6.1 REVIEWING THE THREE-BUS SYSTEM ARCHITECTURE

In one regard a microprocessor is a very simple machine. It endlessly follows the sequence:

1. Fetch the next instruction in sequence from memory.
2. Execute the instruction.
3. Go to step 1.

The fetch cycle is actually a *memory read* operation in which the byte or word "pointed to" by the program counter (or instruction pointer) is transferred from memory to the instruction register in the CPU. Execution of this instruction may require additional memory reads as in the 8086 instruction MOV BX,[MEMWDS]. Other possibilities for the execution phase include a *memory write* (MOV [MEMWDS],BX), an *I/O read* (IN AL,DPORT), an *I/O write* (OUT DPORT,AL), or an *internal* CPU activity (INC AX).

In all, there are five unique operations or *bus cycles* possible:

1. Memory read
2. Memory write
3. I/O read
4. I/O write
5. Bus idle (internal operation not requiring access to memory or I/O)

As we saw in Fig. 1.1, three sets of wires (or buses) are dedicated to this transferring of data between the CPU and the memory and I/O units. In this section we look at the timing between the address, data, and control buses for each of the (active) bus cycle types.

Bus Cycle Timing

Figure 6.1 illustrates bus cycle timing for the four active bus cycle types. Each cycle begins with the output of the memory or I/O port address during the T1 clock cycle. For the 8086 this can be a 20-bit memory address, a 16-bit indirect I/O address (using register DX), or an 8-bit direct I/O address.

Note that in Fig. 6.1 I have not attempted to draw all 20 address lines. Instead, the contents of the address bus is shown to change to a new address sometime during the T1 state. The parallel lines indicate that some of the lines are assumed to be high and others low. The specific address is unimportant for this discussion.

Examining the address lines only, it is not possible to determine if this is a memory

or an I/O address. Neither can you tell the *direction* of the data flow. That is, is the CPU performing a read or a write cycle? For this reason a *control* bus is required. As shown in Fig. 6.1, this bus consists of the four active-low signals

1. $\overline{\text{MEMR}}$
2. $\overline{\text{MEMW}}$
3. $\overline{\text{IOR}}$
4. $\overline{\text{IOW}}$

To see how the three buses shown in Fig. 6.1 work together, consider the sequence of events that occur during a memory read bus cycle.

T1. The processor outputs the 20-bit memory address. The data lines are open circuited and all control lines disabled.

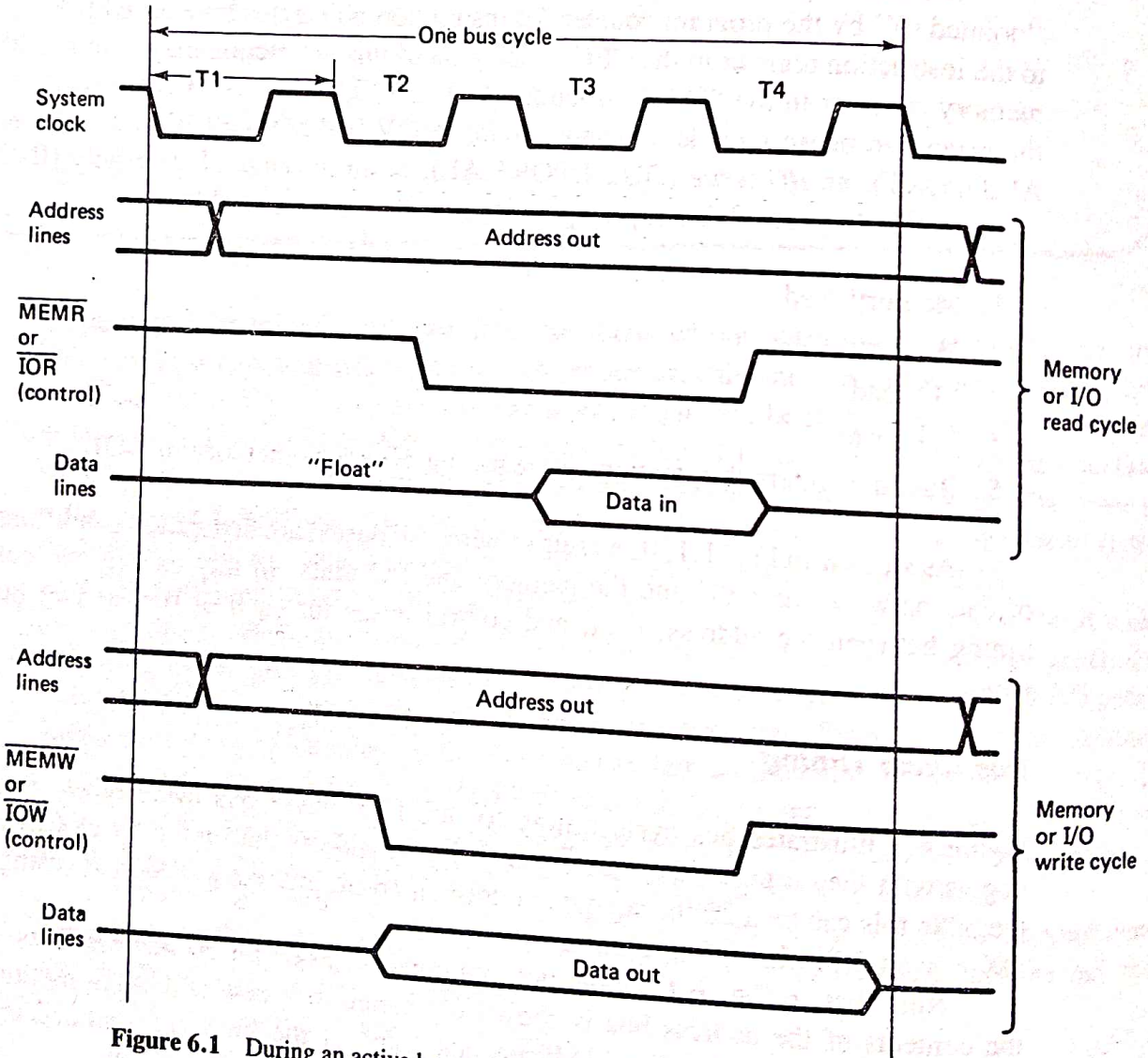

Figure 6.1 During an active bus cycle the microprocessor may perform a memory read, memory write, I/O read, or I/O write operation. The control and address buses are used to specify the memory or I/O address and the direction of data flow on the data bus lines.

T2. The $\overline{\text{MEMR}}$ control line is driven low. The memory unit recognizes this bus cycle as a memory read and prepares to place the addressed byte or word onto the data lines.

T3. The microprocessor configures its data bus lines for input but takes no further action. This state is provided primarily to give the memory time to "look up" the data byte or word.

T4. The microprocessor now expects the data to be on the data bus lines. Therefore, it latches the contents of these lines and releases the memory read control signal. This marks the end of the bus cycle.

The most important point to note is that the microprocessor controls all the bus timing. The memory must be able to supply the selected data byte or word by the time $\overline{\text{MEMR}}$ goes high during the T4 state. If it cannot do so, the CPU will read random information on the data bus lines. This will lead to unpredictable results.[1]

Example 6.1

Describe the contents of the address, data, and control bus lines when the instruction MOV [1000H],BX is executed. Assume that register DS = 09D3H and register BX = 1234H.

Solution The instruction requires a memory write to location 1000H of the data segment. In this case the physical address is 09D30H + 1000H = 0AD30H. This 20-bit address will be placed on the address lines. The control bus signal $\overline{\text{MEMW}}$ will go low during T2 and the data bus will contain 1234H, the contents of register BX. The memory is assumed to latch this data word by the time $\overline{\text{MEMW}}$ returns high in the T4 state.

There is one final point to make about the control bus timing—but it should be obvious. *Only one control signal can be active at a given time*. The processor cannot read from its memory at the same time it is outputting to an I/O device, for example. This is a very important point and the key to the memory and I/O interface circuits covered in Chaps. 7 and 8.

SELF-REVIEW 6.1 (answers on page 269)

6.1.1. During an instruction fetch a _____ _____ bus cycle is performed.

6.1.2. Execution of the instruction OUT DPORT,AX requires a(n) _____ _____ bus cycle.

6.1.3. Under what conditions will the 8086 buses be idle?

6.1.4. Using a logic analyzer the three buses of an 8086 microprocessor are recorded at a particular instant.

Address bus: 47000H
Control bus: $\overline{\text{MEMR}}$ = 0, $\overline{\text{MEMW}}$ = 1, $\overline{\text{IOR}}$ = 1, $\overline{\text{IOW}}$ = 1
Data bus: F3C0H

Explain the type of bus cycle occurring.

[1]In Chap. 7 we see how the memory unit can request "wait" states, effectively extending the T3 state by an integral number of clock cycles. In this way slow-memory devices can be interfaced.