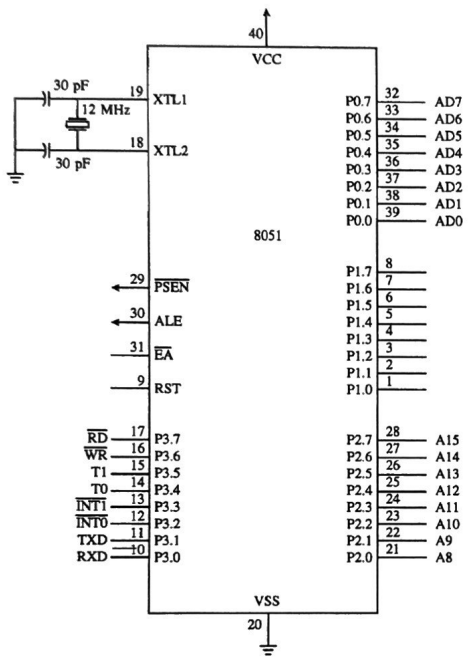
Lecture 2

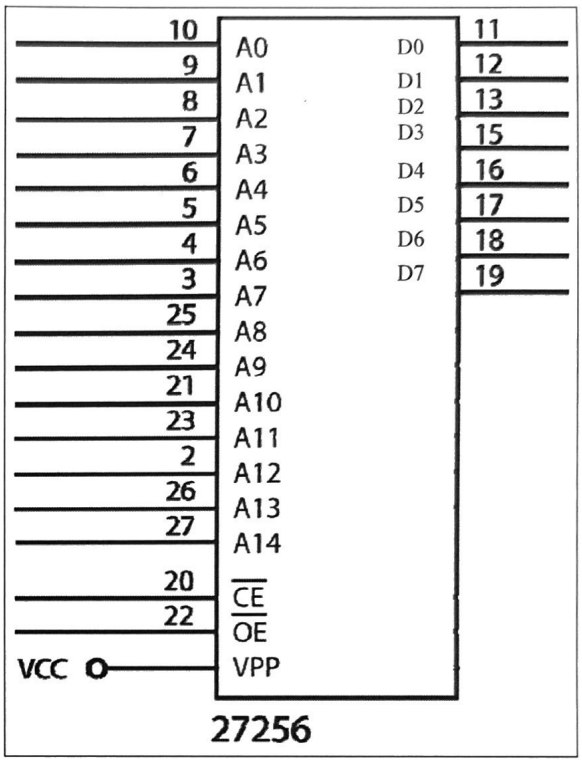
Our first "System Study": The R31JP development board.
What might you want in a microcontroller development? Perhaps:

- Easy to change test code
- Plenty of memory for code and data storage
- Requires little specialized/expensive hardware to get started using the system
- Easy interface to other hardware – remember, we're making embedded systems!

The simplest possible development system in term of chip count:

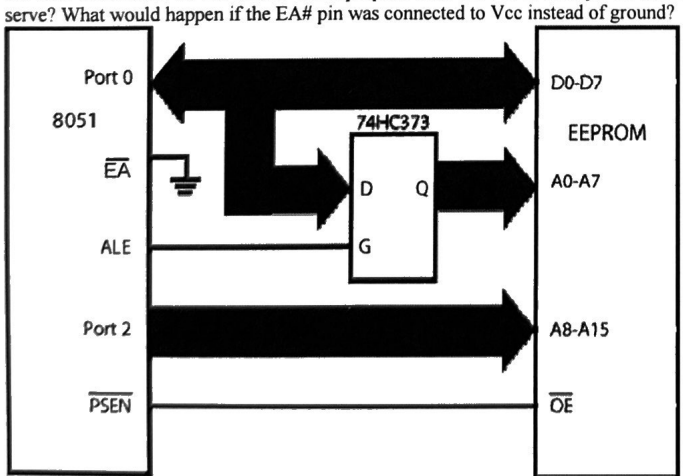


And a typical ROM chip:

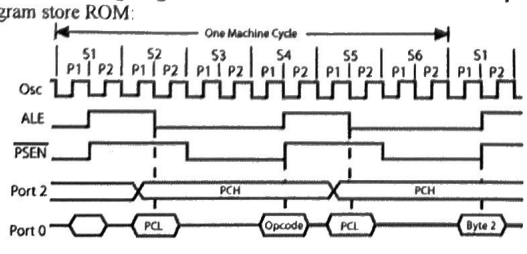


The lines CE# and OE# are again active LOW for this ROM. They perform similar functions as for the RAM chip. Why is there no WR# line?

Here is a simple connection scheme to allow a ROM or EPROM or FLASH/EEPROM to serve as "program store for the 8051. What must be done with the CE# connection on the ROM? What purpose does the 74373 transparent latch serve? What would happen if the EA# pin was connected to Vcc instead of ground?

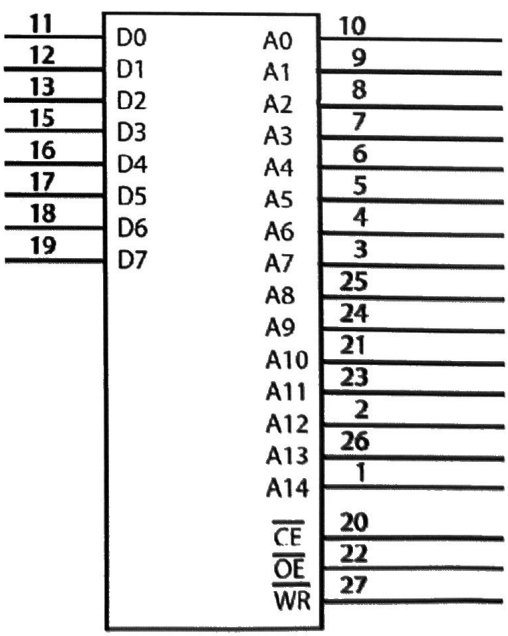


And here is the timing diagram that describes how the 8051 reads op-codes from the program store ROM:



Note: PCH = Program counter high byte PCL = Program counter low

Typical RAM Chip:



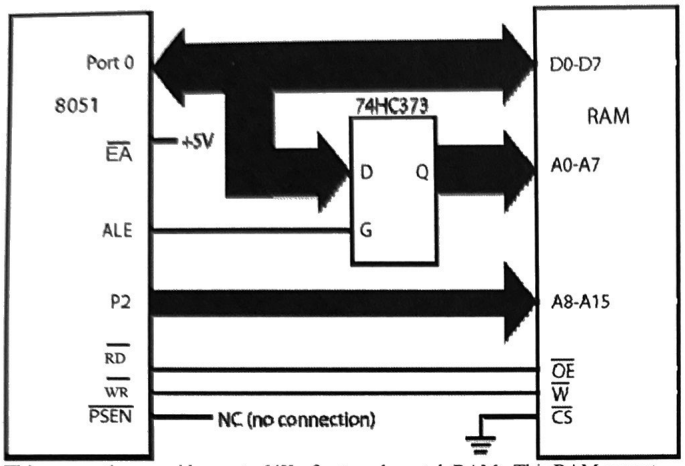
The pins CE#, OE#, and WR# are all active LOW.
These pins control the operation of the chip. The WR# line is used to load a data byte into the RAM. The OE# line "turns on" the data lines, allowing them to drive the data bus so that a data byte can be read from the memory. The CE# line "turns on" the entire chip – it must be active for any read or write operations to occur.

The remainder of the lines shown in this figure constitute the address and data bus.

Some lines are missing!
What are they?

(Gnd, Vcc)

If you don't need external program store, you might still need more scratchpad RAM than is available on the 8051. In this case, the program store might be inside the 8051 on-board ROM, and we might connect an external RAM to provide more "variable" space:



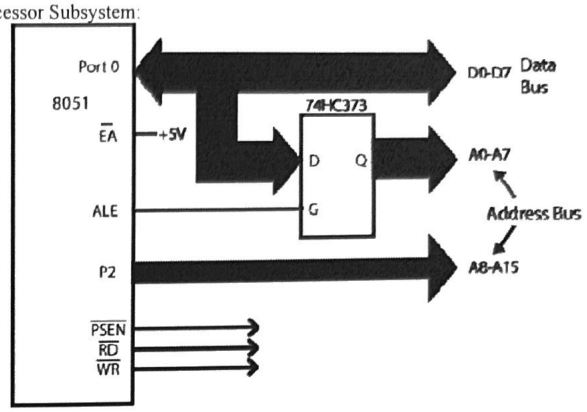
This connection provides up to 64K of external scratch RAM. This RAM cannot hold a program, only data! Notice:

- PSEN# is disconnected
- EA# is HIGH – where is the program stored?
- CS# on the RAM chip is the same as CE# - you'll see both on data sheets
- See the INTEL MCS51 manual for timing diagrams that explain how data is read from or written to the RAM by the 8051.

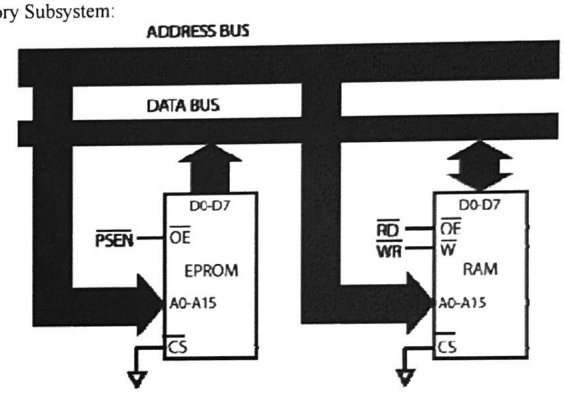
Now, suppose you wanted to have both external program store and additional, external scratch RAM:

HARVARD Architecture: Provides 64K of program store, 64K of data memory.

Processor Subsystem:



Memory Subsystem:



LINEAR Architecture with memory-mapped I/O:

Only the control signals are shown. The address bus and data bus must also be connected to the RAM, ROM, and 8051, but have been removed in the figure for clarity. Provides a 64K linear memory space consisting of 32K RAM and 32K ROM. Either the RAM or ROM can fill the first 32K (0000h – 7FFFh) by flipping the DPDT "MON/RUN" switch. XIO# selects memory-mapped I/O devices.