⇒ Message Digest - 5

→ Hashing algo

→ Ron Rivest is developer

→ o/p is fixed of 128 bits

I] Accept msg & do padding.

message	Padding

i.e. Total len of msg + padding + 64 = multiple of 512

For padding :-

Total length of bits = should be multiple of 512 less 64

Eg: message = 1000 bits

1] 512 × 1 = 512 ×

2] 512 × 2 = 1024 ×

3] 512 × 3 = 1536 ✓ ∵ msg bit is only 1000 & we would subtract 64 ∴ can't take 1024

Padding bits = (Selected - 64) - msg

= (1536 - 64) - 1000

= 1472 - 1000

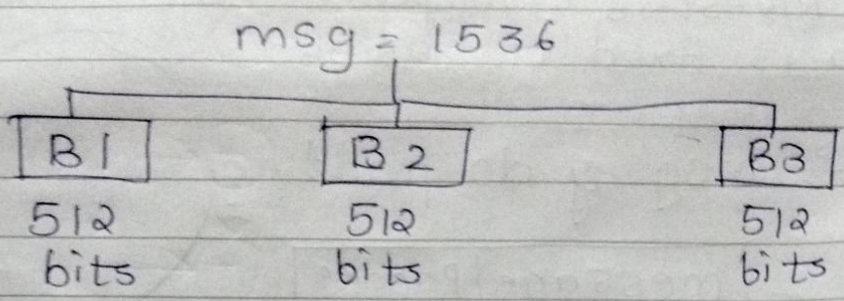
= 472

II] Append random string in msg appended with Padding, random string = 64 bits

NOTE :- Why? To get Randomness

msg	padding	string
1000	472	64

III] Divide above msg block into sub-blocks each of 512 bits



IV] Initialize chaining var i.e. Buffers
There a 4 buffers each of 32 bits

Total buffer length = 4×32 = 128bits

A = a
B = b
C = c
D = d
} Var or Buffers

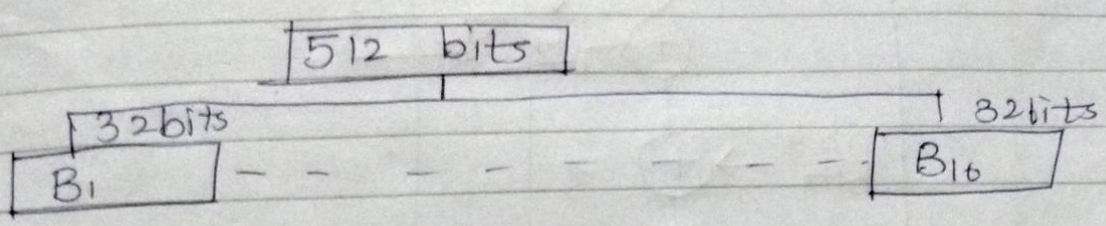
V] Round Explanation:

Each Block of 512 bits will go for 4 rounds & every round has total 16 operations.

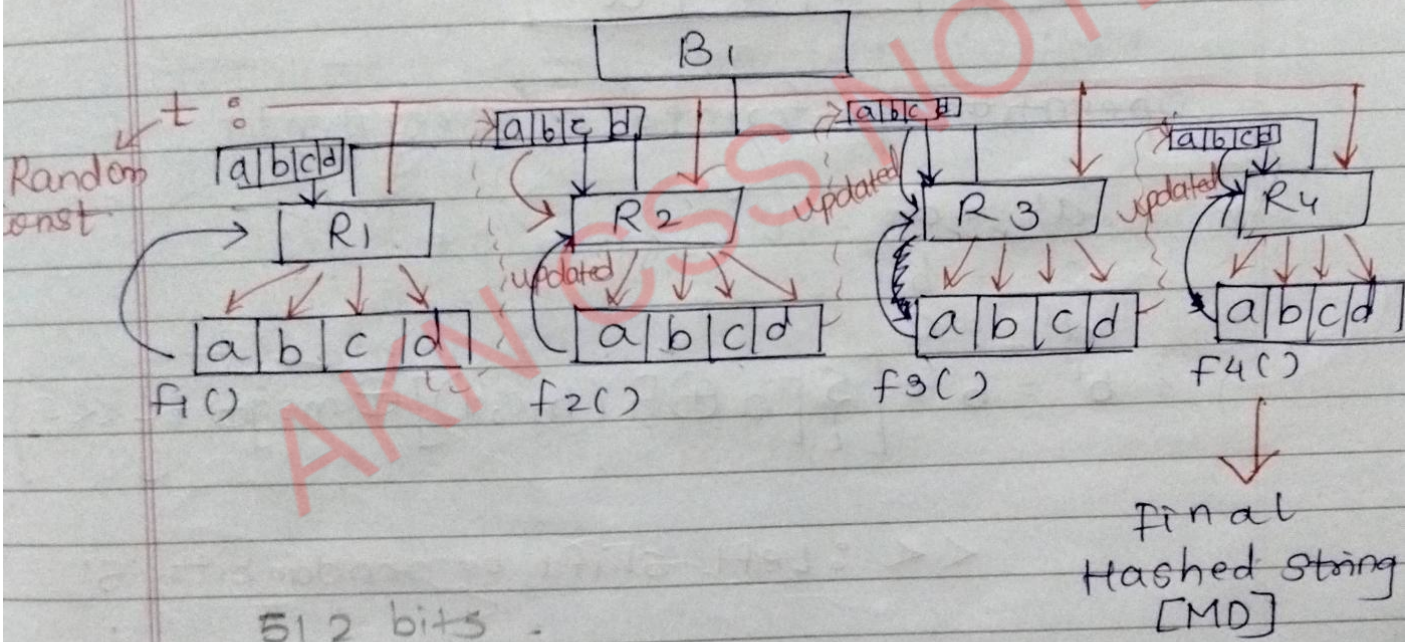
∴ Total operations performed in single Block = 16 × 4 = 64

NOTE: All rounds will have diff operation

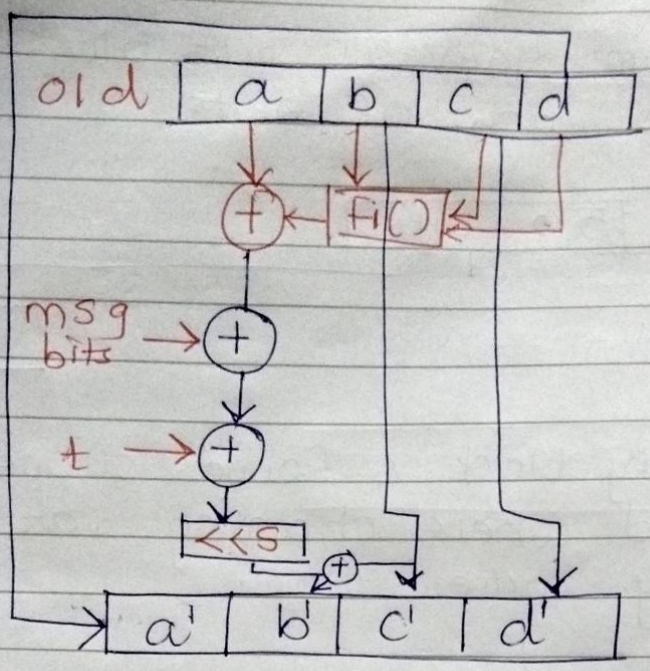
VI] Block Diagram NOW 512 bits Block is divided into 16 blocks each of 32bits



Now on every block 4 Rounds of operation will be performed where chaining var will be updated by some func.



512 bits .
↓
16 blocks
↓
Each block 4 rounds
↓
16 operation in every round.



Operation performed on every Round:

$$a' = d$$
$$c' = b$$
$$d' = c$$
$$b' = b + \left[\{[a \oplus fi(b,c,d)] \oplus msg \oplus t\} << s\right]$$

<< : Left shift by random bits 's'