MOTOROLA
Semiconductor Products Inc.

AN-868
Application Note

USING HIGH-SPEED CMOS LOGIC FOR MICROPROCESSOR INTERFACING

Prepared by
Sherri Craig
MOS Logic Applications
Austin, Texas

INTRODUCTION

This application note explores the possibility of using high-speed CMOS (HSCMOS) devices in microprocessor systems where LSTTL, because of its speed, has generally been used. The system designer is no longer forced to sacrifice low power consumption in order to achieve the higher speeds required by the system, since HSCMOS devices have the higher speeds of LSTTL and yet maintain the low power consumption of metal-gate CMOS.

Because most present-day microprocessors, memories, and peripherals are made with the N-channel metal-oxide semiconductor (NMOS) technology, as opposed to other MOS technologies such as P-channel MOS (PMOS) or complementary MOS (CMOS), this paper begins with a discussion of considerations for interfacing HSCMOS and NMOS technologies. NMOS (also referred to as HMOS or high density N-channel MOS) is used for these devices because higher packing densities (or number of transistor devices per chip) can be achieved with NMOS. Attaining higher packing densities on a chip is important when trying to implement very complex devices, such as microprocessors, on a single chip.

Next, some of the digital logic needs of a microprocessor system are discussed and analyzed. These needs include data and address bus buffering for increasing output drives, address decoding, and the use of latches to meet the various speed requirements of MPU systems. A number of the high-speed CMOS devices available to meet these needs are discussed.

Finally, some recommendations for using HSCMOS devices in NMOS microprocessor systems are given.

Interfacing HSCMOS and NMOS Technologies

The inputs and outputs of NMOS microprocessors, memories, and peripherals are designed for TTL compatibility because the TTL logic family, until now, has done the best job of meeting the cost and speed performance needs for interfacing MPU systems while maintaining a tolerable power consumption level.

Metal-gate CMOS is not a desirable logic family to use for interfacing in NMOS MPU systems for two reasons. Metal-gate CMOS has very slow speeds and very low output drive capability for driving logic other than CMOS logic. This logic family is usually used in systems where low power consumption is most important and speed is of little consequence.

High-speed CMOS logic, however, has LSTTL speeds and greater drive capability than metal-gate CMOS. Voltage levels, drive capability, speed, power, and noise immunity are considerations in interfacing HSCMOS and NMOS technologies.

Voltage Levels

High-speed CMOS logic is not always directly compatible with NMOS devices. There is a voltage level incompatibility that exists when driving HSCMOS inputs from NMOS outputs.

NOTE

This paper only considers interfacing with low voltage NMOS (single positive supply), and does not consider interfacing with high voltage NMOS (more than one positive supply).

Following is an analysis of NMOS inputs and outputs that have TTL characteristics.

NMOS Inputs — The input circuit of a TTL compatible N-channel MOS integrated circuit is shown in Figure 1. Q1 is the driver and Q2 is the load, which provides pullup current for the input. The threshold voltage of the input transistor is 0.5 to 1.5 V.

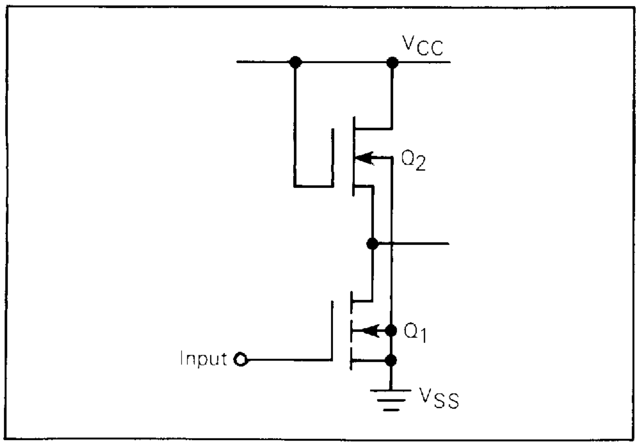


FIGURE 1 — Input Circuit of TTL Compatible NMOS I.C. (Q_1 is an enhancement mode device and Q_2 is a depletion mode device.)

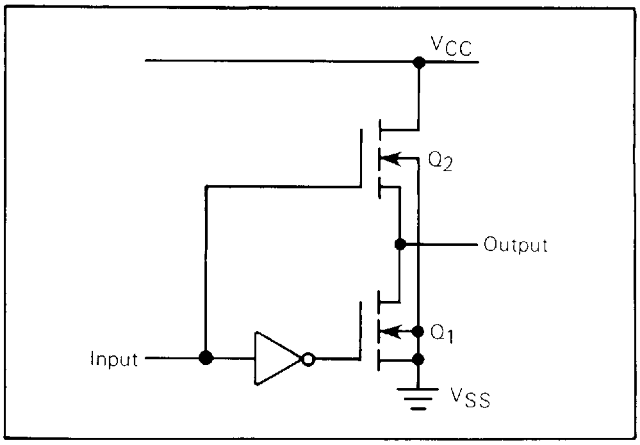


FIGURE 2 — Output Circuit of TTL Compatible NMOS I.C. (Q_1 and Q_2 are enhancement mode MOSFETS)

TABLE 1 — Voltage Levels for NMOS Inputs and HSCMOS Outputs. (V_{CC} = +5 volts ± 10%)

NMOS	*HSCMOS
V_{IH}(min) = 2.0 V	V_{OH}(min) = 4.95 V
V_{IL}(max) = 0.8 V	V_{OL}(max) = 0.05 V

*$I_O \leq$ 1 μA

TABLE 2 — Voltage Levels for NMOS Outputs and HSCMOS Inputs. (V_{CC} = +5 volts ± 10%)

NMOS	*HSCMOS
V_{OH}(min) = 2.4 V	V_{IH}(min) = 3.5 V
V_{OL}(max) = 0.4 V	V_{IL}(max) = 1.0 V

*$I_O \leq$ 1 μA

Table 1 shows the input voltage levels for NMOS and the output voltage levels for HSCMOS.

As can be seen from Table 1, HSCMOS outputs can be directly connected to NMOS inputs without any need for pullup or pulldown resistors. V_{OH} = 4.95 V (min) for HSCMOS is more than adequate for the NMOS input, V_{IH} = 2.0 V (min), to recognize as a logic 1. Also, the HSCMOS V_{OL} = 0.05 V (max) is low enough for the NMOS input, V_{IL} = 0.8 V (max), to recognize as a logic 0.

NMOS Outputs — The output circuit of TTL compatible NMOS is shown in Figure 2.

Q1 is the driver and Q2 is the load, which is an enhancement mode device in this case. The output of this particular configuration will not swing all the way up to V_{CC}, but will only go up to one threshold below V_{CC}.

As can be seen from Table 2, the output high voltage of 2.4 V (min) is insufficient for the HSCMOS input to see as a logic 1 since it requires a V_{IH} (min) of 3.5 V. Therefore, a pullup resistor on the output of the NMOS device to V_{CC} is needed to get a high output voltage of 3.5 V. The pullup resistor chosen should be small enough to assure adequate speed in a low-to-high transition.

The output low voltage of 0.4 V (max) is low enough for the HSCMOS input to recognize as a logic 0. Therefore, the only voltage incompatibility that exists is when NMOS outputs are driving HSCMOS inputs for the high state. Some of the bus compatible devices to be offered in HSCMOS do have TTL compatible voltage levels on the inputs. These are shown in Table 8, at the end of this application note, and are referred to as the HCT series devices.

Drive Capability

An improvement made in high-speed CMOS over metal-gate CMOS is higher output drive capability. Standard high-speed CMOS devices can sink up to 4 mA of current (6 mA for drivers) and still maintain an output voltage level of 0.4 V depending on the temperature series (54 or 74) and the output type. (See Table 3.)

For example, a 74-series high-speed CMOS standard output, such as the 74HC00 two-input NAND gate, can drive ten LSTTL inputs. This kind of drive capability is only important when driving logic other than metal-gate CMOS or HSCMOS logic because HSCMOS could drive hundreds of these inputs due to its high input impedance. Also, HSCMOS devices have symmetrical output drives since they can source the same amount of current that they can sink.

Most NMOS outputs can drive four LSTTL loads plus a specified amount of capacitance (usually 30, 90 or 130 pF). This drive capability is more than adequate for driving

TABLE 3 — Drive Capability of HSCMOS Devices. (For V_{out} = 0.4 V or V_{CC} − 0.8 V V_{in} = V_{CC} or GND).

Device	Output Type	Output Current	LSTTL Fanout
MC54HC	Standard	3.4 mA	8
MC54HC	Bus Driver	5.1 mA	12
MC74HC	Standard	4.0 mA	10
MC74HC	Bus Driver	6.0 mA	15

HSCMOS inputs which only require a maximum of 1 μA of current. Therefore, NMOS outputs can drive many more HSCMOS inputs than LSTTL inputs. For an all CMOS MPU system, the drive capability of HSCMOS devices is many times higher than typically necessary.

Speed Considerations

A problem with metal-gate CMOS is that the outputs cannot drive much capacitance (very long lines, many inputs, etc.) without slowing down considerably. Figure 3 shows propagation delay versus load capacitance for metal-gate CMOS, HSCMOS, and LSTTL. As can be seen from these curves, propagation delay degrades rapidly with load capacitance for standard CMOS. As with LSTTL, HSCMOS speeds degrade much less with load capacitance than metal-gate CMOS. This difference is due to the higher output impedance of the metal-gate CMOS device, since speed is dependent on this impedance. The LSTTL and HSCMOS devices have similar output impedances for the low output state. When in the high state, the LSTTL device has a higher output impedance, thus sourcing a smaller current (400 μA). HSCMOS devices, however, have the same output impedance in the low state as they do in the high state, since the source and sink currents are the same. Speed considerations are important when designing MPU systems, since timing considerations for data and address bus buffering, address decoding, and data latching are critical to the MPU system designer.

As shown in Table 4, HSCMOS devices have propagation delays, rise and fall times, etc., that are similar to LSTTL. The issue of speed is discussed in more detail in the sections that follow. The point here is that these devices will interface quite well in MPU systems where speed is concerned.

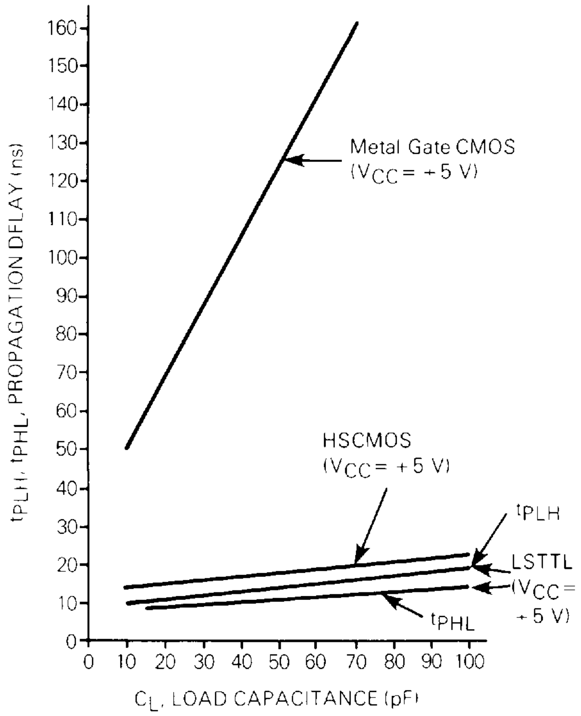


FIGURE 3 — Propagation Delay versus Load Capacitance for Metal-Gate CMOS, HSCMOS, and LSTTL

Power Considerations

Why would HSCMOS be useful in an NMOS microprocessor system? The answer is, of course, lower power dissipation. High-speed CMOS devices only dissipate appreciable power when they are switching states. While in the static state, they dissipate negligible quiescent power; whereas, LSTTL devices dissipate power even in the static state. However, high-speed CMOS devices do have a power dissipation increase as frequency increases, as can be seen in Figure 4. The difference in power dissipation for HSCMOS and LSTTL is quite large for frequencies 5 MHz and below due to the fact that bipolar devices require current to maintain the static state. Average power dissipation per gate for various logic families is shown in Figure 5. The advantage of using HSCMOS is even more evident from a system point of view. The more complex the system and the greater the parts count, the more the user will realize dramatic power savings with CMOS technologies.

In an NMOS MPU system, most of the power dissipation is a result of the NMOS microprocessors, memories, and

TABLE 4 — Comparison of Switching Characteristics for Several HSCMOS and LSTTL Devices

Symbol	Parameter	NAND Gate		Latch		Buffer		Decoder	
		74HC00	74LS00	74HC373	74LS373	74HC240	74LS240	74HC138	74LS138
f_{max}	Maximum Clock Frequency			30 MHZ	35 MHZ				
t_{PHL}/t_{PLH}	Maximum Prop Delay, Clock to Q	15 ns	15 ns	28 ns	30 ns	18 ns	18 ns	40 ns/25 ns	41 ns/20 ns
t_{SU}	Minimum Setup Time, Data to Clock			20 ns	5 ns				
t_h	Minimum Hold Time Clock to Data			0 ns	20 ns				
t_W	Minimum Pulse Width, Clock & Clear			16 ns	15 ns				
t_{PZL}/t_{PZH}	Three-State Enable Time			30 ns	36 ns/28 ns	30 ns	30 ns/23 ns		
t_{PLZ}/t_{PHZ}	Three-State Disable Time			25 ns	25 ns/20 ns	25 ns	25 ns/18 ns		

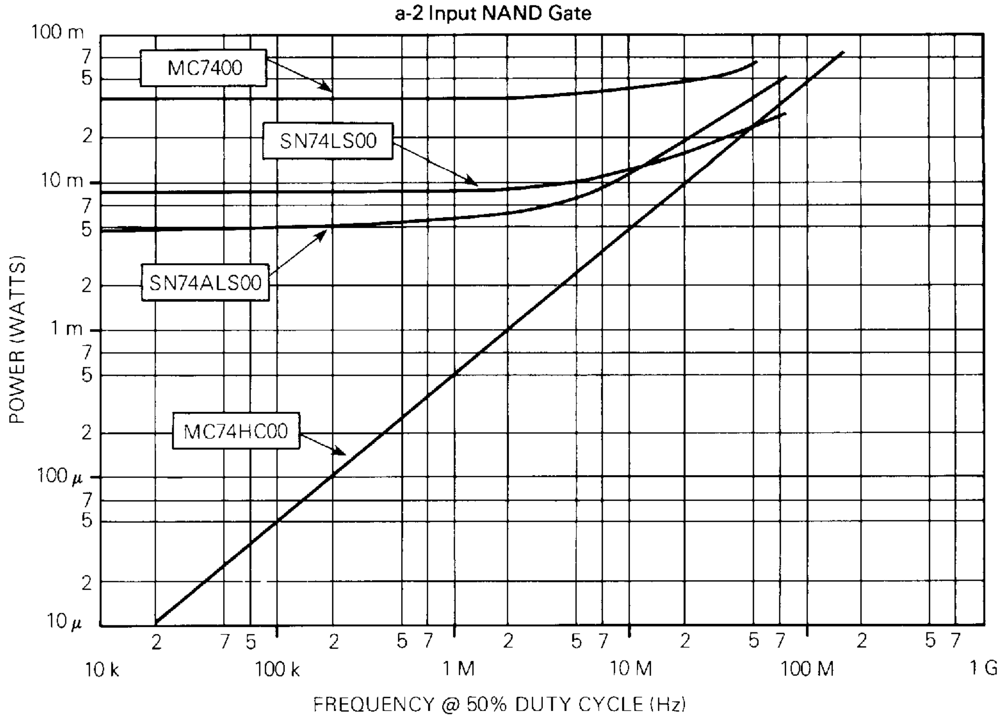


FIGURE 4a — Power Dissipation versus Device Frequency (No Load)
High-Speed CMOS versus TTL, LSTTL and ALS Typical Values

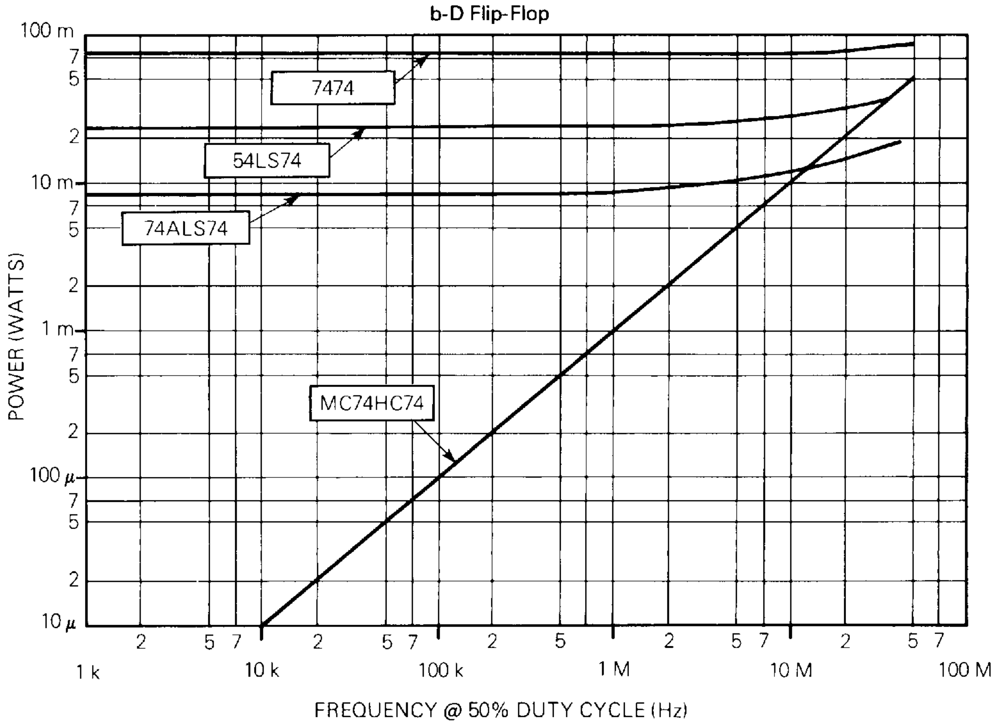


FIGURE 4b — Power Dissipation versus Device Frequency (No Load)
High-Speed CMOS versus TTL, LSTTL and ALS Typical Values

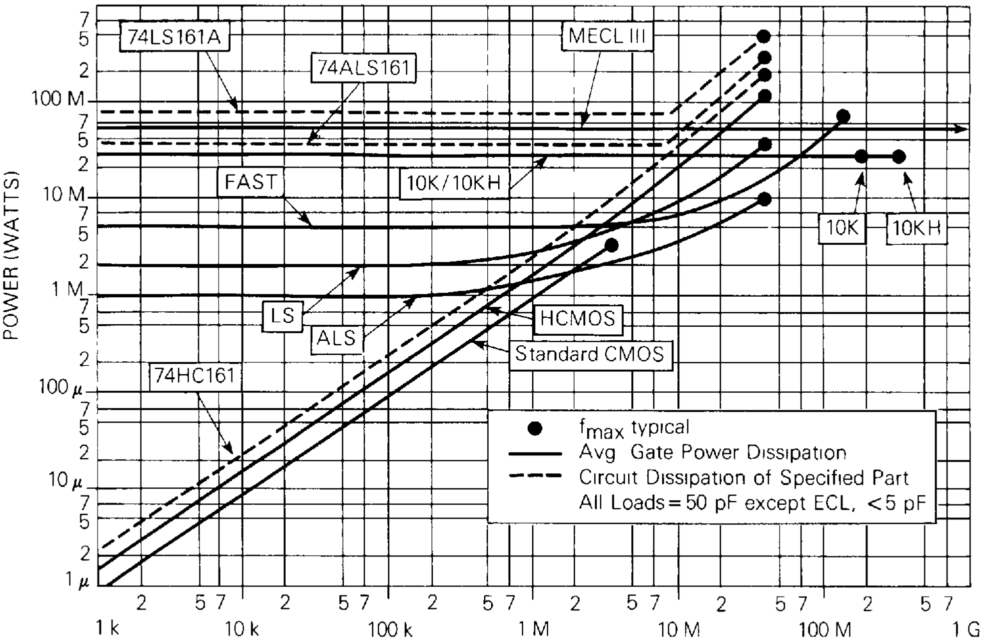


FIGURE 5 — Average Gate Power Dissipation versus Frequency

TABLE 5 — Noise Margin Comparison

Logic Family	Noise Margin	
	$V_{NL} = V_{IL} - V_{OL}$	$V_{NH} = V_{OH} - V_{IH}$
High-Speed CMOS MC74HCXX	0 95 V (19% V_{CC})	1 45 (29% V_{CC})
LSTTL 74LS series	0 40 V (8% V_{CC})	0 70 V (14% V_{CC})
Metal-Gate CMOS MC14000B series	1 45 V (29% V_{DD})	1 45 V (29% V_{DD})

peripherals used in the system. However, LSTTL can comprise a substantial portion of that power dissipation, depending on the size of the sytem, the operating frequency and the number of LSTTL devices in the system. HSCMOS devices could comprise an extremely small portion of the overall power dissipation of the system.

Noise Immunity

Noise margin values for HSCMOS, LSTTL, and metal-gate CMOS are outlined in Table 5. HSCMOS can be used in those MPU systems where noise may be a problem, i.e., industrial environments, automobiles. The noise immunity of NMOS devices is very poor by CMOS standards. Therefore, the noise immunity of an NMOS/HSCMOS system may be dictated by the NMOS devices. High-speed CMOS devices have about the same noise immunity properties as metal-gate CMOS devices.

The V_{NL} (noise margin low) for HSCMOS was intentionally made smaller than that for metal-gate CMOS. The HSCMOS input low voltage level is therefore more compatible with the TTL output voltage level.

HSCMOS devices will offer an improvement in noise margin in an NMOS microprocessor system over LSTTL. Table 6 shows the interface noise margins of CMOS and LSTTL interfaced with NMOS devices. Clearly, the noise

TABLE 6 — Interface Noise Margins for HSCMOS/NMOS and LSTTL/NMOS Interface

Interface	+ Interface Noise Margins	
	"1"	"0"
HSCMOS to NMOS	2 95 V	0 75 V
*NMOS to HSCMOS	1 10 V	0 60 V
LSTTL to NMOS	0 40 V	0 40 V
NMOS to LSTTL	0 40 V	0 40 V

*With pull-up resistor

+Interface Noise Margin

For "1" column — difference between output high level of one device and input high level of next device

For "0" column — difference between output low level of one device and input low level of next device margin of CMOS interfaced with NMOS is much better than LSTTL interfaced with NMOS.

As already shown, HSCMOS devices will interface quite easily with NMOS microprocessor memories and peripherals. It should be noted that these devices are directly compatible with CMOS MPUs and make a natural interface family for CMOS MPU systems. Until now, CMOS MPU systems have used low power Schottky devices for buffering and decoding requirements since the older metal-gate CMOS devices have propagation delays that are too long. Figure 6 shows an example where LSTTL devices would usually be required for the address decode logic since faster speeds are required than can be achieved with metal-gate CMOS. This example is an ideal application for high-speed CMOS devices.

Bus Buffering

There are two basic purposes for bus buffering in microprocessor systems. Generally, three-state buffers are needed to assert data and address information on the bus. However, buffers are also used to increase output drive currents when interfacing to keyboards, displays, etc., that require large drive currents or when driving large capacitances such as long lines and/or many inputs.

Asserting Information on the Bus

Three-state buffers are often used in the design of input ports, which must be able to disconnect themselves from the bus when they are not enabled and are used for asserting information onto the bus to be transferred to the microprocessor.

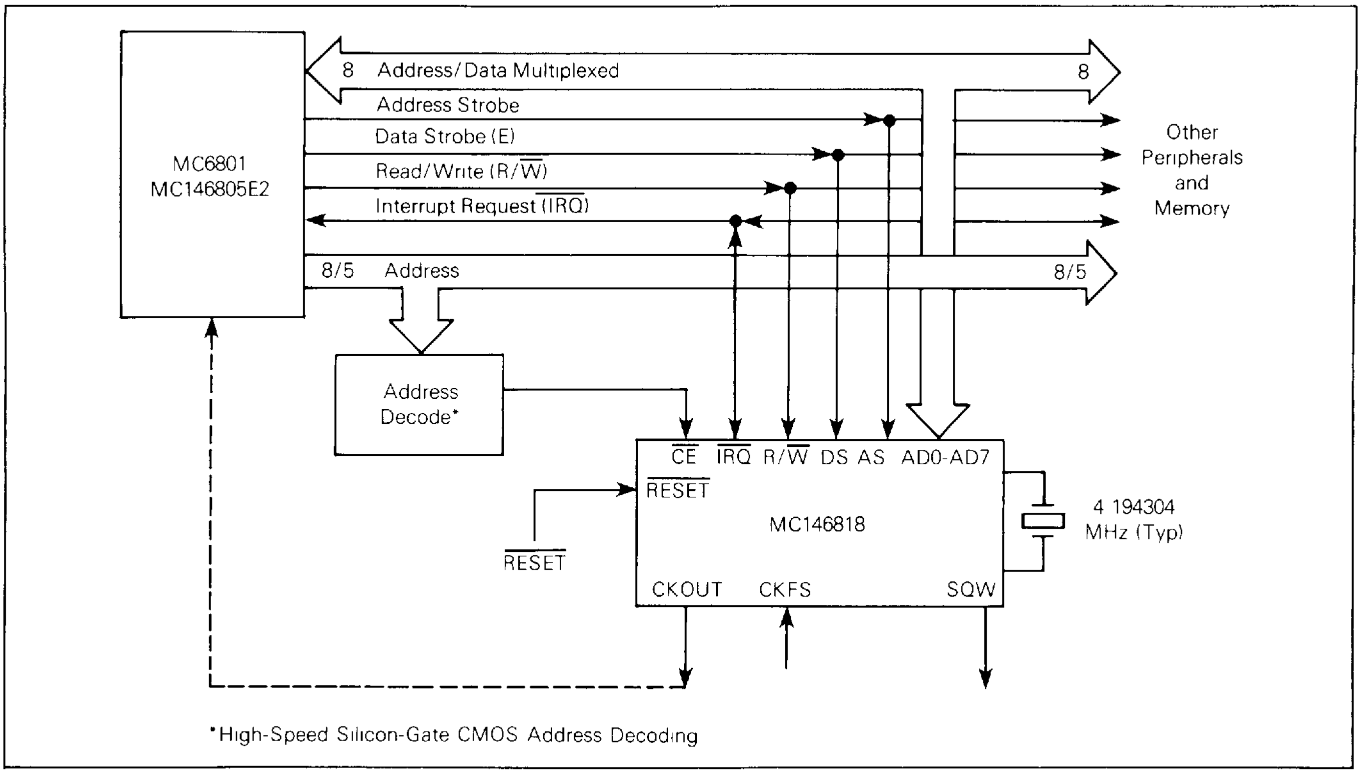
Input ports used to transfer data to the MPU may be constructed using many different three-state devices, such as the MC74HC240/241/244 octal buffers. Pinouts for these devices are shown in Figure 7. The 240 has inverting outputs while the 244 and 241 have noninverting outputs. These devices contain eight buffers in each package, which are divided into two sets of four buffers that are independently controlled by two enable inputs. The enable inputs are active low on the 240 and 244, while the 241 has one active low enable and one active high enable.

Since buffers can pass information from input to output only when they are enabled and represent a high impedance on the output when they are not enabled, they can effectively isolate a device from the bus when it is not being addressed. Figure 8 shows the use of three-state buffers for isolating the microprocessor's data bus from the bus of the Read/ Write (R/W) memory. In this case, a pair of MC74HC243 bus transceivers are used, since the data bus is bidirectional and data must be transmitted in both directions. The R/W memory data bus is isolated from the microprocessor data bus via the transceivers except when the processor does a read or a write of the R/W memory.

The MC74HC243 is a noninverting quad bus transceiver, designed for asynchronous two-way communication between data busses. Figure 9 shows pinouts and truth tables for this device as well as its inverting counterpart, the MC74HC242, and the MC74HC245, which is a noninverting octal transceiver. For the quad transceivers, the states of the control inputs, A-to-B ENABLE and B-to-A ENABLE determine the direction of data flow and the states of the outputs. For the octal transceiver, direction of data flow is controlled by the DIRECTION pin and the ENABLE pin is used to disable the device so that the busses are isolated.

Increasing Drive Capability

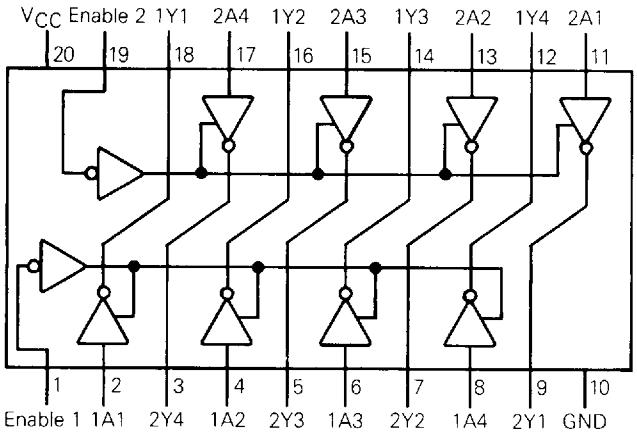
Buffering is also used in microprocessor systems to increase output drive capability for interfacing with devices that require large currents or for driving more loads than the



*High-Speed Silicon-Gate CMOS Address Decoding

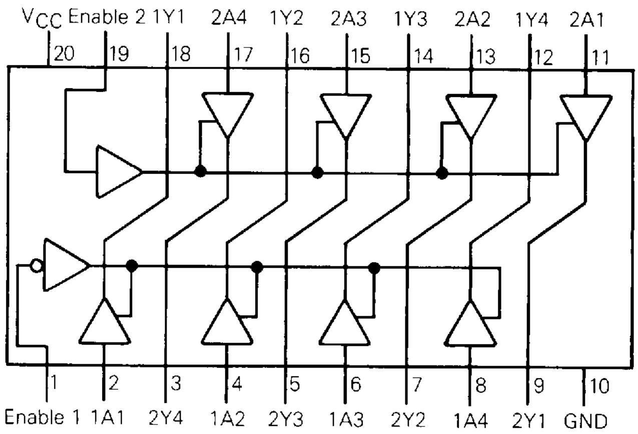
FIGURE 6 — MC146818 Interfaced to Motorola Compatible Multiplexed Bus Microprocessors

MC54HC240•MC74HC240
Inverting, Active Low



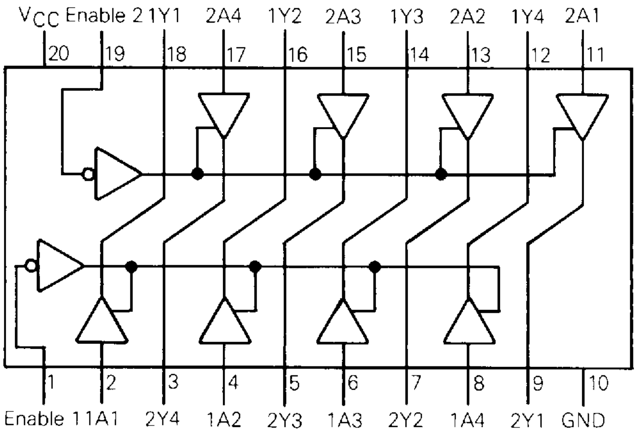
Inputs		Output
Enable 1 Enable 2	A	Y
L	L	H
L	H	L
H	X	Z

MC54HC241•MC74HC241
Noninverting; 4 Active Low, 4 Active High



Inputs		Output	Inputs		Output
Enable 1	A	1Y	Enable 2	A	2Y
L	L	L	L	L	Z
L	H	H	L	H	Z
H	L	Z	H	L	L
H	H	Z	H	H	H

MC54HC244•MC74HC244
Noninverting, Active Low



Inputs		Output
Enable 1, Enable 2	A	Y
L	L	L
L	H	H
H	X	Z

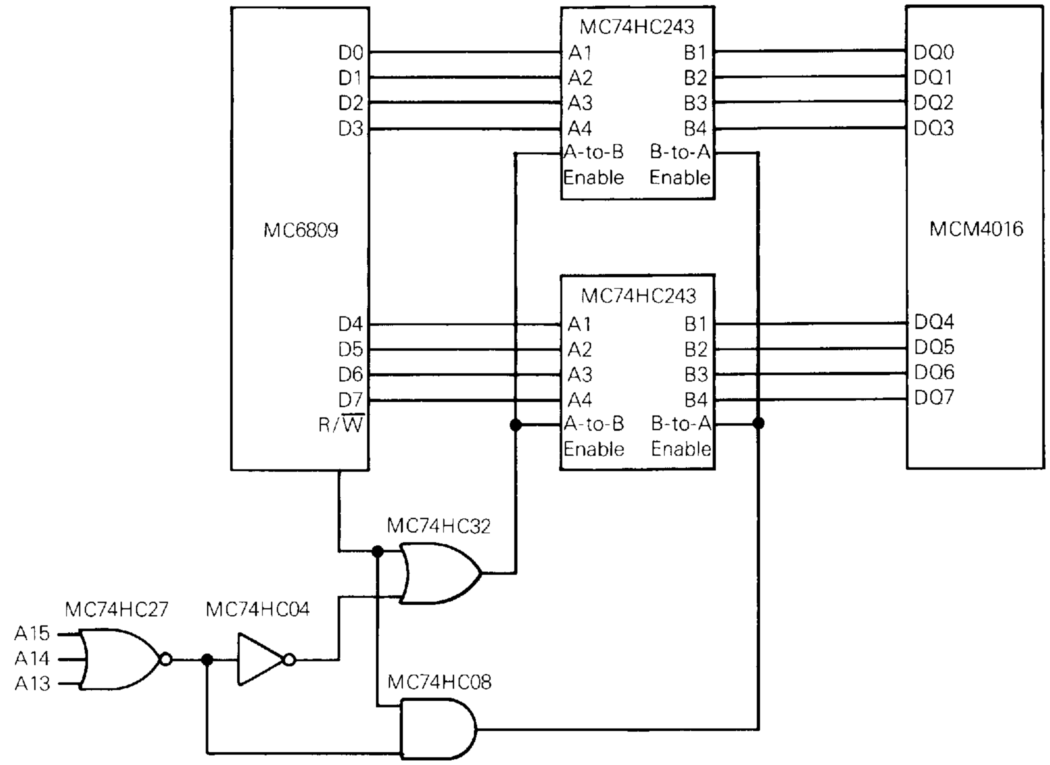
FIGURE 7 — Pinouts and Truth Tables for MC54HC240/241/244

device is specified to drive. Buffers are also used to increase drive on microprocessor address and data bus lines because of the high current and capacitive loading of the system bus.

When designing bus oriented systems, special attention must be given to capacitive loading of the bus. There is a practical limit to the number of devices that may be put on a bus and, also, consideration must be given to speed limitations imposed by capacitive loading. Capacitance acts as a load to the device driving the bus and has an effect on rise/fall times and propagation delays of the device.

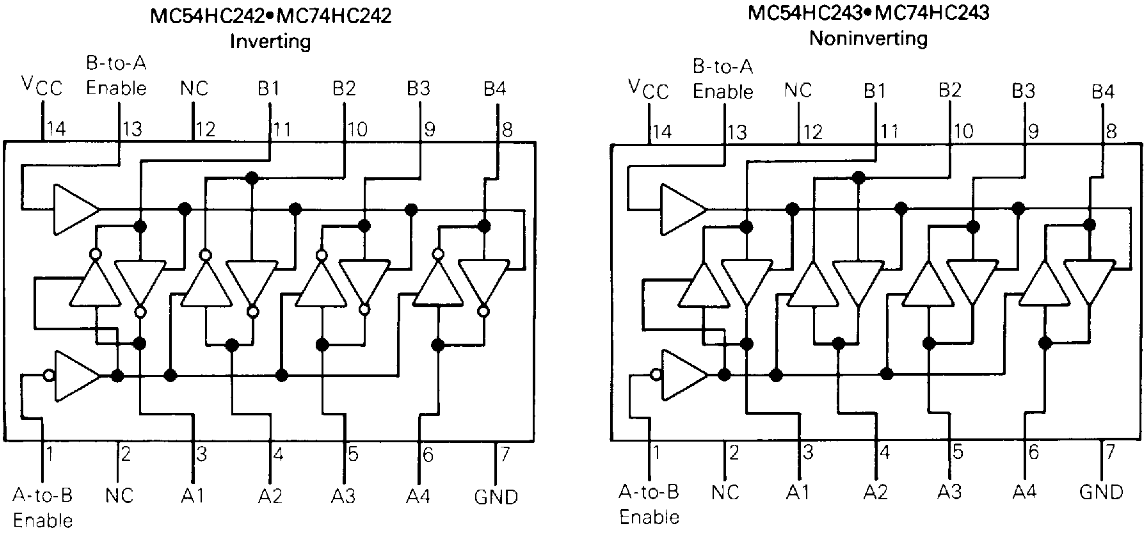
Timing specifications are given for most microprocessors, memories, and peripherals with a certain load capacitance on each line. For example, timing specifications are given for the MC6809 MPU for a load capacitance of 90 pF on each address line (A0-A15) and the R/$\overline{W}$ line, 130 pF for the data lines (D0-D7), and 30 pF for other lines. As the system becomes larger and more devices are put on these lines, a determination of the load capacitance on each line is necessary so that timing specifications are not violated. If the load capacitance is greater than that specified, buffers must be used to increase the drive capability of each line.

For increasing drive capability on data lines, transceivers (MC74HC242/243/245) such as those already described may be used. For increasing current drive on unidirectional lines, such as address lines, buffers such as the MC74HC240/241/244 may be used. All of these HSCMOS devices have three-



*Pullup Resistors Are Needed When Going From NMOS to HSCMOS

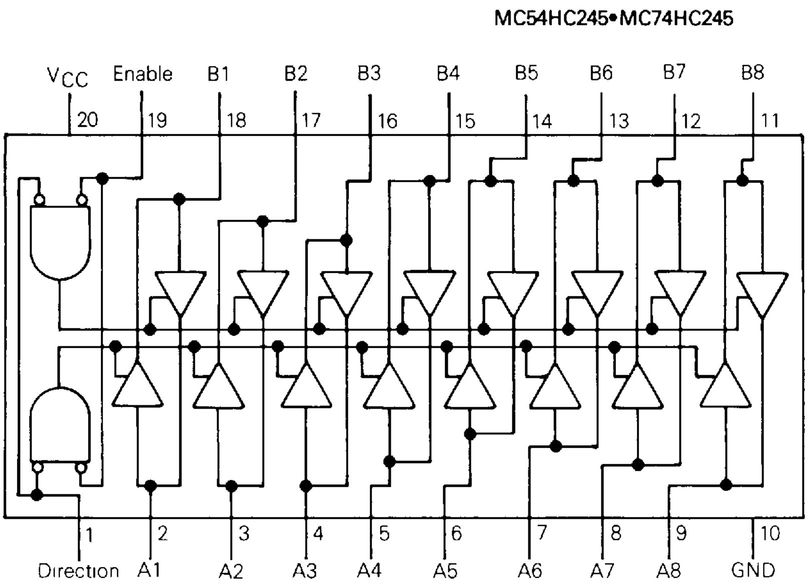
FIGURE 8 — The Use of Three-State Buffers for Isolating Data Busses



Control Inputs		MC54/74HC242 Data Port Status		MC54/74HC243 Data Port Status	
A-to-B Enable	B-to-A Enable	A	B	A	B
1	1	$\overline{O}$	I	O	I
0	1	Z	Z	Z	Z
1	0	Z	Z	Z	Z
0	0	I	$\overline{O}$	I	O

I = Input, O = Output, $\overline{O}$ = Inverting Output, Z = High-Impedance

FIGURE 9 — Pinouts and Truth Tables for MC74HC242/243/245



Control Inputs		Operation
Enable	DIR	245
L	L	B data to A bus
L	H	A data to B bus
H	X	Isolation

H = high level, L = low level, X = irrelevant

FIGURE 9 — Pinouts and Truth Tables for MC74HC242/243/245 (Continued)

state outputs and are available in octal or quad format for easy use with data and address busses. These devices can drive fifteen LSTTL loads for an output drive current of 6 mA. (The 54 series drives twelve LSTTL loads for 5.1 mA.)

In the case of driving a heavily loaded system bus, however, the drive capability of these devices will not suffice and, therefore, standard TTL buffers are better to use for this purpose because of their higher sinking capability of 48 mA. However, the HSCMOS buffers may be connected in parallel to increase the sinking capability.

Timing Considerations

Figure 10 shows the generic bus timing diagram for Motorola's NMOS microprocessors. The usable, or the overall time between a valid address output from the microprocessor and the required input data valid (or read setup time) is the time provided by the microprocessor for address buffers and decoders, data buffers, and memory access times. This access time may be computed using the formula in note 9 of Figure 10 for a given bus rate (1 MHz, 1.5 MHz or 2 MHz). At 1 MHz, Table 7 shows that for the MC6809, 695 ns are available for decoding, buffering, etc. Also, notes 10, 11, and 12 show how to compute usable address buffer time, read data buffer time and write data buffer time, which are included in Table 7.

Using the values in Table 7, available buffer times may be applied to the configuration shown in Figure 11. At 1 MHz, 150 ns are available for the two 74HC240 address buffers, which is ample time. Each buffer has maximum propagation delays of 30 ns. This leaves a margin of 90 ns (150 ns-60 ns) available for more buffering or for addressing decoding.

After the peripheral output data is valid, data transmitted to the MPU must be valid on or before the read data setup time, so that propagation delay through the data bus buffers must be done quickly to ensure valid data to the MPU. The read data buffer time at 1 MHz is 80 ns, leaving a margin of 20 ns after the propagation delays of 30 ns for each MC74HC243. The write data buffer time is 365 ns, which leaves a margin of 305 ns after total propagation delays of 60 ns.

This example clearly shows that the 74HC buffers are fast enough for a 1 MHz MPU system.

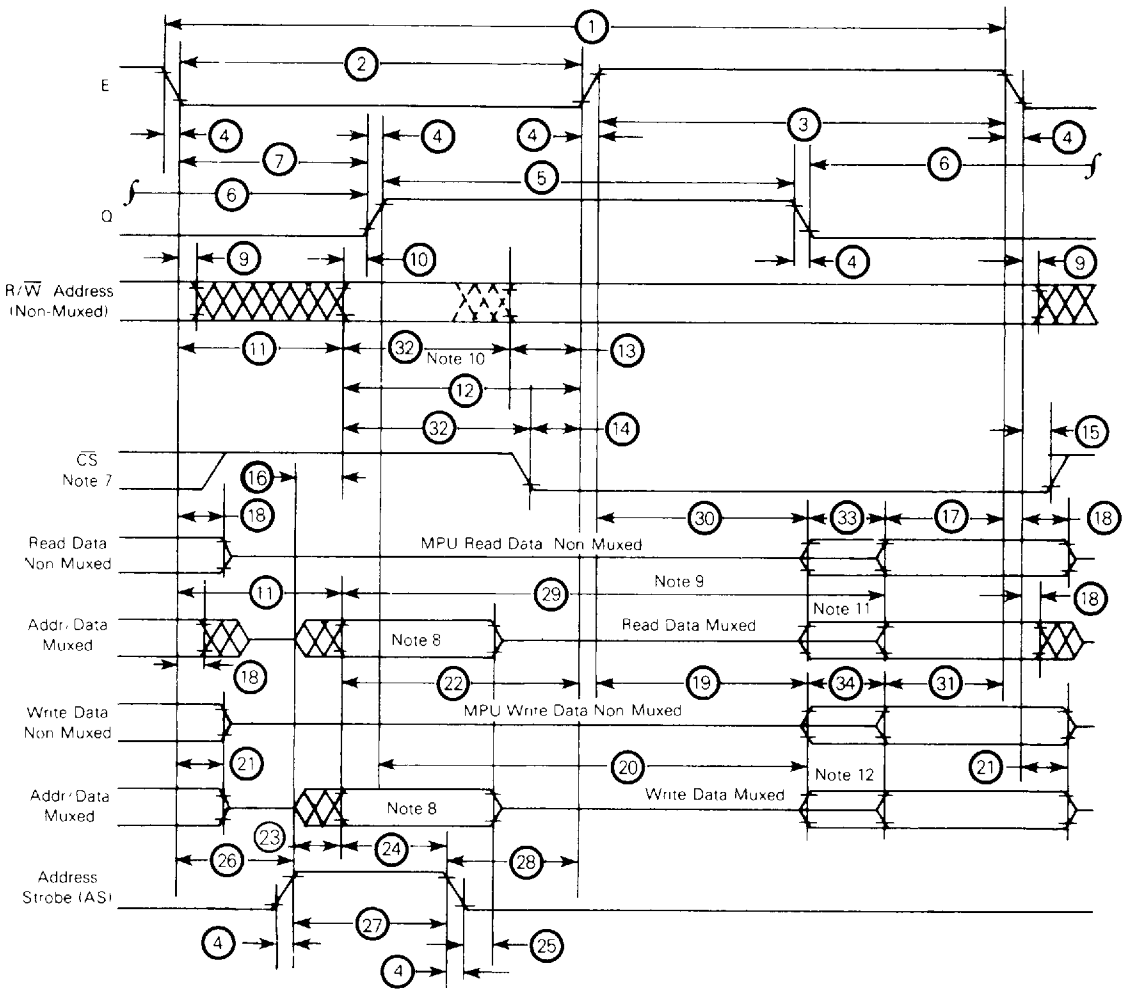
Address Decoding

Address decoding is another important function used in all microprocessor systems. Address decoding is important for achieving adequate communication between the MPU and other devices in the system. It is used for assigning individual addresses to other devices within the system. Some microprocessors, such as those made by Motorola, communicate with other devices (memories, peripherals, etc.) by treating them as separate memory locations. This is called memory mapped I/O and no special I/O instructions are needed by the MPU. Since each device is treated as a separate memory location, each device must be assigned an address. The task of assigning addresses can be accomplished using address decode logic made up of simple gates or special address decoders, depending upon what is required.

Two very popular devices used for address decoding are the MC74HC138 and the MC74HC154 decoders. The pinouts and truth tables for these devices are shown in Figure 12. The 74HC138 is a three-to-eight line decoder, which selects one of eight outputs, depending on the states of the three select inputs (A, B, C) and the three enable inputs (CS1, $\overline{\text{CS2}}$, $\overline{\text{CS3}}$).

The 154 decoder is a four-to-16 line decoder, which selects one of 16 outputs depending on the states of the four inputs (A, B, C, D) and the strobe inputs (CS1, CS2). This decoder provides broad address decoding flexibility and may be used to nonabsolutely decode addresses for 16 devices, as shown in Figure 13. Nonabsolute decoding means that the device being addressed responds to several addresses. The output selected will be low, while the remaining 15 outputs will be high.

Figure 14 shows an example of address decoding using the MC74HC138 decoder in a CMOS MPU system to perform address decoding between the MC146805E2 MPU and the MC146818 real time clock. This system requires some type of high-speed decoding since the time available for decoding is so small, for an internal bus rate of 1 MHz. Figure 15 shows



NOTES
1 Not all signals are applicable to every part
2 Voltage levels shown are $V_L \leq 0\ 4$ V, $V_H \geq 2\ 4$ V, unless otherwise specified
3 Measurement points shown are 0 8 V and 2 0 V unless otherwise specified
4 For MC6800 write data is referenced to DBE not E see M6800 pages
5 For MC6800 address delay is referenced to $\phi 1$ not E see M6800 pages
6 Clock pulse rise and fall time for MC6800 measured to V_{CC} - 0 6 V see M6800 pages
7 CS and $\overline{CS}$ on MC6810 have same timing
8 Address valid on the occurrence of the latest of 11 12, 16 22 or 23
9 Usable access time is computed by 1 – (4 + 11 + 17) or 12 + 3 - 17 see note 8 (except for MC6809 for MC6809 by 1 – 4 – 7 max + 10 – 17)
10 Usable address buffer time is computed by 2 – (11 + 13) see note 8 (except for MC6809, for MC6809, by 2 – 7 max + 10 – 13)
11 Usable read data buffer time is computed by 3 – (17 + 30)
12 Usable write data buffer time is computed by 3 – (19 + 31) except for MC6809 for MC6809 by 1 – (4 + 7 max + 4 + 20 + 31)

FIGURE 10 — NMOS/HMOS Generic Bus Timing Diagram

NMOS/HMOS GENERIC BUS TIMING CHARACTERISTICS FOR 1.0 MHz OPERATION

Ident. Number	Characteristics	Symbol	6800		6801		6802		6809		6821/ 6859		Unit
			Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	
1	Cycle Time	t_{cyc}	1 0	10	1 0	2 0	1 0	10	1 0	10	1 0	10	μs
2	Pulse Width, E Low (See Note 6)	PW_{EL}	405	9500	430	1000	450	5000	430	5000	430	9500	ns
3	Pulse Width, E High (See Note 6)	PW_{EH}	450	9500	450	1000	450	9500	450	9500	450	9500	ns
4	Clock Rise and Fall Time (See Note 6)	t_r, t_f	–	100	–	25	–	25	–	25	–	25	ns
5	Pulse Width, Q High	PW_{QH}	–	–	–	–	–	–	430	5000	–	–	ns
6	Pulse Width, Q Low	PW_{QL}	–	–	–	–	–	–	450	9500	–	–	ns
7	Delay Time, E to Q Rise*	t_{AVQ}	–	–	–	–	–	–	200	250	–	–	ns
9	Address Hold Time	t_{AH}	30	–	20	–	20	–	20	–	10	–	ns
10	Address Valid Time to Q Rise*	t_{AQ}	–	–	–	–	–	–	50	–	–	–	ns
11	Address Delay from E Low (See Note 5)	t_{AD}	–	270	–	–	–		–			–	ns
12	Non Muxed Address Valid Time to E* (MPU)	t_{AV}	–	–	200	–	160	–	–	–	–	–	ns
13	Addres Setup Time Before E (Periph)	t_{AS}	–	–	–	–	–	–	–	–	80	–	ns
14	Chip Select Setup Time Before E	t_{CS}	–	–	–	–	–	–	–		80	–	ns
15	Chip Select Hold Time	t_{CH}	–	–	–	–	–	–	–	–	10	–	ns
16	Non-Muxed Address Delay Time from AS	t_{AD}	–	–	–	–	–	–	–	–	–	–	ns
17	Read Data Setup Time	t_{DSR}	100	–	80	–	100	–	80	–	–	–	ns
18	Read Data Hold Time	T_{DHR}	10	–	10	–	10	–	10	–	20	50	ns
19	Write Data Delay Time (See Note 4)	t_{DDW}	–	225	–	225	–	225	–	–	–	–	ns
20	Data Delay Time from Q	t_{DDQ}	–	–	–	–	–	–	–	200	–	–	ns
21	Write Data Hold Time (See Note 4)	t_{DHW}	10	–	20	–	30	–	30	–	10	–	ns
22	Muxed Address Valid Time to E Rise*	t_{AVM}	–	–	200	–	–	–	–	–	–	–	ns
23	Muxed Address Delay Time from AS	t_{ADAS}	–	–	–	–	–	–	–	–	–	–	ns
24	Muxed Address Valid Time to AS Fall*	t_{ASL}	–	–	60	–	–	–	–	–	–	–	ns
25	Muxed Address Hold Time	t_{AHL}	–	–	20	–	–	–	–	–	–	–	ns
26	Delay Time, AS to E Rise*	t_{ASD}	–	–	90	–	–	–	–	–	–	–	ns
27	Pulse Width AS High*	PW_{ASH}	–	–	220	–	–	–	–	–		–	ns
28	Delay Time, AS to E Rise*	t_{ASED}	–	–	90	–	–	–	–	–	–	–	ns
29	Usable Access Time* (See Note 9)	t_{ACC}	605	–	570	–	605	–	695	–	–	–	ns
30	Peripheral Output Data Delay Time	t_{DDR}	–	–	–	–	–	–	–		–	290	ns
31	Peripheral Input Data Setup Time	t_{DSW}	–	–	–	–	–	–	–	–	165	–	ns
32	Buffer Logic Delay Time Address, $\overline{CS}$, R/$\overline{W}$ (See Note 10)	t_{BDA}	55		120	–	100	–	150	–	–	–	ns
33	Buffer Delay Time Read Data (See Note 11)	t_{BDR}	60	–	80	–	60	–	80	–	–	–	ns
34	Buffer Delay Time, Write Data (See Note 12)	t_{BDW}	60	–	60	–	60	–	365	–	–	–	ns

*At specified cycle time

FIGURE 10 — NMOS/HMOS Generic Busing Timing Diagram (Continued)

TABLE 7 — Available Access Time and Buffer Delay Times For MC6809

	MC6809 (1MHZ)	MC68A09 (1.5MHZ)	MC68B09 (2MHZ)
Usable Access Time	695 ns	442 ns	330 ns
Buffer Logic Delay Time	150 ns	80 ns	60 ns
Buffer Delay Time (Read Data)	80 ns	30 ns	30 ns
Buffer Delay Time (Write Data)	365 ns	230 ns	160 ns

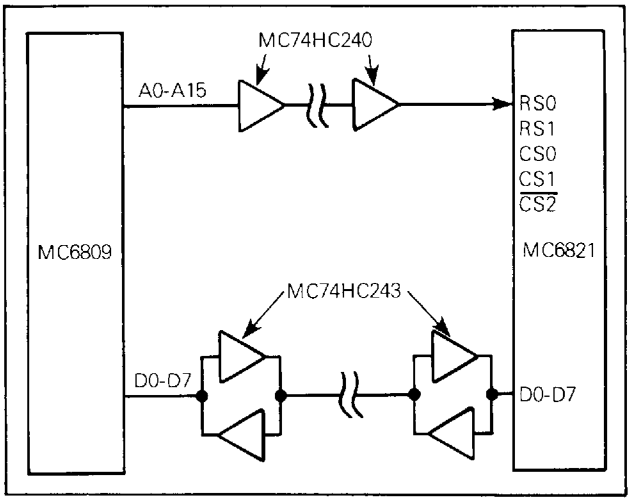
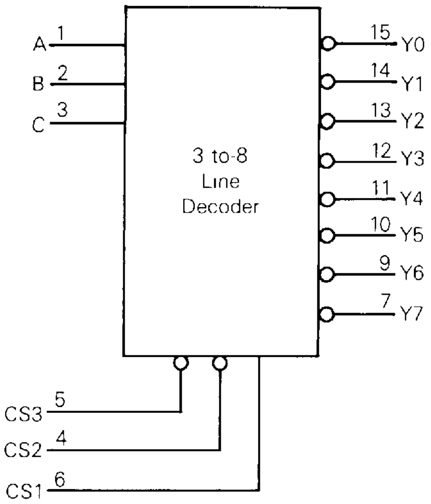


FIGURE 11 — Typical Configuration Using Address and Data Bus Buffers

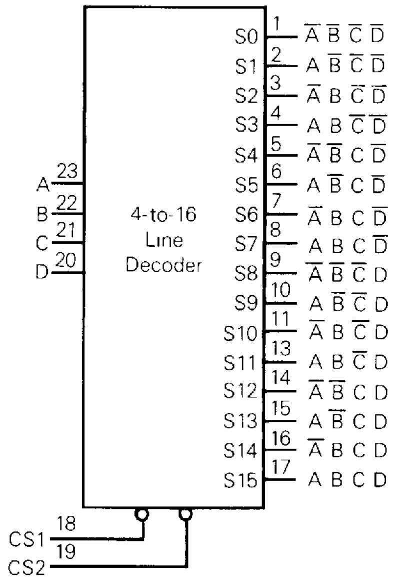


MC74HC138 BLOCK DIAGRAM

TRUTH TABLE

Inputs						Outputs							
Chip Select			Output Select										
CS1	CS2	CS3	C	B	A	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
X	X	H	X	X	X	H	H	H	H	H	H	H	H
X	H	X	X	X	X	H	H	H	H	H	H	H	H
L	X	X	X	X	X	H	H	H	H	H	H	H	H
H	L	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	L	H	H	L	H	H	H	H	H	H
H	L	L	L	H	L	H	H	L	H	H	H	H	H
H	L	L	L	H	H	H	H	H	L	H	H	H	H
H	L	L	H	L	L	H	H	H	H	L	H	H	H
H	L	L	H	L	H	H	H	H	H	H	L	H	H
H	L	L	H	H	L	H	H	H	H	H	H	L	H
H	L	L	H	H	H	H	H	H	H	H	H	H	L

H = high level (steady state)
L = low level (steady state)
X = don't care


A 23
B 22
C 21
D 20
4-to-16
Line
Decoder
S0 1 $\overline{A}\,\overline{B}\,\overline{C}\,\overline{D}$
S1 2 $A\,\overline{B}\,\overline{C}\,\overline{D}$
S2 3 $\overline{A}\,B\,\overline{C}\,\overline{D}$
S3 4 $A\,B\,\overline{C}\,\overline{D}$
S4 5 $\overline{A}\,\overline{B}\,C\,\overline{D}$
S5 6 $A\,\overline{B}\,C\,\overline{D}$
S6 7 $\overline{A}\,B\,C\,\overline{D}$
S7 8 $A\,B\,C\,\overline{D}$
S8 9 $\overline{A}\,\overline{B}\,\overline{C}\,D$
S9 10 $A\,\overline{B}\,\overline{C}\,D$
S10 11 $\overline{A}\,B\,\overline{C}\,D$
S11 13 $A\,B\,\overline{C}\,D$
S12 14 $\overline{A}\,\overline{B}\,C\,D$
S13 15 $A\,\overline{B}\,C\,D$
S14 16 $\overline{A}\,B\,C\,D$
S15 17 $A\,B\,C\,D$
CS1 18
CS2 19

MC74HC154 BLOCK DIAGRAM

TRUTH TABLE

CS1	CS2	Data Inputs				Selected Output at Logic "0"
Inhibit		D	C	B	A	
0	0	0	0	0	0	S0
0	0	0	0	0	1	S1
0	0	0	0	1	0	S2
0	0	0	0	1	1	S3
0	0	0	1	0	0	S4
0	0	0	1	0	1	S5
0	0	0	1	1	0	S6
0	0	0	1	1	1	S7
0	0	1	0	0	0	S8
0	0	1	0	0	1	S9
0	0	1	0	1	0	S10
0	0	1	0	1	1	S11
0	0	1	1	0	0	S12
0	0	1	1	0	1	S13
0	0	1	1	1	0	S14
0	0	1	1	1	1	S15
1	X	X	X	X	X	All Outputs = 1
X	1	X	X	X	X	

X = Don't Care

FIGURE 12 — BLock Diagrams and Truth Tables for MC74HC138/154 Decoders

the bus timing diagram for the MC146805E2 CMOS microprocessor. The time available for address decoding may be derived by the following formula:

Non-mux time = #27 − #16 − #24 (peripheral)

For a bus rate of 200 kHz, a total of 600 ns is available (850 ns-200 ns-50 ns), which is more than adequate time for most decoding schemes. For a bus rate of 1 MHz, there are 25 ns available for address decoding. These are both worst case values. The typical propagation delay for the 74HC138 decoder is only 20 ns and should provide adequate decoding for the example shown in Figure 14.

Data Latching

Finally, a third important function used in microprocessor systems is the latching of information, such as address, data or control information, to be held until a peripheral or the microprocessor itself is ready for it.

Latches and flip-flops are two versatile devices used in MPU systems for performing many different kinds of latching functions. They may be used for constructing output or input ports, for low-order address latching, or for allowing an asynchronous bus to be interfaced to a synchronous bus. These are but a few uses for latches and flip-flops.

Figure 16 shows an example of using an eight-bit D-type flip-flop, such as the MC74HC374/534, to hold input data from an encoded keyboard before being captured by the Peripheral Interface Adapter (PIA), since the PIA does not have an input latch. The active-low strobe latches the data and provides a transition on the control line CA1 of the PIA. Pinouts and truth tables for the 74HC374 (noninverting) and

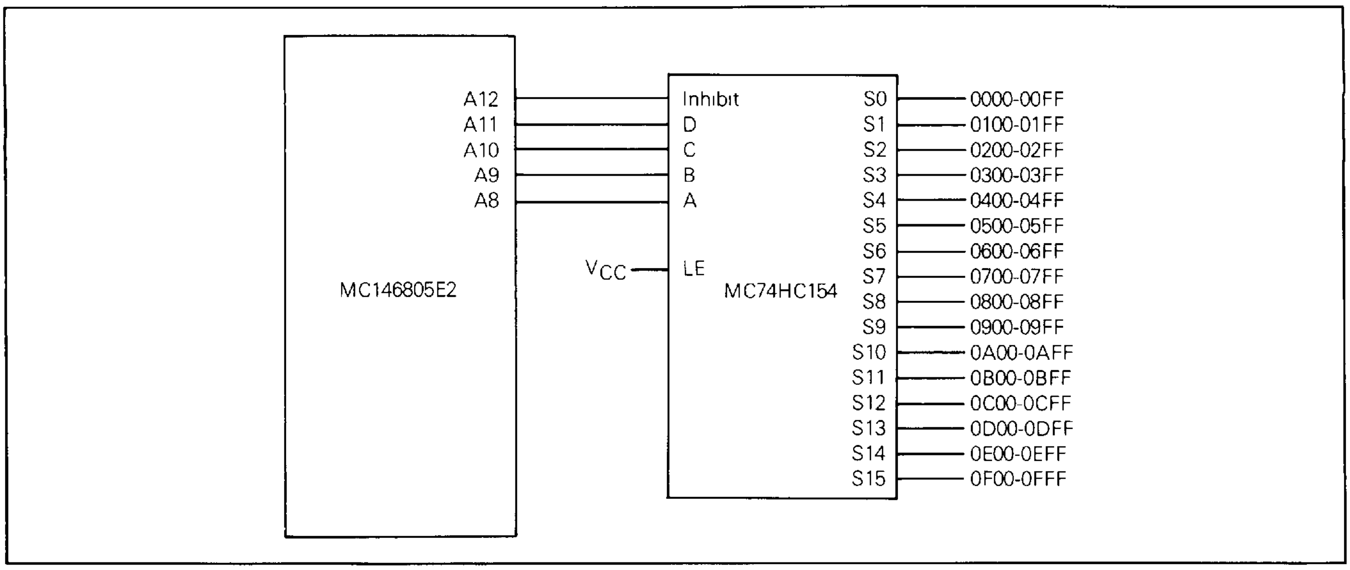


FIGURE 13 — The MC74HC154 Decoder Used to Nonabsolutely Produce 16 Decoded Device Selects

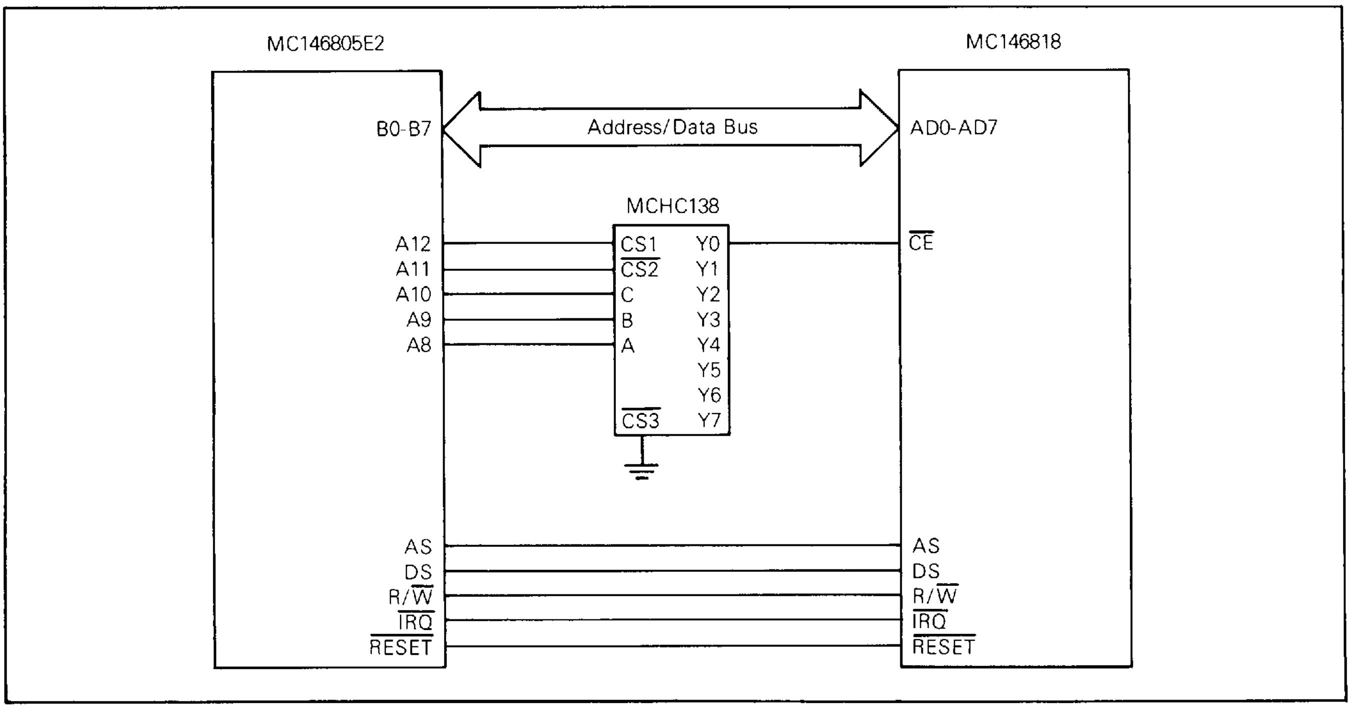
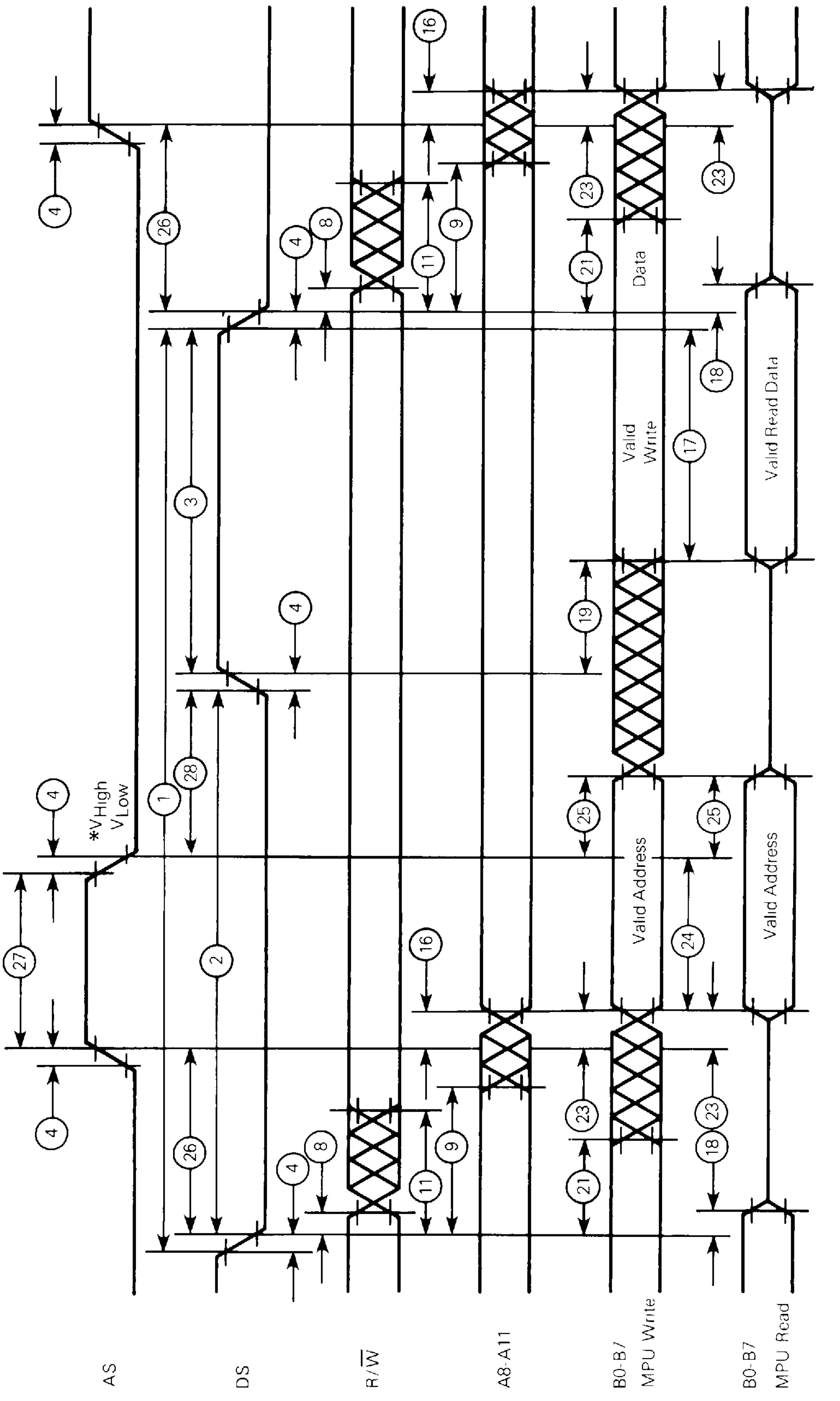


FIGURE 14 — The MC74HC138 Used for Decoding in a CMOS System

Motorola reserves the right to make changes 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



*V_{High} = 2.0 V, V_{Low} = 0.5 V for V_{DD} = 3 V
V_{High} = V_{DD} − 2.0 V, V_{Low} = 0.8 V for V_{DD} = 5 V ± 10%

FIGURE 15 — Bus Timing for the MC146805E2

Bus Timing ($T_A = T_L$ to T_H, $V_{SS} = 0$ V)

Num	Characteristics	Symbol	$f_{osc} = 1$ MHz $V_{DD} = 3.0$ V 50 pF Load		$f_{osc} = 5$ MHz $V_{DD} = 5.0$ V ± 10%, 1 TTL and 130 pF Load		Unit
			Min	Max	Min	Max	
1	Cycle Time	t_{cyc}	5000	dc	1000	dc	ns
2	Pulse Width, DS Low	PW_{EL}	2800	–	560	–	ns
3	Pulse Width, DS High	PW_{EH}	1800	–	375	–	ns
4	Clock Transition	t_r, t_f	–	100	–	30	ns
8	R/$\overline{W}$ Hold	t_{RWH}	10	–	10	–	ns
9	Non-Muxed Address Hold	t_{AH}	800	–	100	–	ns
11	R/$\overline{W}$ Delay from DS Fall	t_{AD}	–	500	–	300	ns
16	Non-Muxed Address Delay from AS Rise	t_{ADH}	0	200	0	100	ns
17	MPU Read Data Setup	t_{DSR}	200	–	115	–	ns
18	Read Data Hold	t_{DHR}	0	1000	0	160	ns
19	MPU Data Delay, Write	t_{DDW}	–	0	–	120	ns
21	Write Data Hold	t_{DHW}	800	–	55	–	ns
23	Muxed Address Delay from AS Rise	t_{BHD}	0	250	0	120	ns
24	Muxed Address Valid to AS Fall	t_{ASL}	600	–	55	–	ns
25	Muxed Address Hold	t_{AHL}	250	750	60	180	ns
26	Delay DS Fall to AS Rise	t_{ASD}	800	–	160	–	ns
27	Pulse Width, AS High	PW_{ASH}	850	–	175	–	ns
28	Delay, AS Fall to DS Rise	t_{ASED}	800	–	160	–	ns

FIGURE 15 — Bus Timing for the MC146805E2 (Continued)

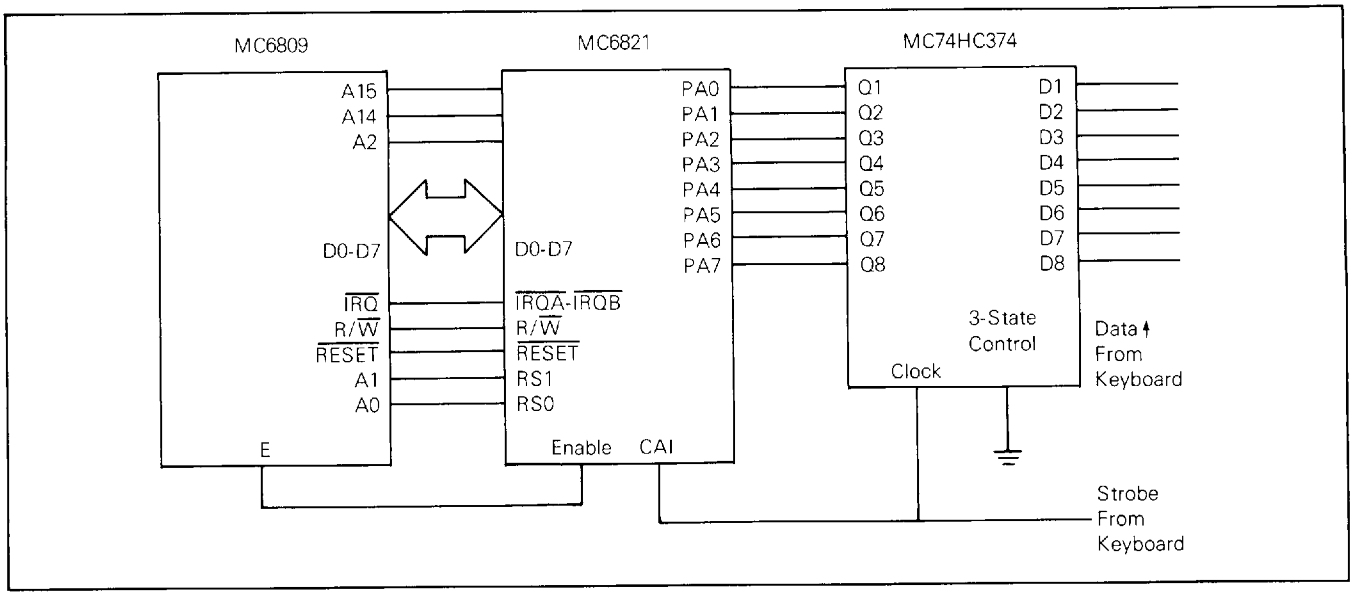


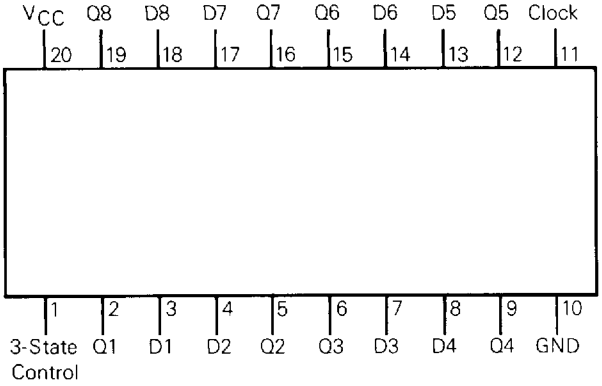
FIGURE 16 — Using the MC74HC374 Octal D Flip-Flop as an Input Port

the 74HC534 (inverting) octal flip-flops are shown in Figure 17. For these flip-flops, data meeting the setup times are clocked to the outputs with the rising edge of the clock. Also, Three-State Control does not affect the state of the flip-flops; however, when Three-State Control is high, the output is forced to the high-impedance state, so that information may be latched even when the device is not selected.

Figure 18 shows an example of using the MC74HC373 octal latch for latching the low-order address from the multiplexed data/address bus of the 146805E2 MPU which is interfaced with the static RAMs. The AS (address strobe) line on the microprocessor is used to indicate the presence of an address on the multiplexed bus and to demultiplex the low order byte of the address from the data bus. The latch is controlled by the AS line and captures the address byte on the negative edge of AS.

Pinouts and truth tables are shown in Figure 19 for the MC74HC373/533 octal latches. These latches are transparent latches, which means that they appear transparent to data (or the outputs change asynchronously) when Latch Control is

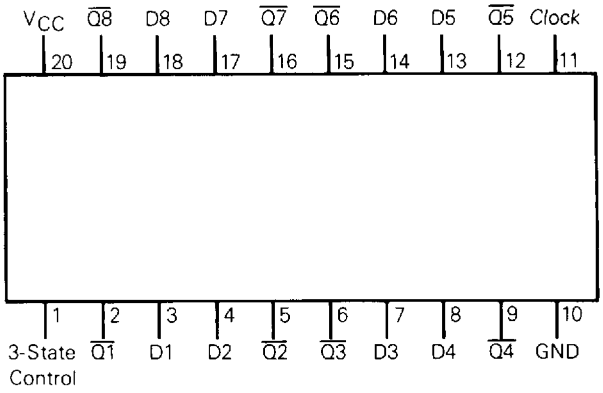
MC54HC374•MC74HC374
Octal D-Type Flip-Flop, Noninverting Output



3-State Control	Latch Control	D	Output
L	↑	H	H
L	↑	L	L
L	L	X	no change
H	X	X	Z

X = don't care
Z = high impedance

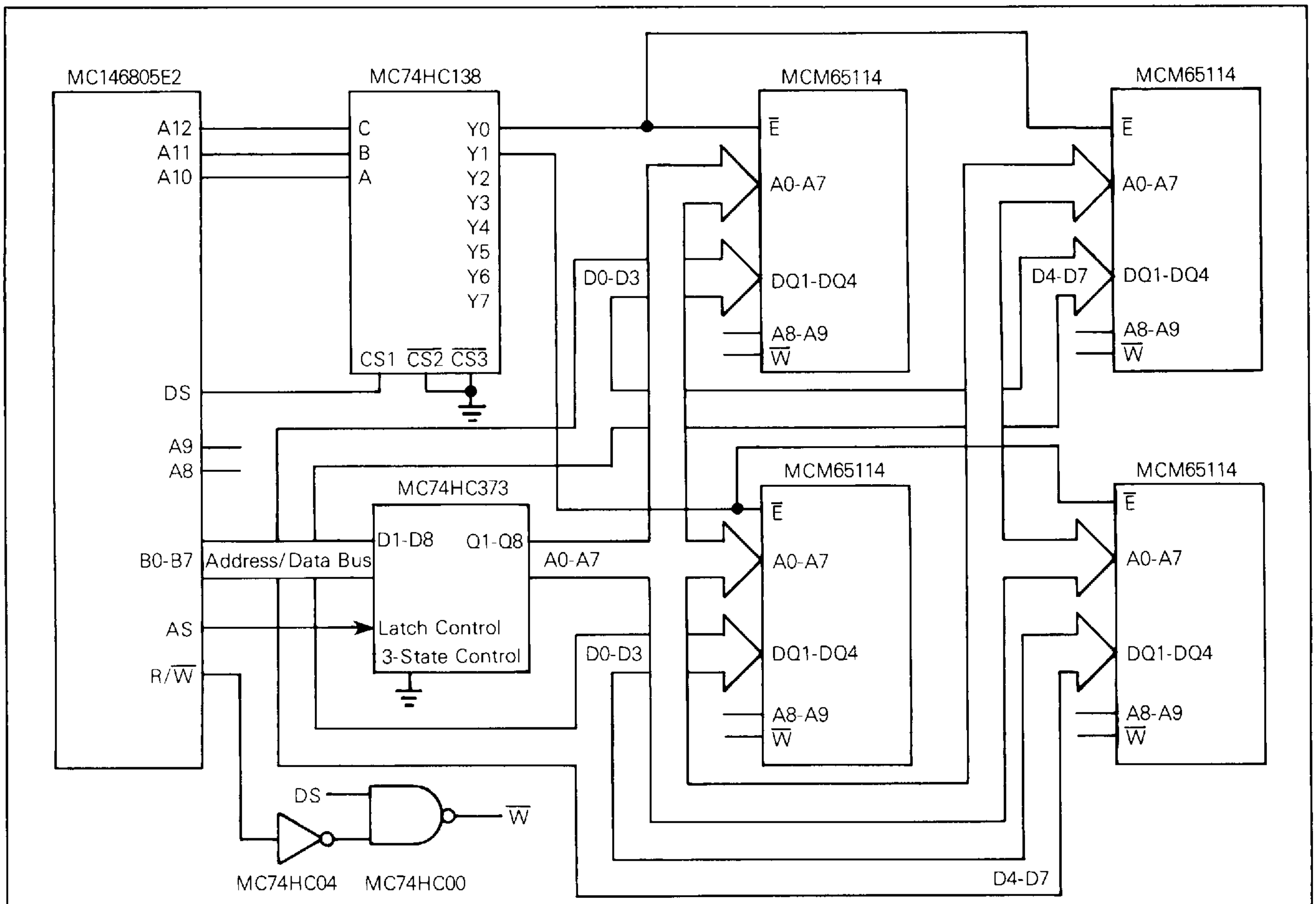
MC54HC534
Octal D-Type Flip-Flop, Inverting Output



3-State Control	Latch Control	D	Output
L	↑	H	L
L	↑	L	H
L	L	X	no change
H	X	X	Z

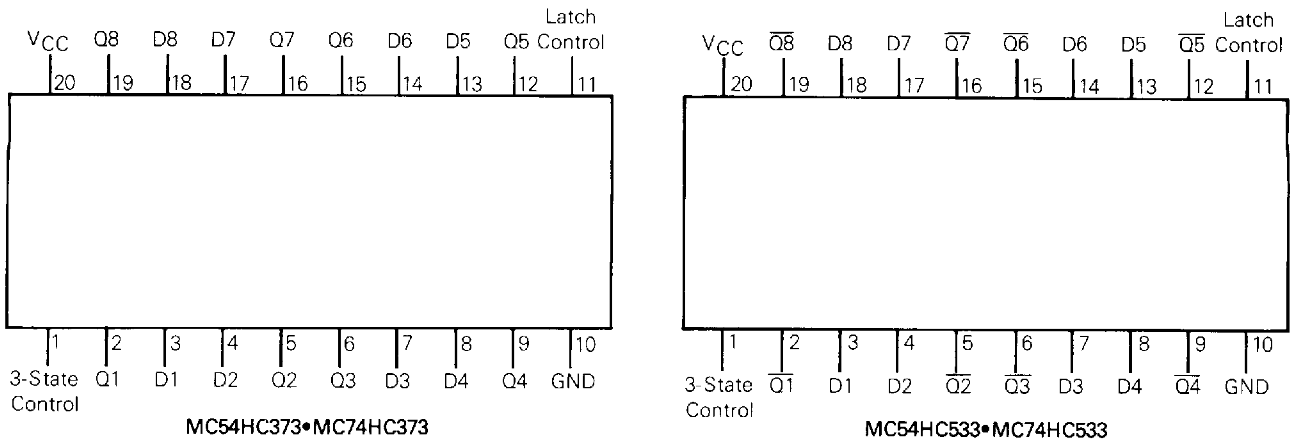
X = don't care
Z = high impedance

FIGURE 17 — Pinouts and Truth Tables For MC74HC374/534 Octal D-Type Flip-Flops



NOTE MCM65114 is a 1024 x 4 Bit CMOS Static RAM The four RAMs are configured to provide a memory size of 2048 x 8 bits

FIGURE 18 — Using the MC74HC373 Latch for Latching the Low-Order Address Off the 146805E2 Data Bus



MC54HC373•MC74HC373
Octal D-Type Transparent Latch, Noninverting Output

3-State Control	Latch Control	D	Output
L	H	H	H
L	H	L	L
L	L	X	no change
H	X	X	Z

X = don't care
Z = high impedance

MC54HC533•MC74HC533
Octal D-Type Transparent Latch, Inverting Output

3-State Control	Latch Control	D	Output
L	H	H	L
L	H	L	H
L	L	X	no change
H	X	X	Z

X = don't care
Z = high impedance

FIGURE 19 — Pinouts and Truth Tables for MC74HC373/533 Octal D-Type Transparent Latches

high. When Latch Control is low, data meeting the setup time is latched. The three-state Control line functions the same as on the MC74HC374/534 octal D flip-flops.

Referring to the timing diagram in Figure 15, addresses for the MC146805E2 MPU are defined to be valid prior to the fall of address strobe by time #24, which is 55 ns for this MPU. This is sufficient time to allow capture of the address by the 74HC373.

All of these devices have maximum propagation delays of 30 ns which are fast enough for most applications.

Conclusion

High-speed CMOS devices offer the system designer yet another alternative for meeting the digital logic interfacing needs of his system. The designer may prefer to use these devices in his NMOS microprocessor system, rather than LSTTL, in order to improve the system's noise margins and to reduce the system's power dissipation. Also, HSCMOS devices can drive more capacitance than metal-gate CMOS devices, making HSCMOS more compatible with NMOS MPU systems. The designer should keep in mind that pullup resistors may be required when going from NMOS devices to HSCMOS devices, unless the HCT series devices with TTL compatible input levels are used.

These devices would ideally be used in CMOS MPU systems where LSTTL has been required previously because of speed. These devices are directly compatible with CMOS and are now fast enough to do the address decoding, latching, and buffering required in these systems. Therefore, a true CMOS MPU system may be realized, which could be used in many low power, high noise applications.

There are other bus compatible devices offered in the high-speed CMOS logic product line that have not been mentioned in this application note. These devices are listed in Table 8.

TABLE 8 — Bus Compatible Devices Offered in High Speed CMOS.

HC173	4-Bit D-Type Register, 3-State
HC240	Octal Buffer/Line Driver/Line Receiver, 3-State, Inverting Output
HC241	Octal Buffer/Line Driver/Line Receiver, 3-State
HC242	Quad Bus Transceiver, 3-State, Inverting Output
HC243	Quad Bus Transceiver, 3-State
HC244	Octal Buffer/Line Driver/Line Receiver, 3-State
HC245	Octal Bus Transceiver, 3-State
HC257	Quad 2-Input Data Selector/Multiplexer, 3-State
HC373	Octal D-Type Transparent Latch, 3-State
HC374	Octal D-Type Flip-Flop, 3-State
HC533	Octal D-Type Transparent Latch, 3-State, Inverting Output
HC534	Octal D-Type Flip-FLop, 3-State, Inverting Output
HC646	Octal Bus Transceiver and Register, 3-State
HC648	Octal Bus Transceiver and Register, 3-State

a. CMOS Input Compatible Interface Circuits

MC54/74HCT245	Octal Three-State Transceiver
MC54/74HCT573	Non-Inverting Three-State Octal Latch
MC54/74HCT574	Non-Inverting Three-State Octal Flip-Flop
MC54/74HCT576	Inverting Three-State Octal Latch
MC54/74HCT580	Inverting Three-State Octal Flip-FLop
MC54/74HCT640	Octal Inverting Three-State Transceiver

b. TTL Input Compatible CMOS Interface Circuits