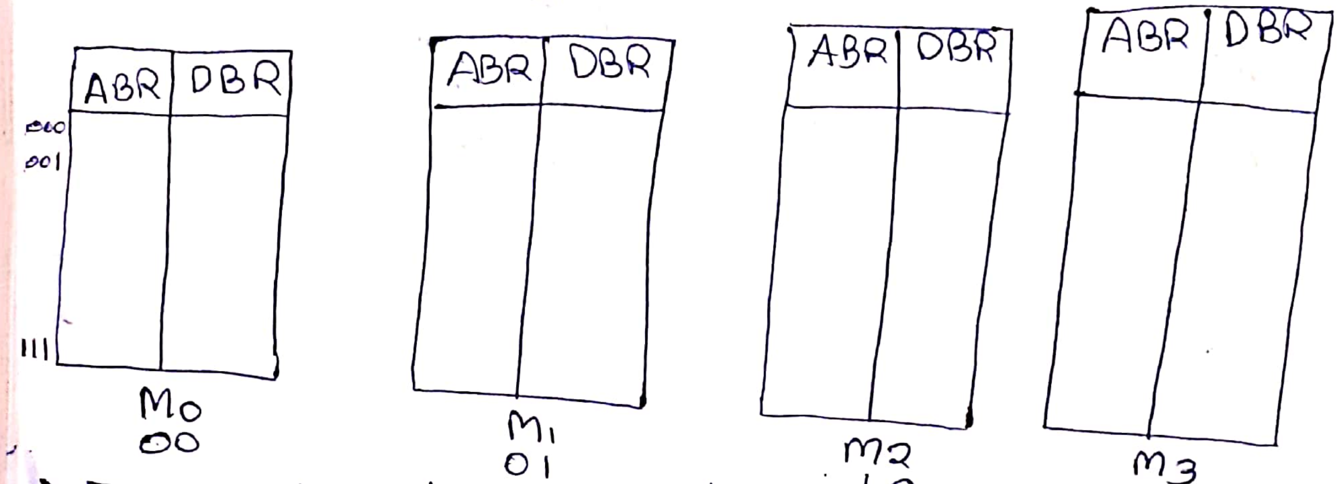
Memory Interleaving (Hamacher)

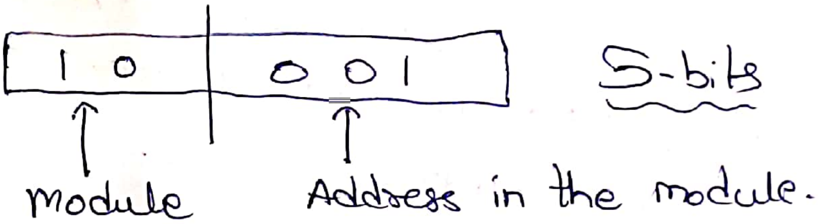
12-03-2020

ABR: Address Buffer Register
DBR: Data Buffer Register



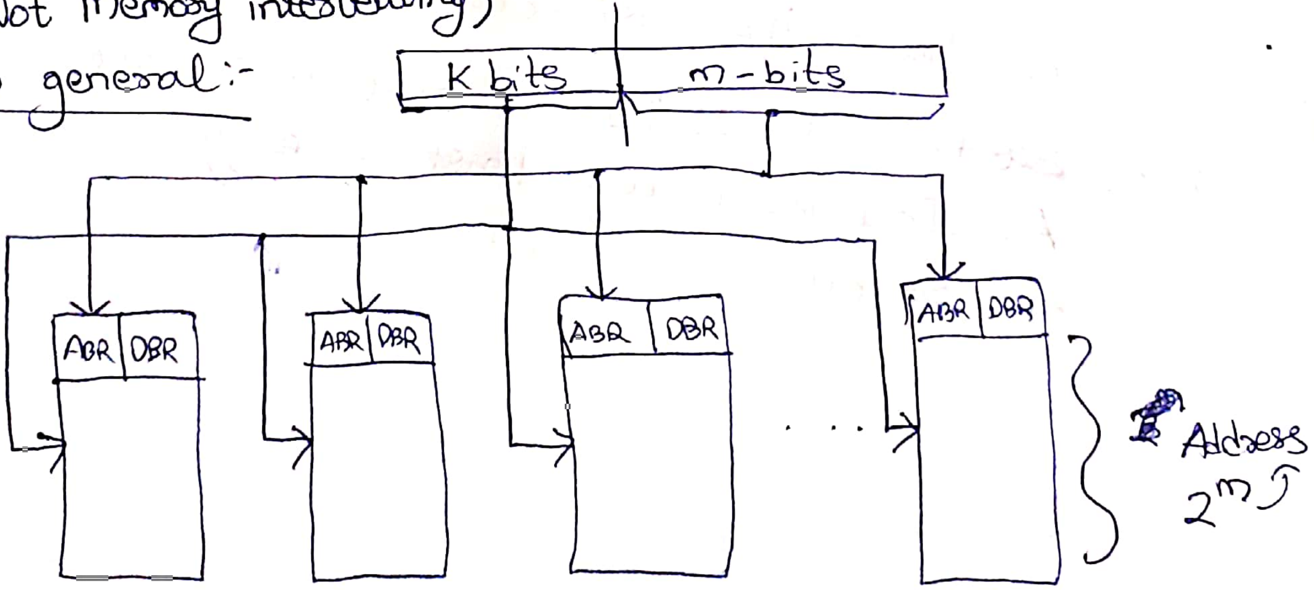
-) Each module has 8 locations
-) Total four modules => Total 32 locations
-> CPU generates 5 bit address.

CPU generates:-



(Not memory interleaving)

In general:-



$\boxed{2^K \text{ modules}}$

Consequtive words are in a module

10	001
10	010
10	011
10	100

#Memory Interleaving Structure

•) In memory interleaving, the lower bits will gives us the module no while the higher (MSB) will give us the address.

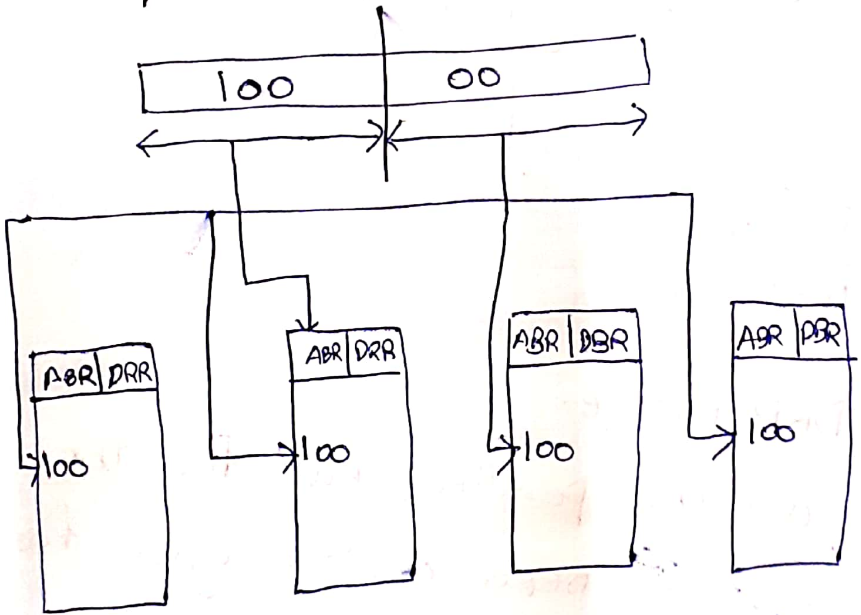
Consecutive words are in diff. modules

•)

100	00
100	01
100	10
100	11

Same address 100 is present in different modules 00, 01, 10, 11.

•)



As the words are available in different modules, so the words can be obtained parallelly (modules work independently).

Eg:- The 100 address will bring up data from diff. modules at same time which reduces the no of cycles (to access & fetch)
Scanned with CamScanner

Numerical:-

•) Cache block-size = 8 words (3-bit)

•) One cycle is required to send MM address from CPU to Main Memory.

•) 8 clock cycle for accessing first word in a module

•) 4 clock cycles for accessing subsequent words from same module

•) 1 clock cycle for sending one word to Cache.

Find Time required to send a block to cache.

A) $$\underbrace{1}_{\text{For one word CPU to mm}} + \underbrace{8}_{\text{For the first word}} + \overbrace{\underbrace{4}_{}}^{\text{In this we are not Calculating Cycle from DBR to Cache as included inside}} + 4 + 4 + 4 + 4 + 4 + 4$$

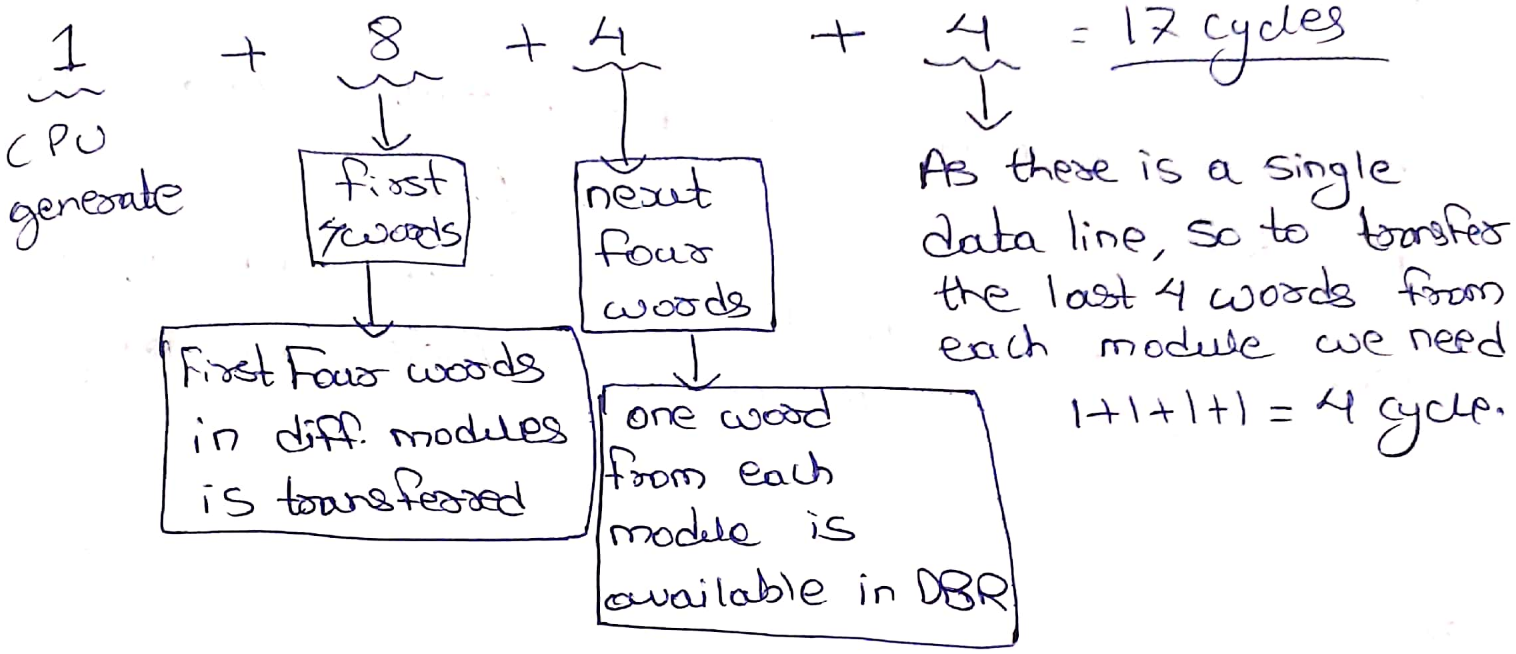
$$+ \underbrace{1}_{} \rightarrow \text{For DBR to Cache.}$$

•) As things are available parallel, but for the last word transfer, we need another cycle for DBR to Cache.

$$\Rightarrow 1 + 8 + (7 \times 4) + 1 = \boxed{38 \text{ cycles}}$$

Scanned with CamScanner

•) In case of memory interleaving;



$\underbrace{1}_{\text{CPU generate}} + \underbrace{8}_{} + \underbrace{4}_{} + \underbrace{4}_{} = 17 \text{ cycles}$

first 4 words

First four words in diff. modules is transferred

next four words

one word from each module is available in DBR

As there is a single data line, so to transfer the last 4 words from each module we need 1+1+1+1 = 4 cycle.

Here, 8 cycles in the above includes parallel data addressing.