CHECKSUM

Checksum = Check + sum

Sender side ⇒ checksum creation.

Receiver side ⇒ checksum validation.

Sender side

1. Break the original message into k (~~4, 8, 16, 32 ... n~~) number of blocks/sections with n bits in each block. (n ⟶ 4, 8, 16, 32 ....)
   [i.e> 20 bits in 4 blocks = 5 bit each blocks).
2. Sum all the 'k' data blocks.
3. Add the carry to the sum, if any. (one's complement arithmetic)
4. Do 1'st complement to the sum = checksum
   <checksum also be n bits>.

Example: 10011001.11100010.00100100.10000100 (Data word)

```
      1 0 0 1 1 0 0 1
      1 1 1 0 0 0 1 0
      0 0 1 0 0 1 0 0
      1 0 0 0 0 1 0 0
   ------------------
   10 0 0 1 0 0 0 1 1
   |
   └──────────────> 1 0
   ------------------
      0 0 1 0 0 1 0 1
```

1's complement : 11011010

∴ checksum : 11011010

Codeword : 10011001·11100010·00100100·10000100·11011010

Receiver side :

1. collect all the data blocks including the checksum
2. Sum all the datablocks and checksum
3. Add the carry to the sum, if any. (one's complement addition)
4. Do 1st complement to the sum = checksum

If the checksum is 0, this means that the data is not corrupted. (Accepted)

~~10011001 · 11100010 · 00100100~~

```
      1001 1001
      1110 0010
      0010 0100
      1000 0100
      1101 1010
   ------------
 10   1111 1101
 |____________10
   ------------
      1111 1111
```

∴ 1's complement ⇒ 00000000 (checksum)

(Accepted)

(সাধারণত 16 bit use করে) **Internet checksum** + Book-300 page

* A sender needs to send the four data items 3456, ABCC, 02BC, and EEEE

Sender site

```
  1 2   2  2
  3  4  5   6
  A  B  C   C
  0  2  B   C
  E  E  E   E
  0  0  0   0   <- checksum (Initial) [optional]
---------------
  D  1  C   C
            1
---------------
  D  1  C   D   => sum
```

Hexadecimal

0–9, A, B, C, D, E, F

0 — 9, 10, 11, 12, 13, 14, 15

6
12
12
14
16)44(2 => carry.
32
12

D → 1101, 1 → 0001, C → 1100, D → 1101

1st complement => 0010 1110 0011 0010

↓ 2, ↓ E, ↓ 3, ↓ 2

15 − C = 3; 15 − D = 2; 15 − D = 2; 15 − 1 = E

checksum => 2E32 (subtracting Each value by 15)

Data word: 3456, ABCC, 02BC, EEEE, 2E32

At the receiver site

~~D1CD~~

~~2E~~

3456	D1CD
ABCC	2E32
02BC	FFFF
EEEE	
2E32	
FFFF	

1111 1111 1111 1111

1st complement: 0000 0000 0000 0000

[Subtracting each value by 15]

Receiver checksum: 0000

Q-1: Find the checksum at the sender site.

Q-2: Find the checksum at the receiver site.

Q-3: ~~send~~ Mr. X send Message ----.

In the receiver site message is ~~[illegible]~~ Accepted/Rejected.