**ECEN 4613/5613 Embedded System Design Week #3**

**Spring 2013 Homework #3**

This assignment should be completed by Wednesday, February 6th

. Note: there is nothing to hand in for

this assignment. In this homework assignment, you will explore:

• External memory interfacing (e.g. EPROMs and SRAMs)

• Logic families and programmable logic

The majority of the assigned reading will be available on the course web site in PDF format.

NOTE: You should always try to use the data sheet provided by the manufacturer of the exact part

that you are using in your circuits, since there can be differences between manufacturers of similar

chips—even standard chips, such as the 74LSxx TTL logic family members. If you cannot find the

correct data sheet, then you may need to look at the data sheet for a similar part/manufacturer.

1. Review the final project assignment, available from the course web site.

2. Review the current topics presentation assignment, available from the course web site.

3. Read pages 1–18 of Philips application note AN457 "80C51 External Memory Interfacing". While

You’re reading it, keep in mind that the processor on the board that you will build will be running at

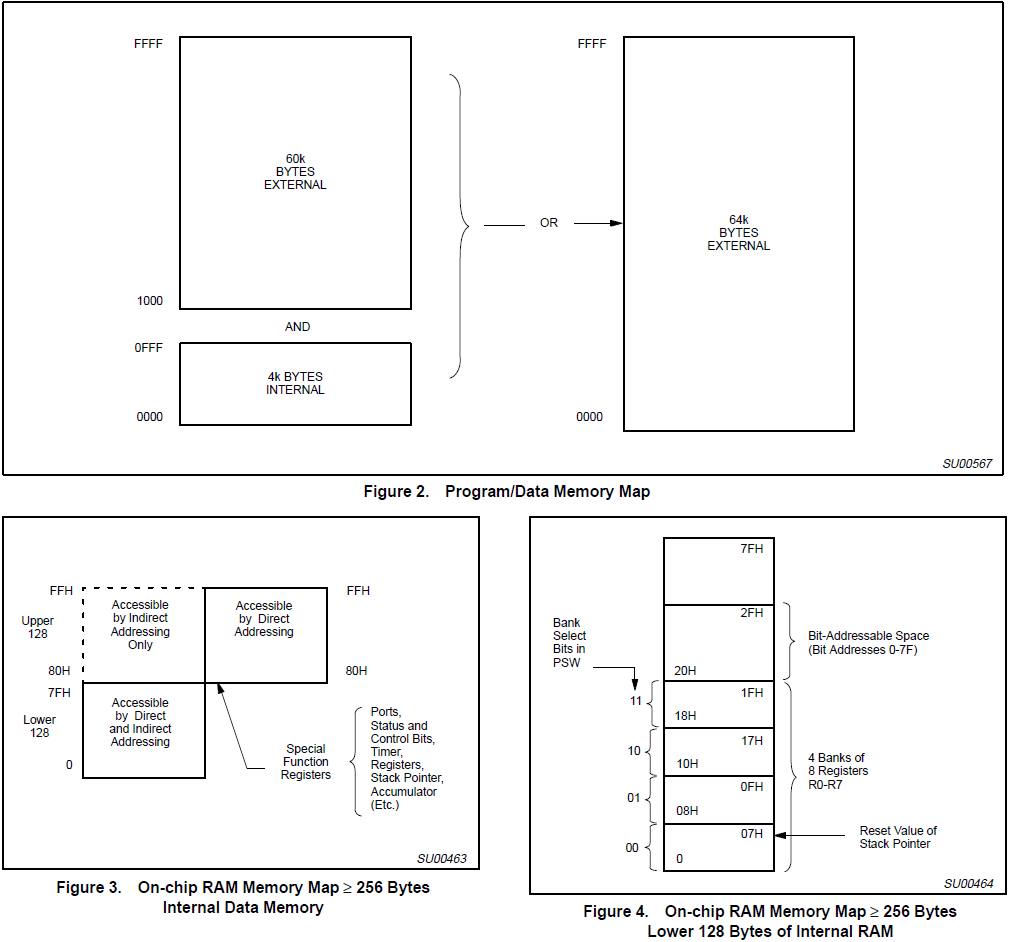
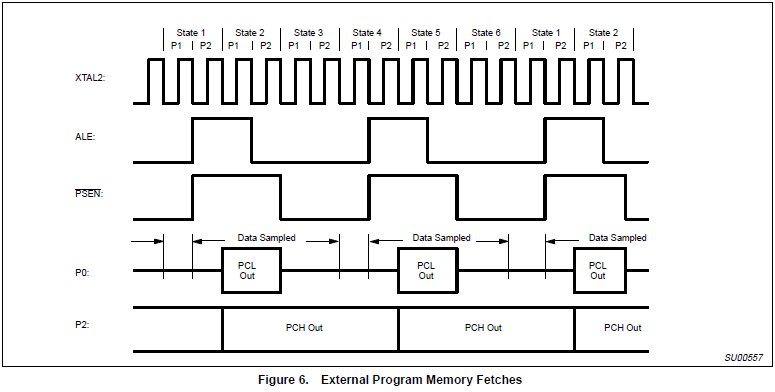
11.0592MHz, a significantly lower speed than 33MHz. **Take time to understand the timing diagrams**

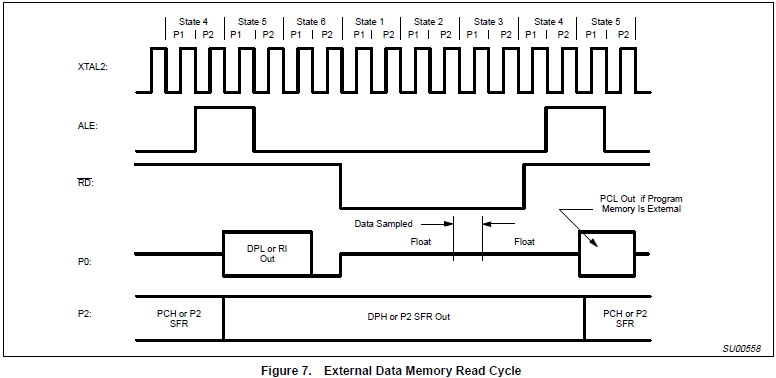
**And what each of the minimum and maximum timing specifications really means**. Take a few minutes

And **review the timing diagrams in the C501 or 80C51** product specification. Remember that **during a**

**Read cycle, the peripheral chip (EPROM or SRAM) is driving the data bus**, while **during a write cycle,**

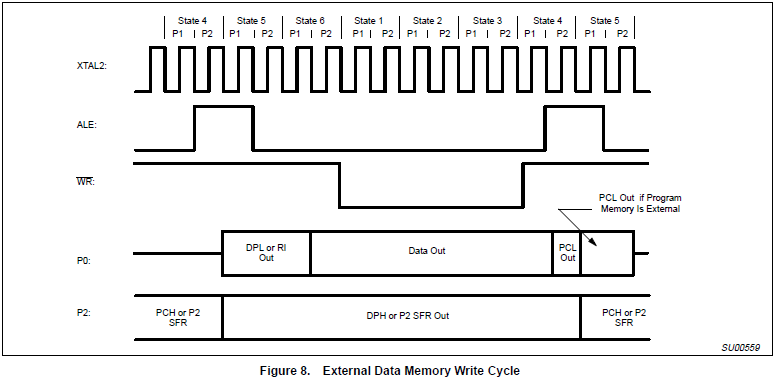
**The processor is driving the data bus**.

* *’51 is logically a ‘Harvard’*
* *EA is asserted, on-chip instruction (but not data) memory is disabled and the entire 64KB of instruction space is accessed externally (this is the only option for the 80C31). Otherwise (EA de-asserted), on-chip instruction memory is enabled and only addresses beyond the end of on-chip instruction memory (i.e., . 1000H for the 8XC51) are accessible externally.*
* *On-chip data memory (i.e., RAM) is somewhat different. First, all ’51 family devices include a basic complement of on-chip RAM comprised of CPU register banks, SFRs (Special Function Registers, i.e., built-in I/O functions such as UART, timer, etc.) and general purpose RAM.*
* *Diagram : *
* *Thus, there are three types of external access – instruction read (PSEN), data read (RD) and data write (WR) as shown in Figures 6, 7, and 8. *
* *ALE is essentially a continuous clock that runs at 1/6 the oscillator frequency regardless of the mix of internal and external accesses. However, note that one ALE cycle is skipped during external data access.*

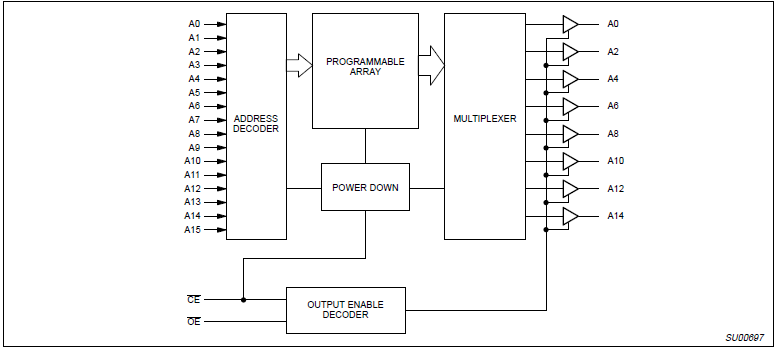
**

* *The skipping of the ALE cycle and assertion of RD or WR only occur during external data access. The MOVX instruction, and only the MOVX instruction, performs external data access. Thus, if the program contains no MOVX instructions, ALE can be used as a timebase (i.e., no skipping) and RD and WR as general purpose outputs.*
* *External program reads (PSEN), whatever the address or cause (i.e., fetch beyond the end of internal instruction memory or EA pin asserted at reset) always use 16-bit addresses and thus require all pins of P0 and P2. [P0 & P2 can be used for general purpose I/O during non-PSEN times? P0 SFR is overwritten by PSEN but what about P2?]*
* *External data accesses affect on P0 and P2 is a little more complicated. 16-bit address accesses (MOVX using DPTR) always drive P0 and P2 with A0–A15. However, 8-bit address accesses*

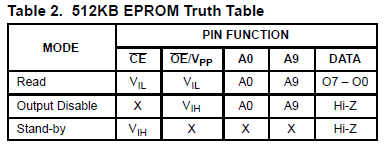
*(MOVX using Ri) instead drive P2 with the value programmed into the P2 SFR, i.e., P2 is essentially general purpose I/O during an 8-bit address external data access.*

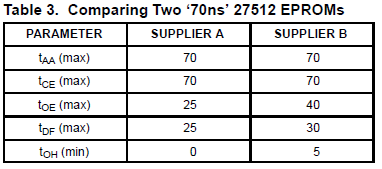
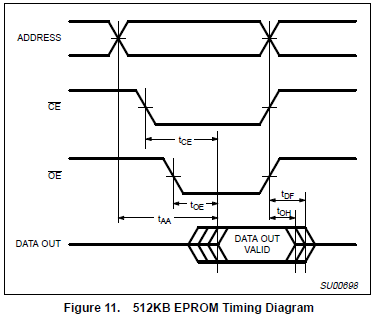
**

*INTERFACING THE EPROM*

**

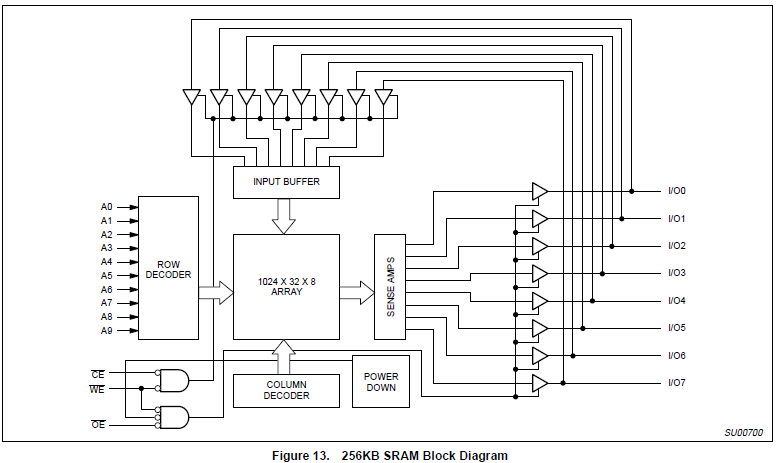
* *functionality is defined by* 
  + *the sixteen address lines A0–A15 that select a byte for output on the eight data lines O0–O7.*
  + *Only two control lines are necessary – Chip Enable (CE) and Output Enable (OE).*
* *De-asserting CE places the EPROM in ‘standby’ mode, consuming typically 1/2 to 1/3 the active (CE asserted) power. Also, observe that both CE and OE must be asserted to enable the data output drivers.*
* *Referring to the timing diagram (Figure 11) shows that operation of the EPROM is quite simple, typically characterized by only a few specifications.*

**

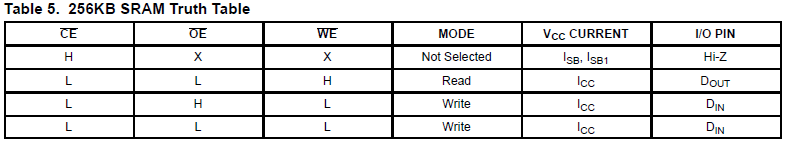
**

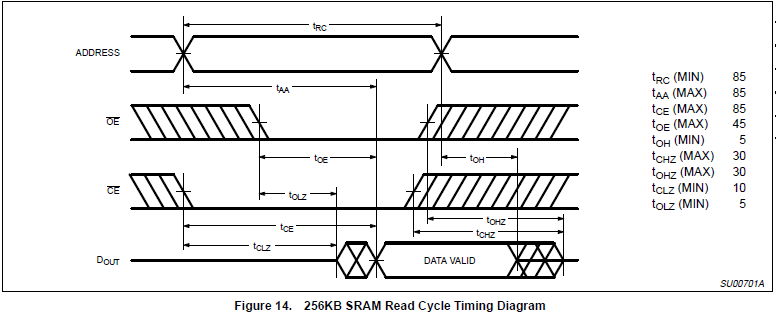
* *The first, tAA, defines the maximum time after the address stabilizes that the EPROM will return valid data. This parameter is commonly referred to as the ‘access time’ of the device.*
* *Similarly, tCE defines the maximum time after the CE input is asserted that the EPROM will return data. This spec is typically, though not always, the same as tAA.*
* *tOE defines the maximum time from OE assertion to valid data output much like tCE. However, because OE only enables the output drivers as opposed to powering up the device, tOE is typically much less than tCE.*
* *Having delivered the data, the remaining two specs deal with what happens at the end of the cycle, i.e., following the deassertion of either CE or OE (remember, both are required to enable the output). tOH indicates the minimum time the data is guaranteed to remain.*

*INTERFACING THE SRAM*

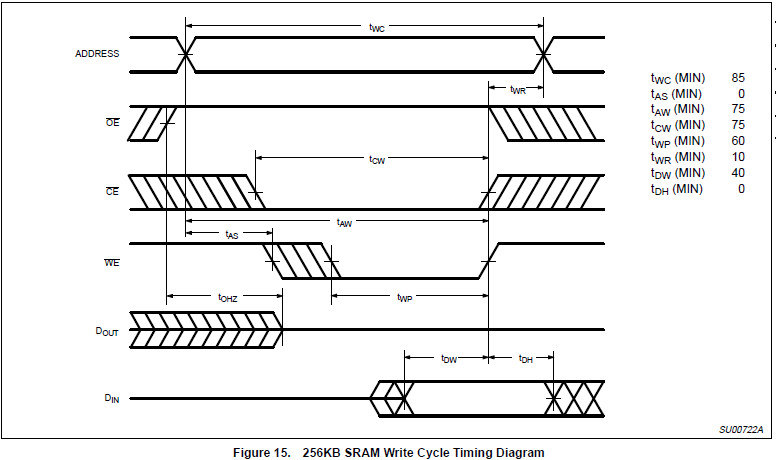
**

* *Reflecting JEDEC standardization, the pinout and functionality is nearly the same as EPROMs, the major difference being addition of a WE (Write Enable) line. Once again, CE is responsible for controlling power consumption and OE simply enables the output reflected in the tCE/tOE differential. For all practical purposes, an SRAM is little distinguishable from an EPROM as far as reads are concerned.*

**

**

* *Figure 14 shows the read cycle timing for an 85ns SRAM which is quite similar to that of an EPROM.*
* *This SRAM (as do some EPROMS), defines separate CE and OE data float specs (tCHZ, tOHZ) instead of a single tDF*
* *Since the SRAM (unlike the EPROM) can be written, it also specifies the other side of the data float coin (i.e., enable to output driven) with tCLZ and tOLZ.*

**

* *Figure 15 shows the write cycle timing which is slightly more complicated than reads. Write time (tWP) is defined as the time during which both CE and WE are asserted.*
* *Thus, various setup and hold timings that are specified relative to the ‘end’ of the write cycle (such as tDW, tDH, tWR, etc.) should refer to whichever signal (CE or WE) terminates first.*
* *tWC simply defines the write cycle time which, along with tRC, is the same as the ‘access time’, i.e., 85ns*
* *tCW and tAW specify the minimum time from valid CE and address inputs to the end of the write cycle (CE or WE high, whichever comes first). They are the write cycle corollary to the read cycles tCE and tAA specs and in this (the usual, but not always as we saw for EPROMs) case are the same. The SRAM also defines an address setup to the beginning of the write cycle (tAS).*
* *tWP simply specifies the minimum write pulse (the overlap of CE and WE) width. tWR specifies a minimum write ‘recovery’ time, essentially an address hold time after the end of write. This SRAM specs tWR at 10ns minimum, but many SRAMs need no address hold (i.e., tWR=0ns minimum) should the spec prove troublesome. Remember, the ‘end’ of the write cycle is defined as the earlier of CE or WE de-assertion.*
* *tDW and tDH specify the input data setup and hold times relative to the end of write. Finally, tOHZ reappears as a reminder that a write cycle shouldn’t drive the bus until previous read data disappears to avoid bus contention.*

4. Obtain and read the following EPROM-related documents:

• Data sheet for an EPROM, including the **AMD Am27C256 (32Kx8)** or **Fairchild FM27C256**

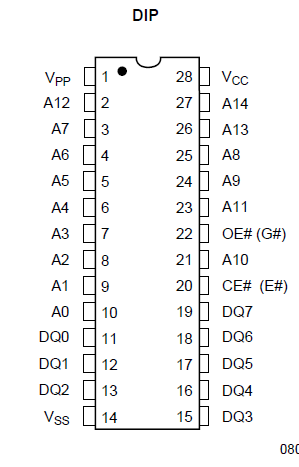
**EPROM.**

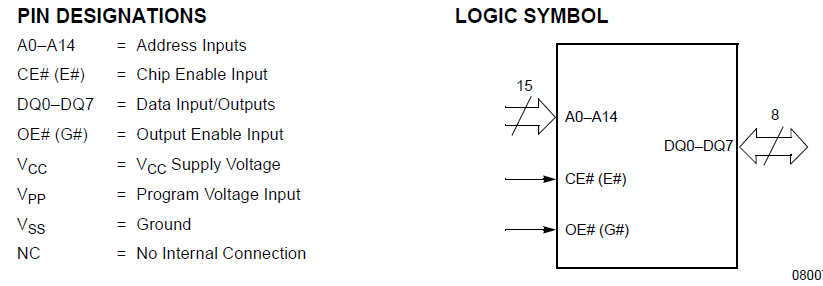
• Technical note "**Programming AMD's CMOS EPROMs**"

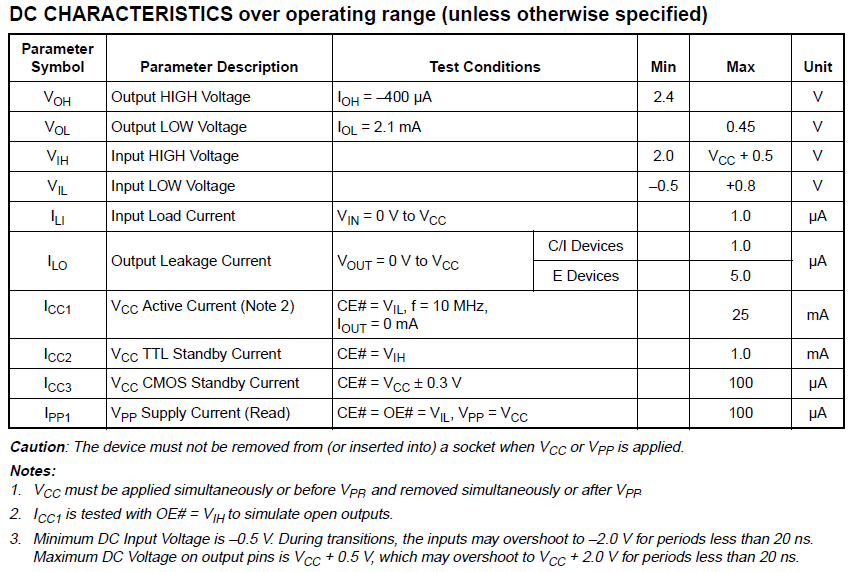
Determine how you would hook up an EPROM to the 80C51. Get a basic understanding for how

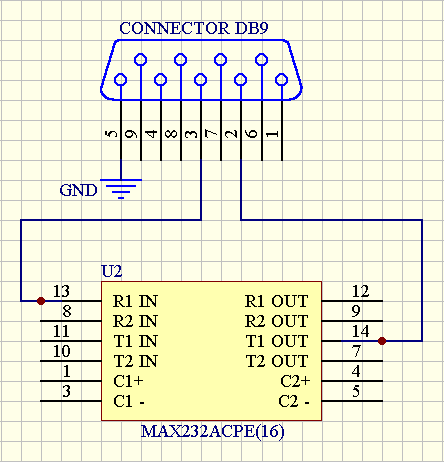
EPROMs work, and how you would program and erase an EPROM. Think about how would you

design a circuit to program an EPROM.









5. Obtain and skim a data sheet for your SRAM on the course web site. Your SRAM may be one of

several types, including the TI BQ4011 NV-SRAM, Cypress CY62256 (32Kx8), Hitachi 62256,

Samsung K6T0808C1D, or Winbond W24257 SRAM. Determine how to hook it up to the 80C51.

Compare the SRAM and EPROM pinouts. Note that we use the 28-pin DIP package in this class.

6. What could happen if you leave the /WE pin on the NV-SRAM floating (not connected)?

7. Understand how each of the following LS TTL chips works. The data sheets are available on the

course web site, but you may also want to store them on a USB drive key, floppy disk or on your PC.

• 74LS00, 74LS02, 74LS04, 74LS08, 74LS74, 74LS138, 74LS156, 74LS244, 74LS245, 74LS373, 74LS374

The On Semiconductor LS TTL General Data document includes interesting technical information.

8. How is 'noise margin' associated with the following logic gate specifications: VOH, VOL, VIH, VIL?

9. Does a TTL totem pole output sink or source current?

10. When a logic high is applied to a TTL gate input, is the input sinking or sourcing current?

11. When a logic low is applied to a TTL gate input, is the input sinking or sourcing current?

12. What is 'fanout' and how does that relate to VOH and VOL?

13. Read about the Atmel ATF16V8C SPLD (programmable logic). Read about WinCUPL. Documents

and links are available on the course web site.

14. Is a decoupling capacitor required when using an EPROM? How about when using an NV-SRAM?© 2004-2013 Linden H. McClure, Ph.D. – 6 – Embedded System Design

15. What is the difference between an OTP EPROM and a UV EPROM?

16. Do more standard values exist for resistors or for capacitors? How might this impact your use of these

components in designs?

17. Why should components be derated when used in circuits? Suppose you have a capacitor that has a

working voltage of +10V. What maximum operating voltage should be applied to this capacitor for

extended periods of time?

18. Suppose you have an RC circuit consisting of a 5% 100KΩ resistor and a 10% 5uF capacitor. What is

the maximum and minimum time constant of this circuit at 25°C?

19. How does the temperature characteristic or profile of a resistor or capacitor affect its operation? Take

a look at the temperature characteristic for an electrolytic capacitor (shown below) and determine

how the component's value changes when operating at 100°C.

20. How do the characteristics of an X7R capacitor compare with a Z5U capacitor?

21. [Optional] Review a logic book or visit a web site such as one of the following and explore logic

families. While you're at the chosen web site, explore a little and try to understand some of the

differences between the different logic families (e.g. LS, S, ALS, FAST, HCT, etc.). Note the

differences in supply voltages, and input and output voltages of some of the different devices. Using

the information from the data sheets or from a text book, compare the fanout, propagation delays,

signal transition times, and power consumption of at least three of the families. Think about the

advantages and disadvantages of using each of the particular families you examined.

• Texas Instruments: http://focus.ti.com/general/docs/scproducts.jsp

• Texas Instruments: http://focus.ti.com/lit/ml/sdyu001z/sdyu001z.pdf

• ON Semiconductor: http://www.onsemi.com/PowerSolutions/ (formerly Motorola logic)

• Toshiba: http://www.toshiba.com/taec/ (see Products > Logic ICs)

• Philips: http://www.philipslogic.com/products/

• Fairchild: http://www.fairchildsemi.com/

For a list of logic manufacturers, see: http://www.interfacebus.com/Standard\_Logic.html