TABLE 6.2 Memory Map of On-Board Data Memory

Memory Range	Range Size	Purpose	Addressing Mode(s)
0x80–0xFF	128 bytes	General-purpose RAM (except 8051)	Indirect only
0x80–0xFF	128 bytes	Special-function registers	Direct only
0x30–0x7F	80 bytes	General-purpose RAM	Direct/indirect
0x20–0x2F	16 bytes	Bit-addressable RAM/general	Direct/indirect
0x18–0x1F	8 bytes	Register bank 3/general	Direct/indirect
0x10–0x17	8 bytes	Register bank 2/general	Direct/indirect
0x08–0x0F	8 bytes	Register bank 1/general	Direct/indirect
0x00–0x07	8 bytes	Register bank 0/general	Direct/indirect

through R7 in the currently selected bank) whose value is used to index into that portion of RAM. The lower 128 bytes of RAM are accessible via both direct and indirect addressing.

The 8051 is a good study in maximizing the capabilities of limited resources. Access to external memory is supported through a variety of indirect and indexed schemes that provide an option to the system designer of how extensive an external bus is implemented. Indirect access to external data memory is supported in both 8- and 16-bit address configurations. In the 8-bit mode, R0 through R7 are used as memory pointers, and the resulting address is driven only on I/O port 0, freeing port 2 for uses other than as an address bus. The DPTR functions as a pointer into data memory in 16-bit mode, enabling a full 64-kB indirect addressing range. Indexed access to external program memory is supported by both the DPTR and the PC. Being program memory (ROM), only reads are supported. Both DPTR and PC can serve as index base address registers, and the current value in the accumulator serves as an offset to calculate a final address of either DPTR+A or PC+A.

The 8051's external bus interface is asynchronous and regulated by four basic control signals: ALE, program storage enable (PSEN*), read enable (RD*), and write enable (WR*). Figure 6.5 shows the interaction of these four control signals and the two bus ports: ports 0 and 2. Recall that ALE causes an external latch to retain A[7:0] that is driven from port 0 during the first half of the access and prior to port 0 transitioning to a data bus role. The timing delays noted are for a standard 12-MHz operating frequency (the highest frequency supported by the basic 8051 devices, although certain newer devices can operate at substantially faster frequencies).*

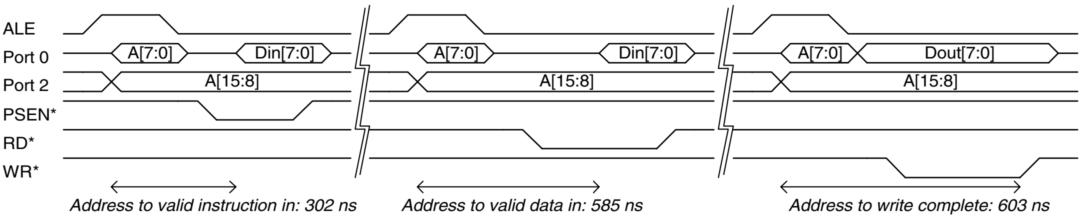
ALE
Port 0
Port 2
PSEN*
RD*
WR*
A[7:0]
Din[7:0]
A[15:8]
A[7:0]
Din[7:0]
A[15:8]
A[7:0]
Dout[7:0]
A[15:8]
Address to valid instruction in: 302 ns
Address to valid data in: 585 ns
Address to write complete: 603 ns

FIGURE 6.5 8051 bus interface timing.

* *Embedded Controller Handbook,* Vol. I, Intel, 1988, pp. 10-6 through 10-9.

Although the specific timing delays of program memory and data memory reads are different, they exhibit the same basic sequence of events. (More time is allowed for data reads than for instruction reads from program memory.) Therefore, if the engineer properly accounts for the timing variations by selecting memory and logic components that are fast enough to satisfy the PSEN* and RD* timing specifications simultaneously, program and data memory can actually be merged into a unified memory space external to the chip. Such unification can be performed by generating a general memory read enable, MRE*, that is the AND function of PSEN* and RD*. In doing so, whenever either read enable is driven low by the 8051, MRE* will be low. This can benefit some applications by turning the 8051 into a more general-purpose computing device that can load a program into its "data memory" and then execute that same program from "program memory." It also enables indexed addressing to operate on data memory, which normally is restricted to indirect addressing as discussed previously.

Timers such as those found in the 8051 are useful for either counting external events or triggering low-frequency events themselves. Each timer can be configured in two respects: whether it is a timer or counter, and how the count logic functions. The selection of timer versus counter is a decision between incrementing the count logic based on the microcontroller's operating frequency or on an external event sensed via an input port pin. The 8051's internal logic runs in a repetitive pattern of 12 clock cycles in which 1 machine cycle consists of 12 clock cycles. Therefore, the count logic increments once each machine cycle when in timer mode. When in counter mode, a low-to-high transition (rising edge) on a designated input pin causes the counter to increment. The counter can be configured to generate an interrupt each time it rolls over from its maximum count value back to its starting value. This interrupt can be used to either trigger a periodic maintenance routine at regular intervals (timer mode) or to take action once an external event has occurred a set number of times (counter mode). If not configured to generate an interrupt, the software can periodically poll the timer to see how many events have occurred or how much time has elapsed.

The timers inherently possess two 8-bit count registers that can be configured in a variety of ways as shown in Fig. 6.6. A timer can be configured as a conventional 16-bit counter, as two 8-bit

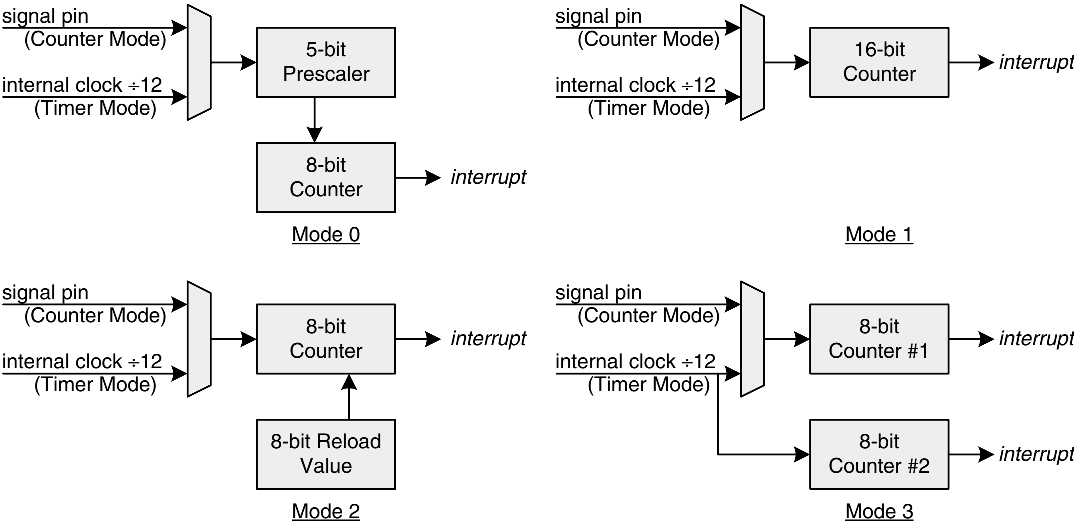

FIGURE 6.6 8051 timer configurations.