μPD71037
Direct Memory Access (DMA) Controller

Description

The μPD71037 is a direct memory access (DMA) controller that provides high-speed data transfers between peripheral devices and memory for microprocessor systems. It is faster and draws less power than its predecessors. The unit has four DMA channels, each with a 64K-byte address area and a transfer byte count function. The channels enable I/O-to-memory and memory-to-memory data transfer.

The μPD71037 is a versatile DMA controller that can be used for the following applications.

- Office automation equipment (personal computers, small business computers, EWS, etc.)
- Communications
- Instrumentation
- Control

Features

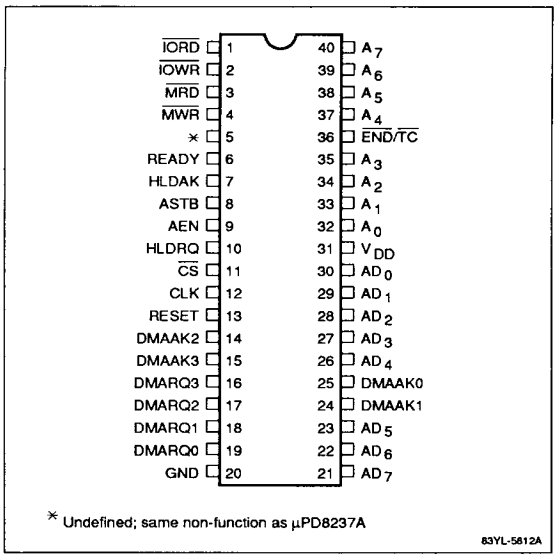
- Processing speed of 10 MHz (twice that of the μPD8237A-5)
- Four independent DMA channels
- Self-initialization for each channel
- Memory-to-memory data transfer
- Block-level memory initialization
- High-speed data transfer
  — 3.2 Mb/s, 10-MHz normal transfer
  — 5.0 Mb/s, 10-MHz compression transfer
- DMA channel count directly expandable in expansion mode
- END input for the end of transfer
- Software DMA request
- CMOS
- Low power consumption

Ordering Information

Part Number	Clock (MHz)	Package
μPD71037CZ-10	10	40-pin plastic DIP (600 mil)
GB-10	10	44-pin plastic QFP
LM-10	10	44-pin PLCC

Pin Configurations

40-Pin Plastic DIP

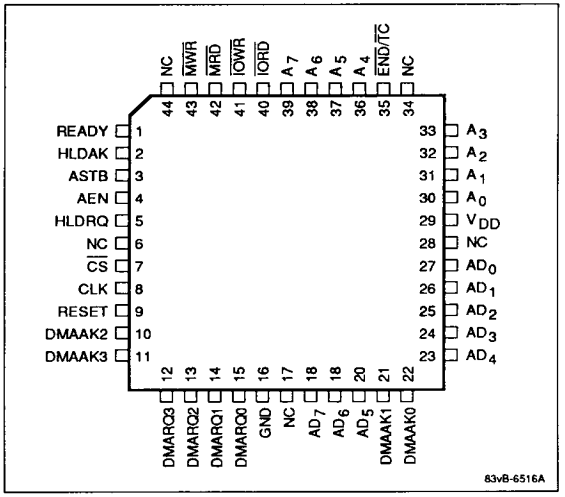


* Undefined; same non-function as μPD8237A

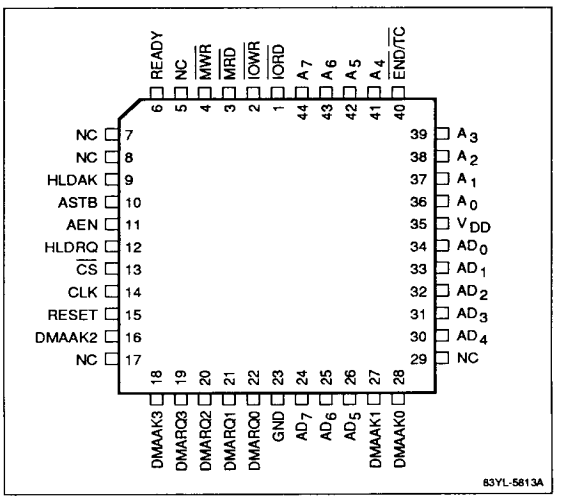
83YL-5812A

5b

44-Pin Plastic QFP



44-Pin PLCC



Pin Identification

Symbol	I/O	Function
A_0-A_3	3-state I/O	Four low-order bits of address bus
A_4-A_7	3-state out	Middle four bits of the output state address bus
AD_0-AD_7	3-state I/O	Eight high-order bits of the address and functions as an 8-bit data bus
AEN	Out	Permits output from an external latch connected to the μPD71037
ASTB	Out	Makes an external latch to the high-order address
CLK	In	Clock input for internal operations and data transfer speeds
$\overline{CS}$	In	Selects the μPD71037 as an I/O device and enables read/write operation
DMAAK0-DMAAK3	Out	Permits peripheral device to perform DMA transfer
DMARQ0-DMARQ3	In	Requests the μPD71037 to perform DMA transfer
$\overline{END/TC}$	I/O	Input that forces the μPD71037 to terminate DMA transfer; output that posts the end of DMA transfer
HLDAK	In	Permits the μPD71037 to hold the bus
HLDRQ	Out	Requests the host CPU to hold the bus
$\overline{IORD}$	3-state I/O	Input that enables the host CPU to read the μPD71037 status; output that enables the μPD71037 to read data from a peripheral device during DMA transfer
$\overline{IOWR}$	3-state I/O	Input that enables the host CPU to write data to the μPD71037; output that enables the μPD71037 to write data to a peripheral device during DMA transfer
$\overline{MRD}$	3-state out	Memory read during DMA transfer
$\overline{MWR}$	3-state out	Memory write during DMA transfer
READY	In	Requests extension of a read/write cycle during DMA transfer
RESET	In	Initializes the μPD71037
NC	—	No connection
V_{DD}	—	Positive power supply voltage
GND	—	Ground reference

PIN FUNCTIONS

A_0-A_3 (Address Bus)

A_0-A_3 input and output the four low-order bits of the address bus. During the inactive cycle, these pins are used as inputs to enable the host CPU to select an appropriate μPD71037 register. During the DMA cycle, A_0-A_3 output an address for memory access.

A_4-A_7 (Address Bus)

A_4-A_7 output the middle bits of the address bus. During the inactive cycle, these pins have high impedance. During the DMA cycle, A_4-A_7 output an address for memory access.

AD_0-AD_7 (Address/Data Bus)

AD_0-AD_7 input and output the high-order byte of an address bus and also function as a data bus. This is done by time multiplexing.

During the DMA cycle, the high-order byte of a memory address is output. During memory-to-memory transfer, memory addresses are output and these pins intermediate the memories for data transfer. During the inactive cycle, these pins act as a data bus when the host CPU reads or writes data from or to the μPD71037.

AEN (Address Enable)

AEN outputs an active-high AEN signal to enable an external latch which latches the high-order byte of an address during the DMA cycle. The AEN signal is fixed at a high level.

ASTB (Address Strobe)

ASTB outputs an active-high strobe signal to make an external latch retain the high-order byte of an address during the DMA cycle.

CLK (Clock)

CLK controls all internal μPD71037 operations and inputs a clock signal to control the DMA transfer speed. A maximum clock signal of 10 MHz can be input.

$\overline{CS}$ (Chip Select)

During the inactive cycle, $\overline{CS}$ inputs an active-low signal to enable the host CPU to handle the μPD71037 as an ordinary I/O device to read data to or write data from it. During the DMA cycle, this input is internally disabled to inhibit reading or writing by the host CPU.

DMAAK0-DMAAK3 (DMA Acknowledge)

DMAAK0-DMAAK3 indicate to peripheral devices that DMA service has been granted. DMAAK0-DMAAK3 respond respectively to DMA channels 0-3. Like the DMARQ pins, the active level of these pins is programmable; however, they can be set to an active-high level by a RESET input.

DMARQ0-DMARQ3 (DMA Request)

DMARQ0-DMARQ3 accept DMA service requests from peripheral devices. DMARQ0-DMARQ3 respond respectively to DMA channels 0-3 and the active level for these pins is programmable. However, they can be set to an active-high level by a RESET input. In fixed nest mode (see Channel Priority), the lowest priority is given to DMARQ3 and the highest to DMARQ0.

$\overline{END}/\overline{TC}$ (End/Terminal Count)

This is a bidirectional pin. The $\overline{END}$ input is used to terminate the current DMA transfer. $\overline{TC}$ indicates the designated cycles of the DMA count transfer have finished. If the low $\overline{END}$ signal is input during the DMA cycle, the μPD71037 forcibly terminates DMA services. If the specified number of data transfers is reached, the μPD71037 outputs a low $\overline{TC}$ signal.

When an $\overline{END}$ is input or $\overline{TC}$ is output, the μPD71037 clears the software DMA request (see Self-Initialization). The contents of the address set register are sent to the effective address register.

If self-initialization is not set, the $\overline{TC}$ bit of the status read register and the mask bit of the mask control register are set by the $\overline{END}/\overline{TC}$ signal. (The corresponding bit of the channel using DMA transfer is set.) In memory-to-memory transfer, $\overline{TC}$ is output when the specified number of DMA transfers is reached in channel 1.

If $\overline{END}$ is not input, pull up this pin to prevent a low signal from causing a malfunction.

HLDAK (Hold Acknowledge)

When active, HLDAK generates an active-high signal to indicate that the host CPU has granted the μPD71037 the use of the system bus. When this signal is input,the μPD71037 enters a DMA cycle.

HLDRQ (Hold Request)

HLDRQ outputs an active-high signal requesting the host CPU to hold the bus.

$\overline{IORD}$ (I/O Read)

$\overline{IORD}$ inputs or outputs an active-low signal to enable the host CPU to read μPD71037 data or to enable the μPD71037 to read from peripheral devices.

During the inactive cycle, the read signal is input to enable the host CPU to read a μPD71037 register. During the DMA cycle, the read signal is output to enable the μPD71037 to read data from a peripheral device.

$\overline{IOWR}$ (I/O Write)

$\overline{IOWR}$ inputs or outputs an active-low signal to enable the host CPU to write μPD71037 data or to enable the μPD71037 to write data to peripheral devices.

During the DMA cycle, the write signal is output to enable the μPD71037 to write data to a peripheral device.

During the inactive cycle, the write signal is input to enable the host CPU to write data to a μPD71037 register.

$\overline{MRD}$ (Memory Read)

During the DMA cycle, $\overline{MRD}$ outputs an active-low $\overline{MRD}$ signal to read data from memory. During the inactive cycle, this pin has high impedance.

$\overline{MWR}$ (Memory Write)

During the DMA cycle, $\overline{MWR}$ outputs an active-low $\overline{MWR}$ signal to write data into memory. During the inactive cycle, this pin has high impedance.

READY (Ready)

READY inputs an active-high signal to mark the end of each data transfer during a DMA operation. If a low-speed peripheral device fails to finish transferring data within a given read/write cycle, the width of the read/write signal output by the μPD71037 can be extended by inputting the low READY signal to this pin. During the extension, the μPD71037 enters a wait cycle (TW).

RESET (Reset)

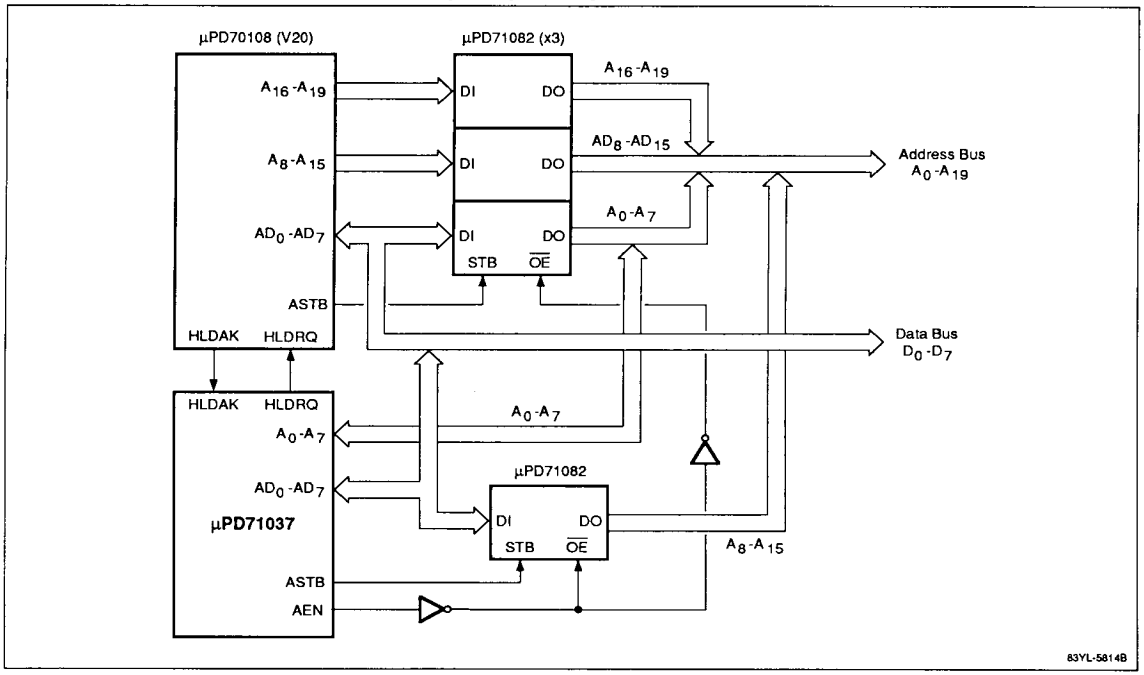
RESET inputs an active-high signal to initialize the internal μPD71037 statuses (register contents, etc.). Inputting this signal returns the μPD71037 to an inactive cycle.When RESET is input, the following control registers are cleared to 00H.

- Device control register
- Status read register
- Request control register
- Temporary data register

The DMA requests (DMARQ pin input) are masked for all four DMA channels. The control registers are described later in the Registers section.

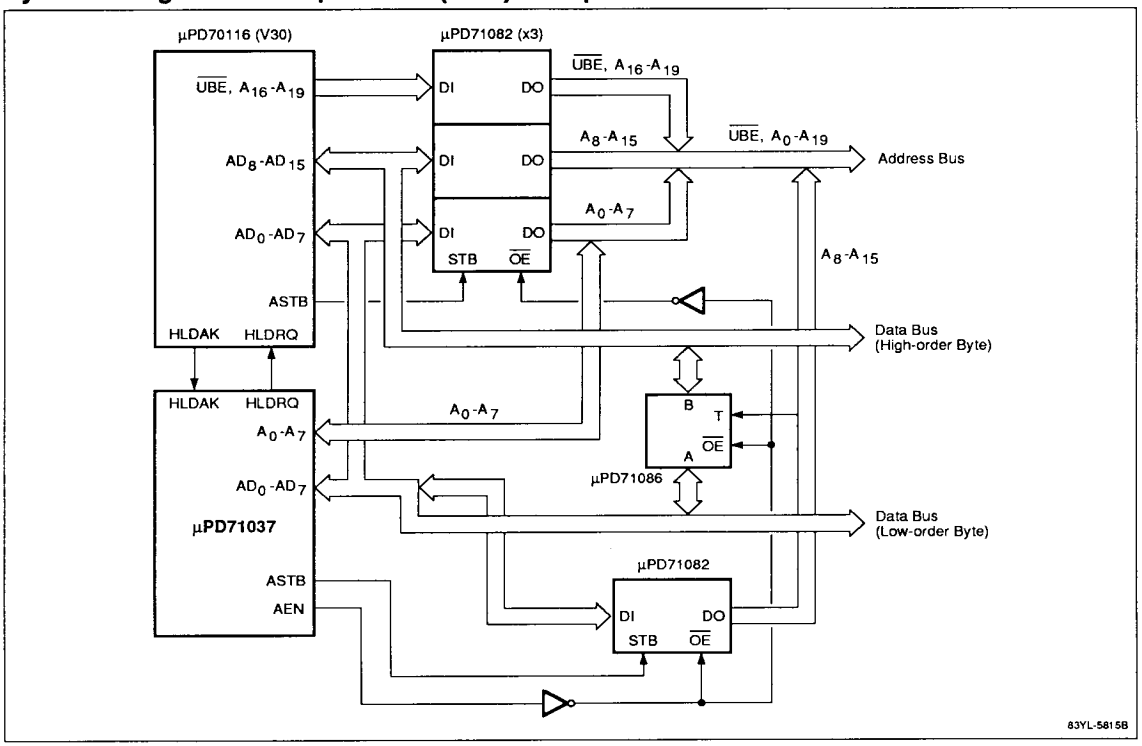
After RESET is input, an address-low byte command (see Commands section) is issued to the μPD71037 and the first byte is placed as the low-order byte in the address/count register.

System Configuration With μPD70108 (V20®) Microprocessor



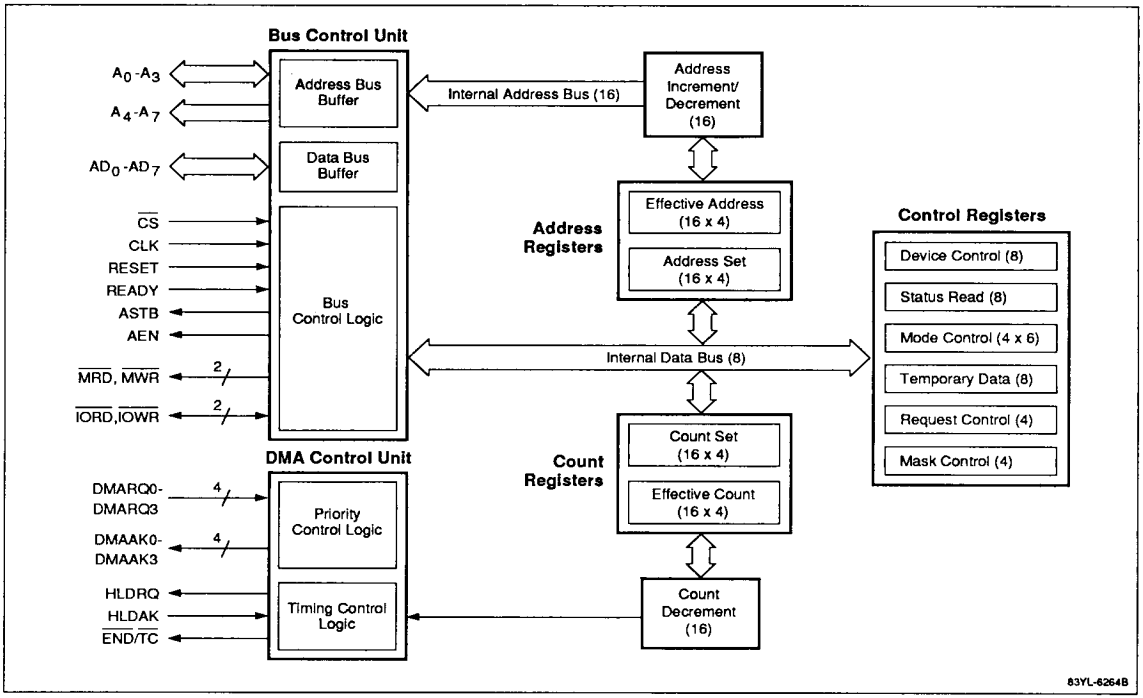
5b

System Configuration With μPD70116 (V30®) Microprocessor



V20 and V30 are registered trademarks of NEC Corporation

μPD71037 Block Diagram



INTERNAL BLOCK FUNCTIONS

The μPD71037 has the following functional units as shown in the block diagram.

- Bus control unit
- DMA control unit
- Address registers
- Address incrementer/decrementer
- Count registers
- Count decrementer
- Control registers

Bus Control Unit

The bus control unit consists of the address and data buffers and bus control logic. The bus control unit generates and receives signals that control addresses and data on the internal address and data buses.

DMA Control Unit

The DMA control unit contains the priority and timing control logic. The priority control logic determines the priority level of DMA requests and arbitrates the use of the bus in accordance with this priority level. The DMA control unit also provides internal timing and controls DMA operations.

Address Registers

Each of the four DMA channels has one address set register and one effective address register. Each register stores a DMA 16-bit address. The effective address register is updated for each single-byte DMA transfer and constantly holds the address to be transferred next. The contents of the address set register remain unchanged until the host CPU writes a new value to it. At self-initialization, the initial DMA address for the next DMA service is transferred from the address set register to the effective address register.

Address Incrementer/Decrementer

The address incrementer/decrementer updates the contents of the current address register whenever a DMA transfer completes.

Count Registers

Each of the four DMA channels has one 16-bit count set register and one 16-bit effective count register that store a DMA transfer byte count. The count set register holds a value written by the CPU. At self-initialization the value is transferrred to the effective address register where it is set as the number of DMA transfers in the next DMA service.

A channel's effective count register is decremented by 1 for each single-byte transfer and constantly holds the remaining number of DMA transfers. If a borrow occurs when this register is decremented, the terminal count is set to mark the end of the specified number of DMA transfers.

Count Decrementer

The count decrementer decrements the contents of the effective count register by 1 when each DMA transfer takes place.

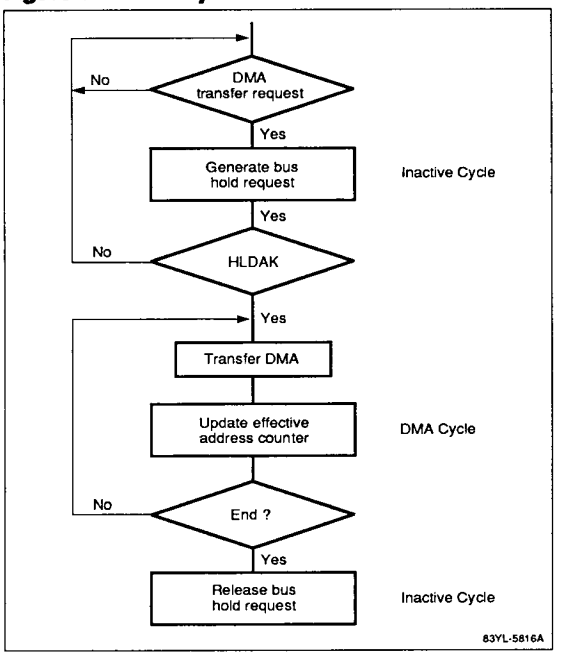
Control Registers

The μPD71037 contains six registers that control the bus mode, pin active levels (DMARQ and DMAAK), and the DMA transfer mode.

DMA OPERATION

The μPD71037 operates in an inactive cycle and a DMA cycle. Figure 1 illustrates basic DMA operation flow.

Figure 1. DMA Operation Flow



Inactive Cycle

During the inactive cycle, the host CPU has authority and the μPD71037 is in one of the following states.

- The μPD71037 has not yet received an effective DMA service request from a peripheral device.
- The μPD71037 has received an effective DMA service request, but has not yet received bus authority from the host CPU.
- The μPD71037 performs the following operations during the inactive cycle.
  - — Detecting a DMA service request
  - — Requesting bus authority
  - — Selecting a DMA channel
  - — Programming

In the inactive cycle, there are no active DMA cycles but there may be one or more active DMA requests. However, the CPU has not yet released the bus.

Detecting a DMA Service Request. The μPD71037 will sample the four DMARQ input pins for each clock signal.

Requesting Bus Authority. When an effective (unmasked) DMA request is received, a bus hold request signal (HLDRQ) is output to the host CPU. The μPD71037 continues to sample DMA requests until it obtains the bus by HLDAK input.

Selecting a DMA Channel. After the CPU returns an HLDAK signal and the μPD71037 obtains the bus, the μPD71037 stops DMA sampling and selects the DMA channel with the highest priority from the valid DMA request signals.

Programming. Before DMA transfer, the transfer addresses, the number of DMA transfers, the DMA transfer mode, and the active levels of the DMARQ and DMAAK pins must be determined.

While the host CPU holds bus authority, μPD71037 programming can be done by inputting a low signal to the $\overline{CS}$ pin. The four low-order address bits (A_0-A_3) specify a register for a read/write operation. Inputting an $\overline{IORD}$/$\overline{IOWR}$ signal performs the operation on the specified register.

DMA Cycle

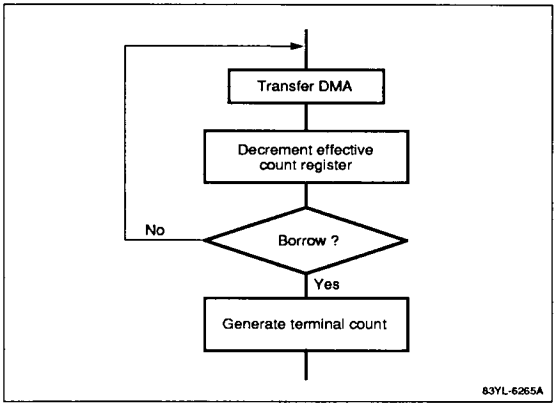
In a DMA cycle, the μPD71037 controls the bus and performs DMA transfer operations based on programmed information.

Terminal Count. External input of an $\overline{END}$ signal or internal generation of a terminal count terminates DMA transfer. The terminal count is generated when a borrow occurs as a result of decrementing the effective count register, which counts the number of DMA transfers in bytes. When this occurs, the μPD71037 outputs the low level pulse $\overline{TC}$.

Figure 2 shows that the effective count register is tested after each DMA operation. A borrow is detected after each DMA transfer is completed. As a result, the actual number of DMA transfers is one greater than the value set in the effective count register.

If self-initialization is not set when DMA service ends, the mask control register bits applicable to the channel where service is ended are set, and the DMARQ input of that channel is masked.

Figure 2. Generation of Terminal Count (TC)



DMA Transfer Type

The type of transfer the μPD71037 performs depends on the following conditions.

- Memory-to-memory transfer enable
- Direction of I/O-to-memory transfer for each channel
- Transfer mode of each channel

Memory-to-Memory Transfer Enable. The μPD71037 performs each DMA transfer (1 byte of data) between an I/O device and memory in a one-bus cycle and between memories in a two-bus cycle.

5b

Memory-to-memory transfer can occur only when bit 0 of the device control register is set to 1. The DMA channels used in memory-to-memory transfers are fixed, with channel 0 as the source channel and channel 1 as the destination channel. Channels 2 and 3 cannot be used in memory-to-memory transfers. The contents of the count registers of each channel should be the same when performing this type of DMA transfer.

The μPD71037 performs the following operations until a channel 0 terminal count or until $\overline{END}$ input is present. Only the block mode is valid for this type of transfer.

(1) The memory data pointed to by the effective address register of channel 0 is read into the temporary data register of the μPD71037 and the effective address register and effective count register of channel 0 are updated.

(2) The temporary register data is written to the memory location shown by the effective address register of channel 1; the effective address register and effective count register are updated.

During memory-to-memory transfers, the address of the transfer source can be fixed using the device control register. In this manner, a range of memory can be padded with the same value (0 or 1) since the contents of the source address never change. During memory-to-memory transfer, the DMAAK signal and channel 0's terminal count ($\overline{TC}$) are not output.

Note: If DMARQ1 (channel 1) becomes active, the μPD71037 will perform memory-to-I/O transfer even though memory-to-memory transfer is selected. Since this may cause erroneous memory-to-memory transfers, mask out channel 1 (DMARQ1) by setting bit 1 of the mask register to 1 before starting memory-to-memory transfers.

Direction of I/O-to-Memory Transfers. All DMA transfers use memory as a reference point. Therefore, a DMA read reads a memory location and writes to an I/O port. A DMA write reads an I/O port and writes the data to a memory location. In memory-to-I/O transfer, use the mode control register to set one of the transfer directions in table 1 for each channel and activate the appropriate control signals.

Table 1. Transfer Direction

Transfer Direction	Activated Signals
Memory to I/O (DMA read)	$\overline{IOWR}$, $\overline{MRD}$
I/O to memory (DMA write)	$\overline{IORD}$, $\overline{MWR}$
Verify (Outputs addresses only. Does not perform a transfer.)	—

Transfer Modes. In I/O-to-memory transfer, the mode control register selects the single, demand, or block mode of DMA transfer for each channel. Table 2 shows the various transfer modes and termination conditions.

Table 2. Transfer Termination

Transfer Mode	End of Transfer Condition
Single	After 1 byte of data is transferred
Demand	$\overline{END}$ input Generation of terminal count When DMARQ for the channel in DMA service becomes inactive
Block	$\overline{END}$ input Generation of terminal count

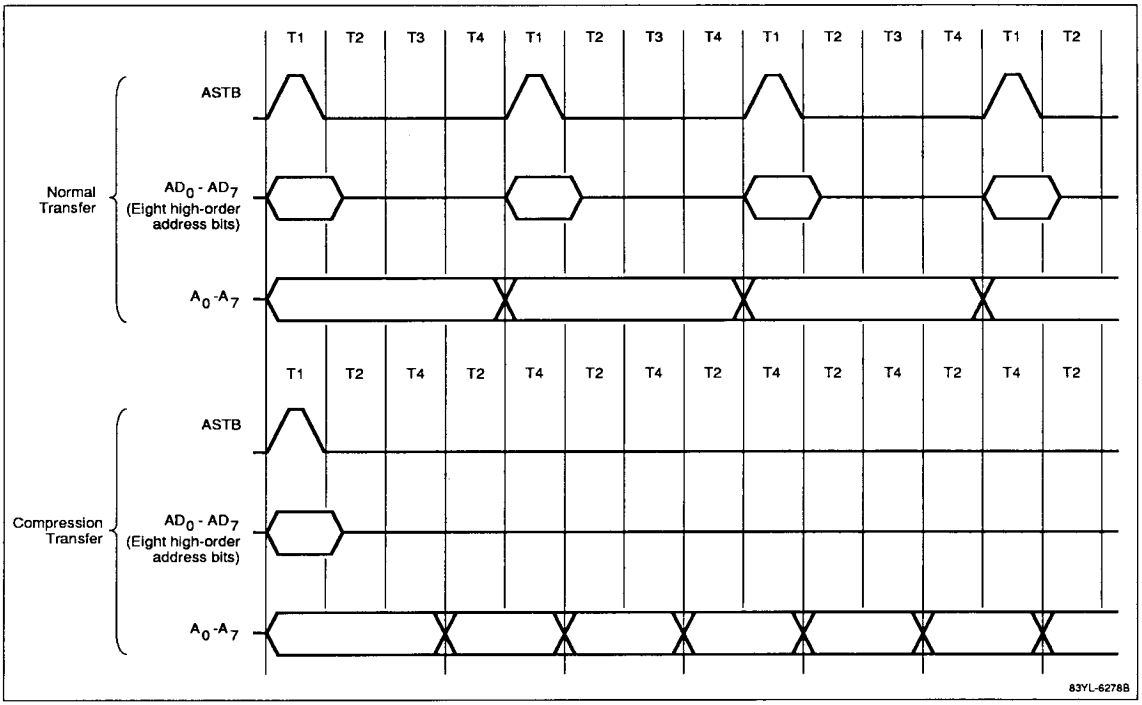
Compressed Timing

DMA transfer cycles are normally executed in four states for each bus cycle. However, when the device control register selects compressed timing, one DMA cycle can be executed in two states (T2 and T4) for each bus cycle. See figure 3.

Compressed timing may be used in block mode or demand mode. (Memory-to-memory transfer is excluded.) In this way, compressed timing permits twice the amount of data to be transferred when compared with normal DMA transfers.

In block mode or demand mode, addresses are output sequentially and the high-order address byte latched in external latches need not be updated except after a carry or borrow from the low-order byte. For this reason, the T1 state is omitted in the bus cycles except during the first bus cycle when the high-order address byte is changed.

Figure 3. Normal and Compression Transfer Timing



Software DMA Requests

The μPD71037 can accept software DMA requests in addition to DMA requests from the four DMARQ pins. Setting the appropriate bit in the request register generates a software request. The mask register does not mask software requests. Software DMA requests operate differently depending on which bus or transfer mode is used.

Single or Demand Mode. When single or demand mode is set, the applicable request bits for the corresponding channel are cleared, and software DMA service ends with the transfer of 1 byte of data (see Request Control Register).

Block Mode (Memory-to-Memory Transfer). When block mode or memory-to-memory mode is set, service continues until $\overline{END}$ is input or a terminal count is generated. In memory-to-memory transfer, only the request bit for channel 0 is cleared.

Self-Initialization

When the mode control register is set to self-initialization, the μPD71037 automatically initializes the address and count registers when $\overline{END}$ is input or a terminal count is generated. The contents of the address set register and the count set register are transferred to the effective address register and the effective count register, respectively. The applicable bit of the mask register is not affected. The applicable bit of the mask register is set for channels not programmed for self-initialization.

Channel Priority

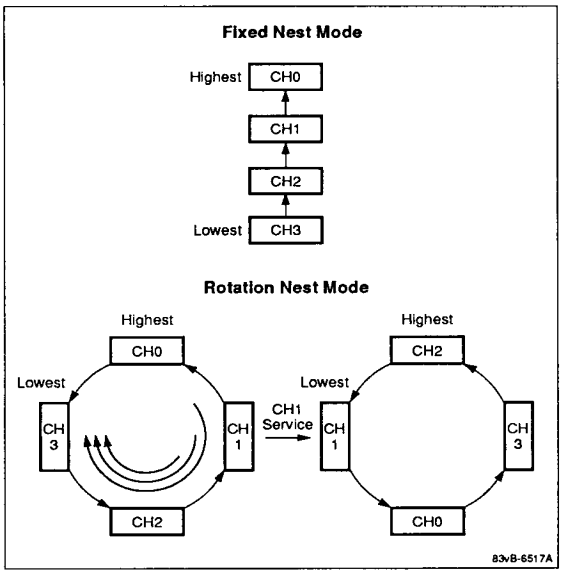
Each of the μPD71037's four DMA channels is assigned a priority. When there are DMA requests from several channels simultaneously, the channel with the highest priority will be serviced. The device control register selects one of two channel priority methods: fixed nest mode or rotation nest mode.

In fixed nest mode, the priority (starting with the highest) is channel 0, 1, 2, and 3, respectively. In rotation nest mode, priority is rotated so that the channel that has just

been given service receives the lowest priority and the next highest channel number is given the highest priority. This method prevents exclusive servicing of some channels.

Figure 4 shows the two priority order methods.

Figure 4. DMA Channel Priorities



Cascade Connection

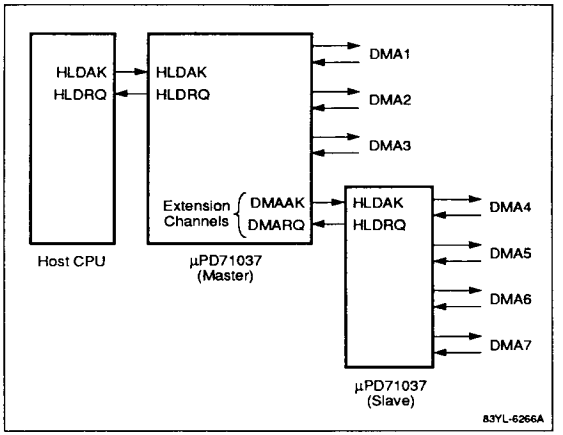
The μPD71037 can be cascaded to expand the system DMA channel capacity. To connect a μPD71037 for cascading (figure 5), perform the following operations.

(1) Connect pins HLDRQ and HLDAK of the second-stage (slave) μPD71037 to pins DMARQ and DMAAK of any channel of the first-stage (master) μPD71037.

(2) To select cascade mode of a particular channel of a master μPD71037, set bits 7 and 6 of that channel's mode control register to 11.

When a channel is set to cascade mode in a master μPD71037, DMARQ, DMAAK, HLDRQ, HLDAK, and RESET are the only valid signals in that master μPD71037. The other signals are disabled. The master cascade channel intermediates only hold request/hold acknowledge between slave and the host CPU.

Figure 5. Cascade Connection Example



DMA Transfer Timing

Figures 6 through 9 are timing diagrams for the DMA transfer operations described previously.

Figure 6. I/O-to-Memory Transfer, Normal Mode Timing

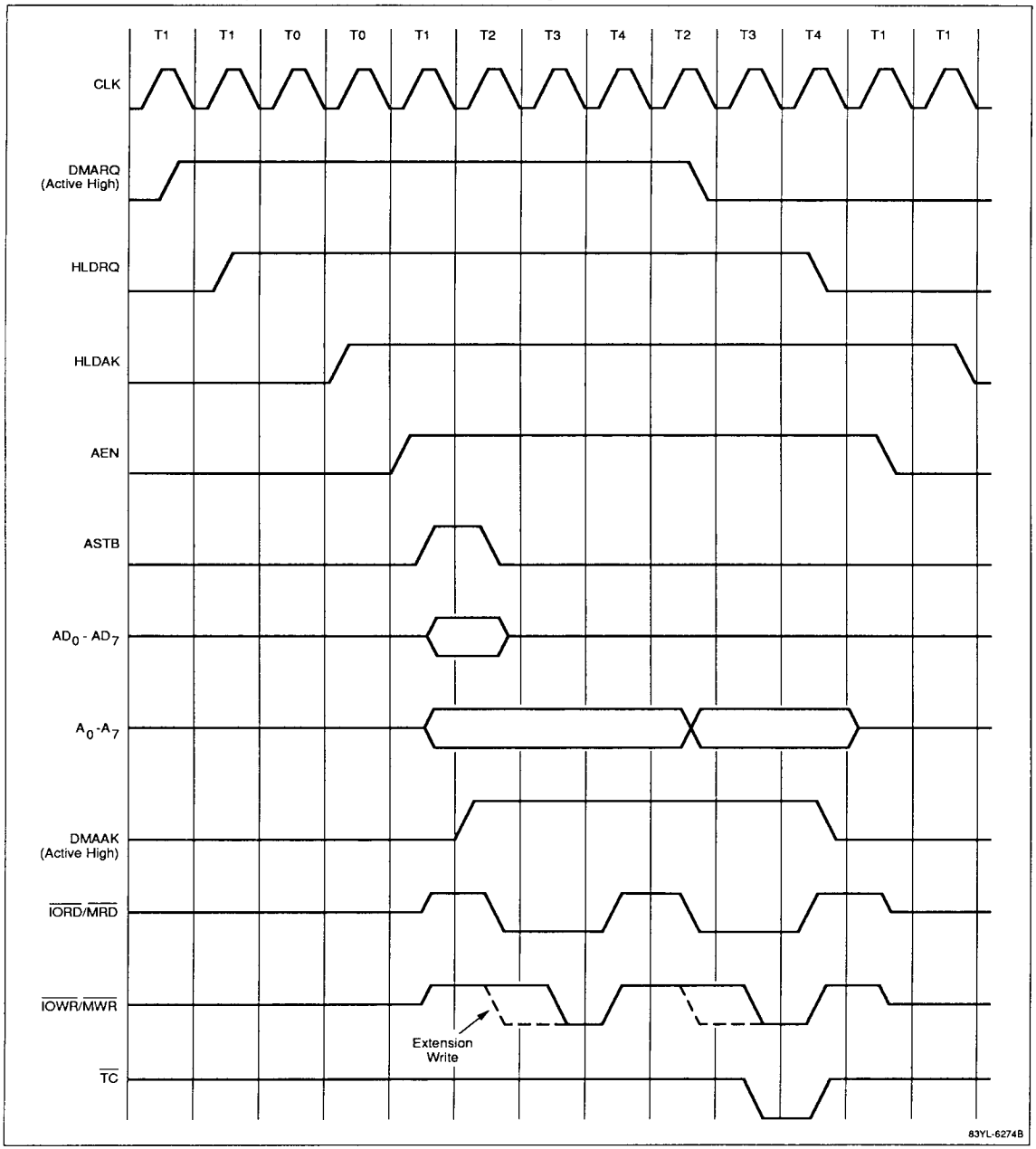


Figure 7. Memory-to-Memory Transfer Timing

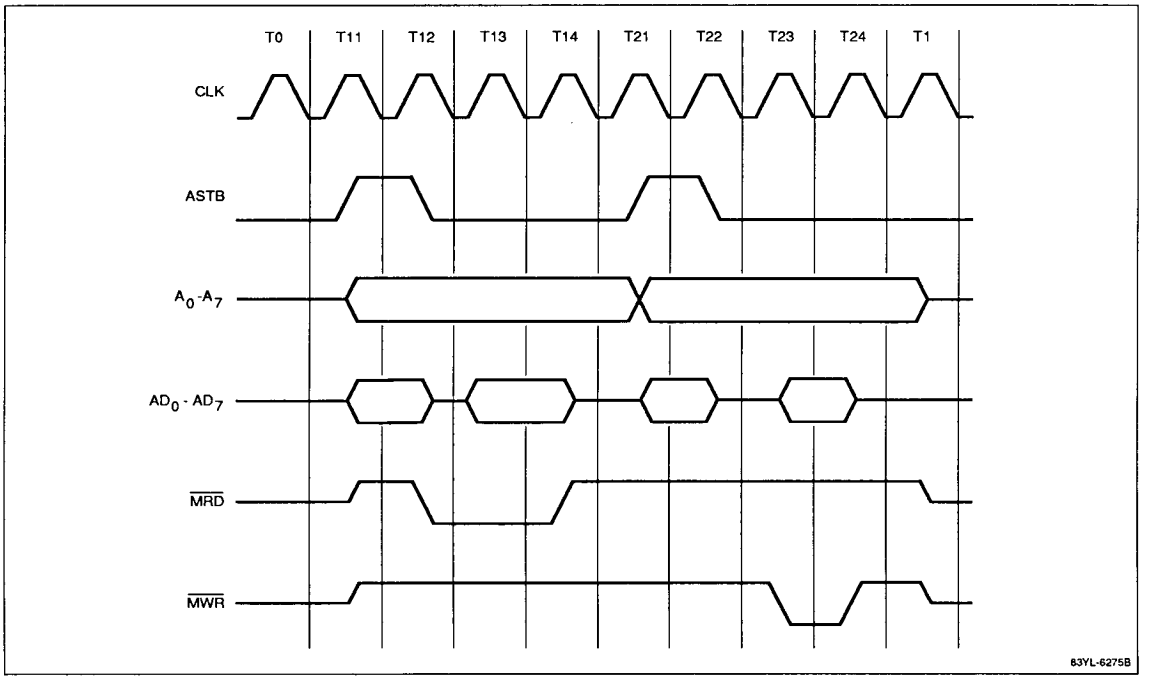


Figure 8. I/O-to-Memory Transfer, Compression Mode Timing

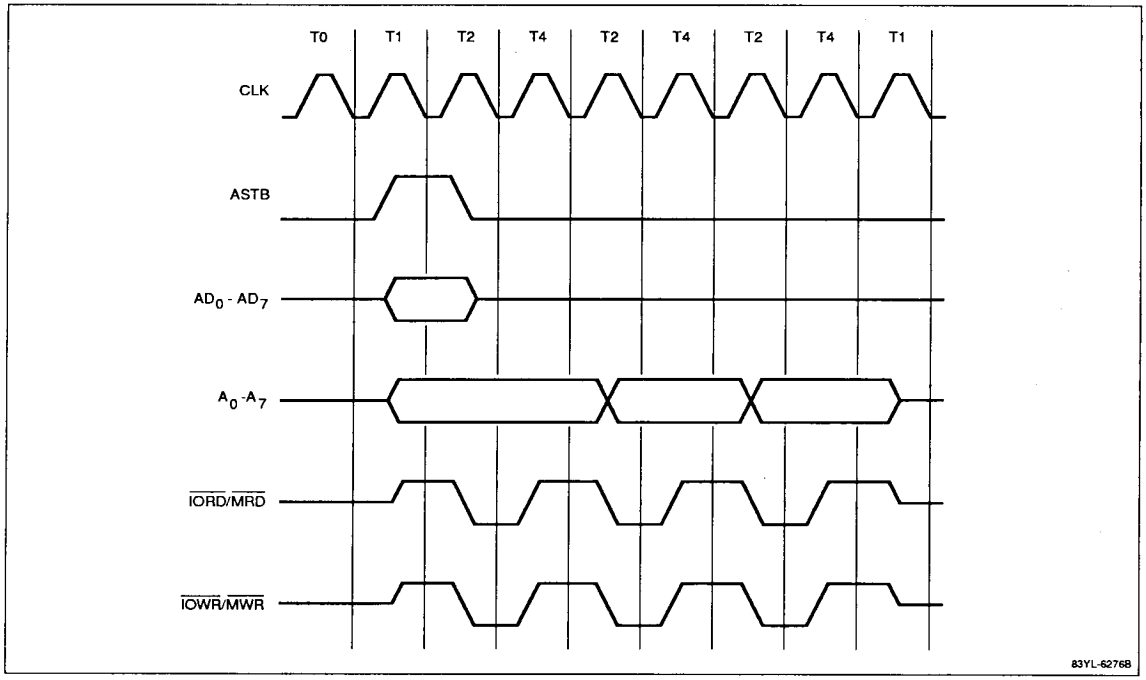
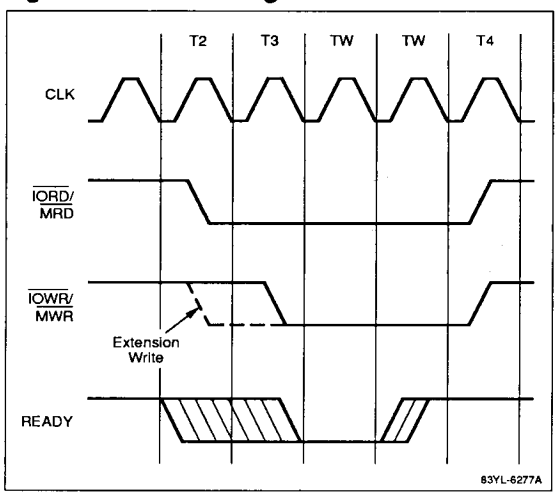


Figure 9. READY Timing



REGISTERS

The μPD71037 registers are considered to be address/count registers or control registers.

- Address/Count Registers
  - — Effective address register
  - — Set address register
  - — Effective count register
  - — Set count register
- Control Registers
  - — Device control register
  - — Mode control register
  - — Request control register
  - — Mask control register for each channel
  - — Mask control register for all channels
  - — Temporary data register
  - — Status read register

Each address/count register consists of 16 bits. A read/write operation for this type of register is performed by accessing 2 bytes of an I/O address. The low-order address byte is accessed first, followed by the high-order address byte. To set a new value in an address/count register, write its low-order byte first by issuing an address low-byte command. The commands you can

use are presented later in Commands. Table 3 lists the address/count registers for each channel. Each control register consists of 8 bits. Table 4 lists the type of output and addresses for the control registers.

Effective Address Register

A 16-bit effective address register is assigned to each channel. It holds the address to be output during DMA transfer. This register is updated by ±1 for each single-byte DMA transfer.

Set Address Register

A 16-bit set address register is assigned to each channel. It contains the initial value of a DMA transfer address set by the host CPU. Unlike the effective address register, this register retains its contents until the host CPU writes a new byte count into it. At self-initialization, the contents of the set address register are transferred to the effective address register at the initial address register. Values are written into the set address register by writing its 2 bytes in succession. However, the host CPU cannot read the address data from this register.

Effective Count Register

The effective count register is a 16-bit register assigned to each channel. It contains the remaining byte count for DMA transfer. This register is decremented by 1 for each single-byte DMA transfer. As in the effective address register, a value is read from or written into the effective count register by reading or writing its 2 bytes in succession.

Table 3. Address/Count Registers

Channel	Register	R/W	Address (Bits A_3-A_0)
0	Set address register	W	0H
	Effective address register	R/W	
	Set count register	W	1H
	Effective count register	R/W	
1	Set address register	W	2H
	Effective address register	R/W	
	Set count register	W	3H
	Effective count register	R/W	
2	Set address register	W	4H
	Effective address register	R/W	
	Set count register	W	5H
	Effective count register	R/W	
3	Set address register	W	6H
	Effective address register	R/W	
	Set count register	W	7H
	Effective count register	R/W	

Notes:

(1) When a new value is written into a set address/count register, it is simultaneously written into an effective address/count register. Therefore, when setting an address and count, you need not consider the differences between a set address/count register and an effective address/count register.

(2) The set address/count registers are used only for writing. If an attempt is made to read these registers, the effective address/count registers are read instead.

Table 4. Control Registers

Register	R/W	Address (Bits A_3-A_0)
Device control register	W	8H
Status read register	R	
Request control register	W	9H
Mask control register (each channel)	W	AH
Mode control register	W	BH
Temporary data register	R	DH
Mask control register (all channels)	W	FH

Notes:

(1) An I/O address other than listed in this table is assigned to address/count registers or commands; otherwise, the address cannot be accessed.

(2) The I/O address DH (in the temporary data register) is assigned to a software reset command when it is written. Refer to the Commands section for details.

Set Count Register

A 16-bit set count register is assigned to each channel. It holds the initial value of the DMA transfer byte count written by the host CPU. Unlike the effective count register, the set count register retains its data until the host CPU writes a new byte count into it. A value is written into the set count register by writing to its 2 bytes in succession. However, this register is read-protected.

Device Control Register

The 8-bit device control register controls the DMA transfer modes, determines whether to permit or inhibit DMA operation, controls the active levels of DMARQ and DMAAK, and determines whether to permit or inhibit memory-to-memory transfer.

Figure 10 shows the format of the device control register and table 5 describes the bits .

Figure 10. Device Control Register

AKL	RQL	EXW	ROT	CMP	DDMA	AHLD	MTM
7	6	5	4	3	2	1	0

Address 8H

AKL	**DMAAK Active Level**
0	Active low
1	Active high
RQL	**DMARQ Active Level**
0	Active high
1	Active low
EXW	**Extension Writing**
0	Normal writing
1	Extension writing
ROT	**Rotation Nesting**
0	Fixed nest mode
1	Rotation nest mode
CMP	**Compression Transfer**
0	Normal transfer mode
1	Compression transfer mode
DDMA	**DMA Operation**
0	Permit
1	Inhibit
AHLD	**Channel 0 Address Hold**
0	Inhibit address fixing
1	Permit address fixing
MTM	**Memory-to-Memory Transfer**
0	Inhibit
1	Permit

Table 5. Device Control Register Bits

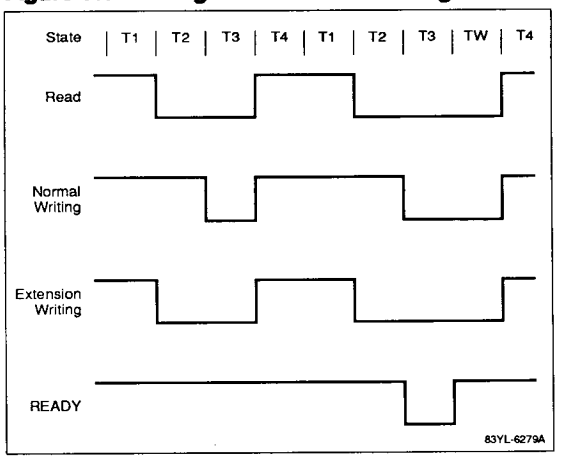
Bit	Symbol	Description
7	AKL	DMA acknowledge active level. This bit specifies the active levels of the four DMAAK output signals. Setting this bit to 1 indicates the active-high DMAAK signals.
6	RQL	DMA request level. This bit specifies the active levels of the four DMARQ input signals. Setting this bit to 0 indicates the active-high DMARQ signals.
5	EXW	Extended write. When this bit is set, the μPD71037 outputs a write signal at the same timing as a read signal (extension writing, figure 11). You cannot specify extension writing during compressed transfer.
4	ROT	Rotate priority. Setting this bit determines DMA channel priorities in rotation nest mode.

5b

Table 5. Device Control Register Bits (cont)

Bit	Symbol	Description
3	CMP	Compressed timing. When this bit is set, DMA transfer in the block or demand modes occurs at compression timing. Compression transfer mode must not be specified during memory-to-memory transfer.
2	DDMA	Disable DMA. While this bit is set, the μPD71037 does not output the HLDRQ signal to the host CPU even if it receives an effective DMA request. It also prevents invalid DMA transfer during μPD71037 programming.
1	AHLD	Channel 0 address hold. If memory-to-memory transfer is permitted, setting this bit fixes the address of the transfer source channel 0. If memory-to-memory transfer is inhibited, this bit is meaningless.
0	MTM	Memory to memory. Setting this bit permits memory-to-memory transfer.

Figure 11. Timing for Extension Writing



Mode Control Register

The mode control register specifies DMA transfer mode for each channel. Figure 12 shows the format and table 6 describes the bits.

Figure 12. Mode Control Register

TMODE		ADIR	SEFI	TDIR		SELCH	
7	6	5	4	3	2	1	0

Address 8H

TMODE	**DMA Transfer Mode**
0 0	Demand mode
0 1	Single mode
1 0	Block mode
1 1	Extension mode
ADIR	**Address Direction**
0	Increment an address
1	Decrement an address
SEFI	**Self-Initialization**
0	Inhibit
1	Permit
TDIR	**Transfer Direction**
0 0	Verify transfer
0 1	I/O-to-memory transfer
1 0	Memory-to-I/O transfer
1 1	Inhibit transfer
SELCH	**Channel Selection**
0 0	Channel 0
0 1	Channel 1
1 0	Channel 2
1 1	Channel 3

Table 6. Mode Control Register Bits

Bits	Symbol	Description
7, 6	TMODE	Transfer mode. These bits indicate the DMA transfer mode for I/O-to-memory transfer (meaningless during memory-to-memory transfer because block mode is automatically selected).
5	ADIR	Address direction. This bit indicates whether the effective address register is incremented or decremented. If set to 0, the register is incremented by 1 for each single-byte transfer. If set to 1, it is decremented by 1 for each single-byte transfer.
4	SEFI	Self-initialization. In memory-to-memory transfer, assign the same value to the SEFI bits of channel 0 (transfer source) and channel 1 (transfer destination).
3, 2	TDIR	Transfer direction. These bits specify the I/O-to-memory transfer direction (meaningless during memory-to-memory transfer).
1, 0	SELCH	Select channel. These bits specify the DMA channel to which DMA transfer modes are specified by bits 7-2 .

Status Read Register

The status read register indicates whether a DMA request or terminal count is generated and whether the $\overline{END}$ signal is externally input. This information is set for each channel. Figure 13 shows the register format and table 7 describes the bits.

Figure 13. Status Read Register

RQ3	RQ2	RQ1	RQ0	TC3	TC2	TC1	TC0
7	6	5	4	3	2	1	0

Address 8H

RQ3-RQ0	Hardware DMA Request (Channels 3-0)
0	Request not generated
1	Request generated

TC3-TC0	Terminal Count (Channels 3-0)
0	Transfer not yet terminated
1	$\overline{END}$ input or terminal count occurs

Table 7. Status Read Register Bits

Bits	Symbol	Description
7-4	RQ3-RQ0	DMA request. These bits specify whether a DMA hardware request is generated by inputting the DMARQ signal. If the corresponding channel is masked, these bits are reset as long as the DMARQ input is active for this channel. A hardware DMA request held by a mask can be detected by sampling these bits.
3-0	$\overline{TC3}$-$\overline{TC0}$	Terminal count. These bits specify whether the $\overline{END}$ signal is input or a terminal count is generated. This information is set for each channel. If a terminal count occurs within a channel or an $\overline{END}$ signal is externally input, the corresponding channel bit is set. These bits are reset each time the status read register is read.

Temporary Data Register

The temporary data register contains the data transferred last during memory-to-memory transfer. Figure 14 shows the register format.

Figure 14. Temporary Data Register

TD7	TD6	TD5	TD4	TD3	TD2	TD1	TD0
7	6	5	4	3	2	1	0

Address DH

Request Control Register

The request control register controls software DMA requests. Figure 15 shows the register format and table 8 describes the bits.

Figure 15. Request Control Register

—	—	—	—	—	SRQ	SELCH	
7	6	5	4	3	2	1	0

Address 9H

SRQ	Software DMA Request
0	Reset DMA request bit
1	Set DMA request bit

SELCH	Select Channel
0 0	Channel 0
0 1	Channel 1
1 0	Channel 2
1 1	Channel 3

Table 8. Request Control Register Bits

Bits	Symbol	Description
2	SRQ	Software DMA request. This bit controls a software DMA request for the channel selected by bits 0 and 1. Setting this bit to 1 sets the request bit of the selected channel. Setting it to 0 resets the request bit.
1, 0	SELCH	Select channel. These bits specify the channel for which software DMA request control is executed.

Mask Control Registers

The mask control register controls the DMA request mask for each channel. Two types of mask control registers are provided. The first is a register that masks each of the channels separately (figure 16). The second is a register that masks all of the channels at once (figure 17). Table 9 describes the mask control register bits.

Figure 16. Mask Control Register (Each Channel)

—	—	—	—	—	MSET	SELCH	
7	6	5	4	3	2	1	0

Address AH

MSET	DMARQ Mask
0	Reset mask
1	Set mask

SELCH	Select Channel
0 0	Channel 0
0 1	Channel 1
1 0	Channel 2
1 1	Channel 3

5b

Figure 17. Mask Control Register (All Channels)

—	—	—	—	M3	M2	M1	M0
7	6	5	4	3	2	1	0

Address FH

M3-M0	DMARQ Mask (Channels 3-0)
0	Reset mask
1	Set mask

Table 9. Mask Control Register Bits

Bits	Symbol	Description
Each Channel		
2	MSET	Mask set. This bit masks the DMA request (DMARQ pin input) for the channel selected in bits 1 and 0.
1, 0	SELCH	Select channel. These bits select the channel to mask.
All Channels		
3-0	M3-M0	Mask. These bits specify whether a mask is set or reset for the four DMA channels. Setting any of the bits to 1 masks the respective channel and the DMA request (DMARQ pin input) for this channel is inhibited. Setting any of these bits to 0 resets the mask.

COMMANDS

The μPD71037 supports three commands (in addition to the registers) to control data transfer operations.

- Address low-byte command
- Software reset command
- Clear-all-masks command

The host CPU writes data into these commands (table 10) to enable them to control μPD71037 operation. Unlike the registers, any data can be written into the commands.

Table 10. List of Commands

Command	R/W	Address (Bits A_3-A_0)	Function
Address low-byte	W	CH	This command must be issued before a new value is set in an address/count register. The low-order byte of this new value is set first and then its high-order byte is set.
Software reset	W	DH	The reset operation performed by this command is the same as the normal hardware reset operation. The reset operation clears the device control register, status read register, request control register, and temporary data register to 00H. Masks are set for all channels.
Clear-all-masks	W	EH	The masks for all channels are released and the reception of a DMA transfer request is permitted.

Notes:

(1) After the reset operation by hardware or software, a value is written in the low-order byte of the address/count register.

(2) If an attempt is made to read I/O address DH, the temporary data register is read instead. See table 4.

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

T_A = 25°C

Supply voltage, V_{DD}	−0.5 to 7.0 V
Input voltage, V_I	−0.5 to V_{DD} + 0.3 V
Output voltage, V_O	−0.5 to V_{DD} + 0.3 V
Operating temperature range, T_{OPT}	−40 to +85°C
Storage temperature range, T_{STG}	−65 to +150°C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage.

Capacitance

T_A = 25°C

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input capacitance	C_I		8	15	pF	fc = 1MHz; unmeasured pins returned to 0 V
Output capacitance	C_O		4	8	pF	
I/O capacitance	C_{IO}		10	18	pF	

DC Characteristics

T_A = −40 to +85°C; V_{DD} = +5.0 V ± 10%

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input voltage, low	V_{IL1}	−0.5		0.6	V	CLK pin
	V_{IL2}	−0.5		0.8	V	All except CLK
Input voltage, high	V_{IH1}	3.3		V_{DD} + 0.3	V	CLK pin
	V_{IH2}	2.2		V_{DD} + 0.3	V	All except CLK
Output voltage, low	V_{OL}			0.4	V	I_{OL} = 2.5 mA; I_{OL} = 4.5 mA (TC only)
Output voltage, high	V_{OH}	0.7 V_{DD}			V	I_{OH} = −400 μA
Input leakage current	I_{LI}			±10	μA	V_I = 0 V to V_{DD}
Output leakage current	I_{LO}			±10	μA	V_O = 0 V to V_{DD}
Power supply current	I_{DD}			20	mA	10-MHz operation

5b

AC Characteristics

T_A = −10 to +70°C; V_{DD} = +5.0 V ± 10%

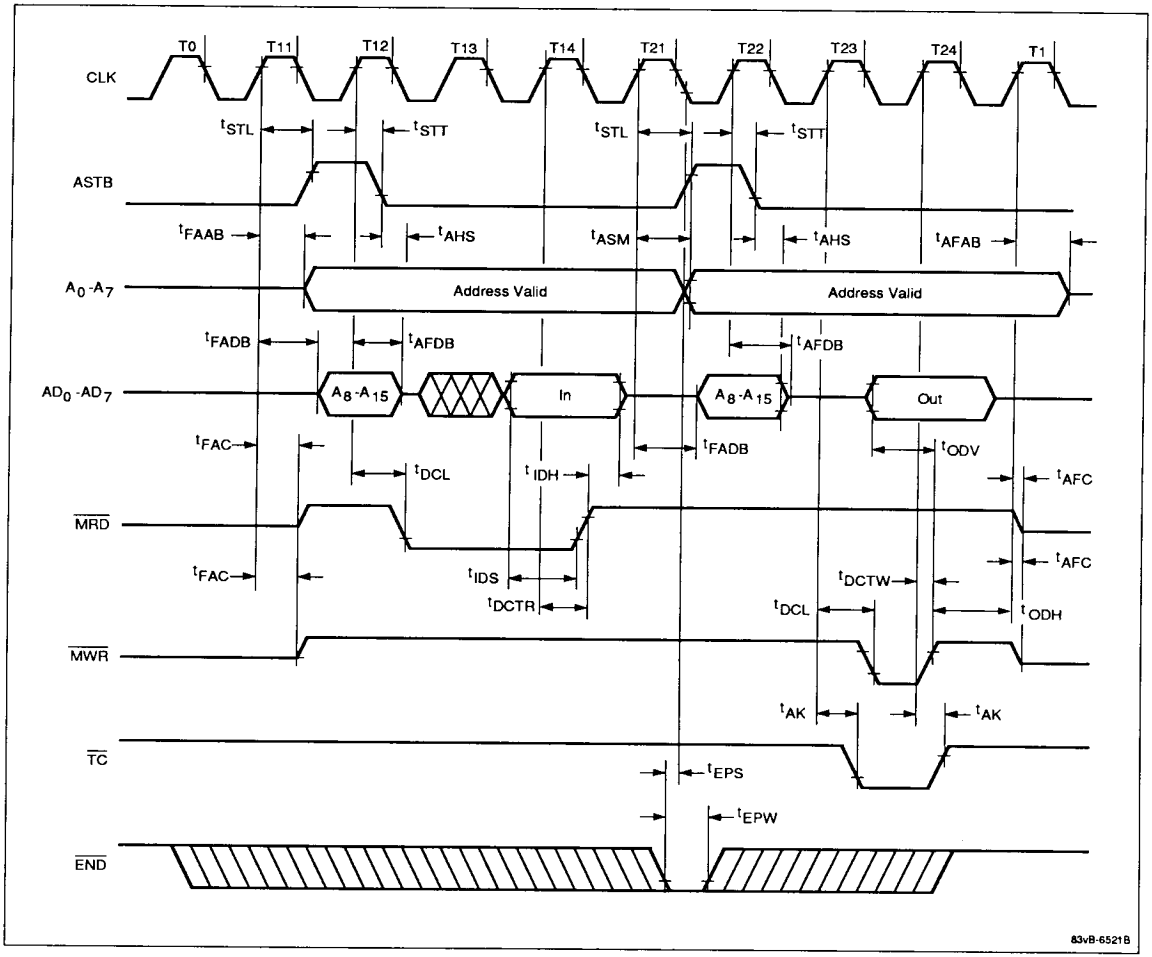
Parameter	Symbol	Min	Max	Unit	Conditions
DMA Master Mode					
AEN high delay time from CLK low	t_{AEL}		100	ns	S1
AEN low delay time from CLK high	t_{AET}		70	ns	S1
Address active to float delay from CLK high	t_{AFAB}		80	ns	
$\overline{IORD}/\overline{MRD}$ or $\overline{IOWR}/\overline{MWR}$ float from CLK high	t_{AFC}		80	ns	
DB active to float delay from CLK high	t_{AFDB}		120	ns	
Address hold time from $\overline{IORD}/\overline{MRD}$ high	t_{AHR}	t_{CY} − 100		ns	
DB hold time from ASTB low	t_{AHS}	20		ns	
Address hold time from $\overline{IOWR}/\overline{MWR}$ high	t_{AHW}	t_{CY} − 40		ns	
DMAAK valid delay time from CLK low	t_{AK}		100	ns	
$\overline{TC}$ high delay time from CLK high	t_{AK}		100	ns	
$\overline{TC}$ low delay time to CLK high	t_{AK}		90	ns	
Address stable from CLK high	t_{ASM}		80	ns	
Data bus setup time to ASTB low	t_{ASS}	40		ns	
Clock pulse width, high	t_{CH}	39		ns	Transitions ≤ 10 ns

AC Characteristics (cont)

Parameter	Symbol	Min	Max	Unit	Conditions
DMA Master Mode (cont)					
Clock pulse width, low	t_{CL}	45		ns	Transitions ≤ 10 ns
CLK cycle	t_{CY}	100	1000	ns	
CLK high delay to $\overline{IORD}/\overline{MRD}$ or $\overline{IOWR}/\overline{MWR}$ low	t_{DCL}	10	80	ns	
$\overline{IORD}/\overline{MRD}$ high delay time from CLK high	t_{DCTR}	10	80	ns	S4
$\overline{IOWR}/\overline{MWR}$ high delay time from CLK high	t_{DCTW}	10	55	ns	S4
HLDRQ valid delay time from CLK high	t_{DQ1}/ t_{DQ2}		70	ns	
$\overline{END}$ low setup time from CLK low	t_{EPS}	25		ns	
$\overline{END}$ pulse width	t_{EPW}	100		ns	
Address float to active delay from CLK high	t_{FAAB}		80	ns	
$\overline{IORD}/\overline{MRD}$ or $\overline{IOWR}/\overline{MWR}$ active from CLK high	t_{FAC}		80	ns	
Data bus float to active delay from CLK high	t_{FADB}		70	ns	
HLDAK valid setup time to CLK high	t_{HS}	50		ns	
Input data hold time from $\overline{MRD}$ high	t_{IDH}	20		ns	
Input data setup time to $\overline{MRD}$ high	t_{IDS}	90		ns	
Output data hold time from $\overline{MWR}$ high	t_{ODH}	10		ns	
Output data valid to $\overline{MWR}$ high	t_{ODV}	65		ns	
DMARQ setup time to CLK low	t_{QS}	20		ns	S1, S4
CLK hold time to READY low	t_{RH}	20		ns	
READY setup time to CLK low	t_{RS}	25		ns	
ASTB high delay time from CLK high	t_{STL}		70	ns	
ASTB low delay time from CLK high	t_{STT}		70	ns	
Peripheral (Slave) Mode					
Address valid or $\overline{CS}$ low to $\overline{IORD}/\overline{MRD}$ low	t_{AR}	35		ns	
Address valid setup time to $\overline{IOWR}/\overline{MWR}$ high	t_{AW}	80		ns	
$\overline{CS}$ low setup time to $\overline{IOWR}/\overline{MWR}$ high	t_{CW}	90		ns	
Data valid setup time to $\overline{IOWR}/\overline{MWR}$ high	t_{DW}	80		ns	
Address or $\overline{CS}$ hold from $\overline{IORD}/\overline{MRD}$ high	t_{RA}	0		ns	
Data access from $\overline{IORD}/\overline{MRD}$ low	t_{RDE}		120	ns	
Data bus float delay from $\overline{IORD}/\overline{MRD}$ high	t_{RDF}	0	70	ns	
Power supply setup time high to RESET low	t_{RSTD}	500		ns	
RESET to first $\overline{IORD}$ or $\overline{IOWR}$	t_{RSTS}	t_{CY}		ns	
RESET pulse width	t_{RSTW}	200		ns	
$\overline{IORD}/\overline{MRD}$ width	t_{RW}	150		ns	
Address hold time from $\overline{IOWR}/\overline{MWR}$ high	t_{WA}	15		ns	
$\overline{CS}$ hold time high from $\overline{IOWR}/\overline{MWR}$ high	t_{WC}	15		ns	
Data hold time from $\overline{IOWR}/\overline{MWR}$ high	t_{WD}	20		ns	
$\overline{IOWR}/\overline{MWR}$ width	t_{WWS}	90		ns	
Access recovery time		125		ns	

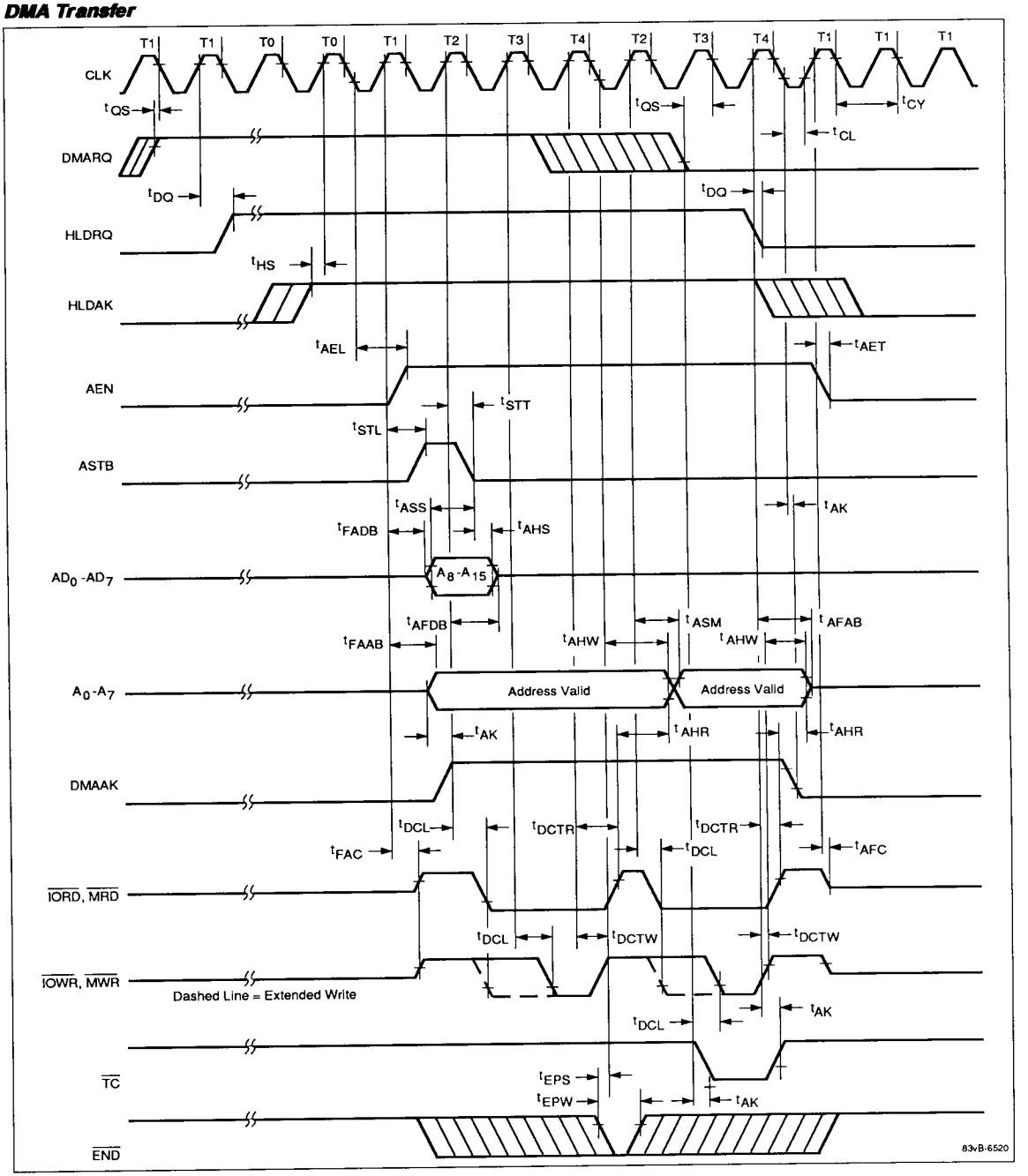
Timing Waveforms

Memory-to-Memory Transfer

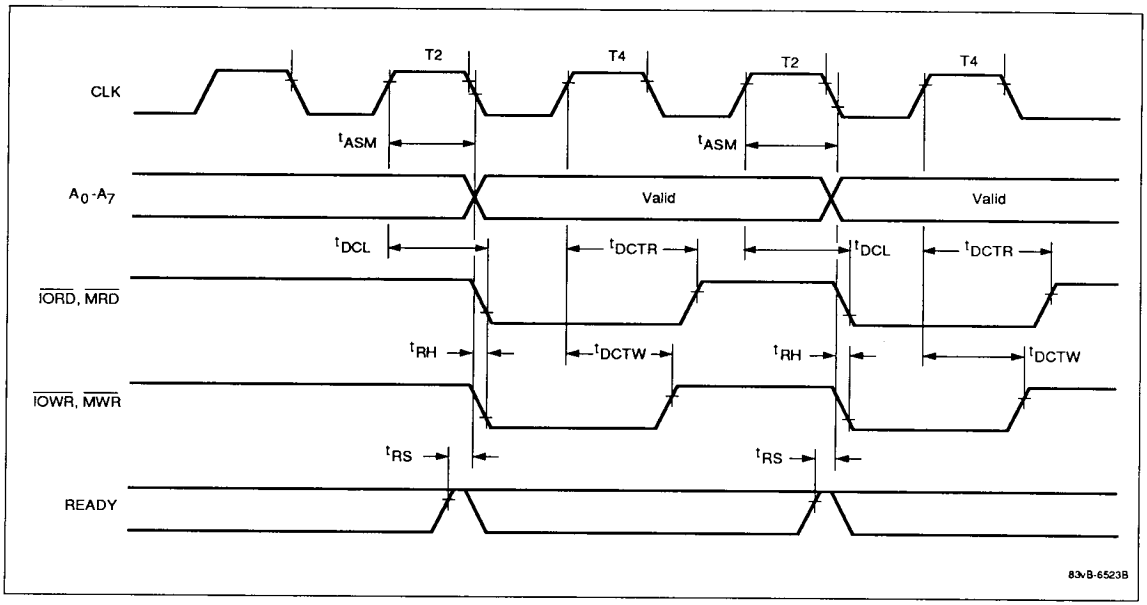


5b

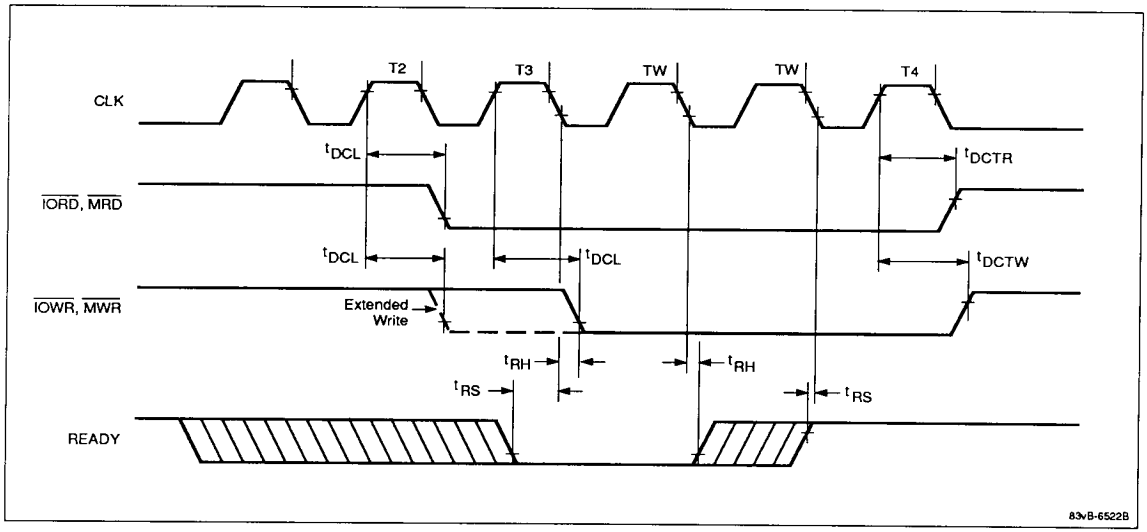
DMA Transfer



Compressed Transfer

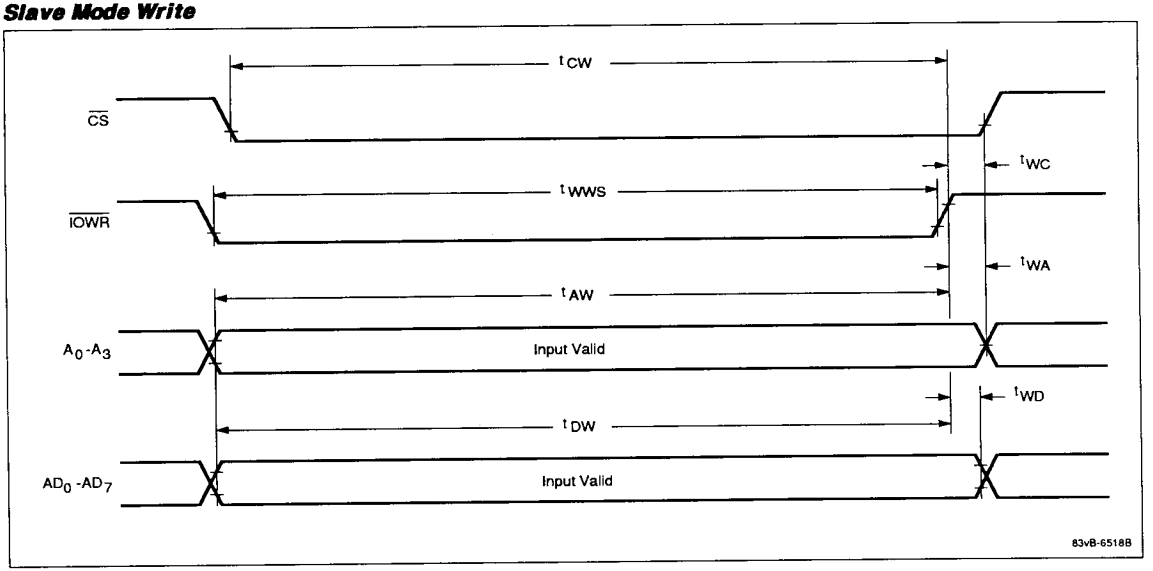


Ready

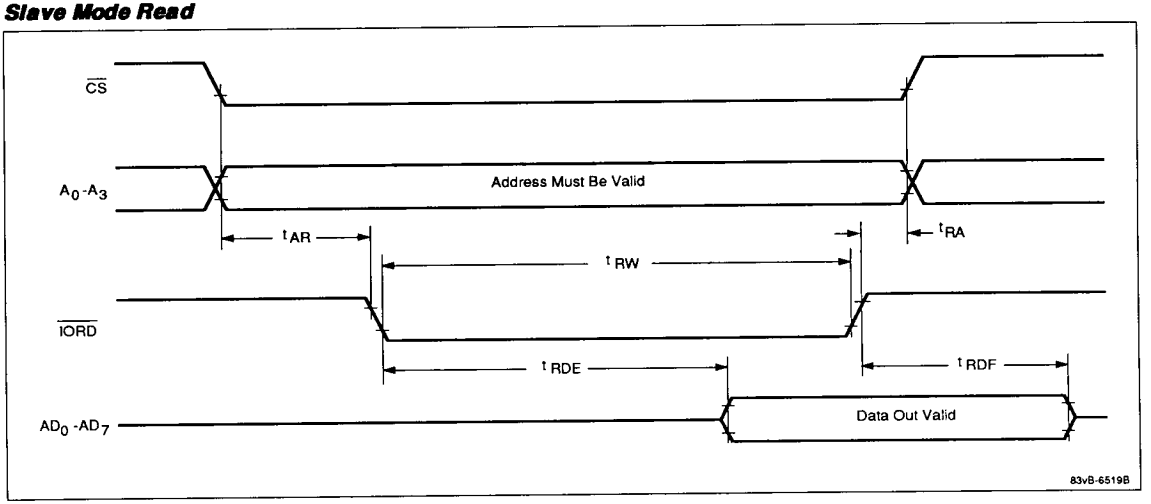


5b

Slave Mode Write



Slave Mode Read



Reset