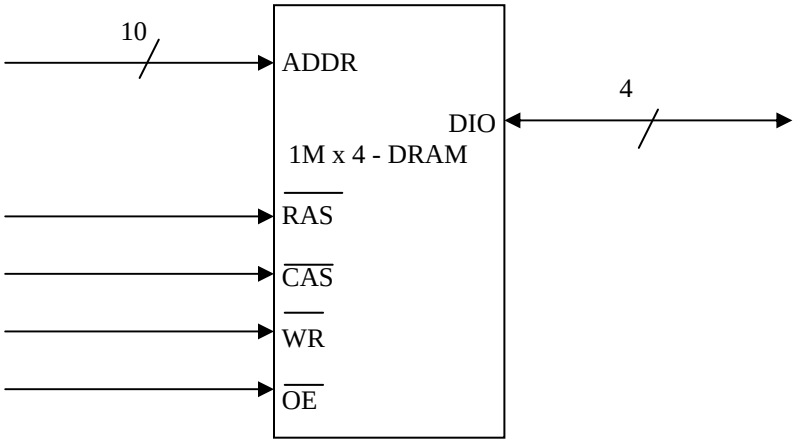
DRAM Timing Relations

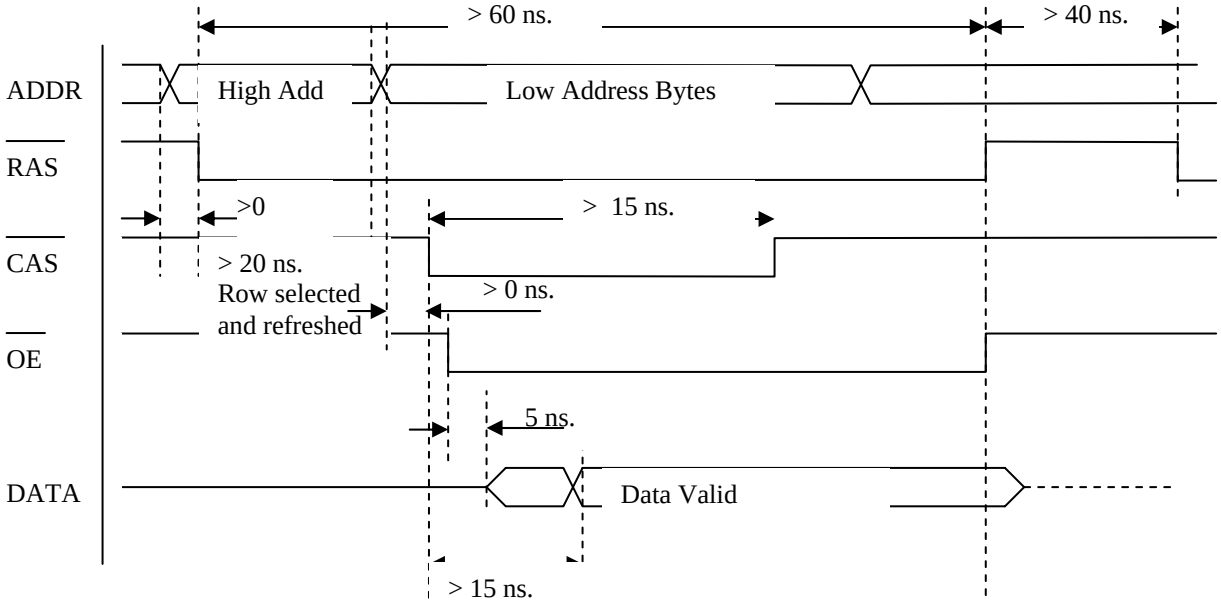
This handout shows the usual timing relations for a non-synchronous 1M x 4 DRAM circa 1994. While DRAMs have gotten much bigger since, their speeds and timing constraints have changed relatively little. The major change has been to synchronous or SDRAM that has extra registers at input and output so reads can be pipelined. The flexibility of this memory makes the timing much more complicated but conceptually little changed.

The 4-Mb device shows you the typical control lines and how they have to be manipulated for READ, WRITE, or REFRESH operations. There are many variations in timing that are possible and many more times that are either restrictions to be observed or intervals that have a minimum specification. However, what is shown here are the principal timing issues. The symbol for the device with the input signal names is:



All data accesses require sequentially applying the most significant half of the address and asserting the Row Address Strobe ($\overline{RAS}$) to set the word line to the memory cell array. After enough time for the whole row to be read and refreshed, one puts the low address out and asserts the Column Address Strobe ($\overline{CAS}$) to read or write the column data as required. The diagram below shows how this works for a READ operation.

Notice that the data remains valid even after the CAS line goes high. This is characteristic of a variant on the basic DRAM called an EDO (Extended Data Output) device that incorporates a transparent latch just before its output drivers.



The WRITE operation is very similar to the READ. The main difference is that the R/W line must be set for writing before the CAS line is asserted. Then the direction of data transfer is to write data placed on the bi-directional DIO lines into the memory during CAS assertion. The initial row refresh and the post-write recovery are the same as the READ cycle equivalents.

The simplest REFRESH cycle only requires asserting CAS before RAS. Then an internal counter supplies the necessary next row refresh address. This refresh timing is shown below.