MOTOROLA
SEMICONDUCTOR
APPLICATION NOTE

Order this document by AN944/D

AN944

MC68020 AND MC68881 PLATFORM BOARD FOR EVALUATION IN A 16-BIT SYSTEM

INTRODUCTION

This application note describes the design of a platform board that contains an MC68020 32-bit microprocessor (with an on-chip cache) and an MC68881 floating-point coprocessor (FPCP). The platform board can be inserted into an MC68000 or MC68010 system to increase the performance of the existing 16-bit system with its more advanced 32-bit microprocessor unit (MPU) and specialized floating-point hardware. When plugged into the MPU socket of the target system, the board draws power, ground, and clock through the socket and runs bus cycles compatible with the host system. This platform board design also can be used to evaluate the programmer's model and instruction set of the MC68020 and MC68881.

The platform board uses the following components:

1 — MC68020 32-bit MPU
1 — MC68881 FPCP
4 — PAL16R4A-15 logic devices
6 — 4.7K pullup resistors

This design emphasizes a low chip count by using programmable array logic devices. The entire system also can be implemented with only small-scale or medium-scale integrated parts.

This application note, detailing the configuration of an MC68020 to simulate an MC68000 while supporting a coprocessor interface, demonstrates the following interface design principles:

- Dynamic bus sizing considerations
- Bus cycle timing modification
- Bus arbitration
- M6800 family device communication
- Coprocessor selection
- Interrupt processing

The remaining paragraphs describe the platform board design and analyze the performance differences between the MC68000 MPU and the MC68020 MPU in a 16-bit system running the EDN and Berkeley benchmark routines.

HARDWARE ARCHITECTURE

This section specifies the logic required for the platform board. It examines some design considerations, and then discusses dynamic bus sizing, bus cycle timing modifications, M6800 family device communication, interrupt processing, bus arbitration, and coprocessor selection.

HARDWARE CONSIDERATIONS

The block diagram of the platform board is shown in Figure 1.

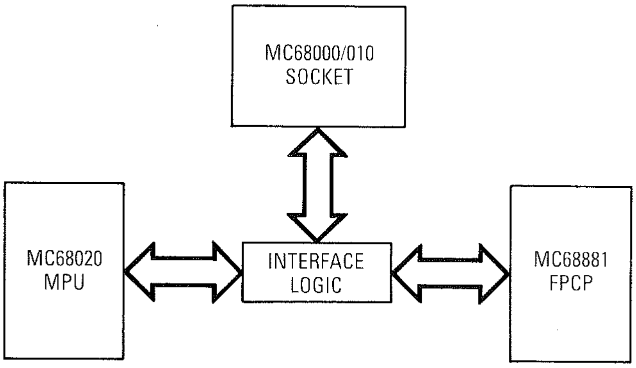


Figure 1. Platform Board Block Diagram

The MC68000 has an asynchronous bus structure. The microprocessor asserts address strobe ($\overline{AS}$) to begin bus cycles; the peripheral or memory device being addressed asserts data transfer acknowledge ($\overline{DTACK}$) to end them. The MC68020 has an asynchronous bus structure that follows the same protocol as the MC68000. However, its dynamic bus sizing capabilities allow bus cycles to end with the assertion of two data transfer and size acknowledge signals ($\overline{DSACK1}$ and $\overline{DSACK0}$) rather than the one $\overline{DTACK}$ of the MC68000. This difference and other timing constraints are discussed later.

The MC68000 has built-in logic that lets it communicate directly with any M6800 family device. The MC68020 does not include this interface. It must be supported by the interface logic to correctly simulate the MC68000.

Communication between the MC68020 and MC68881 must be transparent to the host system because the MC68000 does not have a coprocessor interface and most MC68000 systems do not handle coprocessors.

To support these requirements, the platform board needs a small amount of logic to delay, create, and hold signals passed between the MC68020, the MC68000 system, and the MC68881. A synchronous decade counter also is required to support the M6800 family interface. This state machine is part of the interface circuitry, as is shown in Figure 2.

MOTOROLA

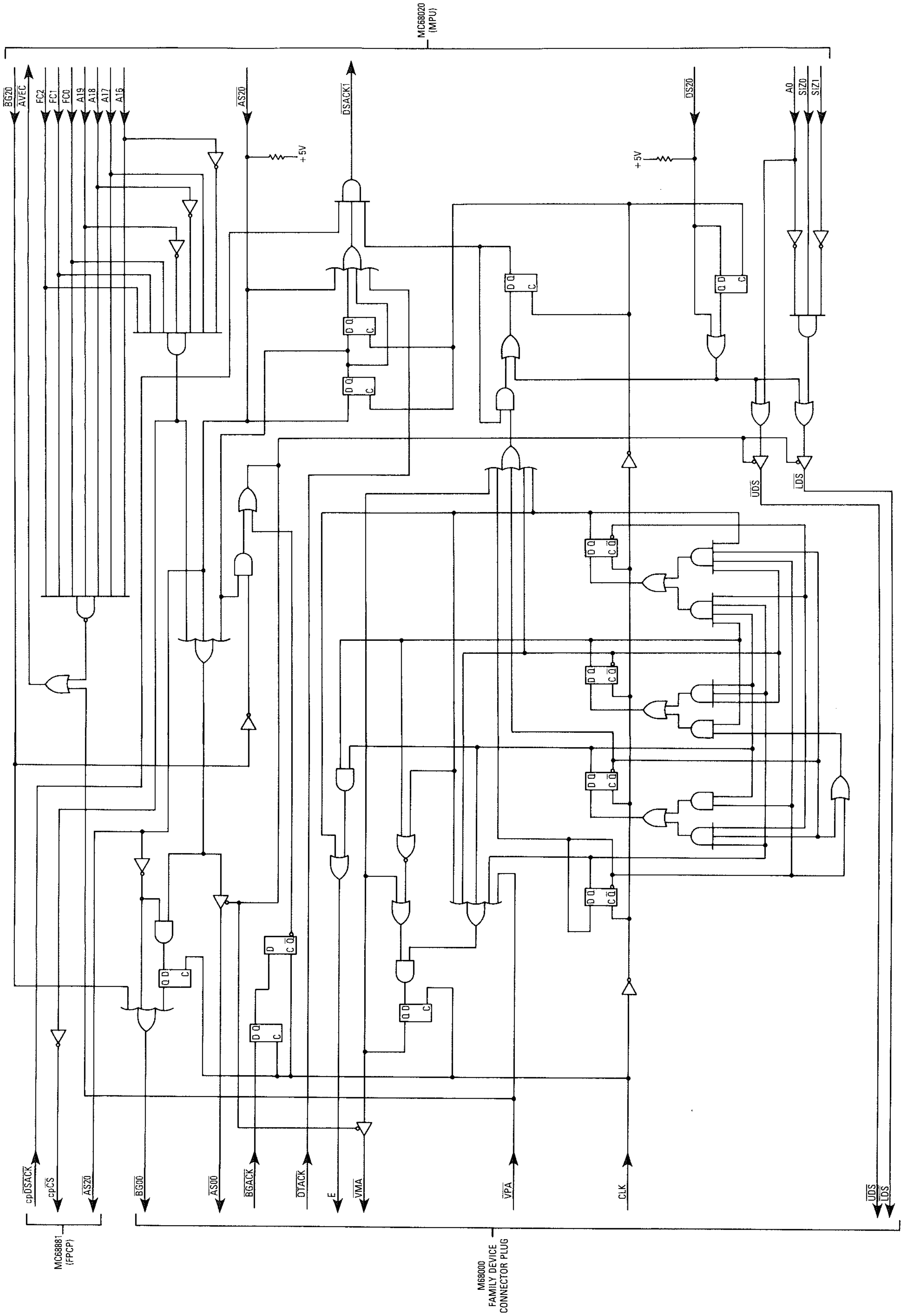


Figure 2. MC68020 and MC68881 Platform Board Inteface Logic

DYNAMIC BUS SIZING

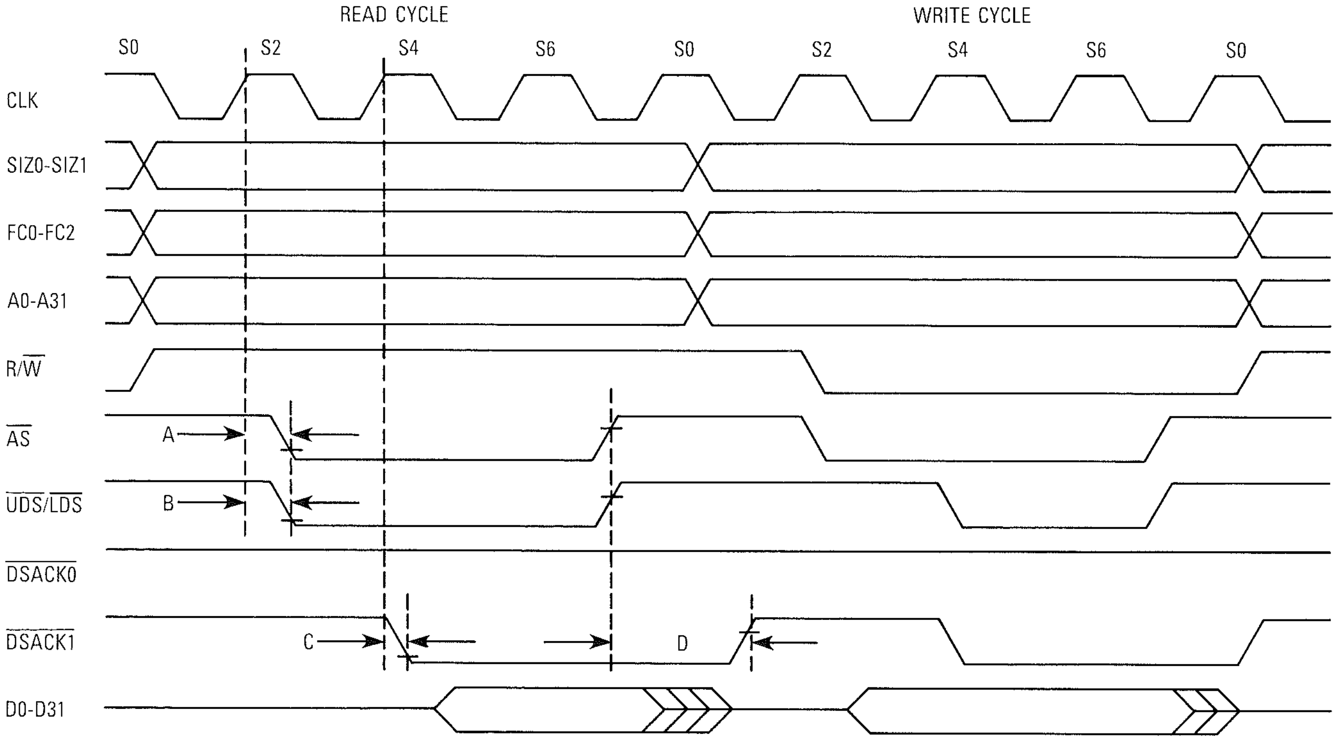
The MC68020 can transfer byte, word, and long word operands using its dynamic bus sizing capabilities. The data transfer and size acknowledge signals (DSACK1 and DSACK0) returned to the processor indicate the size of the addressed peripheral port, allowing the MC68020 to correctly configure the data bus. Only one of these signals is needed when addressing the host system because an MC68000 or MC68010 always uses a 16-bit port (although single bytes can be transferred through the port). In this design, the DSACK0 pin on the MC68020 MPU is tied to V_{CC} (inactive) through a pullup resistor. Since the DSACK0 pin is still essential for communication between the MC68020 and its coprocessor, the DSACK0 output of the MC68881 is wired to the node connecting the DSACK0 input on the MPU to the pullup resistor.

The bus sizing capabilities of the MC68020 also include the transfer size pins (SIZ1 and SIZ0) and the two least significant address lines (A1 and A0). These signals generate the address and data strobes as discussed in the following paragraphs.

BUS CYCLE TIMING MODIFICATIONS

As previously mentioned, the bus cycle timing of the MC68000 and the MC68020 are slightly different. In a no wait-state environment, an MC68000 bus cycle takes four clock periods (eight high and low clock states). In a no wait-state system, the MC68020 bus cycle takes three clock periods (six high and low clock states). Its bus cycle is one clock faster than an MC68000 or MC68010 system can handle.

To correctly simulate MC68000 or MC68010 bus cycles, minor changes must be made to the MC68020 bus cycle timing. The address strobe (AS) asserted by the MC68020 must be delayed from reaching the host system until after the rising edge of state 2 (S2) in the bus cycle, which is shown in Figure 3. This delay results from feeding AS



NOTE:
Deviations from MC68000 system characteristics are shown on the diagram; the appropriate characteristics of the platform board are given below.

Num.	Characteristic	MC680000L8	Platform Board
A	Clock High to AS Low	60 ns	40 ns
B	Clock High to UDS/LDS Low	60 ns	65 ns
C	Clock High to DSACK1 Asserted	—	40 ns
D	AS Negated to DSACK1 Negated	—	50 ns

Figure 3. Bus Cycle Timing Diagram

into a D-type flip-flop and performing the logical OR of the flip-flop output with the input address strobe. The logical OR of the output with the input ensures the delay of the assertion, but not the negation, of the strobe as required by MC68000 bus cycle specificatons. The resulting signal becomes the system address strobe for the target system.

A similar modification must be made with the MC68020 data strobe ($\overline{\mathrm{DS}}$), as shown in Figure 2. The data strobe to the MC68000 system is delayed the same way as $\overline{\mathrm{AS}}$, and then encoded into the upper and lower data strobes ($\overline{\mathrm{UDS}}$ and $\overline{\mathrm{LDS}}$) for the host system. This modification uses the additional bus sizing capabilities of the MC68020.

The MC68020 generates four signals to identify the size and orientation of an item transferred on the data bus: SIZ1, SIZ0, A1, and A0. These signals and the delayed $\overline{\mathrm{DS}}$ are encoded into the upper and lower data strobes for the MC68000 system as shown in Table 1.

As shown in the truth table, $\overline{\mathrm{UDS}}$ coincides with A0 and $\overline{\mathrm{LDS}}$ relates to A0, SIZ1, and SIZ0. A1 is not involved in the generation of the system data strobes for a 16-bit data bus. The following equations define $\overline{\mathrm{UDS}}$ and $\overline{\mathrm{LDS}}$:

$$\overline{\mathrm{UDS}} = \overline{\mathrm{DS}} + \mathrm{A0}$$

$$\overline{\mathrm{LDS}} = \overline{\mathrm{DS}} + \overline{\mathrm{A0}} \bullet \mathrm{SIZ1} \bullet \mathrm{SIZ0}$$

Figure 2 shows the resulting upper and lower data strobe logic.

A logical AND of three signals provides the input to $\overline{\mathrm{DSACK1}}$ on the MC68020. The signals involved are the $\overline{\mathrm{DSACK1}}$ output from the MC68881, the $\overline{\mathrm{DTACK}}$ signal from the host system, and a signal generated by the interface hardware for the M6800 family peripheral cycle.

The coprocessor $\overline{\mathrm{DSACK1}}$ signal does not need modification. Communication between the MC68020 and MC68881 is transparent to the host system, and occurs in a standard three-clock bus cycle over a 32-bit data bus. To ensure transparency, however, the assertion of $\overline{\mathrm{AS}}$ to the MC68000 system must be blocked during coprocessor communication. This requirement is discussed further in the coprocessor selection section of this application note.

The return of $\overline{\mathrm{DSACK1}}$ to the MC68020 must not be asserted before state 3 (S3) in the bus cycle for the processor to run with a minimum four-clock bus cycle in the host system. This delay results from performing the logical OR of the $\overline{\mathrm{DTACK}}$ signal from the target system with the MC68020 $\overline{\mathrm{AS}}$ signal.

M6800 family interface logic generates the third signal in the $\overline{\mathrm{DSACK1}}$ expression. M6800 devices are synchronous and do not produce a $\overline{\mathrm{DTACK}}$ signal, so the MC68000 provides a synchronous bus cycle mechanism to support them. As previously mentioned, the MC68020 does not include this mechanism; external logic emulates it.

A decade counter controls the synchronous communication protocol. This state machine generates the E clock signal every M6800 device needs. The frequency of this signal is one-tenth the frequency of the system clock and the duty cycle is 60% low, 40% high. As part of the interface logic, the decade counter's outputs are available to produce a properly timed $\overline{\mathrm{DSACK1}}$.

The state machine generates the $\overline{\mathrm{DSACK1}}$ signal and the $\overline{\mathrm{VMA}}$ signal required to communicate with M6800 devices. The $\overline{\mathrm{DSACK1}}$ signal is discussed in the following paragraphs. The $\overline{\mathrm{VMA}}$ signal is covered in the M6800 family device communication section.

In this configuration, M6800 interface specifications indicate when data is valid on the data bus so the coprocessor can assert $\overline{\mathrm{DSACK1}}$ on the appropriate state of the clock. The following equations show the logical expressions for the E clock and the third input to the $\overline{\mathrm{DSACK1}}$ AND function:

$$\mathrm{E} = \mathrm{QD} + \mathrm{QC} \bullet \mathrm{QB}$$

$$\mathrm{Z1} \diamond = (\overline{\mathrm{QD}} + \mathrm{QC} + \mathrm{QB} + \mathrm{QA} + \mathrm{VMA}) \bullet \mathrm{Z1} + \mathrm{DS00}$$

In these equations, QX is one of four outputs of the decade counter, Z1 ◇ is the input to a clocked D-type flip-flop that stores the $\overline{\mathrm{DSACK}}$ assertion state information, and DS00 is the delayed data strobe of the MC68020. As shown in Table 2, this logic ensures $\overline{\mathrm{DSACKI}}$ is asserted in the ninth state of the E clock and remains asserted until the bus cycle finishes on the falling edge of E.

Table 1. $\overline{\mathrm{UDS}}$ and $\overline{\mathrm{LDS}}$ Truth Table

Alignment	Size	A1	A0	SIZ1	SIZ0	$\overline{\mathrm{UDS}}$	$\overline{\mathrm{LDS}}$
Long Word	Long	0	0	0	0	0	0
	Byte	0	0	0	1	0	1
	Word	0	0	1	0	0	0
	Three	0	0	1	1	0	0
Odd Byte Even Word	Long	0	1	0	0	1	0
	Byte	0	1	0	1	1	0
	Word	0	1	1	0	1	0
	Three	0	1	1	1	1	0
Odd Word	Long	1	0	0	0	0	0
	Byte	1	0	0	1	0	1
	Word	1	0	1	0	0	0
	Three	1	0	1	1	0	0
Odd Byte, Odd Word	Long	1	1	0	0	1	0
	Byte	1	1	0	1	1	0
	Word	1	1	1	0	1	0
	Three	1	1	1	1	1	0

Table 2. Clock, $\overline{\text{VMA}}$ and $\overline{\text{DSACK1}}$ Generator State Table

QD	QC	QB	QA	E	Comments
0	0	0	0	0	Negate $\overline{\text{VMA}}$ and $\overline{\text{DSACK1}}$
0	0	0	1	0	
0	0	1	0	0	
0	0	1	1	0	Assert $\overline{\text{VMA}}$ if $\overline{\text{VPA}}$ Asserted
0	1	0	0	0	
0	1	0	1	1	
0	1	1	0	1	
0	1	1	1	1	
1	0	0	0	1	Assert $\overline{\text{DSACK}}$ if $\overline{\text{VPA}}$ Asserted
1	0	0	1	0	

When these logic modifications are made, the resulting platform board configuration allows the MC68020 to properly emulate a standard MC68000 or MC68010 bus cycle.

M6800 FAMILY DEVICE COMMUNICATION

The MC68000 and MC68010 support M6800 family devices with a special protocol when an external device asserts valid peripheral address ($\overline{\text{VPA}}$) to the microprocessor. The MC68000 responds to the $\overline{\text{VPA}}$ assertion by synchronizing with the E clock output from the MC68000, asserting valid memory address ($\overline{\text{VMA}}$) to the M6800 device, and ending the cycle on the falling edge of E. The following paragraphs present the remaining aspects of the synchronous interface, E clock synchronization, and $\overline{\text{VMA}}$ generation.

Table 2 lists the states of the E clock generator and the various actions taken on particular state transitions.

The assertion of $\overline{\text{VMA}}$ occurs on the rising edge of the system clock 1.5 clocks before the rise of the E clock to synchronize the MC68000 to an M6800 device. Refer to the MC68000 data sheet for best and worse case synchronization timing in relation to the assertion of $\overline{\text{VPA}}$.

The decoded states of the decade counter generate $\overline{\text{VMA}}$. The MC68000 asserts $\overline{\text{VMA}}$ on the fourth state of the counter if the peripheral generated $\overline{\text{VPA}}$. The return of $\overline{\text{VMA}}$ to the peripheral signals that the processor is synchronized to the E clock. It also completes the two-line handshake initiating M6800 family device communication. $\overline{\text{VMA}}$ must remain asserted during the transfer. The signal is negated on the falling edge of the E clock. The following equation shows the logical expression for $\overline{\text{VMA}}$:.

$$(\overline{\text{VMA}})\Diamond = \overline{\text{QD}}\cdot\overline{\text{QC}}\cdot\text{QB}\cdot\text{QA}\cdot\overline{\text{VPA}} + \overline{\text{VMA}}\cdot(\text{QD}+\text{QC})$$

In this equation, QX is a particular output of the decade counter. QD is the most significant bit and the ◇ designates the input to a clocked D-type flip-flop.

INTERRUPT PROCESSING

The MC68000 can support interrupting devices that cannot supply an interrupt vector. The peripheral device asserts $\overline{\text{VPA}}$ input during an interrupt acknowledge (IACK) cycle to request that the MC68000 automatically produce the vector. The MC68020 provides the same capability, but the peripheral device signals the autovector request by asserting the $\overline{\text{AVEC}}$ input.

The decoded function code lines and address lines A19 through A16 generate the $\overline{\text{AVEC}}$ input signal. When all these lines are high and the interrupting device asserts $\overline{\text{VPA}}$, an autovector is required. Figure 2 shows the logic for the autovector signal. The following equation defines the signal:

$$\overline{\text{AVEC}} = \overline{\text{VPA}}\cdot\text{FC2}\cdot\text{FC1}\cdot\text{FC0}\cdot\text{A19}\cdot\text{A18}\cdot\text{A17}\cdot\text{A16}$$

The MC68000 and the MC68020 handle all other interrupts in the same way, so no additional logic is required.

BUS ARBITRATION

The assertion of $\overline{\text{AS}}$ by the MC68020 must be blocked from the host system during MC68881 accessing to ensure transparency of the bus operations on the platform board. However, blocking the $\overline{\text{AS}}$ assertion may trigger the following events:

- During any host bus cycle, an alternate bus master can ask to use the bus by asserting the bus request ($\overline{\text{BR}}$).
- The MC68020 begins a coprocessor access and simultaneously returns the bus grant ($\overline{\text{BG}}$) signal to the alternate bus master, assuming the alternate bus master will wait until the negation of the address strobe before taking control of the bus.
- With the asserted address strobe blocked from the host system during coprocessor cycles, the requesting device asserts bus grant acknowledgement and begins to drive the bus.
- Bus contention occurs because the platform board control, address, and data pins are connected to the MC68000 system.

To avert this situation, bus arbitration must be inhibited until coprocessor access is complete. The interface logic must prevent the assertion of $\overline{\text{BG}}$ to the alternate bus master until the interface negates $\overline{\text{AS}}$. Also, while the system address strobe is delayed to simulate the MC68000 bus cycle, the MC68020 assertion of $\overline{\text{BG}}$ must be blocked to avoid bus contention. The interface circuit accomplishes this task. The following equation shows its corresponding logical expression:

$$\overline{BG00} = \overline{BG20}\bullet(\overline{AS00} + AS20)\diamond\bullet AS20$$

In this equation, the 00 suffix indicates a signal to the MC68000 system, the 20 suffix indicates a signal from the MC68020, and the ◇ represents a signal delayed through a D-type flip-flop. The $\overline{AS00}$ term includes the coprocessor chip select.

COPROCESSOR SELECTION

In addition to the bus arbitration previously discussed, coprocessor selection also involves generation of the chip select signal.

When the MC68020 communicates with a coprocessor, it encodes the following items:.

- Function codes to 7.
- Coprocessor communications central processing unit (CPU) space type (0010) on address lines A19 through A16.
- Coprocessor identification number on address lines A15 through A13.

In this design, only the coprocessor select field is decoded because the MC68881 is the sole coprocessor on the platform board. The chip select signal is generated by decoding the function codes and address lines A19 through A16. The following equations shows the logical expression for this condition:

$$cp\overline{CS} = FC2\bullet FC1\bullet FC0\bullet\overline{A19}\bullet\overline{A18}\bullet A17\bullet\overline{A16}$$

HARDWARE IMPLEMENTATION

Once all the appropriate MC68020 signals are modified to emulate an MC68000 or MC68010, the interface can be implemented. This section discusses an implementation that emphasizes a low chip count. It uses four PAL16R4A-15 programmable array logic (PAL) devices and six 4.7K resistors. This system also could be implemented with discrete TTL devices. Figure 4 shows the platform board schematic diagram.

PROGRAMMABLE LOGIC DESIGN

As shown in the PLA logic diagram, this design uses 12 D-type flip-flops. There are 18 inputs to the interface, including a system clock. It generates 10 outputs. The inputs and outputs are listed as follows:

Inputs:
CLK, FC2, FC1, FC0, A19, A18, A17, A16, A0, $\overline{AS20}$, $\overline{DS20}$, SIZ1, SIZ0, cp$\overline{DSACK1}$, $\overline{DTACK}$, $\overline{VPA}$, $\overline{BG20}$, and $\overline{BGACK}$.

Outputs:
cp$\overline{CS}$, $\overline{AVEC}$, $\overline{DSACK1}$, $\overline{UDS}$, $\overline{LDS}$, $\overline{VMA}$, E, $\overline{AS00}$, $\overline{AS20}$, and $\overline{BG00}$.

This design uses programmable logic devices because they provide the correct configurations while conserving board space. The PAL device allows 16 inputs (the eight outputs can be fed back into the device as inputs) and provides four registered and four direct outputs. The particular component chosen is the version with high speed and low set-up times. Its gate delays are almost as fast as Schottky TTL devices (15 ns typical and 25 ns maximum) and it offers 15 ns set-up times.

Ideally, all the outputs dependent on a given set of inputs should be on the same programmable device. However, in this design the outputs tend to use a subset of multiple input configurations. Some inputs must be fed to more than one device, and some outputs must be fed into another device.

In this implementation, the $\overline{AS00}$, $\overline{UDS}$, $\overline{LDS}$, and $\overline{VMA}$ outputs must be set in a high-impedance state to allow bus arbitration in multiple bus master systems. The PAL device for this design has high-impedance outputs, so only a control signal is needed to set the output buffers in the high-impedance state. The control signal generation requires one more input to the interface ($\overline{BGACK}$) and produces an additional output because of the input-output configuration of this design.

During an active bus cycle, if $\overline{BG20}$ asserts before $\overline{AS20}$ negates, the four control signals must be set in the high-impedance state before the rising edge of the clock after $\overline{AS20}$ negates. During an idle bus period, the signals need only be set in the high-impedance state within 60 ns (maximum) after $\overline{BG20}$ asserts. Once the alternate bus master releases the bus, the microprocessor drives these four signals again. To meet these two criteria, the high-impedance control signal negates 1.5 clocks after the alternate bus master negates $\overline{BGACK}$.

Figure 5 shows the logic reguired for the high-impedance control signal. Its associated Boolean expression appears in the following equation:

$$\text{HIGH-IMPEDANCE STATE} = \overline{BGACK}\diamond\ \diamond + AS20\bullet\overline{BG20}$$

Figure 6 shows the PAL pinouts used in this platform board.

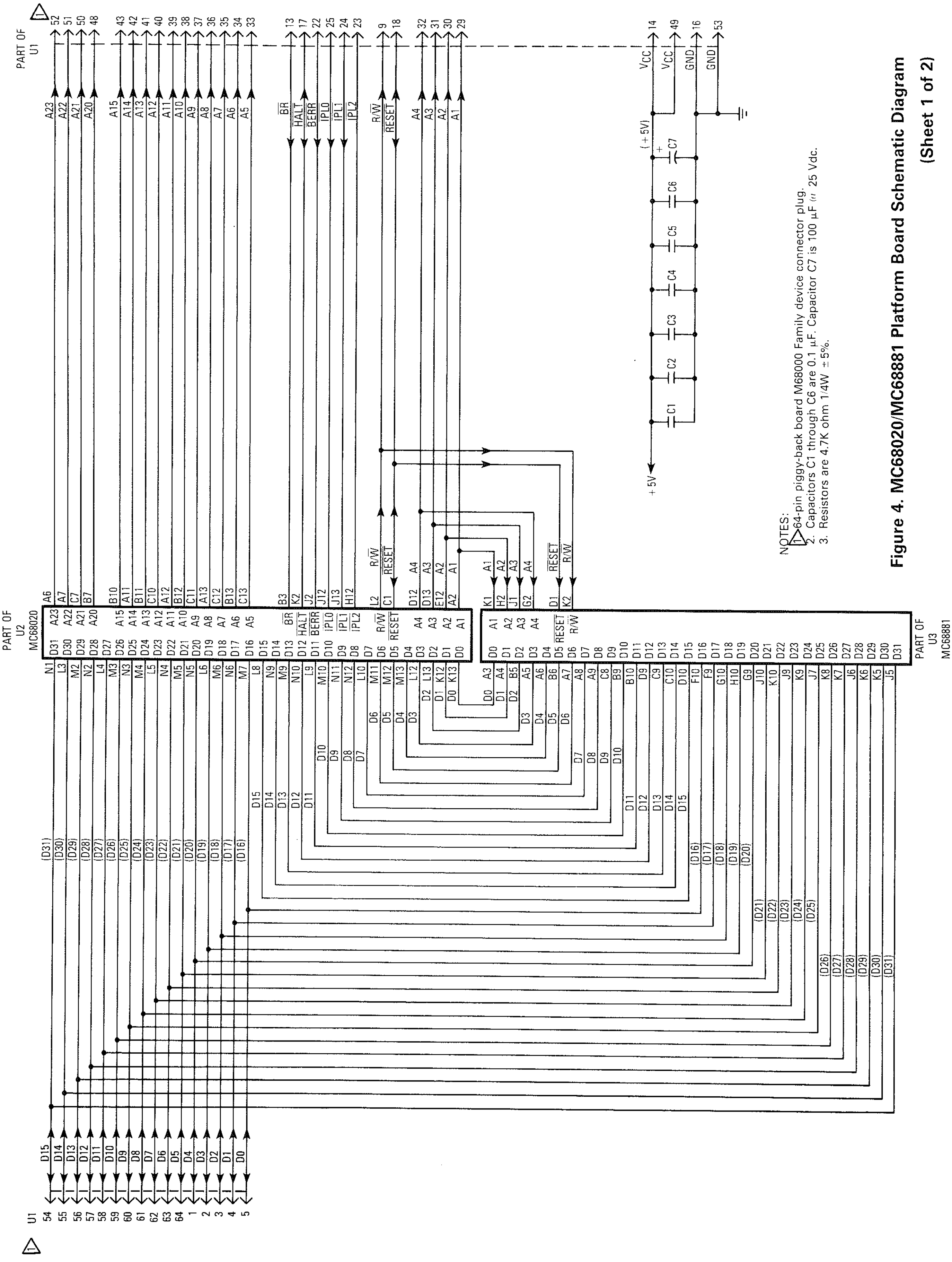
Appendix A presents the PALASM source used with a Data I/O Logic Pak to program the four PALs. File descriptions follow the PALASM syntax in Appendix A as specified by Monolithic Memories Inc. The file begins with a device specification and title line. The second line contains the chip name and date. Next is a file description title, followed by the owner's name and address. The next two lines list the pin designations; pins 1 through 10 appear on the first line and pins 11 through 20 on the second line.

In this design, the outputs of three PALs always are enabled, so pin 11 is tied to ground. The system clock, connected to pin 1, clocks all four devices.

The remaining sections of the file definitions in Appendix A consist of logic equations. The first set defines the nonregistered outputs (identified by the = assignment operator). The second set of equations describe registered outputs (denoted by the := assignment operator). The file definition concludes with a description that identifies the end of the file.

Although this design is intended for execution speeds up to 12.5 MHz, the system was not tested above 10 MHz. The set-up time requirements of the PALs in this prototype can result in an error at 12.5 MHz operation.

Figure 7 illustrates this potential anomaly for the address strobe. In this case, the falling edge of the clock triggers the MC68020 assertion of $\overline{AS}$. The PAL register that uses $\overline{AS}$ as an input is clocked by the rising edge of



NOTES:
1. 64-pin piggy-back board M68000 Family device connector plug.
2. Capacitors C1 through C6 are 0.1 µF. Capacitor C7 is 100 µF @ 25 Vdc.
3. Resistors are 4.7K ohm 1/4W ±5%.

Figure 4. MC68020/MC68881 Platform Board Schematic Diagram (Sheet 1 of 2)

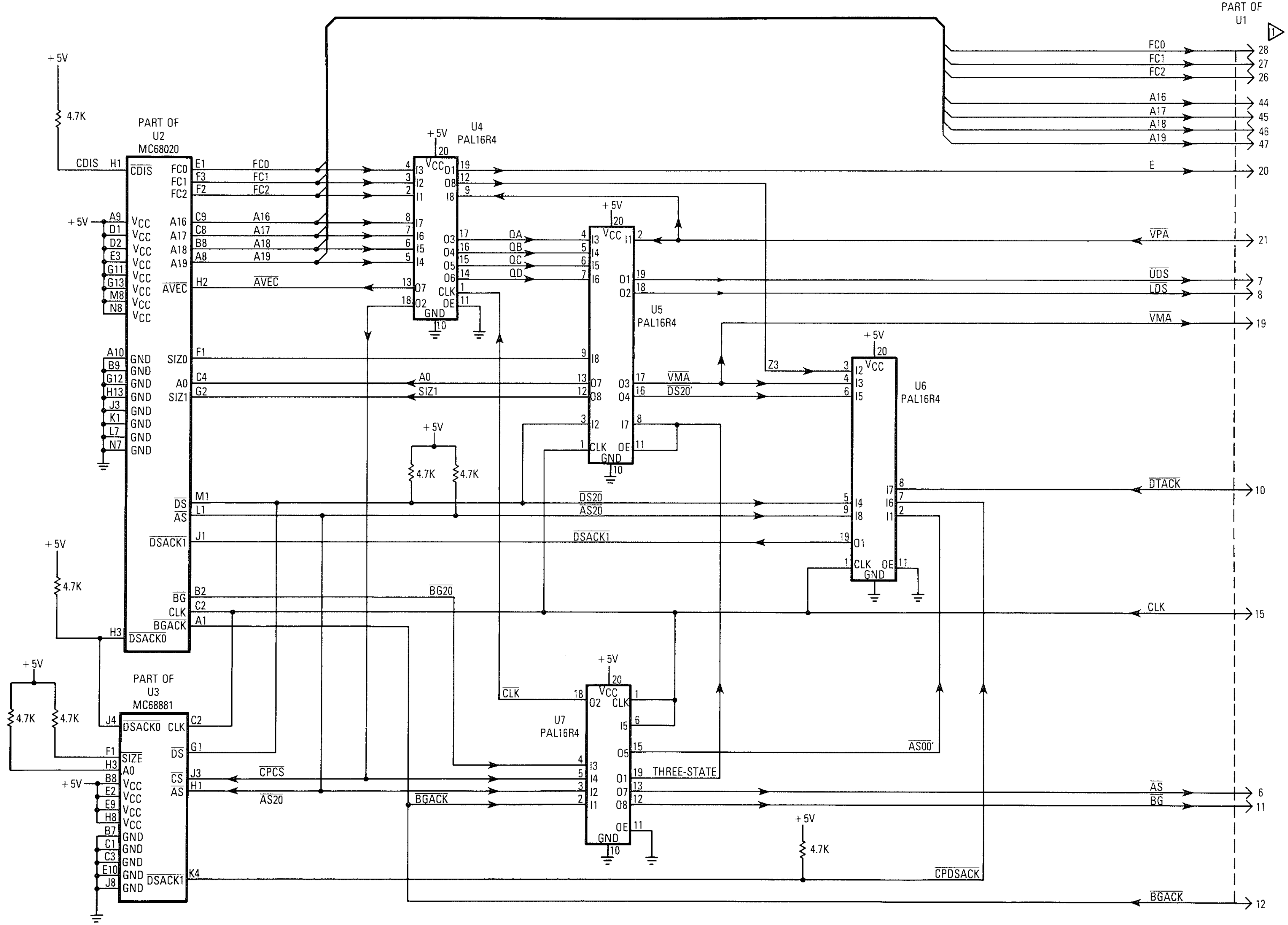


Figure 4. MC68020/MC68881 Platform Board Schematic Diagram (Sheet 2 of 2)

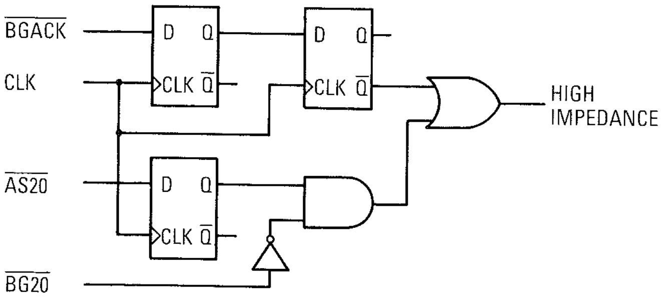
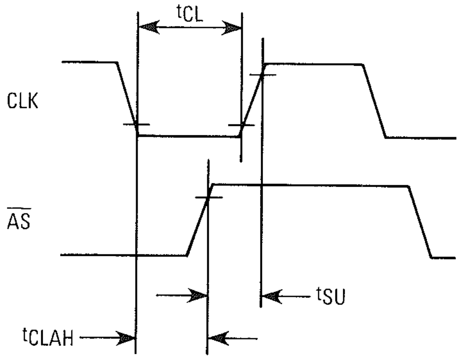


Figure 5. High-Impedance State Control Logic



where:
t_{CL} — is the CLK pulse width
t_{CLAH} — is CLK low to AS high (refer to the *MC68020 User's Manual, Second Edition*, page 10-4, specification #9)
t_{SU} — is the setup time for the PAL

Figure 7. Setup Time Error Example

the clock. In this example, it is possible for a change in AS (or any signal sampled on the clock edge) to be missed if the signal is not within the 15 ns setup time for the PAL. The following equation shows the criteria that must be met:

$$t_{CLAH} + t_{su} \leq t_{CL}$$

At 8 MHz, t_{CL} must be 55 ns minimum and t_{CLAH} must be 40 ns maximum to conform to design specifications. At 10 MHz, t_{CL} drops to 45 ns minimum. An MC68020R16, with a t_{CLAH} of 30 ns maximum, must be used at this speed to ensure proper operation.

At 12.5 MHz, t_{CL} drops again to 35 ns minimum. Operation at 12.5 MHz is not guaranteed as the design was not tested at this speed. However, the MC68020 capacitive derating can be used to calculate a smaller value for t_{CLAH}, theoretically ensuring board operation within specifications. An interface design using Advanced Schottky or FAST components instead of PALs also would ensure proper operation at 12.5 MHz. In that configuration, jumper lines from the host system could provide any additional power and grounding the platform board needed.

TEST VEHICLES

Motorola used a typical MC68000-based system, the M68KVM02-3, to test and analyze this design. This microprocessor, however, does not use autovectoring or bus arbitration. To test the autovector circuitry, the interface was installed within another Motorola Microsystems product, the MEK68KECB (Educational Computer Board). To test the bus arbitration capabilities of the interface, a pulse generator drove the bus request (BR) pin during various system activities and a logic analyzer observed the response.

The M68KVM02 board is not the ideal system in terms of testing performance. It runs with three wait-states on read cycles and two wait-states on write cycles at 8 MHz with an MC68000 microprocessor. At 10 MHz the system requires an additional wait-state on all read accesses. In this configuration the MC68020 runs with up to five wait-states.

PERFORMANCE ANALYSIS

Motorola ran benchmark routines to compare the performance of the MC68020 to the MC68000 in the same host configuration. This section explains the software run and presents the results of those tests.

The software benchmarks run were a subset of the EDN benchmarks for microprocessors and the Pascal versions of the Berkeley benchmark routines. The EDN benchmarks consisted of the following: E (string search), F (bit

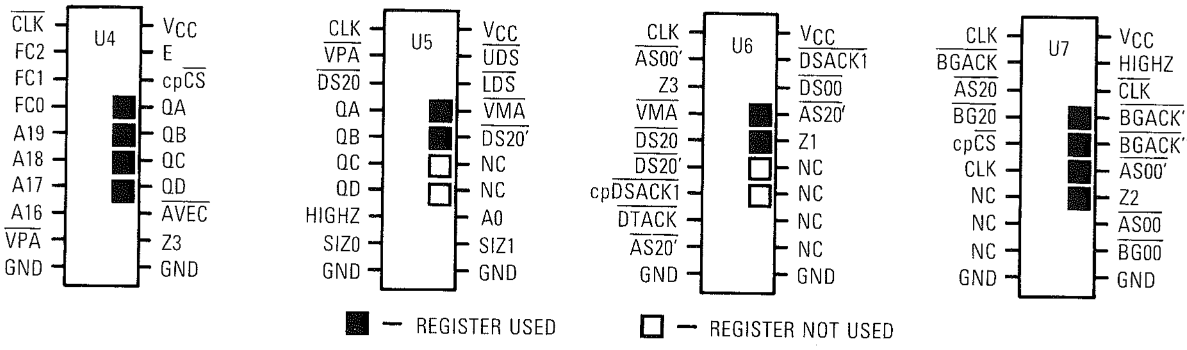


Figure 6. PAL Pinout Definitions

test, set, and clear), H (linked list insertion), I (quick sort), and K (matrix inversion routines). The Berkeley software consisted of the following: Acker (Ackermann Function), Benche (string concatenation program), Sieve (Sieve of Erastosthenes), Towers (Towers of Hanoi routine), and Puzzle (a Stanford routine).

The routines ran at 8 and 10 MHz for two configurations: the MC68000 and the MC68020 with cache enabled. Tables 3 through 6 and Figures 8 and 9 summarize these tests.

The software shows a performance increase for the MC68020 with on-chip cache enabled, except for the EDN benchmark F and the Berkeley Benche routine. The two exceptions have little or no looping constructs, and consequently fail to take advantage of the on-chip cache.

The MC68020 has several mechanisms that allow it to perform four to six times faster than the MC68000 for the same programs. These include its on-chip instruction cache, the MC68020 internal barrel shifter, a faster and more efficient arithmetic-logic unit (ALU), and a more sophisticated internal architecture.

The performance increases shown by the platform board benchmarks are limited to approximately two times for the following reasons:

- Operation on a 16-bit bus rather than a 32-bit bus.
- Execution of an extra prefetch cycle (on a 16-bit bus) during jumps and branches because the MC68020 always prefetches 32 bits of instruction stream at a time.
- Operation of the MC68020 at a speed slower than the nomimal clock frequency of 16.67 MHz.
- No use of the new instructions and addressing modes of the MC68020.

There is little performance difference between running the platform board at 8 and 10 MHz. The M68KVM02 is limited to 10 MHz, but some performance increase can be expected when using the board in systems capable of the maximum MC68000 and MC68010 frequency of 12.5 MHz.

Table 3. 8 MHz END Benchmark Times

Device	Execution Times (ms)				
	E	F	H	I	K
MC68000	400.8	113.8	263.1	43.4	705.6
MC68020 (Cache On)	237.4	149.3	211.3	20.2	361.7

Table 4. 10 MHz END Benchmark Times

Device	Execution Times (ms)				
	E	F	H	I	K
MC68000	362.2	102.0	236.0	39620.0	627.3
MC68020 (Cache On)	205.2	134.3	188.2	17260.0	306.8

Table 5. 8 MHz Berkely Benchmark Times

Device	Execution Times (ms)				
	Acker	Benche	Sieve	Towers	Puzzle
MC68000	9.975	0.348	1.150	31.060	28.610
MC68020 (Cache On)	5.721	0.374	0.765	18.030	14.360

Table 6. 10 MHz Berkely Benchmark Times

Device	Execution Times (ms)				
	Acker	Benche	Sieve	Towers	Puzzle
MC68000	8.953	0.347	1.067	27.590	25.450
MC68020 (Cache On)	4.958	0.373	0.720	15.860	12.160

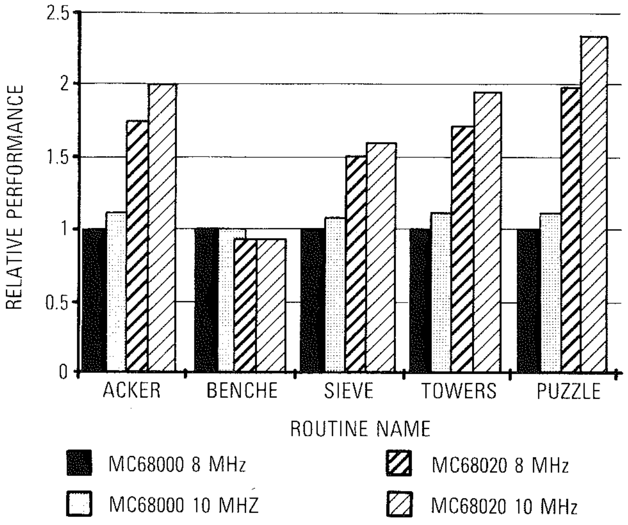


Figure 8. EDN Benchmark Summary

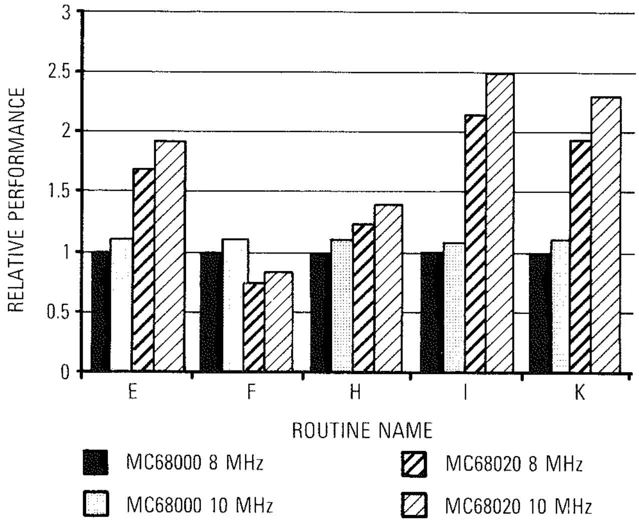


Figure 9. Berkely Benchmark Summary

APPENDIX A

1) PAL16R4A-15 PAL DESIGN SPECIFICATION
U4 8/19/86

MC68020/MC68000 INTERFACE; MOTOROLA INC., AUSTIN, TEXAS

CLKN	FC2	FC1	FC0	A19	A18	A17	A16	VPA	GND
GND	Z3	AVEC	QD	QC	QB	QA	CPCS	E	V_{CC}

$\overline{\mathrm{E}}=(\overline{\mathrm{QD}})(\overline{\mathrm{QC}})+(\overline{\mathrm{QD}})(\overline{\mathrm{QB}})$
$\overline{\mathrm{CPCS}}=(\mathrm{FC2})(\mathrm{FC1})(\mathrm{FC0})(\overline{\mathrm{A19}})(\overline{\mathrm{A18}})(\mathrm{A17})(\overline{\mathrm{A16}})$
$\overline{\mathrm{AVEC}}=(\mathrm{FC2})(\mathrm{FC1})(\mathrm{FC0})(\mathrm{A19})(\mathrm{A18})(\mathrm{A17})(\mathrm{A16})(\overline{\mathrm{VPA}})$
$\overline{\mathrm{Z3}}=\overline{\mathrm{QD}}+\mathrm{QC}+\mathrm{QB}+\mathrm{QA}$

$\overline{\mathrm{QA}}:=\mathrm{QA}$
$\overline{\mathrm{QB}}:=(\overline{\mathrm{QB}})(\overline{\mathrm{QA}})+(\mathrm{QB})(\mathrm{QA})+(\mathrm{QD})$
$\overline{\mathrm{QC}}:=(\overline{\mathrm{QC}})(\overline{\mathrm{QA}})+(\overline{\mathrm{QC}})(\overline{\mathrm{QB}})+(\mathrm{QC})(\mathrm{QB})(\mathrm{QA})$
$\overline{\mathrm{QD}}:=(\overline{\mathrm{QC}})(\overline{\mathrm{QD}})+(\overline{\mathrm{QB}})(\mathrm{QA})+(\overline{\mathrm{QD}})(\overline{\mathrm{QA}})$

DESCRIPTION: DECADE COUNTER, 6800 CLOCK, AUTOVECTOR, AND COPROCESSOR SELECT LOGIC.

2) PAL16R4A-15 PAL DESIGN SPECIFICATION
U5 8/19/86

MC68020/MC68000 INTERFACE; MOTOROLA INC., AUSTIN, TEXAS

CLK	VPA	DS20	QA	QB	QC	QD	HIGHZ	SIZ0	GND
HIGHZ	SIZ1	A0	NC	NC	DS20′	VMA	LDS	UDS	V_{CC}

IF $(\overline{\mathrm{HIGHZ}})$ $\overline{\mathrm{UDS}}=(\overline{\mathrm{DS20}})(\overline{\mathrm{DS20'}})(\overline{\mathrm{A0}})$
IF $(\overline{\mathrm{HIGHZ}})$ $\overline{\mathrm{LDS}}=(\overline{\mathrm{DS20}})(\overline{\mathrm{DS20'}})(\overline{\mathrm{SIZ1}})+(\overline{\mathrm{DS20}})(\overline{\mathrm{DS20'}})(\overline{\mathrm{SIZ0}})+(\overline{\mathrm{DS20}})(\overline{\mathrm{DS20'}})(\mathrm{A0})$

$\overline{\mathrm{VMA}}:=(\overline{\mathrm{QD}})(\overline{\mathrm{QC}})(\mathrm{QB})(\mathrm{QA})(\overline{\mathrm{VPA}})+(\overline{\mathrm{VMA}})(\mathrm{QD})+(\overline{\mathrm{VMA}})(\mathrm{QC})$
$\overline{\mathrm{DS20'}}:=\overline{\mathrm{DS20}}$

DESCRIPTION: $\overline{\mathrm{VMA}}$, $\overline{\mathrm{UDS}}$, $\overline{\mathrm{LDS}}$ LOGIC INCLUDING HIGHZ CONTROL FOR $\overline{\mathrm{UDS}}$ AND $\overline{\mathrm{LDS}}$

3) PAL16R4A-15 PAL DESIGN SPECIFICATION
U6 8/19/86

MC68020/MC68000 INTERFACE; MOTOROLA INC., AUSTIN, TEXAS

CLK	AS00′	Z3	VMA	DS20	DS20′	cpDSACK1	DTACK	AS20	GND
GND	NC	NC	NC	NC	Z1	AS20′	DS00	DSACK1	V_{CC}

$\overline{\mathrm{DS00}}=(\overline{\mathrm{DS20}})(\overline{\mathrm{DS20'}})$
$\overline{\mathrm{DSACK1}}=\overline{\mathrm{cpDSACK1}}+\overline{\mathrm{Z1}}+(\overline{\mathrm{AS20}})(\overline{\mathrm{DTACK}})(\overline{\mathrm{AS00'}})(\overline{\mathrm{AS20}})$

$\overline{\mathrm{AS20'}}:=\overline{\mathrm{AS00'}}$
$\overline{\mathrm{Z1}}:=(\overline{\mathrm{DS00}})(\overline{\mathrm{Z1}})+(\overline{\mathrm{DS00}})(\mathrm{Z3})(\overline{\mathrm{VMA}})$

DESCRIPTION: SYSTEM DATA STROBES, BUS GRANT, AND INTERNAL LOGIC

4) PAL16R4A-15 PAL DESIGN SPECIFICATION
U7 8/19/86

MC68020/MC68000 INTERFACE; MOTOROLA INC., AUSTIN, TEXAS

CLK	BGACK	AS20	BG20	CPCS	CLK	NC	NC	NC	GND
GND	BG00	AS00	Z2	AS00′	BGACK″	BGACK′	CLKN	HIGHZ	V_{CC}

$\text{HIGHZ} = (\text{BGACK}'')(\overline{\text{AS00}'}) + (\text{BGACK}'')(\text{BG20})$
IF (HIGH Z) $\overline{\text{AS00}} = (\text{CPCS})(\overline{\text{AS20}})(\overline{\text{AS00}'})$
$\overline{\text{BG00}} = (\overline{\text{BG20}})(\overline{\text{Z2}})(\text{AS20})$
$\overline{\text{CLKN}} = \text{CLK}$

$\overline{\text{BGACK}'} := \overline{\text{BGACK}}$
$\overline{\text{BGACK}''} := \overline{\text{BGACK}'}$
$\overline{\text{AS00}'} = \overline{\text{AS20}}$
$\overline{\text{Z2}} = (\overline{\text{AS00}}) + (\text{AS20})$

DESCRIPTION: HIGH IMPEDANCE CONTROL, SYSTEM BUS GRANT, AND ADDRESS STROBE

Motorola reserves the right to make changes without further notice to any products herein to improve reliability, function or design. Motorola does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others. Motorola and Ⓜ are registered trademarks of Motorola, Inc. Motorola, Inc. is an Equal Employment Opportunity/ Affirmative Action Employer.

Literature Distribution Centers:

USA: Motorola Literature Distribution; P.O. Box 20912; Phoenix, Arizona 85036.
EUROPE: Motorola Ltd.; European Literature Center; 88 Tanners Drive, Blakelands Milton Keynes, MK145BP, England.
ASIA PACIFIC: Motorola Semiconductors H.K. Ltd.; P.O. Box 80300; Cheung Sha Wan Post Office; Kowloon Hong Kong.

MOTOROLA

A18672 PRINTED IN USA 9-87 IMPERIAL LITHO 51197 18,000 AN944/D