Module-2, Unit-2
Main Memory

Question 1: What are some of the main semiconductor memories which were used for designing the main memory? Also, discuss their pros and cons.

Solution 1:

Main Memory is generally implemented in the form of RAM (Random Access Memory). RAM is volatile

The following are the major kinds of RAMs which were developed and utilized as main memory. Also, their pros and cons for usage as main memory are as follows.

- **SRAM (Static RAM):** Access time (i.e., read and write) is fastest in SRAM compared to other variations. However, it is expensive because it requires comparatively larger area (i.e., 6-transistor circuit for memory cell). Due to its cost it is not very suitable for main memory (and is generally used for cache memory).

- **DRAM (Dynamic RAM)**: Access time is higher than SRAM, but is cheaper as it can be implemented using only a single capacitor and single transistor. There is another issue with this memory, because capacitors lose their charge and hence DRAM needs to be refreshed every few milliseconds. As of now, DRAM is the most widely used for main memory implementation.

The two well-known variants of DRAM are SDRAM and DDR SDRAM.

SDRAM (Synchronous DRAM), as the name suggest, is synchronised with the clock of the CPU's system bus

DDR SDRAM (Double-Data Rate SDRAM) is an optimisation of SDRAM that allows data to be transferred on both the rising edge and falling edge of a clock signal, thereby doubling the amount of data that can be transferred in a period of time.

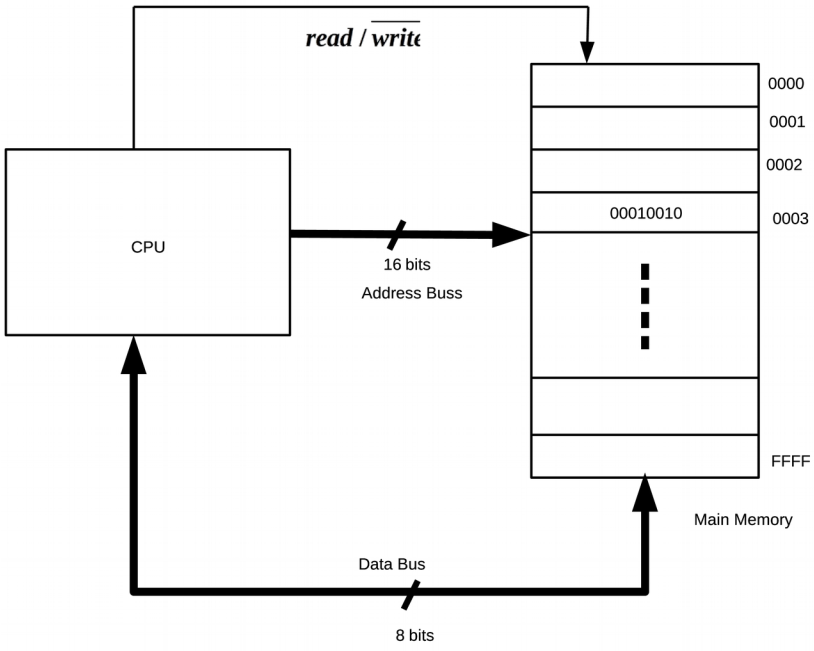
Question 2: Assume a hypothetical CPU connected to the main memory whose size is 64K x 8 Bits. Now an instruction LOAD Acc, 0003_H is executed which loads the value present in memory location 0003_H to the accumulator in the CPU. Explain how this operation is executed by illustrating the values in the system buses connecting the CPU to the memory.

Solution 2:
The configuration of the main memory is 64K x 8 Bits. It implies that there are 64 x 1024 (=65536) memory locations and each location (called word) has 8 bits.

65536 locations are referenced from 0 to 65535. First location is referenced by Binary number 0000 0000 0000 0000 and the last location is referenced as 1111 1111 1111 1111; so 16 Bit binary number is required to address this memory. ***So the address bus is of size 16 Bits.*** For ease of representation in Assemble level instructions we use in Hexadecimal notation, where we reference the first location as 0000_H and the last location as $FFFF_H$.

As each location has 8 bits (i.e., data) the data bus is 8 bits.

Also, there is a single bit control signal to configure the memory as read (i.e., transfer data from a memory location, whose address in given in address bus, to data bus) or write (i.e., transfer data from data bus to a memory location, whose address in given in address bus).
The figure given below illustrates the details of the hypothetical CPU connected to the main memory.

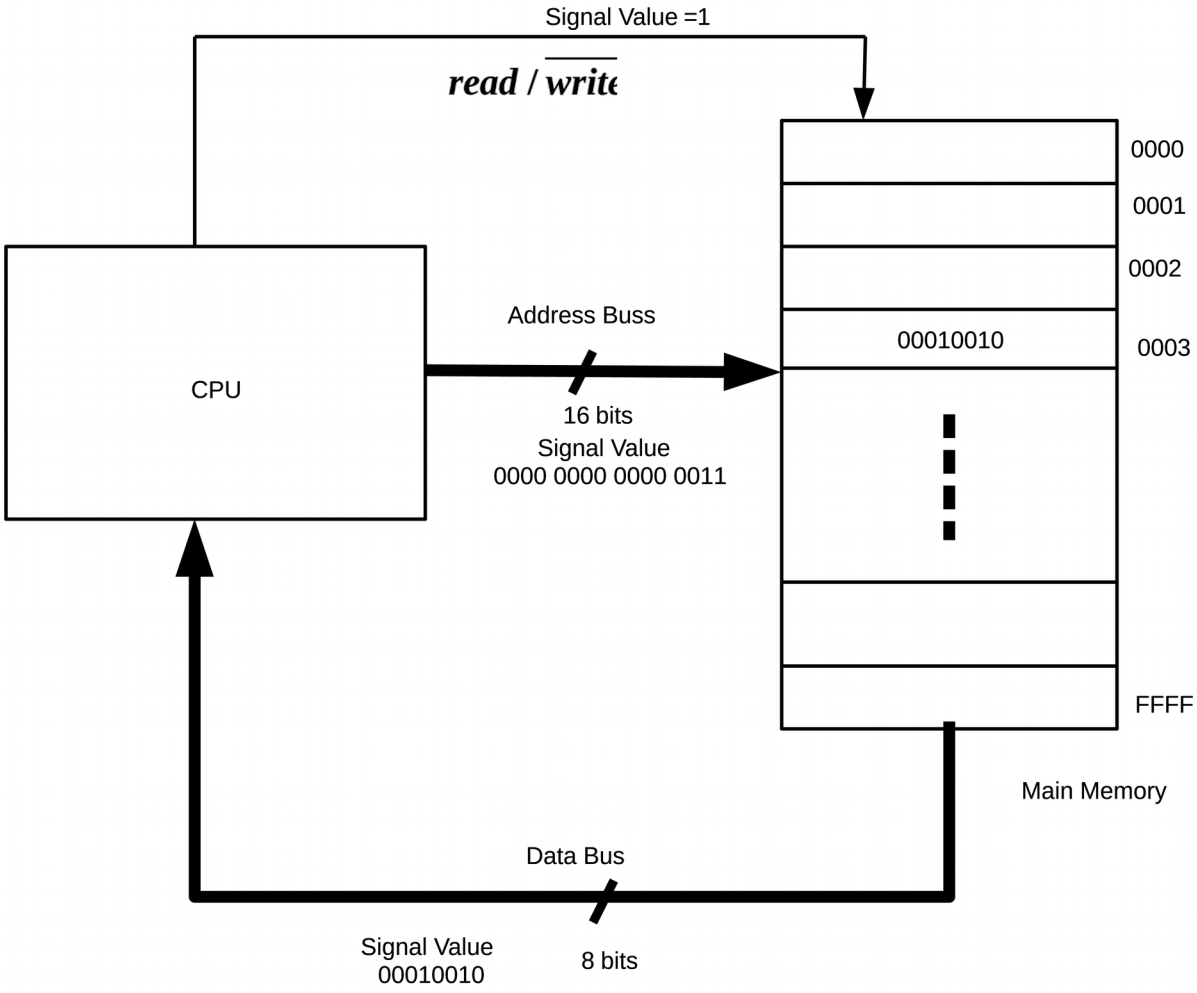


Now, when instruction LOAD Acc, 0003_H is executed the values in the system buses connecting the CPU to the memory are as follows:

- Address Bus: 0000 0000 0000 0011 (i.e., 0003_H)
- Control line: 1 (i.e., read from memory)

- Data Bus : 0001 0010 (i.e., contents of memory location 0003_H)

The figure given below illustrates the values in the system buses when instruction LOAD Acc, 0003_H is executed



Question 3.

The capacity of a main memory is 2GB (Giga Byte). Provide the size of Data Bus and Address Bus with proper explanation if the main memory is arranged as:

a)Bit Organized

b)Byte Organized

c)16 bit in each memory location:

d)32 bit in each memory location:

Solution 3

Total size of the data is 2GB =2^{31} Bytes (as 1 G Byte = 2^{30} Bytes)
So, 2GB = 8 * 2^^{31} Bits = 2^{34} Bits.

a) Bit Organized:

Size of data bus is 1 (only one bit at every location in the memory)
No. of addresses=$2^{34}/1=2^{34}$.
Therefore size of address bus=34 Bits

b) Byte Organized:

Size of data bus=1*8=8 bits (8 bits at every location in the memory)
No. of addresses=$(2^{34})/8=2^{31}$
Size of address bus =31 Bits

c) Double Byte (16 bit) in each memory location:

Size of data bus=16bits
No. of addresses=$(2^{34})/16=2^{30}$
Size of address bus=30 bits.

d)32 bits in each memory location:

Size of data bus=32 bits
No. of addresses=$(2^{34})/32=2^{29}$
Size of address bus=29 bits.

Question 4.

We need to design a memory of configuration 4 K x 16 Bits. However, we have memory chips of configuration 1 K x 8 Bits and standard digital circuit blocks like decoder, multiplexes etc. Explain the design.

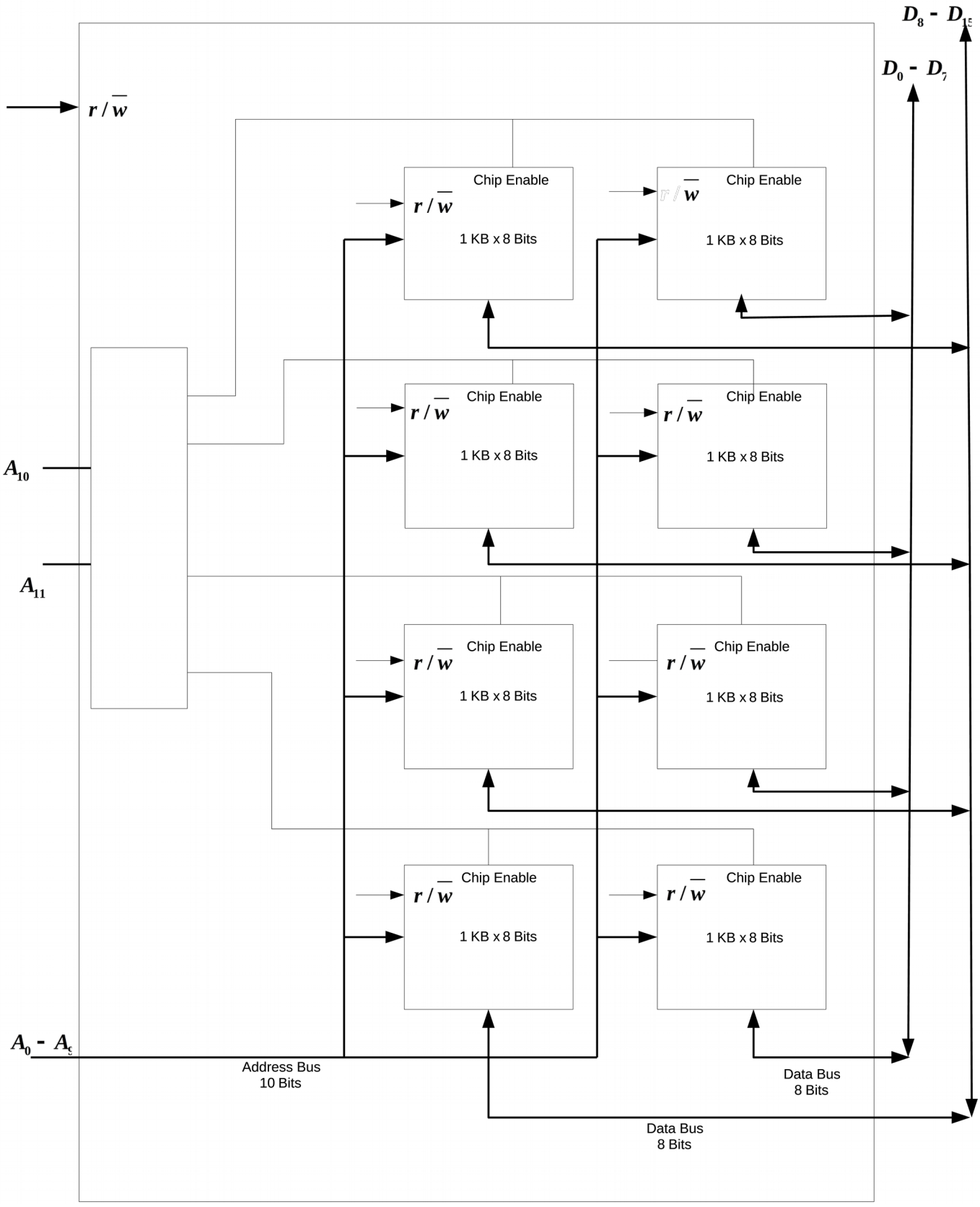
Solution 4:

The final memory size is 8 time the size of the chips available. So we require 8 memory chips of configuration 1 K x 8 Bits to implement 4 K x 16 Bits memory. It may be noted that in the memory we want to implement, we require

- 16 bits in each location i.e., twice the number in the available memory chips. So to implement this requirement we need to put two memory chips side by side; the 8 bits of the left chip would comprise the 8 MSBs (i.e., D_{15}-A_8) and the 8 bits of the right chip would comprise the 8 LSBs (i.e., D_7-A_0).
- 4 K of memory locations i.e., four times the number in the available memory chips. So to implement this requirement we need to put four memory chips arranged in a columnar fashion. For locations 0 – 1023 we use the first chip, 1024-2047 we use the second chip and so on. To implement this philosophy we use the chip enable line of the memory chips. Only when the chip enable line is 1 the corresponding memory chip is functional (i.e., ON), else it is OFF and do not send/receive data through the data-Bus. 4 K locations imply that the address bus size is 12 bits. Among these 12 bits, the LSB 10 Bits (i.e., A_0-A_9) are connected to the data Bus of the memory chips. The MSB 2 Bits (i.e., A_{10}-A_{11}) are connected to 2-4 Decoder, whose outputs are connected to the chip enable lines of the memory chips.

The figure given below shows interconnections of 8 memory chips of configuration 1 K x 8 Bits and the 2-4 decoder which implements the 4 K x 16 Bits memory.

For example, if we need to read memory location FFF_H (i.e., 4095^{th} Location) then the address bus has the value 1111 1111 1111. The 10 Bits LSB of the data bus selects the last locations of all the memory chips. But, MSB 2 Bits selects the 4^{th} line of the decoder (because A_{10}=1 and A_{11}=1) thereby enabling ONLY the memory chips of the last row. The data bus outputs 8 MSBs (i.e., D_{15}-A_8) from the left chip and 8 LSBs (i.e., D_7-A_0) form the right chip of the last row.


D_8 - D_{15}
D_0 - D_7
$r / \overline{w}$
Chip Enable
1 KB x 8 Bits
A_{10}
A_{11}
A_0 - A_9
Address Bus
10 Bits
Data Bus
8 Bits
Data Bus
8 Bits