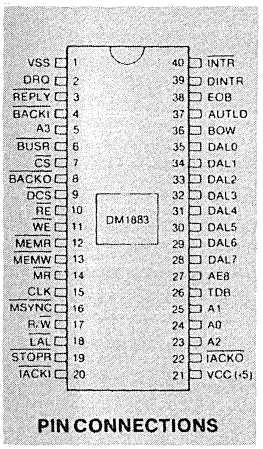
DM1883A/B Direct Memory Access Controller

FEATURES

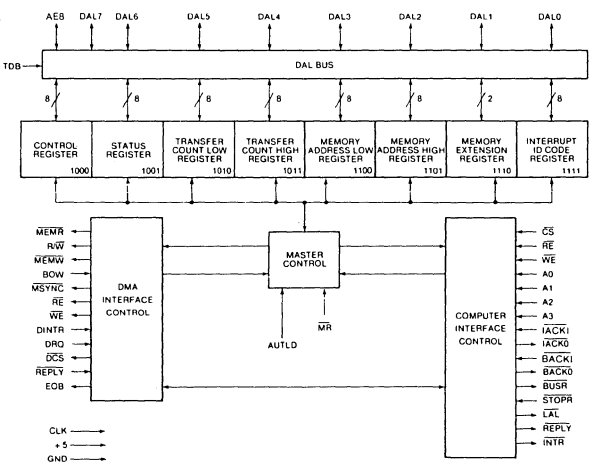
- AUTOMATIC DAISY CHAINING OF BUS AND INTERRUPT ACKNOWLEDGE SIGNALS
- AUTO LOAD OPTION
- SINGLE +5 VDC POWER SUPPLY
- 8 BIT BI-DIRECTIONAL DATA BUS
- TRUE OR COMPLEMENT DATA BUS
- 8 CPU ADDRESSABLE DMAC REGISTERS
- 8 CPU ADDRESSABLE DEVICE REGISTERS
- AUTOMATIC GENERATION OF DEVICE CS DURING DMA AND CPU DEVICE ACCESSES
- 256K MEMORY ADDRESSING
- 64K PROGRAMMABLE PAGE PROTECTION
- BYTE OR WORD DMA TRANSFERS
- INTERRUPT AND BUS REQUEST CAPABILITIES
- END-OF-BLOCK SHUT OFF BY DMAC
- TIME-OUT INTERRUPT CAPABILITY
- SINGLE CLOCK INPUT
- CS, RE, WE, A0-A3 ADDRESSING
- STOP REQUEST INPUT TO DELAY INTERRUPT OR BUS REQUESTS
- COMPATIBLE WITH OUR FLOPPY DISC CONTROLLERS
- 8 BIT PROGRAMMABLE INTERRUPT ID CODE

GENERAL DESCRIPTION

The DM1883 Direct Memory Access Controller (DMAC) is packaged in a 40 pin standard dual in-line package. The chip requires a single +5 power supply input and a single clock input. The device contains 8 CPU addressable registers, and allows for up to 8 CPU addressable device registers if the automatic device chip select feature is used. Byte or word transfers can be programmed, and all memory DMA operations are handshaked for compatibility with a variety of bus structures. Up to 256K bytes of memory can be accessed directly with 64K page protection and nonexistent memory interrupt as options. Bus and Interrupt Acknowledge signals are internally daisy chained, and a STOP REQUEST input prevents new requests while a current request is active. Device accesses are not handshaked, and a BUS HOLD feature is present for high speed devices. Device interrupt input, end-of-block output, and I/O read/write output pins simplify hardware interfacing to the device and the CPU bus. The AUTO LOAD feature allows automatic boot-loading of up to 64K bytes or words into memory starting at location zero. An 8 bit interrupt ID code is also provided.



PIN CONNECTIONS



DM1883 BLOCK DIAGRAM

INTERFACE SIGNALS DESCRIPTIONS

PIN NUMBER	SIGNAL NAME	SYMBOL	FUNCTION
1	GROUND	VSS	Ground
2	DATA REQUEST	DRQ	Data service request input from the peripheral device. A DMA transfer is initiated when this signal goes high.
3	$\overline{\text{REPLY}}$	$\overline{\text{REPLY}}$	Active low bi-directional handshake signal for both CPU and DMA transfers.
4	$\overline{\text{BACK IN}}$	$\overline{\text{BACKI}}$	Bus acknowledge in. An active low input signal from the CPU or a previous device in the $\overline{\text{BACK}}$ daisy chain. When low this signal will initiate a DMA transfer if the DMAC was requesting a DMA cycle.
5, 23, 24, 25	REGISTER SELECTS	A0-A3	These inputs select one of eight DMAC registers or one of eight device registers. When A3 is high the DMAC is selected. When A3 is low the DMAC is deselected and $\overline{\text{DCS}}$ is made low by the DMAC to activate device transfers. $\overline{\text{CS}}$ input to the DMAC must be made low before either the DMAC or the device may be selected by the CPU.
6	$\overline{\text{BUS REQUEST}}$	$\overline{\text{BUSR}}$	Active low output signal to initiate a CPU bus request and to latch A8-A15, A17 of the 18 bit DMA transfer address from DAL0-DAL7, AE8 into an external register.
7	$\overline{\text{CHIP SELECT}}$	$\overline{\text{CS}}$	Active low chip select input signal for CPU controlled operations.
8	$\overline{\text{BACK OUT}}$	$\overline{\text{BACKO}}$	Bus acknowledge out. An active low output signal used to pass $\overline{\text{BACKI}}$ along the daisy chain when the DMAC is not requesting a DMA cycle. This output is not affected by $\overline{\text{STOPR}}$.
9	$\overline{\text{DEVICE SELECT}}$	$\overline{\text{DCS}}$	Active low device chip select output signal for CPU and DMAC controlled operations.
10	$\overline{\text{READ ENABLE}}$	$\overline{\text{RE}}$	Active low bi-directional read enable for the DMAC and the device.
11	$\overline{\text{WRITE ENABLE}}$	$\overline{\text{WE}}$	Active low bi-directional write enable for the DMAC and the device. $\overline{\text{RE}}$ and $\overline{\text{WE}}$ are inputs during CPU controlled operations, and outputs to the device during DMAC controlled operations.
12	$\overline{\text{MEMORY READ}}$	$\overline{\text{MEMR}}$	Active low output to initiate a memory read during DMA transfers to the peripheral device.
13	$\overline{\text{MEMORY WRITE}}$	$\overline{\text{MEMW}}$	Active low output to initiate a memory write during DMA transfers from the peripheral device.
14	$\overline{\text{MASTER RESET}}$	$\overline{\text{MR}}$	Active low master reset signal to initialize the DMAC.
15	CLOCK	CLK	Clock input
16	$\overline{\text{MEMORY SYNC}}$	$\overline{\text{MSYNC}}$	Active low memory sync output to initiate a memory access during DMA transfers.
17	READ/$\overline{\text{WRITE}}$	R/$\overline{\text{W}}$	This output indicates the direction of transfer for the peripheral device. High for device-to-memory transfers (READ), and low for memory to device transfers (WRITE). Tied directly to Control Register bit 4.
18	$\overline{\text{LOAD ADDRESS LOW}}$	$\overline{\text{LAL}}$	Active low output signal to latch A0-A7, A16 of the 18-bit DMA transfer address from DAL0-DAL7, AE8 into an external register. $\overline{\text{BUSR}}$ and $\overline{\text{LAL}}$ are compatible with INTEL 8212 devices.

PIN NUMBER	SIGNAL NAME	SYMBOL	FUNCTION
19	STOP REQUEST	STOPR	Active low input that prevents INTR and BUSR from going low even if a request becomes active. An active INTR or BUSR request will not be affected by this input going low. This signal is used to speed up daisy chaining of bus and interrupt acknowledge inputs, and to prevent new requests while some other request is in the process of being serviced.
20	IACK IN	IACKI	Interrupt acknowledge in. An active low input signal from the CPU or a previous device in the IACK daisy chain. The DMAC is selected when INTR is low and this signal goes low. If RE also goes low while the DMAC is selected via this signal then the interrupt ID code is gated onto DAL0-DAL7.
21	POWER SUPPLY	V_{CC}	+5 VDC power supply input
22	IACK OUT	IACKO	Interrupt acknowledge out. An active low output signal used to pass IACKI along the daisy chain when the DMAC is not requesting an interrupt. This output is not affected by STOPR.
26	TRUE DATA BUS	TDB	This input selects a true data bus on the DAL lines when high or open, and a complemented data bus on the DAL lines when low.
27	ADDRESS EXTENSION	AE8	Address extension bit output. Used during DMA operations to extend the address to 18 bits. This bit is true if TDB is high and complemented if TDB is low.
28-35	DATA ACCESS LINES	DAL0-DAL7	An 8-bit bi-directional three-state bus for CPU and DMAC controlled transfers to and from the DMAC. These signals remain in a three-state mode if the peripheral device is selected via A3 instead of the DMAC.
36	BYTE OR WORD	BOW	Byte or word DMA transfer mode input. When high memory addresses are incremented by one after every DMA transfer. When low memory addresses are incremented by two after every DMA transfer and the LSB of the memory address is forced to zero.
37	AUTO LOAD	AUTLD	Active high input to initiate a non-programmed 64K device to memory data transfer.
38	END OF BLOCK	EOB	Active high output to shut off the peripheral device when the transfer count goes to zero.
39	DEVICE INTERRUPT	DINTR	Interrupt service request input from the peripheral device. An interrupt request is generated by the DMAC if this input is high and the device interrupt enable bit in the command register is also set.
40	INTERRUPT REQUEST	INTR	Active low interrupt service request output. This output goes low if: 1) Any one of the three interrupt conditions is active, and 2) The STOPR input is high, and 3) The corresponding interrupt enable bit for the interrupting condition is set.

NOTE: The following pins float when not active low and require an external pull-up resistor of 10 KΩ (or greater) to +5 VDC:

INTR, REPLY, RE, WE, MEMR, MEMW, MSYNC

The following pins have internal 10 KΩ pull-up resistors to +5 VDC:

TBD, DRQ, DINTR

WIRE-ORABLE SIGNALS

The following output signals can be wired together with a single common pull-up resistor if multiple DMAC chips exist on the same board:

$\overline{MSYNC}$, $\overline{MEMR}$, $\overline{MEMW}$, $\overline{INTR}$

REGISTER SELECTION

A 4-bit address input (A0, A1, A2, A3) is used to select one of 8 internal DMAC registers or to generate a device chip select ($\overline{DCS}$) output signal for selection of up to 8 peripheral device registers. The following table details the selection process.

INPUTS					OUTPUT	SELECTED REGISTER
$\overline{CS}$	A3	A2	A1	A0	$\overline{DCS}$	
L	L	X	X	X	L	One of 8 peripheral device registers
L	H	L	L	L	H	DMAC control register (0)
L	H	L	L	H	H	DMAC status register (1)
L	H	L	H	L	H	DMAC TC low register (2)
L	H	L	H	H	H	DMAC TC high register (3)
L	H	H	L	L	H	DMAC MA low register (4)
L	H	H	L	H	H	DMAC MA high register (5)
L	H	H	H	L	H	DMAC MA ext. register (6)
L	H	H	H	H	H	DMAC ID code register (7)

NOTE: L = Low voltage level, H = High voltage level, X = don't care.

TRANSFER COUNT REGISTER (TCR)

A 16-bit counter register that holds the two's complement of the transfer count (words or bytes) for DMA transfer operations. The low order 8 bits are in TC low, and the high order 8 bits are in TC high. The count is incremented by one after every DMA transfer. When the count reaches zero bit 3 of the Status Register is set to a one. If bit 3 in the Command Register is also a one then $\overline{INTR}$ will go low (providing $\overline{STOPR}$ is also high). TCR is set to a one on a MASTER RESET to allow a 64K transfer count during auto load.

MEMORY ADDRESS REGISTER (MAR)

An 18-bit counter register that occupies 3 DMA registers. Bits 0-7 are in MA low, bits 8-15 are in MA high, and bits 16-17 are in MA ext. The carry from bit 15 to 16 is enabled if and only if bit 6 of the Command Register is set to a one. If the BOW input pin is high then the MAR is incremented by one after every DMA transfer. If the BOW input pin is low then the MAR is incremented by two after every transfer and bit 0 is forced to a zero. This register is cleared to all zeros on a MASTER RESET.

During a DMA operation the DMA address is gated onto the DAL lines in two 9-bit bytes. The first byte out contains MAR 8-15 on DAL 0-7 and MAR 17 or AE8. The second byte out contains MAR 0-7 on DAL 0-7 and MAR 16 on AE8. The first byte is valid on the trailing edge of $\overline{BUSR}$, and the second byte is valid on the trailing edge of $\overline{LAL}$. Note that the address can easily be extended to 24 bits by decoding the address of the 2-bit extension register externally and gating the 6 unused bits into an external latch. This would give the system 16 Mbytes of addressing with either 65K or 256K bytes of paging.

REGISTER DEFINITIONS

DMAC CONTROL REGISTER (CR)

7	6	5	4	3	2	1	0
N/A	AECE	HBUS	IOM	TCIE	TOIE	DIE	RUN

BIT	SYMBOL	FUNCTION
0	RUN	Run/stop bit. A 1 places the DMAC in the run mode. A 0 terminates DMAC operation.
1	DIE	Device interrupt enable. A 1 allows a high input on DINTR to set the $\overline{INTR}$ output low.
2	TOIE	Time-out interrupt enable. A 1 allows the time-out one-shot to set the $\overline{INTR}$ output low. The time-out interrupt is set during a DMA transfer if $\overline{REPLY}$ does not go low within 5 usec of $\overline{MSYNC}$ going low.
3	TCIE	Transfer count zero interrupt enable. A 1 allows a zero in the transfer count register to set the $\overline{INTR}$ output low.
4	IOM	Input or output mode. A 1 sets READ mode (from the peripheral device to memory), and a 0 sets WRITE mode (from memory to the peripheral device). This bit also appears as an ungated output on the R/$\overline{W}$ pin.
5	HBUS	Hold bus. A 1 informs the DMAC to hold onto the bus for the entire block instead of releasing the bus after each byte or word transfer.

BIT	SYMBOL	FUNCTION
6	AECE	Address extension carry enable. A 1 allows a carry from DMA address bit 15 to propogate into bit 16.
7	N/A	Not used.

NOTE: Bits 1, 2, 3 set $\overline{\text{INTR}}$ low on an active condition if and only if the $\overline{\text{STOPR}}$ input is high.

DMAC STATUS REGISTER (SR)

7	6	5	4	3	2	1	0
BUSY	AECE	HBUS	IOM	TCZI	TOI	DINT	BOW

BIT	SYMBOL	FUNCTION
0	BOW	Byte or word data channel. A read only bit that indicates the status of the BOW input pin. A 1 bit indicates byte mode, and the DMA memory address is incremented by one after each DMA transfer. A 0 bit indicates word mode, and the DMA memory address is incremented by two (bit 0 is forced to a 0) after every DMA transfer.
1	DINT	If set a device interrupt has occurred. This is a read/write bit. Resetting this bit to a zero will reset $\overline{\text{INTR}}$.
2	TOI	If set a time-out interrupt has occurred. This is a read/write bit. Resetting this bit to a zero will reset $\overline{\text{INTR}}$.
3	TCZI	If set a transfer count equals zero interrupt has occurred. A read only bit. Sets EOB output when set.
4	IOM	Input-output mode. This bit reflects the status of bit 4 in the Command Register. A read only bit.
5	HBUS	Hold bus. This bit reflects the status of bit 5 in the Command Register. A read only bit.
6	AECE	Address extension carry enable. This bit reflects the status of bit 6 in the Command Register. A read only bit.
7	BUSY	Busy (data transfer not completed). A read only bit that reflects the status of bit 0 (RUN) in the Command Register.

NOTE: Bits 1, 2, 3 are set if the corresponding condition occurs. The enable bits in the CR affect only the $\overline{\text{INTR}}$ output, and not the Status Register.

ID CODE REGISTER (IDR)

An 8-bit programmable interrupt ID code register that gives the system an efficient way to establish a jump or vector address during a DMAC interrupt. The register is cleared to all zeros during a MASTER RESET, and must be loaded by the program during system initialization. If $\overline{\text{INTR}}$ is low, and $\overline{\text{IACKI}}$ and $\overline{\text{RE}}$ go low then the contents of this register are gated onto DAL 0-7. $\overline{\text{IACKI}}$ and $\overline{\text{CS}}$ must not be allowed to be low at the same time.

MASTER RESET

All register bits are reset to a zero during a MASTER RESET except the following which are set to ones: TCR bit 0, CR4, CR5, CR6, SR4, SR5, and SR6. This sets up the DMAC for a 64K transfer from the peripheral device to memory starting at address 0. The hold bus mode is also enabled. Execution of an Auto Load will begin DMA transfers under the above conditions.

AUTO LOAD

If the AUTLD input is made active after a MASTER RESET then bits CR3, CR1, and CR0 are also set. This places the DMAC in run mode, and enables two of the interrupt conditions. The DMAC will initiate data transfers, and will continue until either the transfer count reaches zero or a device interrupt occurs. Either event will terminate transfers and generate an interrupt.

WRITE PROTECT FEATURE

During CPU controlled transfers to the DMAC, if the RUN bit is set then any attempt to write into any of the Memory Address or Transfer Count registers will result in a NOP. $\overline{\text{REPLY}}$ will be made low in any case.

CPU CONTROLLED DATA TRANSFERS

During a CPU controlled transfer the CPU must have control of the system bus. When a CPU cycle is

initiated the system decodes the address on the bus. If the DMAC or its associated peripheral device is selected then $\overline{CS}$ to the DMAC is made low. The DMAC looks at the A3 input. If A3 is low the peripheral device is selected, and $\overline{DCS}$ is made low. The DMAC will not respond to an active $\overline{RE}$ or $\overline{WE}$ if A3 is low, and the DAL bus will stay in a high impedance state. This allows the DMAC DAL bus and the device DAL bus to be tied together if the device DAL bus is also in a high impedence state when the device is not selected.

If A3 is high when $\overline{CS}$ is low then the DMAC is selected and will respond to an active low $\overline{RE}$ or $\overline{WE}$. A0-A2 selects the DMAC as described under the REGISTER SELECTION section. If $\overline{RE}$ goes low the DMAC places the contents of the selected register on the DAL bus and activates $\overline{REPLY}$ to inform the CPU that valid data is on the bus. If $\overline{WE}$ goes low the DMAC gates the contents of the DAL bus into the selected register and activates $\overline{REPLY}$ to inform the CPU that data has been accepted.

If the peripheral device has more than 8 registers, or the device has fewer than 8 registers and there are one or more auxiliary registers external to the device, then it may be easier for the user to separate DMAC and device chip selects. In this mode $\overline{CS}$ to the DMAC is activated if and only if the DMAC is selected and A3 is tied to +5 VDC. The chip select to the device from a CPU controlled data transfer is ORed with $\overline{DCS}$ out of the DMAC. In this mode $\overline{DCS}$ will go low if and only if a DMA transfer is in effect and can be used by the controller as a "DMA ACTIVE" signal. Note that in any case actual data transfers to and from the CPU and the peripheral device are done by way of the device's DAL bus, not the DMAC's DAL bus.

DMAC CONTROLLED DATA TRANSFERS

When the DMAC is in RUN mode (CR0=1) it waits for a Data Request (DRQ) input from the peripheral device. When DRQ becomes active the DMAC requests the bus from the CPU by activating $\overline{BUSR}$. If $\overline{STOPR}$ was active when DRQ went active then the DMAC would wait until $\overline{STOPR}$ went high before activating $\overline{BUSR}$. When $\overline{BACKI}$ goes low in response to an active $\overline{BUSR}$ the request has been granted and the DMAC controls data transfers between the peripheral device and memory. The direction of the transfer is determined by the status of the READ/$\overline{WRITE}$ (R/$\overline{W}$) output pin. Note that R/$\overline{W}$ is tied directly to CR4.

1.) DEVICE-TO-MEMORY DMA TRANSFERS (CR4=1)

Once the DMAC has been granted the bus the following occurs:

A.) The DMAC places the high byte of the memory address on the DAL lines, activates $\overline{DCS}$, and then raises $\overline{BUSR}$. The trailing edge of $\overline{BUSR}$ can be used to latch the address into an external buffer.

B.) The DMAC places the low byte of the memory address on the DAL lines while activating $\overline{LAL}$, and then activates $\overline{MSYNC}$. The trailing edge of $\overline{LAL}$ can be used to latch the address into an external buffer

C.) The DAL lines are placed into a high impedence state in anticipation of a data transfer across the bus.

D.) The DMAC activates $\overline{RE}$ and then activates $\overline{MEMW}$.

E.) The DMAC waits for $\overline{REPLY}$ to go low. When $\overline{REPLY}$ is active the DMAC deactivates $\overline{MEMW}$ and then deactivates $\overline{RE}$.

F.) If the DMAC is *not* in hold bus mode (CR5=1) then the DMAC deactivates $\overline{DCS}$ and gives up control of the bus. If the DMAC is in hold bus mode then $\overline{DCS}$ remains low until after the completion of the final data transfer. Note that $\overline{BUSR}$ still cycles for every transfer.

G.) After the completion of every data transfer the memory address register is incremented by one in byte mode or two in word mode.

H.) After the completion of every data transfer the transfer count is incremented by one. Transfers are considered to be completed when the transfer count equals zero.

2.) MEMORY-TO-DEVICE DMA TRANSFERS (CR4=0)

Once the DMAC has been granted the bus it goes through the same steps as in the DEVICE-TO-MEMORY mode with the exception of steps "D" and "E" which are as follows:

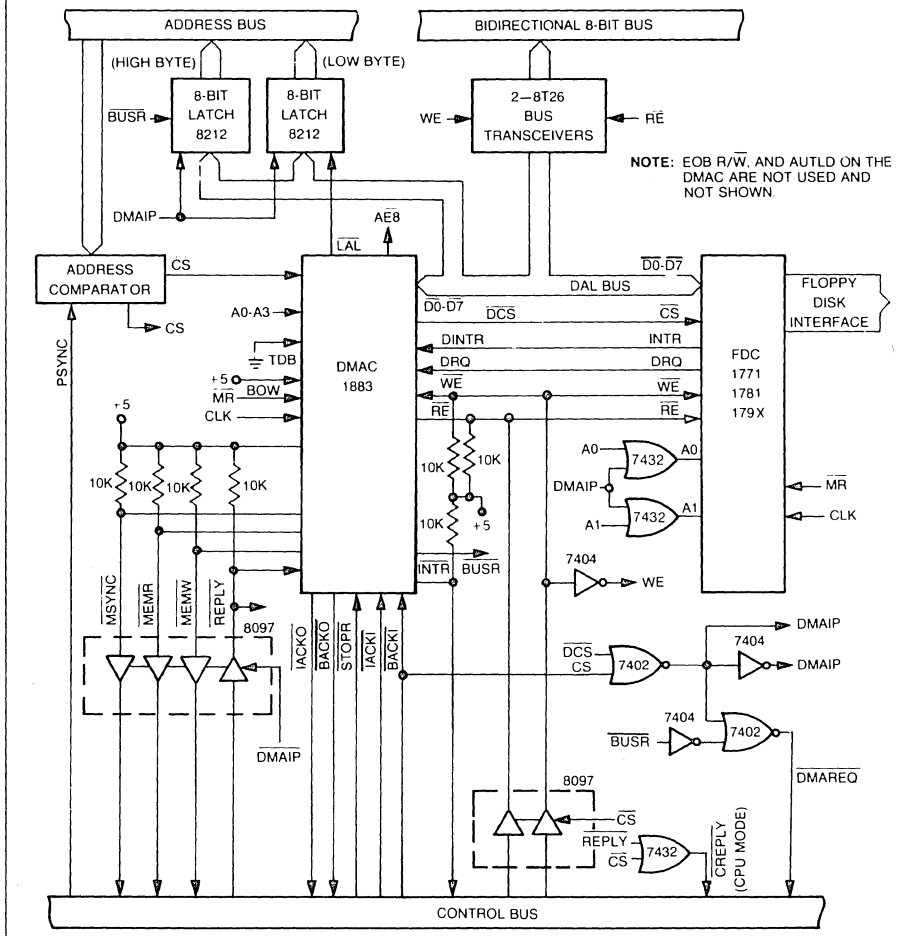
D.) The DMAC activates $\overline{MEMR}$ and then activates $\overline{WE}$.

E.) The DMAC waits for $\overline{REPLY}$ to go low. When REPLY is active the DMAC deactivates $\overline{WE}$ and then deactivates $\overline{MEMR}$.

In either mode $\overline{BACKI}$ will be gated out to $\overline{BACKO}$ as soon as the DMAC deactivates $\overline{DCS}$. This allows other devices in the chain to gain access to the bus immediately.

INTERRUPTS

There are three individually enabled interrupt conditions. If any of the conditions occurs it will set its corresponding bit in the Status Register. If the appropriate enable bit in the Command Register is set then $\overline{INTR}$ is also activated. Note that these are independent functions. When $\overline{INTR}$ is active then the DMAC can be selected by an active $\overline{IACKI}$ instead of an active $\overline{CS}$. $\overline{CS}$ and $\overline{IACKI}$ must not both be active at the same time.



TYPICAL DMAC TO FDC APPLICATION

Once an interrupt condition sets its corresponding bit in the status register the bit stays set until a CPU write to the status register occurs with a zero in the bit position.* If any one (or more) of the three interrupt condition bits in the Status Register is set then IACKI will not be gated out to IACKO even if the interrupt is *not* enabled.

NOTE: For a transfer-count-equals-zero interrupt condition to be cleared the Transfer Count Register must be loaded with a non-zero count.

The three interrupt conditions are as follows:

1.) DEVICE INTERRUPT (DINT)

A device interrupt condition occurs when the DINTR input is made high. This sets SR1 and, if CR1 is set, it activates INTR. The RUN bit is also reset thus terminating all subsequent DMA transfers. A device interrupt could be generated by a number of causes, and the program will have to test the device's Status Register to determine the cause of the interrupt. The DINT status bit in the DMAC Status Register must be cleared by the program as a part of the interrupt service routine.

2.) TRANSFER COUNT EQUALS ZERO INTERRUPT (TCZI)

When the TCR is incremented to zero after a DMA transfer the TCZI status bit (SR3) is set and the RUN bit (CR0) is reset. This terminates all DMA operations and, if CR3 is set, activates INTR. SR3 can be cleared only by loading a non-zero value into the TCR. The EOB output pin is high whenever SR3 is set.

3.) TIME-OUT INTERRUPT (TOI)

During any DMA transfer the leading edge of MSYNC triggers an internal time delay of approximately 5 microseconds. If the DMAC does not receive an active low REPLY input within that time delay then the DMA operation is terminated, the RUN bit is reset, and the TOI status bit (SR2) is set. If CR2 is set then INTR is activated. SR2 can only be cleared by writing a zero into that position of the Status Register.

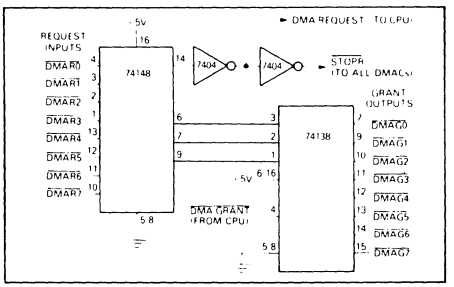
INTERRUPT OPERATION

When the DMAC activates INTR the CPU responds by activating IACKI. This signal can be daisy chained through all devices. The first device in the chain that has any bit in SR1-SR3 set will block the gating of IACKI out to IACKO. In addition, if INTR is active an IACKI will select the DMAC. An active RE after an IACKI select will gate the contents of the interrupt ID code register onto the DAL lines. The ID code stays active on the DAL lines as long as IACKI and RE are active. This code, which is cleared to zero by a MASTER RESET and loaded by the program during system initialization, can be used by the system to create a JUMP or VECTOR address for the device interrupt routine. Note that an active CS during a DMAC select via an active IACKI will cause unspecified results. Note also that no condition can activate INTR unless its corresponding enable bit is set and STOPR is high. If STOPR is active when the interrupt condition occurs then the DMAC will hold INTR inactive until STOPR goes inactive. At that time the DMAC will activate INTR automatically.

DMA PRIORITY SYSTEMS

Fixed Priority

A fixed priority can be established in two ways: through a parallel request-grant system or through a CPU controlled daisy chain system. A typical asynchronous parallel DMA priority system is shown. In this system any request generates an active STOPR, which is gated to all devices, and an active DMA request to the CPU. The CPU DMA grant generates a grant to the requesting device with the highest priority. If more than one request is received at the same time then the grants are honored from the highest to the lowest priority. In most cases, however, grants are not received simultaneously. The highest priority devices, therefore, will receive most of the immediate grants with the others being delayed by an active STOPR.



ASYNCHRONOUS PARALLEL DMA PRIORITY SYSTEM

Establishing a fixed priority system through a daisy chain approach requires the CPU monitor a "DMA IN PROGRESS" signal on the bus. This signal can be generated from DCS during a DMA transfer (i.e., DCS·CS). In this mode the CPU activates BACKI and STOPR in response to some bus request. STOPR is tied to all DMA controllers to prevent new bus requests while BACKI is propagating through all non-requesting DMAC devices. When the requesting DMAC gains control over the bus and activates DCS the CPU drops BACKI*. When DCS is deactivated the CPU deactivates STOPR to allow new requests. In this manner the device physically closest to the CPU on the daisy chain has highest priority for all request cycles.

NOTE: BACKI and STOPR can be dropped at the same time with no effect on the priority scheme, but the CPU may have to capture new requests until DCS goes high.

Rotating Priority

This is a daisy chain approach that prevents one device from getting most of the bus grants if multiple devices are active at the same time. In this mode any device requesting the bus causes the CPU to activate $\overline{\text{BACKI}}$. This signal is tied to the $\overline{\text{BACKI}}$ and $\overline{\text{STOPR}}$ inputs of the first DMAC. The $\overline{\text{BACKO}}$ output of the first DMAC goes to the $\overline{\text{BACKI}}$ and $\overline{\text{STOPR}}$ inputs of the second DMAC, and so on. The $\overline{\text{BACKO}}$ output of the last DMAC in the chain goes back to the CPU to reset its $\overline{\text{BACKI}}$ output. In this mode the first device cannot request again until all other requesting devices in the chain have also been serviced.

In any case, if the CPU has to have the DMA request held active throughout the DMA cycle then the user will have to create this signal on the controller thusly: DMAREQ = BUSR + (DCS·$\overline{\text{CS}}$). If the device and DMAC chip selects are generated on the controller separately then the $\overline{\text{CS}}$ can be eliminated from the equation. It is needed only to distinguish a CPU chip select from a DMA cycle chip select. Note that in either case the second term in the equation is equivalent to "DMA CYCLE IN PROGRESS" (DMAIP).

SPECIFICATIONS

Absolute Maximum Ratings

Ambient Temperature Under Bias... 0°C to +70°C
Voltage on Any Pin with Respect to Ground ..... -0.5V to +7V
Power Dissipation ..... 0.6 Watt

NOTE: Maximum ratings indicate limits beyond which permanent damage may occur. Continuous operation at these limits is not intended and should be limited to those conditions specified under dc electrical characteristics.

DC Electrical Characteristics

T_A = 0°C to +70°C; V_{CC} = 5.0V ±5%; GND =0V

SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNIT	TEST CONDITIONS
V_{IL}	Input Low Voltage	-0.5		0.8	V	
V_{IH}	Input High Voltage	2.4		V_{CC}	V	
V_{OL}	Output Low Voltage			0.45	V	I_{OL} = 1.6 mA
V_{OH}	Output High Voltage	2.4			V	I_{OH} = -100 µA
I_{DL}	Data Bus Leakage			-50	µA	V_{IN} = 0.45V
				10	µA	V_{IN} = V_{CC}
I_{IL}	Input Leakage			10	µA	V_{IN} = V_{CC}
I_{CC}	Power Supply Current		45	90	mA	

NOTE: VOL ≤0.4V when interfacing with low power Schottky parts (I_{OL} <1 mA).

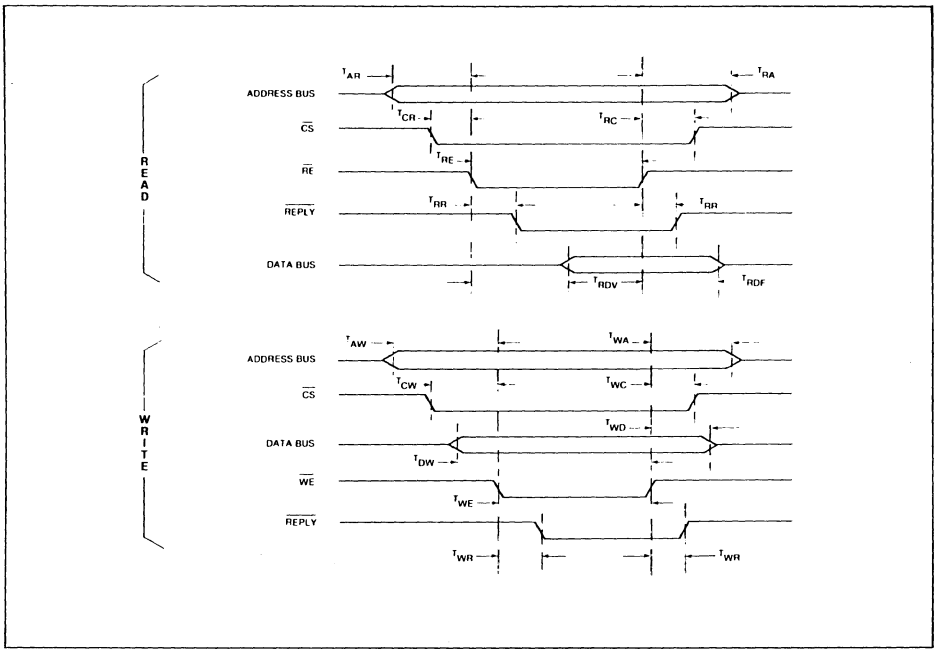
Capacitance

T_A = 25°C; V_{CC} = GND = 0V

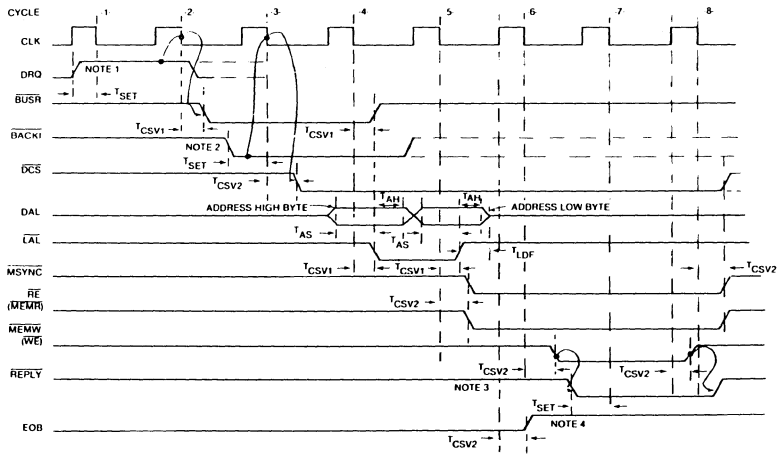
SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNIT	TEST CONDITIONS
C_{IN}	Input Capacitance			10	pF	f_C = 1 MHz
$C_{I/O}$	I/O Capacitance			20	pF	Unmeasured pins returned to GND.

System Clock (CLK) Characteristics

Maximum Frequency	2.0 MHz
Minimum Pulse Width	250 ns
Maximum Pulse Width	50% of duty cycle



CPU CONTROLLED TRANSFER



NOTES: 1 $\overline{BUSR}$ ↑ FOLLOWS SECOND CLK ↑ AFTER DRQ ↑.
2 a CYCLE 2 IS SKIPPED FOR ALL SUBSEQUENT TRANSFERS WHEN IN THE HOLD MODE. I E., $\overline{DCS}$ REMAINS LOW FOR ENTIRE BLOCK OF TRANSFERS.
b FOLLOWING $\overline{BUSR}$ ↑ WITH $\overline{DCS}$ HIGH (I E., REQUESTING BUS CONTROL), THE DMA WILL ADD WAIT CYCLES BETWEEN CYCLES 2 AND 3 UNTIL $\overline{BACK1}$ ↑ IS RECEIVED.
3 CYCLE 7 WILL BE REPEATED FOR EACH ADDITIONAL PERIOD THAT $\overline{REPLY}$ ↑ IS DELAYED TO THE DMA. CYCLE 7 WILL BE SKIPPED (I E., $\overline{MEMW}$ ($\overline{WE}$) PULSE WIDTH = 1/2 CLK PERIOD) IF $\overline{REPLY}$ IS MADE LOW PRIOR TO CLK ↑ OF CYCLE 6 (E G., $\overline{REPLY}$ TIED TO GND DURING DMA TRANSFER).
4 EOB IS ACTIVATED ONLY FOR THE TRANSFER WHERE TCR INCREMENTS FROM 11 1 TO 00 0 (I E., END OF BLOCK).

DMA CONTROLLED TRANSFER TIMING

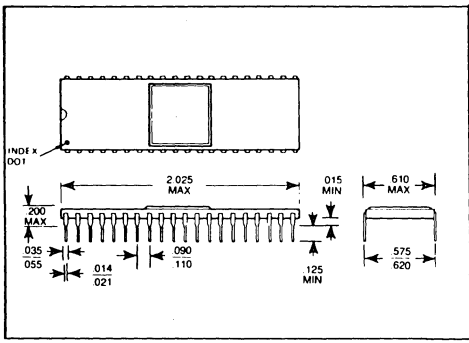
AC Electrical Characteristics

T_A = 0°C to +70°C; V_{CC} 5.0V ±5%; GND 0V

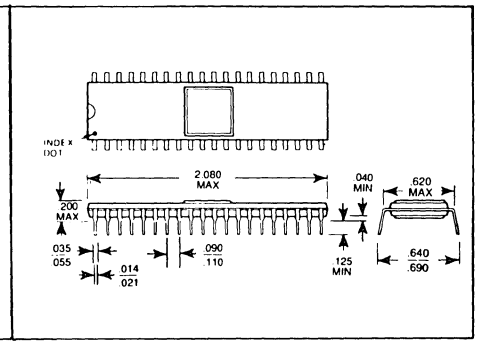
SYMBOL	DESCRIPTION	MIN	MAX.	UNIT	COND
CPU CONTROLLED TRANSFER TIMING - READ					
T_{AR}	Address Valid to $\overline{RE}$ ↓	80		ns	
T_{CR}	$\overline{CS}$ ↓ to $\overline{RE}$ ↓	0		ns	
T_{RE}	$\overline{RE}$ Pulse Width	300		ns	
T_{RDV}	$\overline{RE}$ ↓ to Data Valid		375	ns	CL = 50 pF
T_{RR}	$\overline{RE}$ ↓ (↓) to $\overline{REPLY}$ ↓ (↓)	50	350	ns	CL = 50 pF
T_{RA}	Address Hold from $\overline{RE}$ ↑	30		ns	
T_{RC}	$\overline{CS}$ Hold from $\overline{RE}$ ↑	0		ns	
T_{RDF}	Data Float from $\overline{RE}$ ↑		200	ns	
CPU CONTROLLED TRANSFER TIMING - WRITE					
T_{AW}	Address Valid to $\overline{WE}$ ↓	80		ns	
T_{CW}	$\overline{CS}$ ↓ to $\overline{WE}$ ↓	0		ns	
T_{DW}	Data Valid to $\overline{WE}$ ↑	300		ns	CL = 50 pF
T_{WE}	$\overline{WE}$ Pulse Width	300		ns	
T_{WR}	$\overline{WE}$ ↓ (↓) to $\overline{REPLY}$ ↓ (↓)	50	350	ns	CL = 50 pF
T_{WA}	Address Hold from $\overline{WE}$ ↑	30		ns	
T_{WC}	$\overline{CS}$ Hold from $\overline{WE}$ ↑	0		ns	
T_{WD}	Data Hold from $\overline{WE}$ ↑	30		ns	

SYMBOL	DESCRIPTION	MIN	TYP	MAX.	UNIT	COND
DMA CONTROLLED TRANSFER TIMING						
T_{CSV1}	Indicated CLK Edge to Indicate Signal Valid		150	250	ns	CL = 50 pF
T_{CSV2}	Indicated CLK Edge to Indicated Signal Valid		250	400	ns	CL = 50 pF
T_{AS}	DAL Set Up to $\overline{BUSR}$ ↑ or $\overline{LAL}$ ↑ (↓)	80			ns	CL = 50 pF
T_{AH}	DAL Hold from $\overline{BUSR}$ ↑ or $\overline{LAL}$ ↑ (↓)	50			ns	CL = 50 pF
T_{LDF}	$\overline{LAL}$ ↑ to DAL Float			250	ns	CL = 50 pF
T_{SET}	Indicated Signal Setup to Indicated CLK Edge	80			ns	
MISCELLANEOUS TIMING (τ = 1 CLOCK PERIOD)						
$\overline{CS}$ ↑ (↓) To $\overline{DCS}$ ↑ (↓) Propogation Delay (for A3 low)			150	250	ns	CL = 50 pF
$\overline{IACKI}$ ↑ (↓) to $\overline{IACKO}$ ↑ (↓) Propogation Delay when Not Requesting Interrupt			150	250	ns	CL = 50 pF
$\overline{BACKI}$ ↑ (↓) to $\overline{BACKO}$ ↑ (↓) Propogation Delay when Not Requesting Bus			150	250	ns	CL = 50 pF
$\overline{MR}$ Pulse Width		2τ				
DINTR, AUTLD, DRQ, $\overline{REPLY}$ Pulse Width		1τ				
BOW ↑ (↓) or TDB ↑ (↓) Set Up		500			ns	
Waiting $\overline{INTR}$ ↑ or $\overline{BUSR}$ ↑ from $\overline{STOPR}$ ↑				1τ +400	ns	CL = 50 pF
$\overline{INTR}$ ↑ from DINTR ↑				1.5τ +400	ns	CL = 50 pF

NOTE: A 1 TTL load is assumed on all output signals



DM1883 CERAMIC PACKAGE



DM1883 PLASTIC PACKAGE

This is a preliminary specification with tentative device parameters and may be subject to change after final product characterization is completed.

Information furnished by Western Digital Corporation is believed to be accurate and reliable. However, no responsibility is assumed by Western Digital Corporation for its use; nor any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Western Digital Corporation. Western Digital Corppration reserves the right to change said circuitry at any time without notice.

WESTERN DIGITAL CORPORATION
3128 REDHILL AVENUE, BOX 2180
NEWPORT BEACH, CA 92663 (714) 557-3550,TWX 910-595-1139