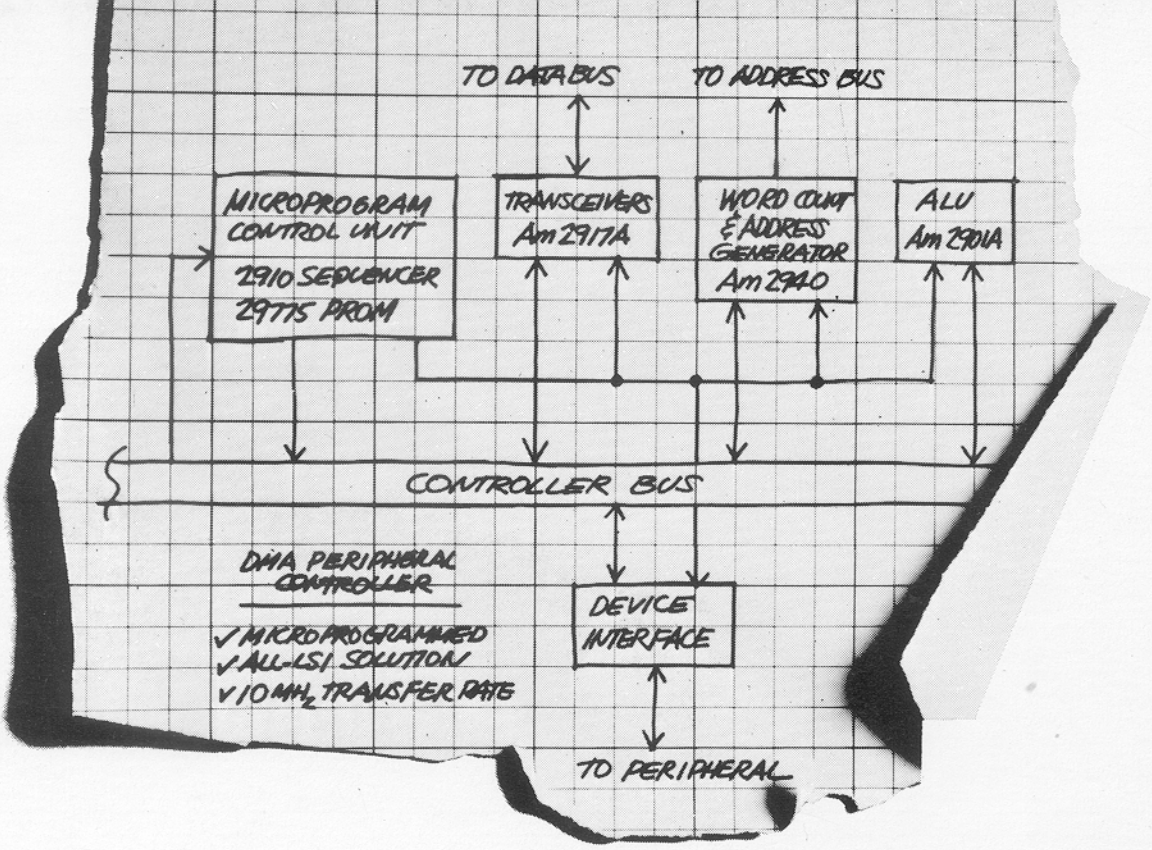


Build A Microcomputer

Chapter VII
Direct Memory Access

Advanced Micro Devices

Advanced Micro Devices

Build A Microcomputer

Chapter VII
Direct Memory Access

AM-PUB073-7

Introduction

The transfer of data between the microcomputer and the peripheral devices is generally referred to as Input/Output (I/O). What is desired is a high speed technique of transferring data between the peripherals and the memory. Generally speaking, there is a minimum of three types of I/O. These are, Programmed I/O, Memory Mapped I/O and Direct Memory Access I/O. All of these schemes are common in today's currently available minicomputers. A basic understanding of these I/O techniques is helpful in fully comprehending DMA. The first two of these types of I/O can be interrupt driven. That is, programmed I/O or memory mapped I/O can be initiated by an interrupt from the peripheral device.

Programmed I/O

In this type of I/O, all operations are controlled by the CPU program. In other words, the peripheral device performs the functions of inputting or outputting data as it is controlled by the CPU. Normally, the machine will include a set of I/O instructions which are used to transfer data to or from the peripheral devices via an Input/Output port. All data for the peripheral devices passes through these I/O ports to the CPU and the resources of the CPU must be utilized in order to effect an I/O transfer. Figure 1 shows the Block Diagram of a programmed I/O system used in a typical microcomputer. Figure 2 shows an example of that portion of the program used to output data to the peripheral device.

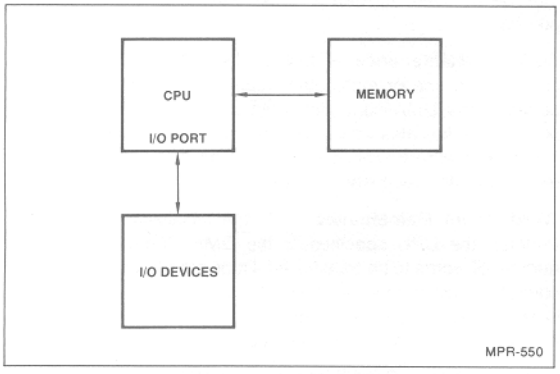


Figure 1. Programmed I/O System.

CPU Program	Comments
–	–
–	–
Load R, M	Load CPU Register R with the Contents of Memory Address M
Out D, R	Transfer the Contents of CPU Register R to I/O Device D via the I/O port.
–	–

Figure 2. Example Output Program – Programmed I/O.

Programmed I/O is simple to implement and does not require the utilization of any memory addresses for its realization. In addition, special instructions are available to the programmer to execute the peripheral data transfers. Programmed I/O is also low cost relative to other types of I/O; however, it has the following disadvantages. Since I/O device operation is asynchronous with respect to CPU operation, the CPU has no way of knowing when a peripheral device is ready to transfer data and must periodically poll the device to determine its readiness. This results in an inefficient I/O transfer. Also, since the CPU must be used to effect the I/O transfer, the CPU resources are tied up during the time of transfer and the time of polling and cannot be used for other tasks. For these reasons, Programmed I/O is generally limited to use with low speed devices.

Perhaps, one of the best known programmed I/O microcomputers in the industry today is the Am9080A. This device features two instructions for either inputting data or outputting data to any one of 256 Input/Output ports.

Memory Mapped I/O

Memory Mapped I/O is a technique whereby the transfer of data to and from peripheral devices is accomplished by using some of the normally available memory space. In this technique, memory addresses are decoded within the peripheral devices and are thus used to determine when a specific device is being addressed. Usually, each type of function within the peripheral device is assigned a memory address and can then be accessed by the CPU. For example, the peripheral device may contain a command register, a status register, a data in register and a data out register. Thus, four memory addresses might be utilized in performing I/O to this peripheral. Figure 1 is also the block diagram for a Memory Mapped I/O scheme.

The chief advantage of Memory Mapped I/O is that all of the memory reference instructions are usually available to perform the I/O function. Consequently, no special I/O instructions are required in the machine. The key disadvantage of this technique is that a block of the memory addressing range must be set aside for assignment to the peripheral devices. Thus, the overall memory addressing range of the machine is reduced by the size of this block. Again, the resources of the CPU are tied up while the I/O is being performed. A well known machine using only Memory Mapped I/O is the PDP 11. In it the upper 4k of memory space is usually used for the I/O devices.

Interrupt Driven I/O

Interrupts are means by which a peripheral device can stop the normal flow of the CPU instruction execution and force the CPU to temporarily suspend its current program. Then, the program "jumps" to a different program which executes an I/O transfer. Typically, this eliminates the need for polling the peripheral devices to determine if an I/O transfer is ready. Thus, the interrupt driven scheme provides a more efficient I/O transfer technique. However, there is an overhead burden associated with interrupts in that the CPU must store away and later restore all of the parameters required to resume the interrupted program. This overhead degrades the CPU performance. Depending on the overall interrupt structure, the CPU still may have to do some polling of devices which may be tied to the same interrupt level.

It should be pointed out that both Programmed I/O and Memory Mapped can take advantage of the interrupt technique. That is, an interrupt can be used to initiate the peripheral data transfer in either type of system. The CPU still must control the transfer of the data between the memory and the peripheral device and the CPU resources are unavailable for executing other instructions during this time.

What is DMA?

DMA is a technique for data transfer which provides a direct path between the I/O device and the memory without CPU intervention. With this path, a peripheral device has "Direct Memory Access" and can transfer data directly to or from the memory. The

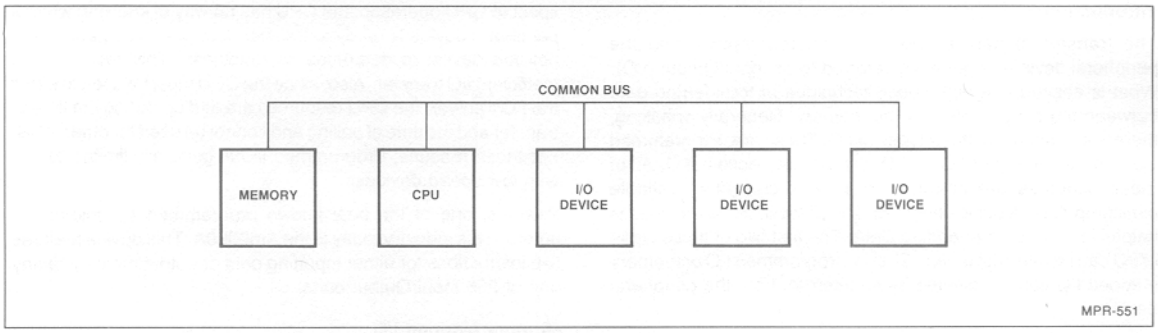


Figure 3. DMA I/O System.

purpose of the DMA is to relieve the CPU of the task of controlling the I/O transfer, thereby freeing it to perform other tasks during this time, and to provide a means by which data can be transferred between an I/O device and memory at very high speed. Figure 3 shows the Block Diagram of a system where several I/O devices can perform DMA transfers into memory. Note that the CPU and peripheral devices share a common bus to the memory and that the CPU and peripheral devices cannot access memory during the same cycle. DMA can also be designed to perform memory-to-memory transfers or I/O-to-I/O transfers.

Several DMA transfer methods exist, such as the CPU halt method, the memory timeslice method, and the "cycle steal" method. In the CPU halt method, the CPU is halted and switched off the bus while a DMA transfer occurs. This is the most straightforward method. However, it takes a relatively long time to switch the CPU on and off the bus, and the CPU cannot do anything during the transfer.

The memory timeslice method works by splitting each memory cycle into two timeslots; one is reserved for the CPU and the other for DMA. This method provides the highest CPU execution rate as well as the highest DMA transfer rate because both the CPU and DMA are guaranteed access to memory during every memory cycle. The disadvantage of this method is that high speed, costly memories must be used.

The "cycle steal" method is a cost/performance compromise between the low cost of the CPU halt method and the high performance of the memory timeslice method. Cycle stealing refers to a DMA device "stealing" a CPU memory cycle in order to execute a DMA transfer. CPU program execution continues during the DMA transfer (the CPU is not halted), resulting in an overlap of CPU program execution with DMA transfer. If the CPU and a DMA device require a memory cycle at the same time, priority is granted to the DMA device and the CPU waits until the DMA cycle is completed. DMA causes CPU performance degradation only in those applications where the CPU uses the entire memory bandwidth. In many applications the CPU is slow relative to memory cycle time and "cycle stealing" provides satisfactory performance at relatively low cost.

How is DMA Implemented?

In order to relieve the CPU of the I/O transfer control task, circuitry external to the CPU must be added. This circuitry is called the DMA Controller and performs the following functions.

Address Line Control – In a DMA system, the memory address lines are driven by either the CPU or a DMA device, depending on which is using the memory during a given cycle. The DMA controller must switch the appropriate address onto the memory address lines.

Data Transfer Control – The DMA Controller must provide the control signals required to transfer data directly between memory and an I/O device. As with the address lines, these control signals must be switched onto and off of the memory control lines appropriately.

Address Maintenance – Just as the CPU has the program counter and one or more other registers for memory address pointers, the DMA controller must also maintain an address pointer that indicates where the next word of data will be read or written in memory. This pointer must be incremented or decremented after each word transfer.

Word Count Maintenance – At the initialization of a DMA transfer, the CPU specifies to the DMA Controller the total number of words to be transferred. During the transfer, the DMA controller must maintain a count of the number of words that have been transferred and terminate the transfer when the specified number of words has been reached.

Mode Control – Certain aspects of a DMA transfer, such as direction of data flow, method of termination, etc., may vary from one DMA transfer to the next. For this reason, a number of DMA modes may be required. Mode control logic contained in the DMA controller, is set by the CPU at the initialization of a DMA transfer.

A DMA Controller can be placed in each I/O device (Distributed DMA) or DMA control circuitry for a number of I/O devices can be placed in a separate unit (Centralized DMA). The former provides the advantage of incremental cost; DMA control circuitry is added only as I/O devices are added. The latter provides the advantages of consolidation.

At DMA initialization, the CPU normally specifies the mode, the starting memory address and the number of words to be transferred (word count) to the DMA controller. In some applications, it is desirable to repeat a DMA transfer over and over again without disturbing the CPU. This capability is called Repetitive DMA, and can be implemented by adding two registers to the DMA controller. One register saves the starting address and the other the starting word count. This allows the DMA Controller to automatically reinitialize itself after the transfer of the data has been completed, thereby eliminating the need for CPU intervention.

The Am2940 DMA ADDRESS GENERATOR

The design of the Address Line Control, Data Transfer Control and Mode Control circuitry of a DMA Controller is dependent upon system architecture and timing; therefore, it varies considerably from system to system. However, the address maintenance and word count maintenance circuitry is independent of these variables, and is common to almost all DMA Controllers. The Am2940 DMA Address Generator is designed for use in DMA Controllers and provides the Address and Word Count maintenance circuitry that is common to most. It combines the advantages of high speed bipolar LSI with the flexibility and general purpose usefulness of microprogrammed control.

Am2940 GENERAL DESCRIPTION

The Am2940, a 28-pin member of Advanced Micro Devices Am2900 family of Low-Power Schottky bipolar LSI chips, is a high-speed, cascadable, eight-bit wide Direct Memory Access Address Generator slice. Any number of Am2940s can be cascaded to form larger addresses.

The primary function of the device is to generate sequential memory addresses for use in the sequential transfer of data to or from a memory. It also maintains a data word count and generates a DONE signal when a programmable terminal count has been reached. The device is designed for use in peripheral controllers with DMA capability or in any other system which transfers data to or from sequential locations of a memory.

The Am2940 can be programmed to increment or decrement the memory address in any of four control modes, and executes eight different instructions. The initial address and word count are saved internally by the Am2940 so that they can be restored later in order to repeat the data transfer operation.

Am2940 ARCHITECTURE

As shown in the Block Diagram of Figure 4, the Am2940 consists of the following:

- A three-bit Control Register.
- An eight-bit Address Counter with input multiplexer.
- An eight-bit Address Register.
- An eight-bit Word Counter with input multiplexer.
- An eight-bit Word Count Register.
- Transfer complete circuitry.
- An eight-bit wide data multiplexer with three-state output buffers.
- Three-state address output buffers with external output enable control.
- An instruction decoder.

Control Register

Under instruction control, the Control Register can be loaded or read from the bidirectional DATA lines D_0-D_7. Control Register bits 0 and 1 determine the Am2940 Control Mode, and bit 2 determines whether the Address Counter increments or decrements. Figure 5 defines the Control Register format.

Address Counter

The Address Counter, which provides the current memory address, is an eight-bit, binary, up/down counter with full look-ahead carry generation. The Address Carry Input ($\overline{ACI}$) and Address Carry Output ($\overline{ACO}$) allow cascading to accommodate larger addresses. Under instruction control, the Address Counter can be enabled, disabled, and loaded from the DATA inputs, D_0-D_7, or the Address Register. When enabled and the $\overline{ACI}$ input is LOW, the Address Counter increments/decrements on the LOW to HIGH transition of the CLOCK input, CP. The Address Counter output can be enabled onto the three-state ADDRESS outputs A_0-A_7 under control of the Output Enable input, $\overline{OE}_A$.

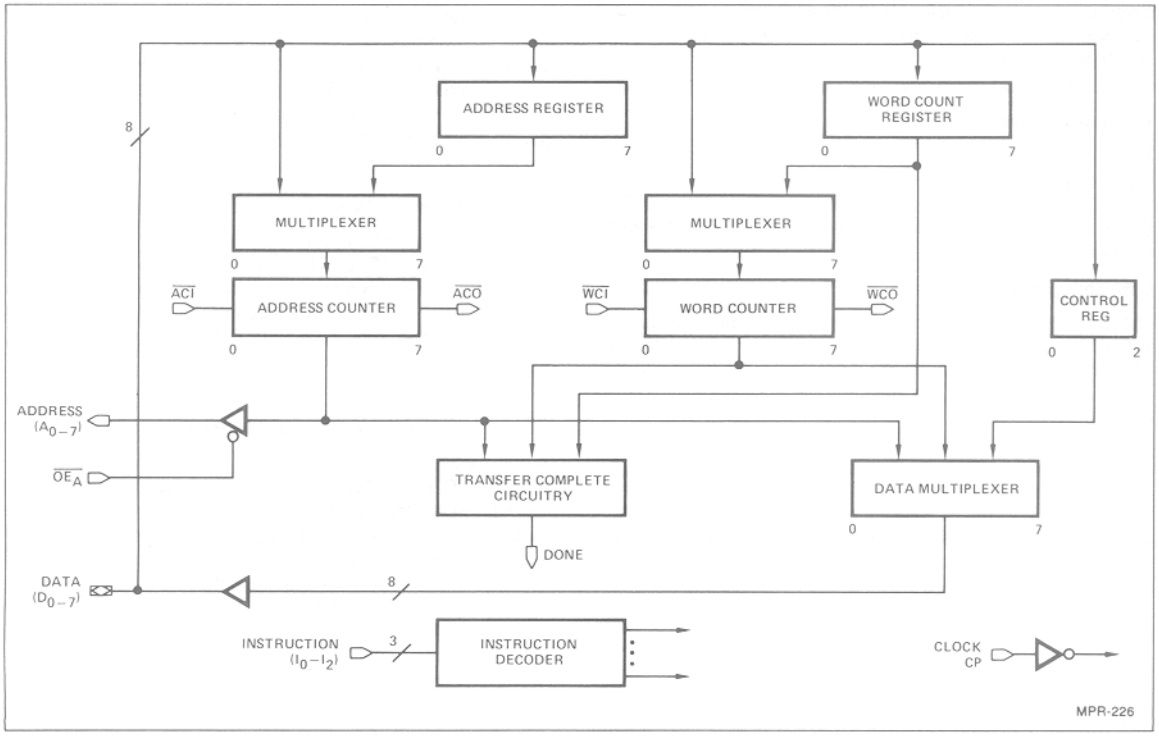


Figure 4. Am2940 DMA Address Generator.

Control Register

CR_2	CR_1	CR_0

CR_1	CR_0	Control Mode Number	Control Mode Type	Word Counter	Done Output Signal	
					$\overline{WCI}$ = LOW	$\overline{WCI}$ = HIGH
L	L	0	Word Count Equals Zero	Decrement	HIGH when Word Counter = 1	HIGH when Word Counter = 0
L	H	1	Word Count Compare	Increment	HIGH when Word Counter + 1 = Word Count Register	HIGH when Word Counter = Word Count Register
H	L	2	Address Compare	Hold	HIGH when Word Counter = Address Counter	
H	H	3	Word Counter Carry Out	Increment	Always LOW	

CR_2	Address Counter
L	Increment
H	Decrement

H = HIGH
L = LOW

Figure 5. Control Register Format Definition.

Address Register

The eight-bit Address Register saves the initial address so that it can be restored later in order to repeat a transfer operation. When the LOAD ADDRESS instruction is executed, the Address Register and Address Counter are simultaneously loaded from the DATA inputs, D_0-D_7.

Word Counter and Word Count Register

The Word Counter and Word Count Register, which maintain and save a word count, are similar in structure and operation to the Address Counter and Address Register, with the exception that the Word Counter increments in Control Modes 1 and 3, decrements in Control Mode 0, and is disabled in Control Mode 2. The LOAD WORD COUNT instruction simultaneously loads the Word Counter and Word Count Register.

Transfer Complete Circuitry

The Transfer Complete Circuitry is a combinational logic network which detects the completion of the data transfer operation in three Control Modes and generates the DONE output signal. The DONE signal is an open-collector output, which can be dot-anded between chips.

Data Multiplexer

The Data Multiplexer is an eight-bit wide, 3-input multiplexer which allows the Address Counter, Word Counter, and Control Register to be read at the DATA lines, D_0-D_7. The Data Multiplexer and three-state Data output buffers are instruction controlled.

Address Output Buffers

The three-state Address Output Buffers allow the Address Counter output to be enabled onto the ADDRESS lines, A_0-A_7, under external control. When the Output Enable input, $\overline{OE}_A$, is LOW, the Address output buffers are enabled; when $\overline{OE}_A$ is HIGH, the ADDRESS lines are in the high-impedance state. The address and Data Output Buffers can sink 24mA output current over the commercial operating range.

Instruction Decoder

The Instruction Decoder generates required internal control signals as a function of the INSTRUCTION inputs, I_0-I_2 and Control Register bits 0 and 1.

Clock

The CLOCK input, CP, is used to clock the Address Register, Address Counter, Word Count Register, Word Counter, and Control Register, all on the LOW to HIGH transition of the CP signal.

Am2940 CONTROL MODES

Control Mode 0 — Word Count Equals Zero Mode

In this mode, the LOAD WORD COUNT instruction loads the word count into the Word Count Register and Word Counter. When the Word Counter is enabled and the Word Counter Carry-in, $\overline{WCI}$, is LOW, the Word Counter decrements on the LOW to HIGH transition of the CLOCK input, CP. Figure 5 specifies when the DONE signal is generated in this mode.

Control Mode 1 — Word Count Compare Mode.

In this mode the LOAD WORD COUNT instruction loads the word count into the Word Count Register and clears the Word Counter. When the Word Counter is enabled and the Word Counter Carry-in, $\overline{WCI}$, is LOW, the Word Counter increments on the LOW to HIGH transition of the clock input, CP. Figure 5 specifies when the DONE signal is generated.

Control Mode 2 — Address Compare Mode

In this mode, only an initial and final memory address need be specified. The initial Memory Address is loaded into the Address Register and Address Counter and the final memory address is loaded into the Word Count Register and Word Counter. The Word Counter is always disabled in this mode and serves as a holding register for the final memory address. When the Address Counter is enabled and the $\overline{ACI}$ input is LOW, the Address Counter increments or decrements (depending on Control Register bit 2) on the LOW to HIGH transition of the CLOCK input, CP. The Transfer Complete Circuitry compares the Address Counter with the Word Counter and generates the DONE signal during the last word transfer, i.e., when the Address Counter equals the Word Counter.

Control Mode 3 – Word Counter Carry Out Mode

For this mode of operation, the user can load the Word Count Register and Word Counter with the two's complement of the number of data words to be transferred. When the Word Counter is enabled and the $\overline{WCI}$ input is LOW, the Word Counter increments on the LOW to HIGH transition of the CLOCK input, CP. A Word Counter Carry Out signal, $\overline{WCO}$, indicates the last data word is being transferred. The DONE signal is not required in this mode and, therefore, is always LOW.

Am2940 INSTRUCTIONS

The Am2940 instruction set consists of eight instructions. Six instructions load and read the Address Counter, Word Counter and Control Register, one instruction enables the Address and Word Counters, and one instruction reinitializes the Address and Word Counters. The function of the REINITIALIZE COUNTERS, LOAD WORD COUNT, and ENABLE COUNTERS instructions vary with the Control Mode being utilized. Table 1 defines the Am2940 Instructions as a function of Instruction inputs I_0-I_2 and the four Am2940 Control Modes.

The WRITE CONTROL REGISTER instruction writes DATA input D_0-D_2 into the Control Register; DATA inputs D_3-D_7 are "don't care" inputs for this instruction. The READ CONTROL REGISTER instruction gates the Control Register outputs to DATA lines, D_0-D_2. DATA lines D_3-D_7 are in the HIGH state during this instruction.

The Word Counter can be read using the READ WORD COUNTER instruction, which gates the Word Counter ouputs to DATA lines D_0-D_7. The LOAD WORD COUNT instruction is Control Mode dependent. In Control Modes 0, 2, and 3, DATA inputs D_0-D_7 are written into both the Word Count Register and Word Counter. In Control Mode 1, DATA inputs D_0-D_7 are written into the Word Count Register and the Word Counter is cleared.

The READ ADDRESS COUNTER instruction gates the Address Counter outputs to DATA lines D_0-D_7, and the LOAD ADDRESS instruction writes DATA inputs D_0-D_7 into both the Address Register and Address Counter.

In Control Modes 0, 1, and 3, the ENABLE COUNTERS instruction enables both the Address and Word Counters; in Control Mode 2, the Address Counter is enabled and the Word Counter holds its contents. When enabled and the carry input is active, the counters increment on the LOW to HIGH transition of the CLOCK input, CP. Thus, with this instruction applied, counting can be controlled by the carry inputs.

The REINITIALIZE COUNTERS instruction also is Control Mode dependent. In Control Modes 0, 2, and 3, the contents of the Address Register and Word Count Register are transferred to the respective Address Counter and Word Counter; in Control Mode 1, the content of the Address Register is transferred to the Address Counter and the Word Counter is cleared. The REINITIALIZE COUNTERS instruction allows a data transfer operation to be repeated without reloading the address and word count from the DATA lines.

Am2940 Timing

Various computations must be performed by the designer to determine how fast the Am2940 can be operated reliably in a given design. The exercises of this section demonstrate how these computations are performed.

Worst case A.C. characteristics, over the full temperature and voltage operating range should be used in these computations. Since, at the time of this writing, the Am2940 is still being characterized, only typical A.C. characteristics are available. These typicals are used here merely to demonstrate how the computations are performed; the designer must use worst-case characteristics. Figure 6 shows the characteristics of a Schottky register and a memory which are assumed for this exercise.

Figures 7A, B, and C show the typical cycle time calculations for the 16-bit Am2940 configuration. The typical delay along the longest path for any of the eight Am2940 instructions determines the typical cycle time. In each case, delays are computed from the LOW to HIGH transition of a clock through an entire microcycle to the next LOW to HIGH transition of a clock. The typical cycle time for a 16-bit Am2940 configuration is 64ns.

TABLE I. Am2940 INSTRUCTIONS

I_2	I_1	I_0	Octal Code	Function	Mnemonic	Control Mode	Word Reg.	Word Counter	Address Reg.	Address Counter	Control Register	Data D_0-D_7
L	L	L	0	WRITE CONTROL REGISTER	WRCR	0, 1, 2, 3	HOLD	HOLD	HOLD	HOLD	D_0-D_2→CR	INPUT
L	L	H	1	READ CONTROL REGISTER	RDCR	0, 1, 2, 3	HOLD	HOLD	HOLD	HOLD	HOLD	CR→D_0-D_2 (Note 1)
L	H	L	2	READ WORD COUNTER	RDWC	0, 1, 2, 3	HOLD	HOLD	HOLD	HOLD	HOLD	WC→D
L	H	H	3	READ ADDRESS COUNTER	RDAC	0, 1, 2, 3	HOLD	HOLD	HOLD	HOLD	HOLD	AC→D
H	L	L	4	REINITIALIZE COUNTERS	REIN	0, 2, 3	HOLD	WCR→WC	HOLD	AR→AC	HOLD	Z
						1	HOLD	ZERO→WC	HOLD	AR→AC	HOLD	Z
H	L	H	5	LOAD ADDRESS	LDAD	0, 1, 2, 3	HOLD	HOLD	D→AR	D→AC	HOLD	INPUT
H	H	L	6	LOAD WORD COUNT	LDWC	0, 2, 3	D→WR	D→WC	HOLD	HOLD	HOLD	INPUT
						1	D→WR	ZERO→WC	HOLD	HOLD	HOLD	INPUT
H	H	H	7	ENABLE COUNTERS	ENCT	0, 1, 3	HOLD	ENABLE COUNT	HOLD	ENABLE COUNT	HOLD	Z
						2	HOLD	HOLD	HOLD	ENABLE COUNT	HOLD	Z

CR = Control Reg.
AR = Address Reg.
AC = Address Counter
WCR = Word Count Reg.
WC = Word Counter
D = Data
L = LOW
H = HIGH
Z = High Impedance

Note 1:
Data Bits D_3-D_7 are high during this instruction.

	Min.	Typ.	Max.
Schottky Register			
Clock to Output Delay		9	15
Input Set-Up Time	5	2	
Memory			
Address Set-Up Time	20	10	

Figure 6. Assumed AC Characteristics.

Figure 8 shows the address output enable time computations. Since the Am2940 has an asynchronous address output enable control, the address output enable time may not be related to the Am2940 cycle time.

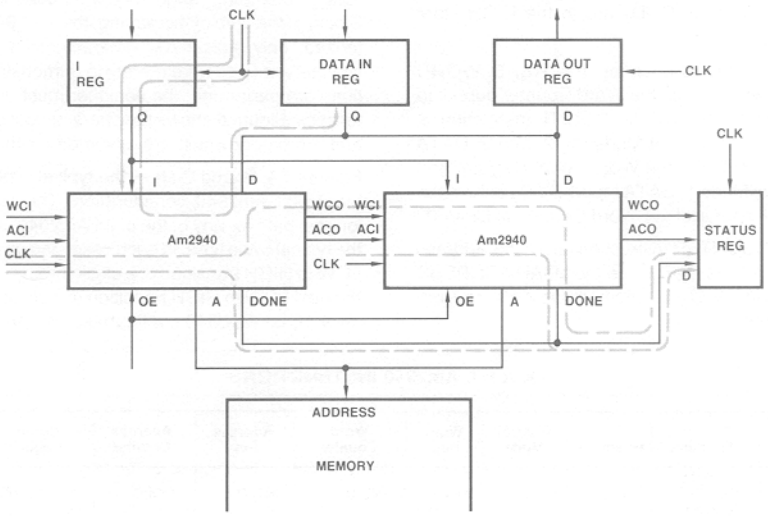
Figure 9 shows the typical cycle time calculation for an 8-bit Am2940 configuration. The path shown is the longest path and determines an 8-bit typical cycle time of 52ns.

The typical cycle time calculation for a 24-bit Am2940 configuration is shown in Figure 10. The path shown is the longest path and determines a 24-bit typical cycle time of 76ns.

Figure 11 is a summary of typical Am2940 cycle times for the 8, 16 and 24-bit configurations.

a)

CALCULATIONS FOR READ CONTROL REG, READ ADDRESS COUNTER, READ WORD COUNTER INSTRUCTIONS



DEVICE TYPE	DEVICE PATH	PATH 1	PATH 2	PATH 3	PATH 4
Schottky Reg.	CLK to Q	9	9		
2940	Inst. Set-Up	33			
2940	Inst. to Data		21		
Schottky Reg.	D Set-Up		2		
2940	CLK to DONE			50	
Schottky Reg.	D Set-Up			2	
2940	CLK to WCO				35
2940	WCI to DONE				27
Schottky Reg.	D Set-Up				2
TOTAL-ns		42	32	52	64

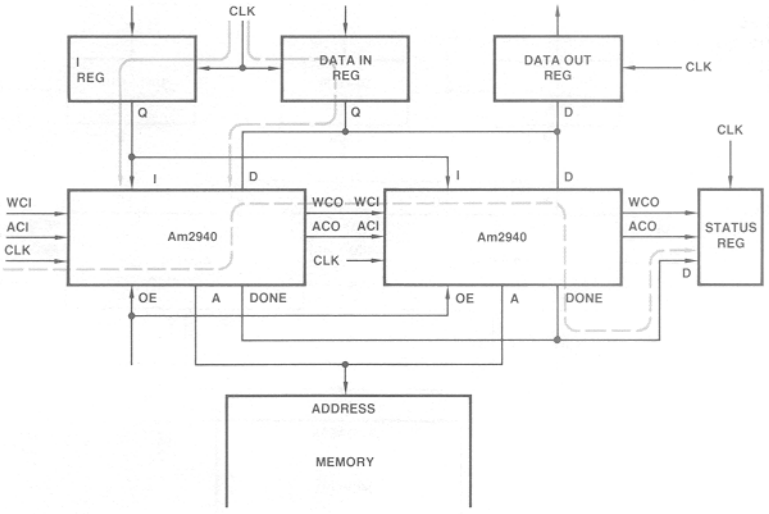
PATH 1
PATH 2
PATH 3
PATH 4

MPR-552

Figure 7. 16-Bit Typical Cycle Time Computations.

b)

CALCULATIONS FOR WRITE CONTROL REG, LOAD WORD COUNT, LOAD ADDRESS INSTRUCTIONS



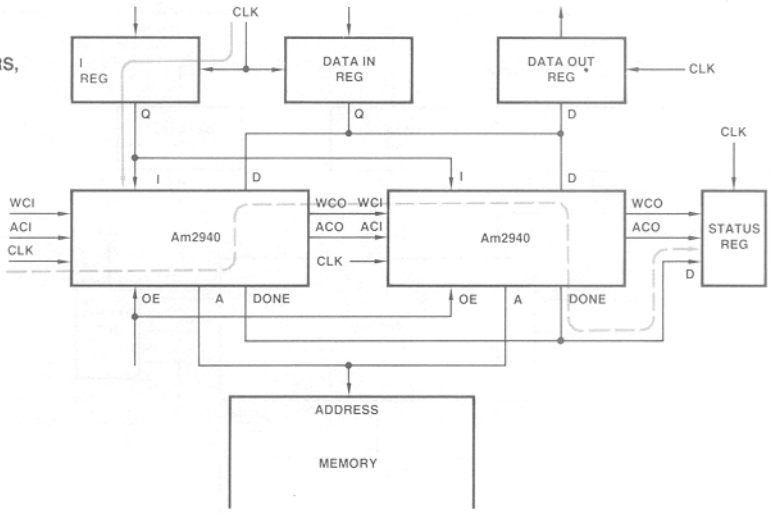
MPR-553

DEVICE TYPE	DEVICE PATH	PATH 1	PATH 2	PATH 3
Schottky Reg.	CLK to Q	9	9	
2940	Inst. Set-Up	33		
2940	Data Set-Up		13	
2940	CLK to WCO			35
2940	WCI to DONE			27
Schottky Reg.	D Set-Up			2
TOTAL-ns		42	22	64

PATH 1
PATH 2
PATH 3

c)

CALCULATIONS FOR REINITIALIZE COUNTERS, ENABLE COUNTERS INSTRUCTIONS



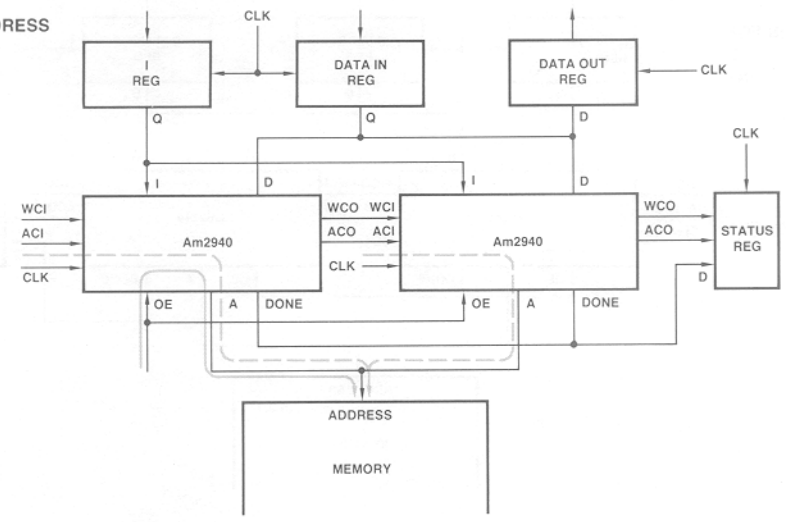
DEVICE TYPE	DEVICE PATH	PATH 1	PATH 2
Schottky Reg.	CLK to Q	9	
2940	Inst. Set-Up	33	
2940	CLK to WCO		35
2940	WCI to DONE		27
Schottky Reg.	D Set-Up		2
TOTAL-ns		42	64

PATH 1
PATH 2

MPR-554

Figure 7. 16-Bit Typical Cycle Time Computations. (Cont.)

CALCULATIONS FOR ENABLE MEMORY ADDRESS (ASYNCHRONOUS) INSTRUCTIONS

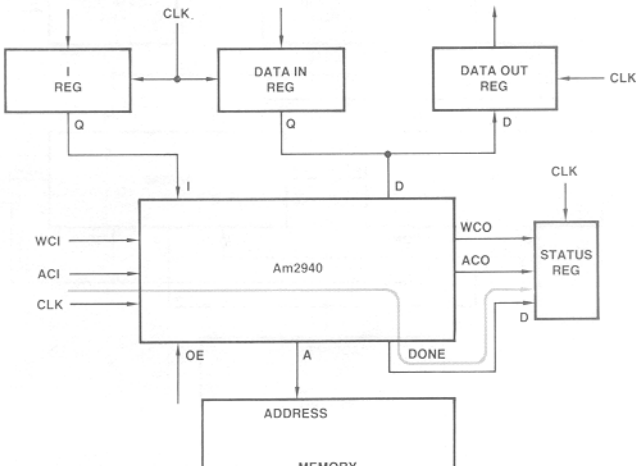


DEVICE TYPE	DEVICE PATH	PATH 1	PATH 2
2940	OE to A	19	
2940	CLK to A		35
Memory	ADR Set-Up	10	10
TOTAL-ns		29	45

PATH 1 ——————
PATH 2 — — — — —

MPR-555

Figure 8. Speed Computations.

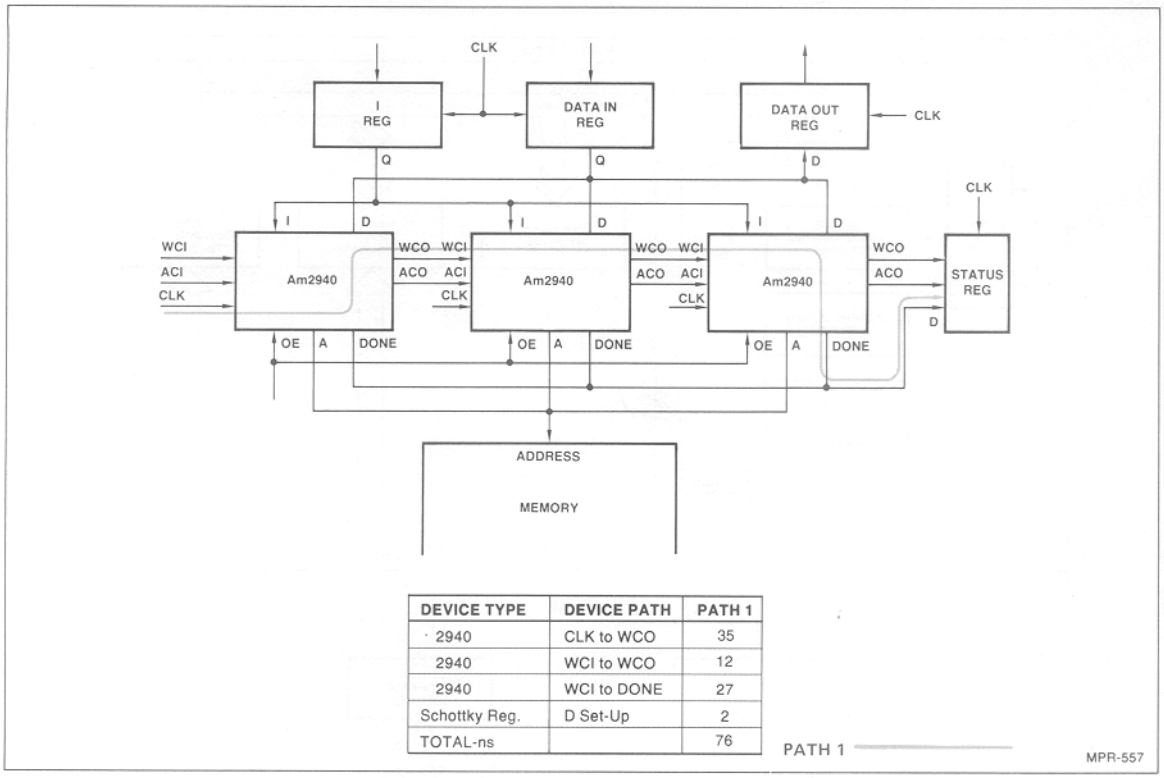


DEVICE TYPE	DEVICE PATH	PATH 1
2940	CLK to DONE	50
Schottky Reg.	D Set-Up	2
TOTAL-ns		52

PATH 1 ——————

MPR-556

Figure 9. 8-Bit Typical Cycle Time Computation.



DEVICE TYPE	DEVICE PATH	PATH 1
2940	CLK to WCO	35
2940	WCI to WCO	12
2940	WCI to DONE	27
Schottky Reg.	D Set-Up	2
TOTAL-ns		76

Figure 10. 24-Bit Typical Cycle Time Computation.

	Typical Cycle Time
8-Bit Configuration	52ns
16-Bit Configuration	64ns
24-Bit Configuration	76ns

Figure 11. Summary of Am2940 Cycle Times.

AN EXAMPLE DESIGN

The Am2940 is designed for use in high speed peripheral Controllers using DMA and provides the address and word count maintenance circuitry that is common to most. As indicated previously, DMA Control can be placed in each I/O Controller (Distributed DMA) or DMA Control for a number of I/O devices can be centralized in a separate unit.

Figure 12 shows a block diagram of a microprogrammed I/O Controller which is designed for use in a Distributed DMA system. The Am2910 Microprogram Sequencer, Microprogram Memory and the Microinstruction Register form the microprogram control portion of this I/O Controller. The Am2940 maintains the memory address and word count required for DMA operation. An internal three-state bus provides the communication path between the Microinstruction Register, the Am2917 Data Transceivers, the Am2940, the Am2901A Microprocessor, and the Device Interface Circuitry. The Address Line Control, Data Transfer Control and Mode Control functions of this DMA Controller are incorporated into the I/O Controller Microprogram and the Asynchronous Interface Control Circuitry. The I/O Controller Microprogram also controls the Am2940.

The Am2940 interconnections are shown in detail in Figure 13. Two Am2940s are cascaded to generate a sixteen-bit address. The Am2940 ADDRESS and DATA output current sink capability is 24mA over the commercial operating range. This allows the Am2940s to drive the System Address Bus and Internal Three-State Bus directly, thereby eliminating the need for separate bus drivers. Three bits in the Microinstruction Register provide the Am2940 Instruction Inputs, I_0-I_2. The microprogram clock is used to clock the Am2940s and, when the ENABLE COUNTERS instruction is applied, address and word counting is controlled by the CNT bit of the Microinstruction Register.

Asynchronous interface control circuitry generates System Bus control signals and enables the Am2940 Address onto the System Address Bus at the appropriate time. The open-collector DONE outputs are dot-anded and used as a test input to the Am2910 Microprogram Sequencer.

The I/O controller read operation is flowcharted in Figure 14. The CPU initializes the I/O controller by sending a read command, the starting memory address, the word count and any other parameters required to perform the operation. The I/O Controller then obtains a word of data from the I/O device and requests use of the system bus for a DMA transfer. When the bus is granted, the I/O Controller requests a memory data transfer. Upon receipt of the memory acknowledge signal, which indicates the memory trans-

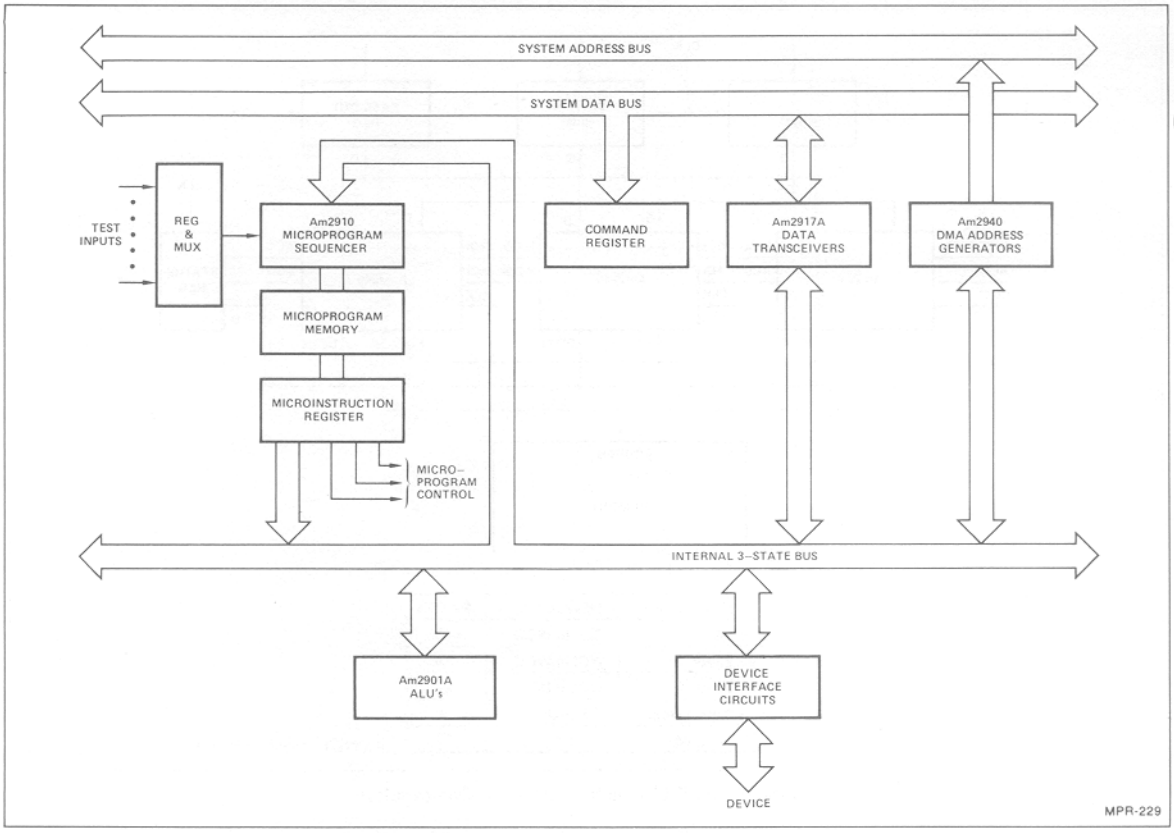


Figure 12. DMA Peripheral Controller Block Diagram.

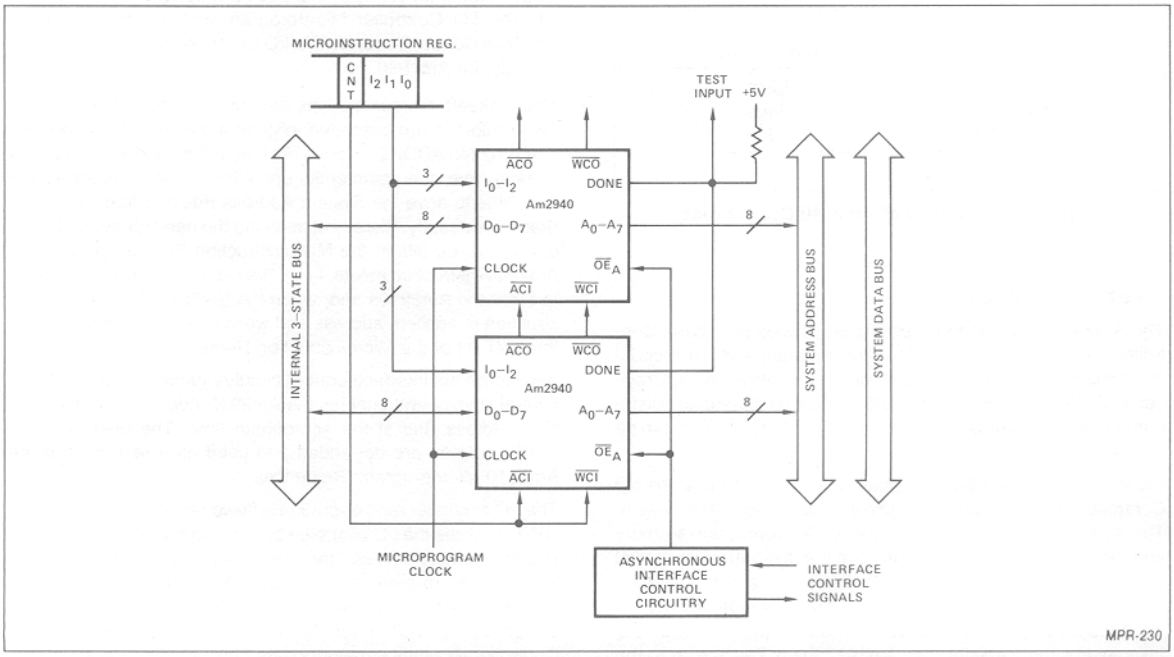


Figure 13. Am2940 Interconnections.

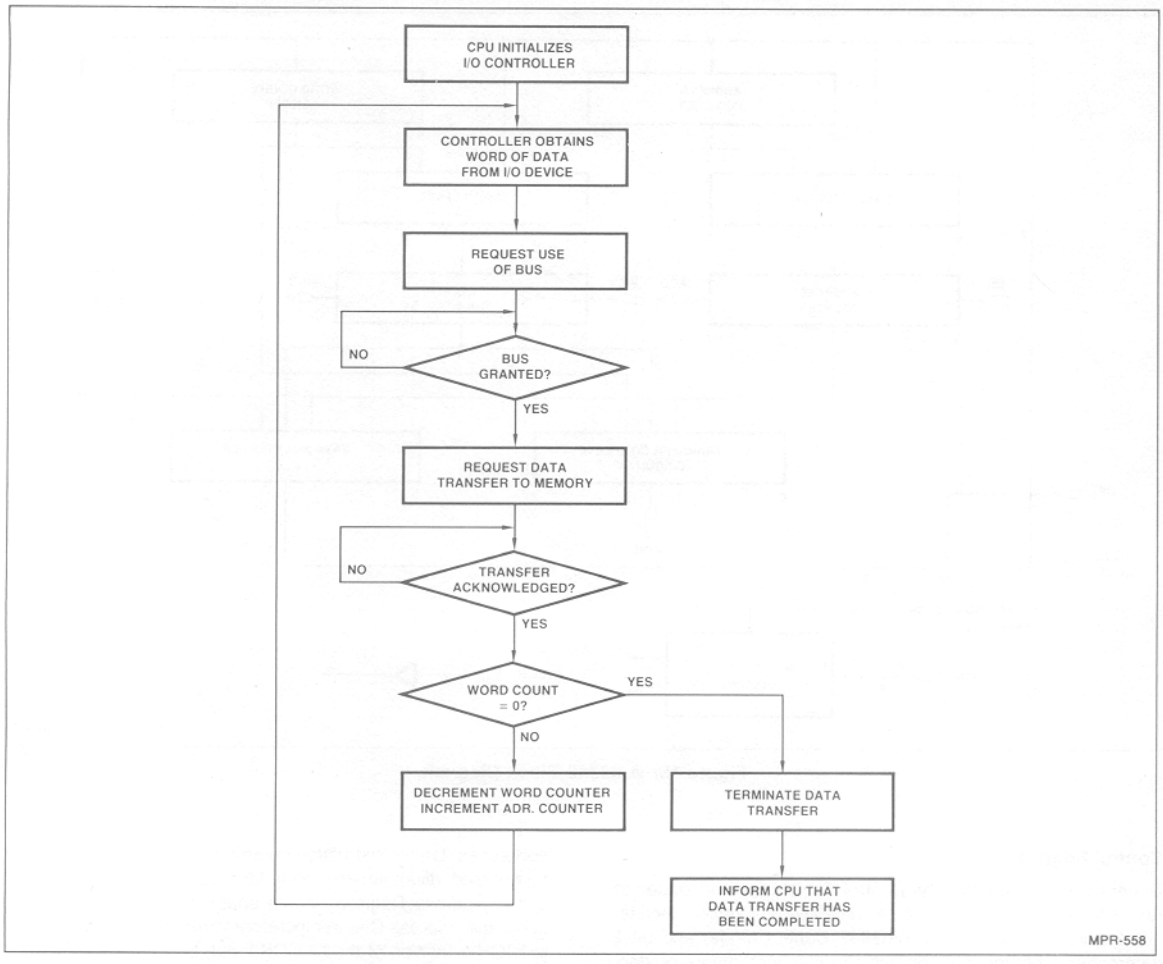


Figure 14. Read Control Flowchart.

fer is complete, the I/O Controller tests the word count. If the word count is not equal to zero, the word counter is decremented, the address counter is incremented and another data word is transferred. When the word count reaches zero, the I/O Controller terminates the data transfer and informs the CPU that the transfer has been completed.

THE Am2942 PROGRAMMABLE TIMER/COUNTER, DMA ADDRESS GENERATOR.

GENERAL DESCRIPTION

The Am2942, a 22-pin version of the Am2940, can be used as a high-speed DMA address Generator or Programmable Timer/Counter. It provides multiplexed Address and Data lines, for use with a common bus, and additional Instruction Input and Instruction Enable pins. The Am2942 executes 16 instructions; eight are the same as the Am2940 instructions, and eight instructions facilitate the use of the Am2942 as a Programmable Timer/Counter. The Instruction Enable input allows the sharing of the Am2942 instruction field with other devices.

When used as a Timer/Counter, the Am2942 provides two independent, programmable, eight-bit, up-down counters in a 22-pin package. The two on-chip counters can be cascaded to form a single chip, 16-bit counter. Also, any number of chips can be cascaded – for example three cascaded Am2942s form a 48-bit timer/counter.

Reinitialization instructions provide the capability to reinitialize the counters from on-chip registers. Am2942 Programmable Control Modes, identical to those of the Am2940, offer four different types of programmable control.

Am2942 ARCHITECTURE

As shown in the Block Diagram, the Am2942 consists of the following:

- A three-bit Control Register.
- An eight-bit Address Counter with input multiplexer.
- An eight-bit Address Register.
- An eight-bit Word Counter with input multiplexer.
- An eight-bit Word Count Register.
- Transfer complete circuitry.
- An eight-bit wide data multiplexer with three-state output buffers.
- An instruction decoder.

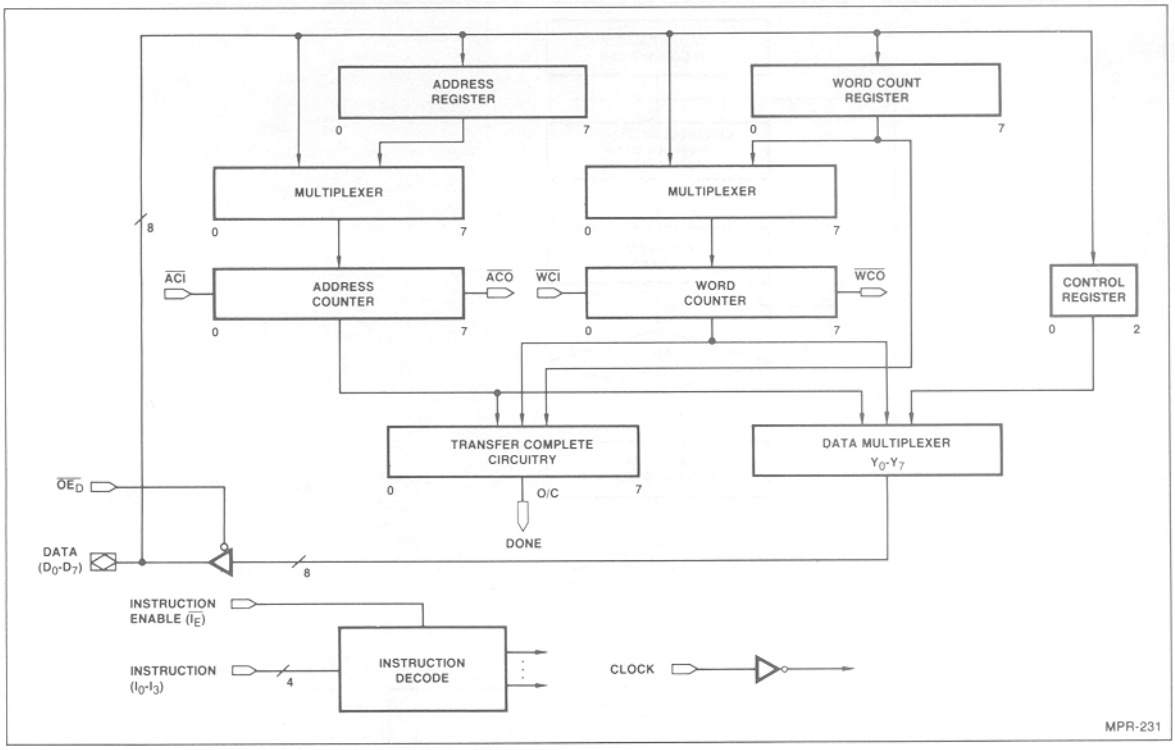


Figure 15. Am2942 Block Diagram.

Control Register

Under instruction control, the Control Register can be loaded or read from the bidirectional DATA lines, D_0-D_7. Control Register bits 0 and 1 determine the Am2942 Control Mode, and bit 2 determines whether the Address Counter increments or decrements. Figure 16 defines the Control Register format.

Address Counter

The Address Counter, which provides the current memory address, is an eight-bit, binary, up/down counter with full look-ahead carry generation. The Address Carry input ($\overline{ACI}$) and Address Carry Output ($\overline{ACO}$) allow cascading to accommodate larger addresses. Under instruction control, the Address Counter can be enabled, disabled, and loaded from the DATA inputs, D_0-D_7, or the Address Register. When enabled and the ACI input is LOW, the Address Counter increments/decrements on the LOW to HIGH transition of the CLOCK input, CP.

Address Register

The eight-bit Address Register saves the initial address so that it can be restored later in order to repeat a transfer operation. When the LOAD ADDRESS instruction is executed, the Address Register and Address Counter are simultaneously loaded from the DATA inputs, D_0-D_7.

Control Register

CR_2	CR_1	CR_0

CR_1	CR_0	Control Mode Number	Control Mode Type	Word Counter	Done Output Signal	
					$\overline{WCI}$ = LOW	$\overline{WCI}$ = HIGH
L	L	0	Word Count Equals Zero	Decrement	HIGH when Word Counter = 1	HIGH when Word Counter = 0
L	H	1	Word Count Compare	Increment	HIGH when Word Counter + 1 = Word Count Register	HIGH when Word Counter = Word Count Register
H	L	2	Address Compare	Decrement	HIGH when Word Counter = Address Counter	
H	H	3	Word Counter Carry Out	Increment	Always LOW	

CR_2	Address Counter
L	Increment
H	Decrement

H = HIGH
L = LOW

Figure 16. Control Register Format Definition.

Word Counter And Word Count Register

The Word Counter and Word Count Register, which maintain and save a word count, are similar in structure and operation to the Address Counter and Address Register, with the exception that the Word Counter increments in Control Modes 1 and 3 and decrements in Control Modes 0 and 2. The LOAD WORD COUNT instruction simultaneously loads the Word Counter and Word Count Register.

Transfer Complete Circuitry

The Transfer Complete Circuitry is a combinational logic network which detects the completion of the data transfer operation in three Control Modes and generates the DONE output signal. The DONE signal is an open-collector output, which can be dot-anded between chips.

Data Multiplexer

The Data Multiplexer is an eight-bit wide, three-input multiplexer which allows the Address Counter, Word Counter and Control Register to be read at DATA lines D_0-D_7. The Data Multiplexer output, Y_0-Y_7, is enabled onto DATA lines D_0-D_7 if, and only if, the Output Enable input, $\overline{OE}_D$, is LOW. (Refer to Figure 17.)

$\overline{OE}_D$	D_0-D_7
L	DATA MULTIPLEXER OUTPUT, Y_0-Y_7
H	HIGH Z

Figure 17. Data Bus Output Enable Function.

Instruction Decoder

The Instruction Decoder generates required internal control signals as a function of the INSTRUCTION inputs, I_0-I_3 Control Register bits 0 and 1, and the INSTRUCTION ENABLE input, I_E.

Clock

The clock input, CP, is used to clock the Address Register, Address Counter, Word Count Register, Word Counter, and Control Register, all on the LOW to HIGH transition of the CP signal.

Am2942 CONTROL MODES

Control Mode 0 – Word Count Equals Zero Mode

In this mode, the LOAD WORD COUNT instruction loads the word count into the Word Count Register and Word Counter. When the Word Counter is enabled and the Word Counter Carry-in, $\overline{WCI}$, is LOW, the Word Counter decrements on the LOW to HIGH transition of the CLOCK input, CP. Figure 16 specifies when the DONE signal is generated in this mode.

Control Mode 1 – Word Count Compare Mode

In this mode the LOAD WORD COUNT instruction loads the word count into the Word Count Register and clears the Word Counter. When the Word Counter is enabled and the Word Counter Carry-in, $\overline{WCI}$, is LOW, the Word Counter increments on the LOW to HIGH transition of the clock input, CP. Figure 16 specifies when the DONE signal is generated.

Control Mode 2 – Address Compare Mode

In this mode, only an initial and final memory address need to be specified. The initial Memory Address is loaded into the Address Register and Address Counter and the final memory address is loaded into the Word Count Register and Word Counter. The Word Counter serves as a holding register for the final memory address. When the Address Counter is enabled and the $\overline{ACI}$ input is LOW, the Address Counter increments or decrements (depending on Control Register bit 2) on the LOW to HIGH transition of the CLOCK input, CP. The Transfer Complete Circuitry compares the Address Counter with the Word Counter and generates the DONE signal during the last word transfer, i.e., when the Address Counter equals the Word Counter.

Control Mode 3 – Word Counter Carry Out Mode

For this mode of operation, the user can load the Word Count Register and Word Counter with the two's complement of the number of data words to be transferred. When the Word Counter is enabled and the $\overline{WCI}$ input is LOW, the Word Counter increments on the LOW to HIGH transition of the CLOCK input, CP. A Word Counter Carry Out signal, $\overline{WCO}$, indicates the last data word is being transferred. The DONE signal is not required in this mode and, therefore, is always LOW.

Am2942 INSTRUCTIONS

The Am2942 instruction set consists of sixteen instructions. Eight are DMA instructions and are the same as the Am2940 instructions. The remaining eight instructions are designed to facilitate the use of the Am2942 as a Programmable Timer/Counter. Figures 18 and 19 define the Am2942 Instructions.

Instructions 0-7 are DMA instructions. The WRITE CONTROL REGISTER instruction writes DATA input D_0-D_2 into the Control Register; DATA inputs D_3-D_7 are "don't care" inputs for this instruction. The READ CONTROL REGISTER instruction gates the Control Register to Data Multiplexer outputs Y_0-Y_2. Outputs Y_3-Y_7 are HIGH during this instruction.

The Word Counter can be read using the READ WORD COUNTER instruction, which gates the Word Counter to Data Multiplexer outputs, Y_0-Y_7. The LOAD WORD COUNT instruction is Control Mode dependent. In Control Modes 0, 2 and 3, DATA inputs D_0-D_7 are written into both the Word Count Register and Word Counter. In Control Mode 1, DATA inputs D_0-D_7 are written into the Word Count Register and the Word Counter is cleared.

The READ ADDRESS COUNTER instruction gates the Address Counter to Data Multiplexer outputs, Y_0-Y_7, and the LOAD ADDRESS instruction writes DATA inputs D_0-D_7 into both the Address Register and Address Counter.

In Control Modes 0, 1, and 3, the ENABLE COUNTERS instruction enables both the Address and Word Counters; in Control Mode 2, the Address Counter is enabled and the Word Counter holds its contents. When enabled and the carry input is active, the counters increment on the LOW to HIGH transition of the CLOCK input, CP. Thus, with this instruction applied, counting can be controlled by the carry inputs.

The REINITIALIZE COUNTERS instruction also is Control Mode dependent. In Control Modes 0, 2, and 3, the contents of the Address Register and Word Count Register are transferred to the respective Address Counter and Word Counter; in Control Mode 1, the content of the Address Register is transferred to the Address Counter and the Word Counter is cleared. The REINITIALIZE COUNTERS instruction allows a data transfer operation to be repeated without reloading the address and word count from the DATA lines.

$\overline{I_E}$	I_3	I_2	I_1	I_0	HEX CODE		
0	0	0	0	0	0	WRITE CONTROL REGISTER	DMA INSTRUCTIONS
0	0	0	0	1	1	READ CONTROL REGISTER	
0	0	0	1	0	2	READ WORD COUNTER	
0	0	0	1	1	3	READ ADDRESS COUNTER	
0	0	1	0	0	4	REINITIALIZE COUNTERS	
0	0	1	0	1	5	LOAD ADDRESS	
0	0	1	1	0	6	LOAD WORD COUNT	
0	0	1	1	1	7	ENABLE COUNTERS	
1	0	X	X	X	0-7	INSTRUCTION DISABLE	
0	1	0	0	0	8	WRITE CONTROL REGISTER, T/C	TIMER/COUNTER INSTRUCTIONS
0	1	0	0	1	9	REINITIALIZE ADDRESS COUNTER	
0	1	0	1	0	A	READ WORD COUNTER, T/C	
0	1	0	1	1	B	READ ADDRESS COUNTER, T/C	
0	1	1	0	0	C	REINITIALIZE ADDRESS & WORD COUNTERS	
0	1	1	0	1	D	LOAD ADDRESS, T/C	
0	1	1	1	0	E	LOAD WORD COUNT, T/C	
0	1	1	1	1	F	REINITIALIZE WORD COUNTER	
1	1	X	X	X	8-F	INSTRUCTION DISABLE, T/C	

0 = LOW 1 = HIGH X = DON'T CARE

Notes: 1. When I_3 is tied LOW, the Am2942 acts as a DMA circuit: When I_3 is tied HIGH, the Am2942 acts as a Timer/Counter circuit.
2. Am2942 instructions 0 through 7 are the same as Am2940 instructions.

Figure 18. Am2942 Instructions

When $\overline{I_E}$ is HIGH, Instruction inputs, I_0-I_2, are disabled. If I_3 is LOW, the function performed is identical to that of the ENABLE COUNTERS instruction. Thus, counting can be controlled by the carry inputs with the ENABLE COUNTERS instruction applied or with Instruction Inputs I_0-I_2 disabled.

Instructions 8-F facilitate the use of the Am2942 as a Programmable Timer/Counter. They differ from instructions 0-7 in that they provide independent control of the Address Counter, Word Counter and Control Register.

The WRITE CONTROL REGISTER, T/C instruction writes DATA input D_0-D_2 into the Control Register. DATA inputs D_3-D_7 are "don't care" inputs for this instruction. The Address and Word Counters are enabled, and the Control Register contents appear at the Data Multiplexer output.

The REINITIALIZE ADDRESS COUNTER instruction allows the independent reinitialization of the Address Counter. The Word Counter is enabled and the contents of the Address Counter appear at the Data Multiplexer output.

The Word Counter can be read, using the READ WORD COUNTER, T/C instruction. Both counters are enabled when this instruction is executed.

When the READ ADDRESS COUNTER, T/C instruction is executed, both counters are enabled and the address counter contents appear at the Data Multiplexer output.

The REINITIALIZE ADDRESS and WORD COUNTERS instruction provides the capability to reinitialize both counters at the same time. The Address Counter contents appear at the Data Multiplexer output.

DATA inputs D_0-D_7 are loaded into both the Address Register and Counter when the LOAD ADDRESS, T/C instruction is executed. The Word Counter is enabled and its contents appear at the Data Multiplexer output.

The LOAD WORD COUNT, T/C instruction is identical to the LOAD WORD COUNT instruction with the exception that Address Counter is enabled.

The Word Counter can be independently reinitialized using the REINITIALIZE WORD COUNTER instruction. The Address Counter is enabled and the Word Counter contents appear at the Data Multiplexer output.

When the $\overline{I_E}$ input is HIGH, Instruction inputs, I_0-I_2, are disabled. The function performed when I_3 is HIGH is identical to that performed when I_3 is LOW, with the exception that the Word Counter contents appear at the Data Multiplexer output.

EXAMPLE DESIGNS

Figure 20 shows an Am2942 used as two independent, programmable eight-bit timer/counters. In this example, an Am2910 Microprogram Sequencer provides an address to Am29775 512 x 8 Registered PROMs. The on-chip PROM output register is used as the Microinstruction Register.

The Am2942 Instruction input, I_3 is tied HIGH to select the eight Timer/Counter instructions. The $\overline{I_E}$, I_0-I_2, and $\overline{OE_D}$ inputs are provided by the microinstruction, and the D_0-D_7 data lines are connected to a common Data Bus. GATE WC and GATE AC are separate enable controls for the respective Word Counter and Address Counter. The DONE, $\overline{ACO}$ and $\overline{WCO}$ output signals indicate that a pre-programmed time or count has been reached.

$\overline{I_E}$	$I_3I_2I_1I_0$ (Hex)	Function	Mnemonic	Control Mode	Word Reg.	Word Counter	Adr. Reg.	Adr. Counter	Control Reg.	Data Multiplexer Output
L	0	WRITE CONTROL REGISTER	WRCR	0, 1, 2, 3	HOLD	HOLD	HOLD	HOLD	$D_{0-2} \rightarrow CR$	FORCED HIGH
L	1	READ CONTROL REGISTER	RDCR	0, 1, 2, 3	HOLD	HOLD	HOLD	HOLD	HOLD	CONTROL REG.
L	2	READ WORD COUNTER	RDWC	0, 1, 2, 3	HOLD	HOLD	HOLD	HOLD	HOLD	WORD COUNTER
L	3	READ ADDRESS COUNTER	RDAC	0, 1, 2, 3	HOLD	HOLD	HOLD	HOLD	HOLD	ADR. COUNTER
L	4	REINITIALIZE COUNTERS	REIN	0, 2, 3	HOLD	WR → WC	HOLD	AR → AC	HOLD	ADR. CNTR.
				1	HOLD	ZERO → WC	HOLD	AR → AC	HOLD	ADR. CNTR.
L	5	LOAD ADDRESS	LDAD	0, 1, 2, 3	HOLD	HOLD	D → AR	D → AC	HOLD	WORD COUNTER
L	6	LOAD WORD COUNT	LDWC	0, 2, 3	D → WR	D → WC	HOLD	HOLD	HOLD	FORCED HIGH
				1	D → WR	ZERO → WC	HOLD	HOLD	HOLD	FORCED HIGH
L	7	ENABLE COUNTERS	ENCT	0, 1, 3	HOLD	ENABLE	HOLD	ENABLE	HOLD	ADR. CNTR.
				2	HOLD	HOLD	HOLD	ENABLE	HOLD	ADR. CNTR.
H	0-7	INSTRUCTION DISABLE	–	0, 1, 3	HOLD	ENABLE	HOLD	ENABLE	HOLD	ADR. CNTR.
				2	HOLD	HOLD	HOLD	ENABLE	HOLD	ADR. CNTR.
L	8	WRITE CONTROL REGISTER, T/C	WCRT	0, 1, 2, 3	HOLD	ENABLE	HOLD	ENABLE	$D_{0-2} \rightarrow CR$	CONTROL REG.
L	9	REINITIALIZE ADR. COUNTER	REAC	0, 1, 2, 3	HOLD	ENABLE	HOLD	AR → AC	HOLD	ADR. COUNTER
L	A	READ WORD COUNTER, TC	RWCT	0, 1, 2, 3	HOLD	ENABLE	HOLD	ENABLE	HOLD	WORD COUNTER
L	B	READ ADDRESS COUNTER, T/C	RACT	0, 1, 2, 3	HOLD	ENABLE	HOLD	ENABLE	HOLD	ADR. COUNTER
L	C	REINITIALIZE ADDRESS AND WORD COUNTERS	RAWC	0, 2, 3	HOLD	WR → WC	HOLD	AR → AC	HOLD	ADR. CNTR.
				1	HOLD	ZERO → WC	HOLD	AR → AC	HOLD	ADR. CNTR.
L	D	LOAD ADDRESS, T/C	LDAT	0, 1, 2, 3	HOLD	ENABLE	D → AR	D → AC	HOLD	WORD COUNTER
L	E	LOAD WORD COUNT, T/C	LWCT	0, 2, 3	D → WR	D → WC	HOLD	ENABLE	HOLD	FORCED HIGH
				1	D → WR	ZERO → WC	HOLD	ENABLE	HOLD	FORCED HIGH
L	F	REINITIALIZE WORD COUNTER	REWC	0, 2, 3	HOLD	WR → WC	HOLD	ENABLE	HOLD	WD. CNTR.
				1	HOLD	ZERO → WC	HOLD	ENABLE	HOLD	WD. CNTR.
H	8-F	INSTRUCTION DISABLE, T/C	–	0, 1, 3	HOLD	ENABLE	HOLD	ENABLE	HOLD	WD. CNTR.
				2	HOLD	HOLD	HOLD	ENABLE	HOLD	WD. CNTR.

WR = WORD REGISTER
WC = WORD COUNTER
AR = ADDRESS REGISTER
AC = ADDRESS COUNTER
CR = CONTROL REGISTER
D = DATA

Figure 19. Am2942 Function Table.

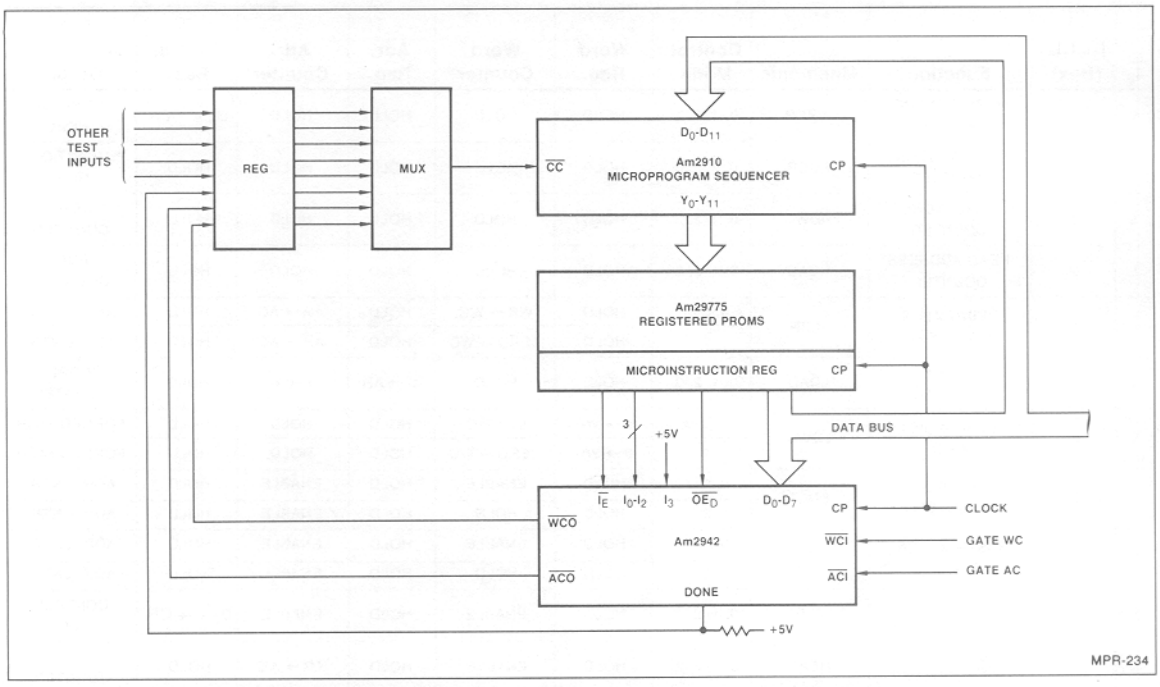


Figure 20. Two 8-Bit Programmable Counters/Timers in a 22-Pin Package.

Figure 21 shows an Am2942 used as a single 16-bit, programmable timer/counter. In this example, the Word Counter carry-out, $\overline{WCO}$, is connected to the Address Counter carry-in, $\overline{ACI}$, to form a single 16-bit counter which is enabled by the GATE signal.

Figure 22 shows two Am2942s cascaded to form a 32-bit programmable timer/counter. The two Word Counters form the low order 16 bits, and the two Address Counters form the high order bits. This allows the timer/counter to be loaded and read 16 bits at a time.

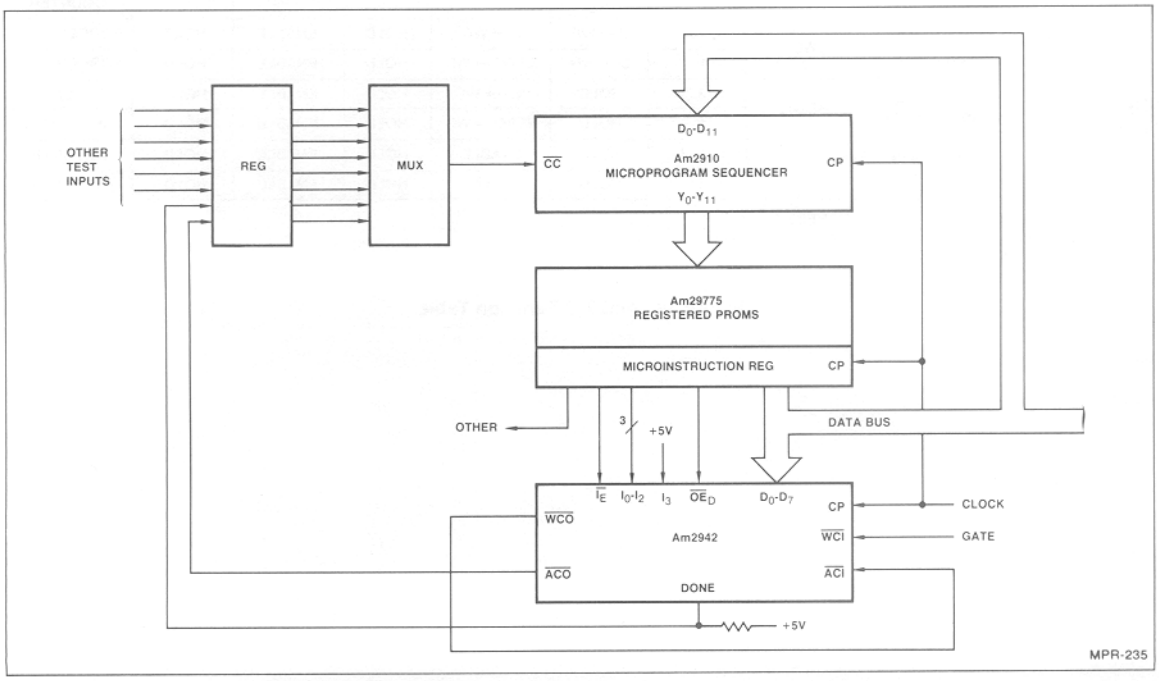


Figure 21. 16-Bit Programmable Counter/Timer Using a Single Am2942.

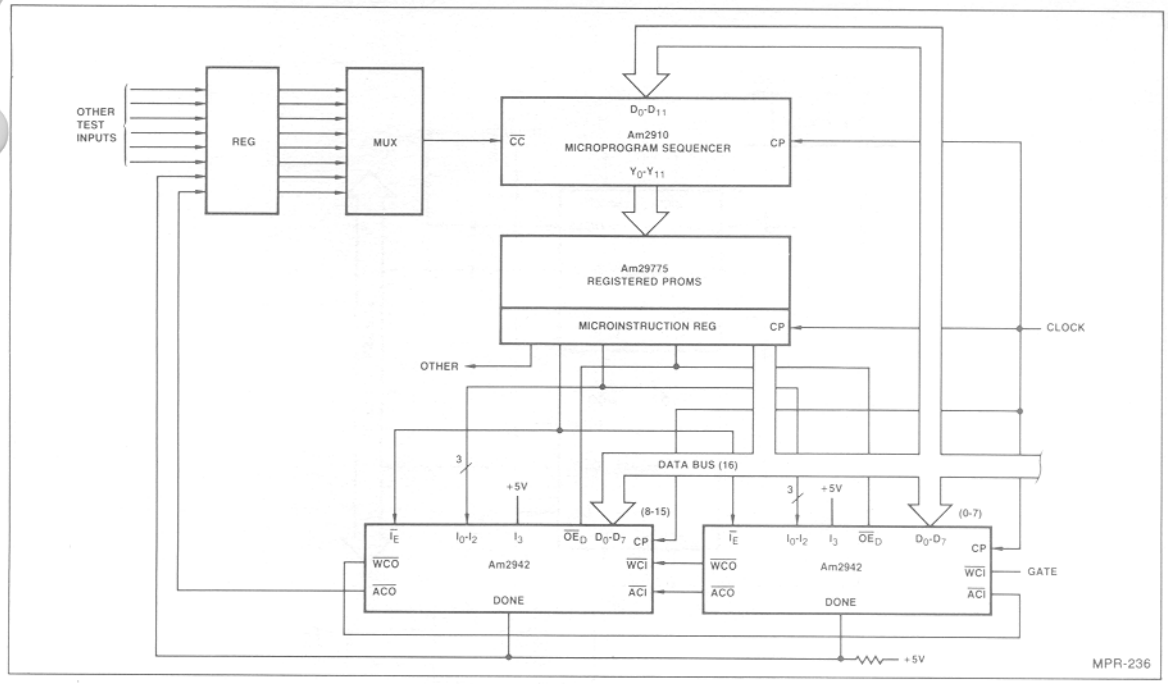


Figure 22. 32-Bit Programmable Counter/Timer Using Two Am2942s.

In Figure 23, two Am2942s are shown cascaded to form dual 16-bit counters/timers. GATE WC and GATE AC are separate enable controls for the respective Word Counter and Address Counter. Using the 16-bit Data Bus, each 16-bit counter can be loaded or read in parallel.

Figure 24 shows two Am2942s used as DMA address Generators on a common DATA/ADDRESS bus. The common bus allows the use of the Am2942 multiplexed data and address pins, D_0-D_7. The Am2942 is in a 22 pin package whereas the Am2940, which has separate address and data pins, requires a 28 pin package.

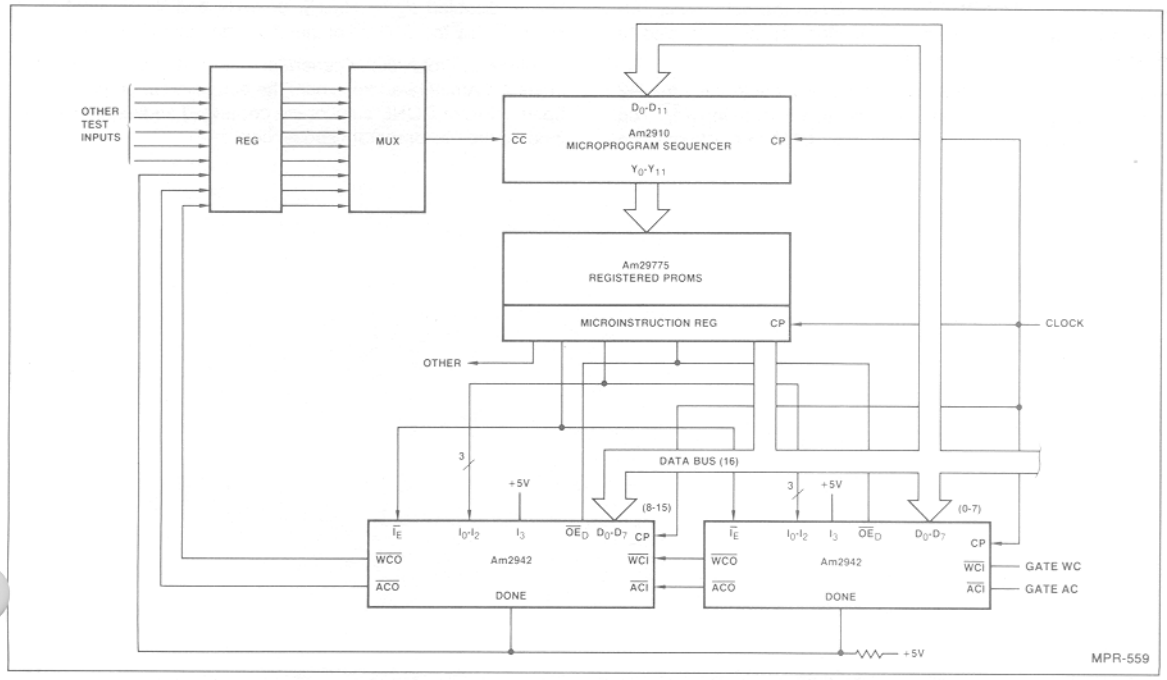


Figure 23. Dual 16-Bit Programmable Counters/Timers.

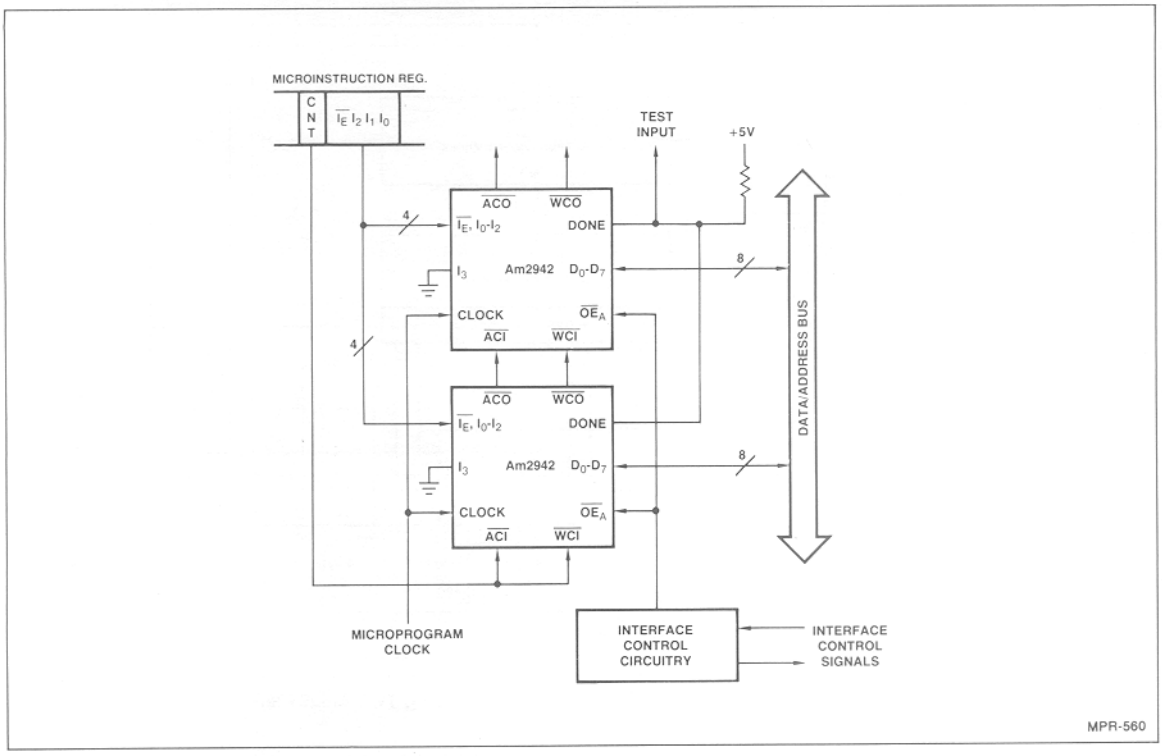


Figure 24. Am2942s Used as DMA Address Generator on Common Bus.

In this example the Am2942 Address Counter, Word Counter and Control Register are loaded and read directly from the CPU via the DATA/ADDRESS bus. Since the bus carries addresses as well as data, the D_0-D_7 pins can be used also to enable the address onto the bus.

Four bits in the Microinstruction Register provide the Am2942 Instruction Inputs, I_0-I_2 and the Instruction Enable input $\overline{I_E}$. The I_4 input is tied LOW, selecting the eight DMA instructions. The microprogram clock is used to clock the Am2942s, and when the ENABLE COUNTERS instruction is applied or the instruction is disabled ($\overline{I_E}$ = HIGH), address and word counting is controlled by the CNT bit of the Microinstruction Register.

Interface control circuitry generates bus control signals and enables the Am2942 address onto the bus at the appropriate. The open-collector DONE outputs are dot-anded and used as a test input to the microprogram sequencer.

ADVANCED MICRO DEVICES, INC.
901 Thompson Place
Sunnyvale
California 94086
(408) 732-2400
TWX: 910-339-928
TELEX: 34-6306
TOLL FREE
(800) 538-8450

11-78