf Load) Delay***		110		90
7	TdC(SNn)	Clock ↑ to Segment Number Valid Delay	20		10	
8	TdC(Bz)	Clock ↑ to Bus Float Delay		65		50
9	TdC(A)	Clock ↑ to Address Valid Delay		100		90
10	TdC(Az)	Clock ↑ to Address Float Delay		65		50
11	TdA(DI)	Address Valid to Data In Required Valid Delay	400		305	
12	TsDI(C)	Data In to Clock ↓ Setup Time	20		15	
13	TdDS(A)	$\overline{DS}$ ↑ to Address Active Delay	80		45	
14	TdC(DO)	Clock ↑ to Data Out Valid Delay		100		90
15	ThDI(DS)	$\overline{DS}$ ↑ to Data In Hold Time	0		0	
16	TdDO(DS)	Data Out Valid to $\overline{DS}$ ↑ Delay	230		200	
21	TdDO(SW)	Data Out Valid to $\overline{DS}$ ↓ (Write) Delay	55		35	
24	TdC(ASf)	Clock ↑ to $\overline{AS}$ ↓ Delay		70		60
25	TdA(AS)	Address Valid to $\overline{AS}$ ↑ Delay	50		35	
26	TdC(ASr)	Clock ↓ to $\overline{AS}$ ↑ Delay		80		60
27	TdAS(DI)	$\overline{AS}$ ↑ to Data In Required Valid Delay		300		220
28	TdDS(AS)	$\overline{DS}$ ↑ to $\overline{AS}$ ↓ Delay	75		35	
29	TwAS	$\overline{AS}$ Width (Low)	80		60	
30	TdAS(A)	$\overline{AS}$ ↑ to Address Valid Delay	60		45	
31	TdAz(DSR)	Address Float to $\overline{DS}$ (Read) ↓ Delay	0		0	
32	TdAS(DSR)	$\overline{AS}$ ↑ to $\overline{DS}$ ↓ (Read) Delay	75		40	
33	TdDSR(DI)	$\overline{DS}$ (Read) ↓ to Data In Required Valid Delay	165		155	
34	TdC(DSr)	Clock ↓ to $\overline{DS}$ ↑ Delay		70		65
35	TdDS(DO)	$\overline{DS}$ ↑ to Data Out (Write Only) and Status Valid (Read and Write) Delay	85		45	
36	TdA(DSR)	Address Valid $\overline{DS}$ (Read) ↓ Delay	120		110	
37	TdC(DSR)	Clock ↑ to $\overline{DS}$ (Read) ↓ Delay		60		60
38	TwDSR	$\overline{DS}$ (Read) Width (Low)	275		185	
39	TdC(DSW)	Clock ↓ to $\overline{DS}$ (Write) ↓ Delay		60		60
40	TwDSW	$\overline{DS}$ (Write) Width (Low)	160		150	
41	TdDSI(DI)	$\overline{DS}$ (Input) ↓ to Data In Required Valid Delay		325		210
42	TdC(DSf)	Clock ↓ to $\overline{DS}$ (I/O) ↓ Delay		60		60
43	TwDS	$\overline{DS}$ (I/O) Width (Low)	150*		150	
47	TdC(S)	Clock ↑ to Status Valid Delay		110		80
48	TdS(AS)	Status Valid to $\overline{AS}$ ↑ Delay	60		35	
62	TsWT(C)	$\overline{WAIT}$ to Clock ↓ Setup Time	20		20	
63	ThWT(C)	$\overline{WAIT}$ to Clock ↓ Hold Time	30		30	
96	TdC(SNr)	Clock ↑ to SN7/MMUSYNC ↑ Delay**		110		110
97	TdC(SNf)	Clock ↑ to SN7/MMUSYNC ↓ Delay**	20	110		110

NOTES:

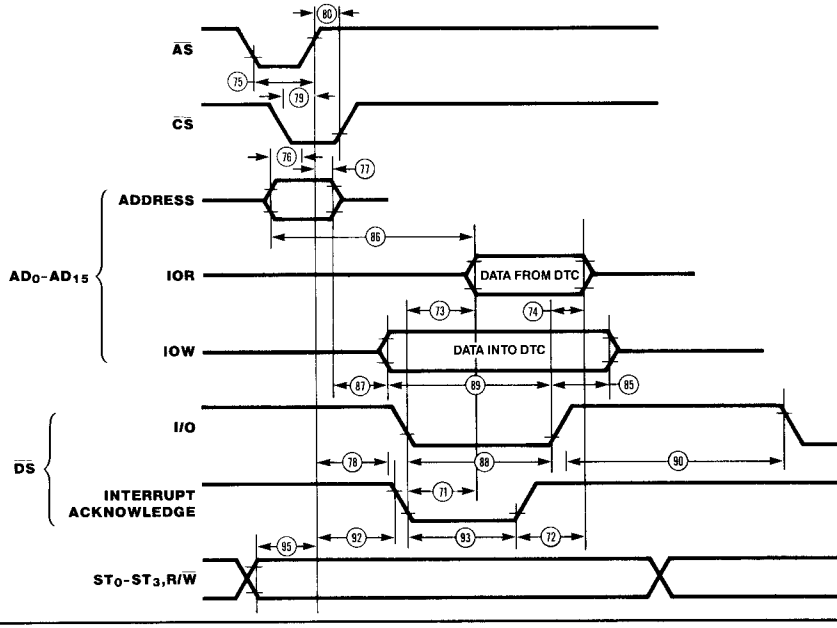
*Wait states should be inserted by programming a hardware when accessing slow peripherals.

**Logical Addressing only.

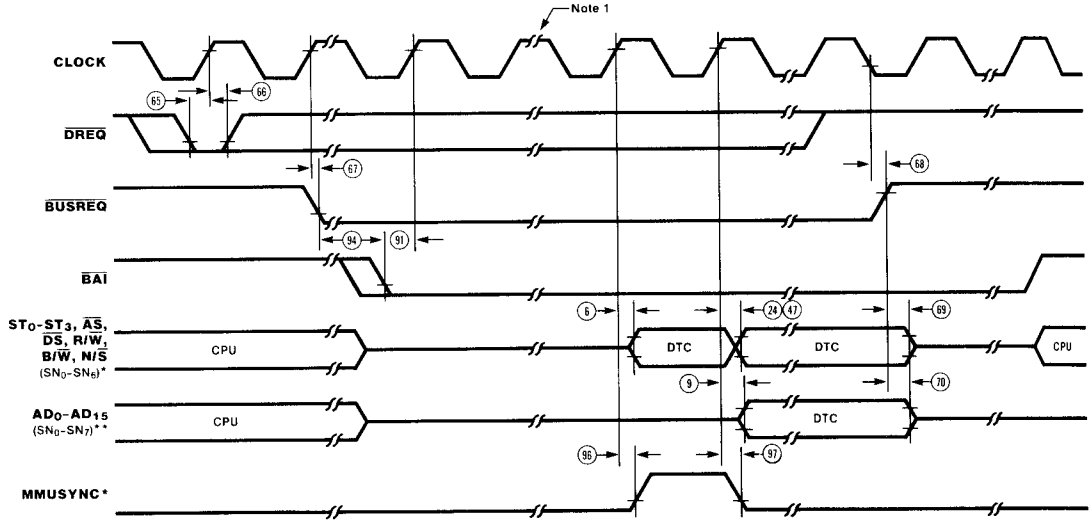
***130 ns max with Logical Addressing.

†Units in nanoseconds (ns).

INACTIVE STATE TIMING



BUS EXCHANGE TIMING



*For logical addressing only.
**For physical addressing only.
Note 1: The DTC will begin driving the bus on the clock cycle following the clock cycle in which the set-up parameters are met.

AC CHARACTERISTICS†

Timing for DTC as Bus Slave and CPU-DTC Bus Exchange

Number	Symbol	Parameters	4 MHz Min	4 MHz Max	6 MHz Min	6 MHz Max
64	TwDRQ	$\overline{DREQ}$ Pulse Width (Single Transfer Mode)	20		20	
65	TsDRQ(C)	$\overline{DREQ}$ Valid to Clock ↑ Setup Time	60		50	
66	ThDRQ(C)	Clock ↑ to $\overline{DREQ}$ Valid Hold Time	20		20	
67	TdC(BRQf)	Clock ↑ to $\overline{BUSREQ}$ ↓ Delay		150		120
68	TdC(BRQr)	Clock ↓ to $\overline{BUSREQ}$ ↑ Delay		165		150
69	TdBRQ(BUSc)	$\overline{BUSREQ}$ ↑ to Control Bus Float Delay		140		110
70	TdBRQ(BUSd)	$\overline{BUSREQ}$ ↑ to AD Bus Float Delay		140		110
71	TdDSA(RDV)	$\overline{DS}$ ↓ (Acknowledge) to Data Output Valid Delay		135		120
72	TdDSA(RDZ)	$\overline{DS}$ ↑ (Acknowledge) to Data Output Float Delay		80		75
73	TdDSR(DOD)	$\overline{DS}$ ↓ (IOR) to Data Output Driven Delay		135		120
74	TdDSR(RDZ)	$\overline{DS}$ ↑ (IOR) to Data Output Float Delay		80		75
75	TwAS	$\overline{AS}$ Low Width	70		50	
76	TsA(AS)	Address Valid to $\overline{AS}$ ↑ Setup Time	30		10	
77	ThAS(Av)	$\overline{AS}$ ↑ to Address Valid Hold Time	50		40	
78	TdAS(DS)	$\overline{AS}$ ↑ to $\overline{DS}$ ↓ Delay (I/O)	50		40	
79	TsCS(AS)	$\overline{CS}$ Valid to $\overline{AS}$ ↑ Setup Time	0		0	
80	ThCS(AS)	$\overline{AS}$ ↑ to $\overline{CS}$ Valid Hold Time	40		30	
81	TwAS(DS)	$\overline{AS}$ and $\overline{DS}$ Simultaneously Low Time (Reset)	3TcC		3TcC	
82	TdBAI(Az)	$\overline{BAI}$ ↑ to SN_0-SN_7, AD_0-AD_{15} Float Delay (Reset)		135		120
83	TdBAI(ST)	$\overline{BAI}$ ↑ to ST_0-ST_3, R/$\overline{W}$, B/$\overline{W}$, N/$\overline{S}$ Float Delay (Reset)		100		80
84	TdBAI(DS)	$\overline{BAI}$ ↑ to $\overline{DS}$, $\overline{AS}$ Float Delay (Reset)		100		85
85	TdDS(Dn)	$\overline{DS}$ ↑ (IOW) to Data Valid Hold Time	40		40	
86	TdAC(DRV)	Address Valid to Data (IOR) Required Valid Delay		540		345
87	TdAZ(DS)	Address Float to DS ↓ (IOR) Delay	0		0	
88	TwDS(IO)	$\overline{DS}$ (IO) Low Width	150*		150	
89	TsD(DS)	Data (IOW) Valid to $\overline{DS}$ ↑ Setup Time	40		40	
90	TrDS(W)	$\overline{DS}$ ↑ (IOW) to $\overline{DS}$ ↓ (IOW) (Write Recovery Time applies only for issuing Command)	4TcC		4TcC	
91	TsBAK(C)	$\overline{BAI}$ Valid to Clock ↑ Setup Time	60		50	
92	TdAS(DS)	$\overline{AS}$ ↑ to DS ↓ (ACK) Delay	100		100	
93	TwDS(AK)	$\overline{DS}$ (ACK) Low Width	150		150	
94	TdBRQ(BAI)	$\overline{BUSREQ}$ ↓ to $\overline{BAI}$ ↓ Required Delay	0		0	
95	TsS(AS)	Status Valid to $\overline{AS}$ ↑ Setup Time	40		0	
98	TdBAI (BAO)	$\overline{BAI}$ ↑, ↓ to $\overline{BAO}$ ↑, ↓ Delay		80		70
99	TdIEI(IEO)	IEI ↑, ↓ to IEO ↑, ↓ Delay		80		60

NOTES:
*2000 ns for reading slow-readable registers (worst case)
†Units in nanoseconds (ns).

ABSOLUTE MAXIMUM RATINGS

Voltages on all pins with respect to GND . . . . . −0.3V to +7.0V
Operating Ambient Temperature . . . . . See Ordering Information
Storage Temperature . . . . . −65°C to +150°C

Stresses greater than those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; operation of the device at any condition above those indicated in the operational sections of these specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

STANDARD TEST CONDITIONS

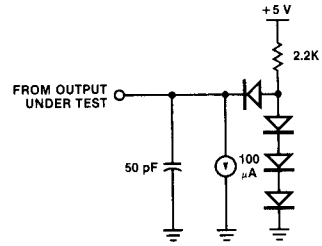
The DC characteristics and capacitance sections below apply for the following standard test conditions, unless otherwise noted. All voltages are referenced to GND. Positive current flows into the referenced pin.

Standard conditions are as follows:

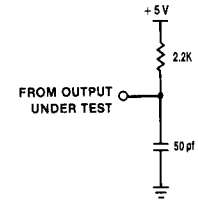
- $+4.75V \leq V_{CC} \leq +5.25V$
- GND = 0V
- T_A as specified in Ordering Information

All AC parameters assume a load capacitance of 50 pf max.

The Ordering Information section lists package temperature ranges and product numbers. Package drawings are in the Package Information section. Refer to the Literature List for additional documentation.



Standard Test Load



Open-Drain Test Load

DC CHARACTERISTICS

Symbol	Parameter	Min	Max	Unit	Condition
V_{CH}	Clock Input High Voltage	V_{CC}-0.4	V_{CC}+0.3	V	Driven by External Clock Generator
V_{CL}	Clock Input Low Voltage	-0.3	0.45	V	Driven by External Clock Generator
V_{IH}	Input High Voltage	2.0	V_{CC}+0.3	V	
V_{IL}	Input Low Voltage	-0.3	0.8	V	
V_{OH}	Output High Voltage	2.4		V	I_{OH} = −250 μA
V_{OL}	Output Low Voltage		0.4	V	I_{OL} = +2.0 mA
I_{IL}	Input Leakage		±10	μA	$0.4 \leq V_{IN} \leq V_{CC}$
I_{OL}	Output Leakage		±10	μA	$0.4 \leq V_{IN} \leq +V_{CC}$
I_{CC}	V_{CC} Supply Current		350	mA	T_A = 0°C

NOTE: V_{CC} = 5 V ± 5% unless otherwise specified.

CAPACITANCE

Symbol	Parameter	Min	Max	Unit
C_{CLOCK}	Clock Capacitance		40	pf
C_{IN}	Input Capacitance		5	pf
C_{OUT}	Output Capacitance		10	pf

T_A = 25°C, f = 1 MHz.
Unmeasured pins returned to ground.

ORDERING INFORMATION

Z8016 Z-DTC, 4.0 MHz	Z8016A Z-DTC, 6.0 MHz
48-pin DIP	**48-pin DIP**
Z8016 PS	Z8016A PS
Z8016 CS	Z8016A CS
Z8016 PE	Z8016A PE
Z8016 CE	Z8016A CE

Codes

First letter is for package; second letter is for temperature.

C = Ceramic DIP
P = Plastic DIP
L = Ceramic LCC
V = Plastic PCC

R = Protopack
T = Low Profile Protopack
DIP = Dual-In-Line Package
LCC = Leadless Chip Carrier
PCC = Plastic Chip Carrier (Leaded)

TEMPERATURE
S = 0°C to +70°C
E = −40°C to +85°C
M*= −55°C to +125°C

FLOW
B = 883 Class B

Example: PS is a plastic DIP, 0°C to +70°C.

†Available soon.
*For Military Orders, contact your local Zilog Sales Office for Military Electrical Specifications.