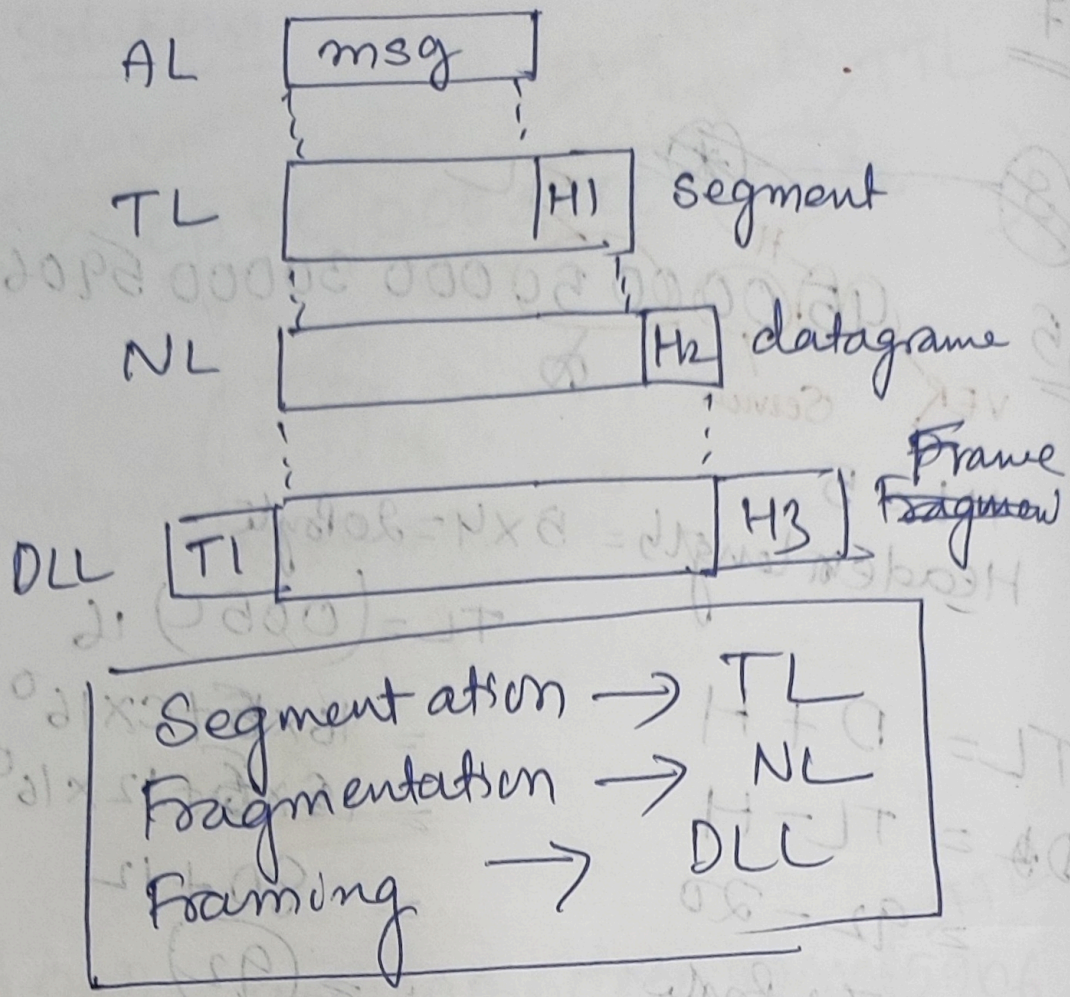
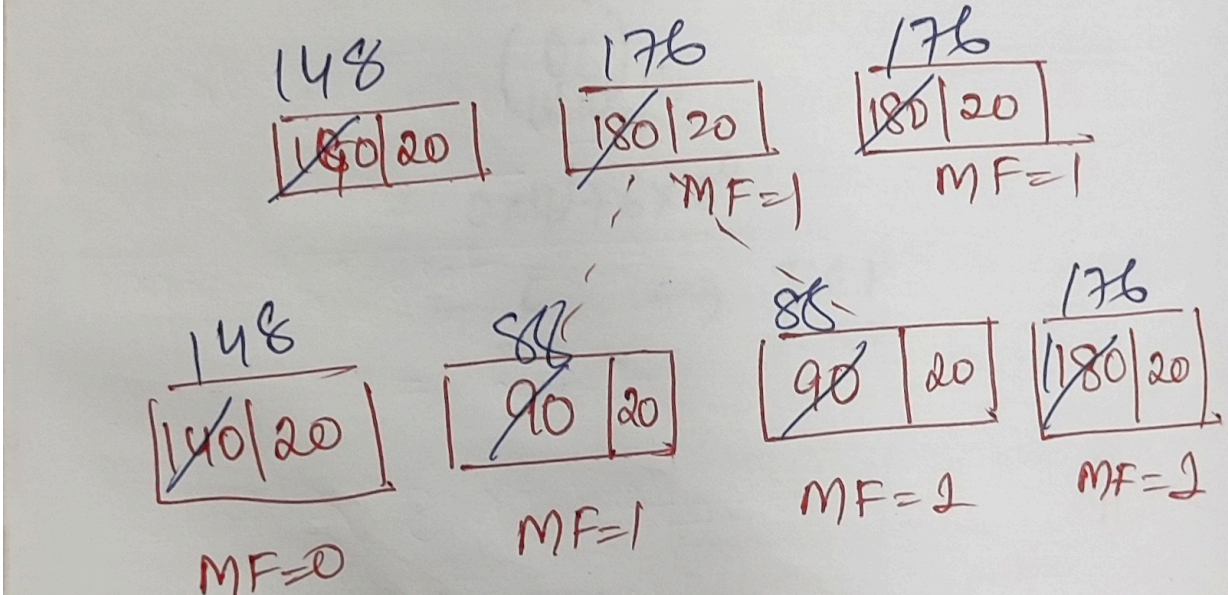
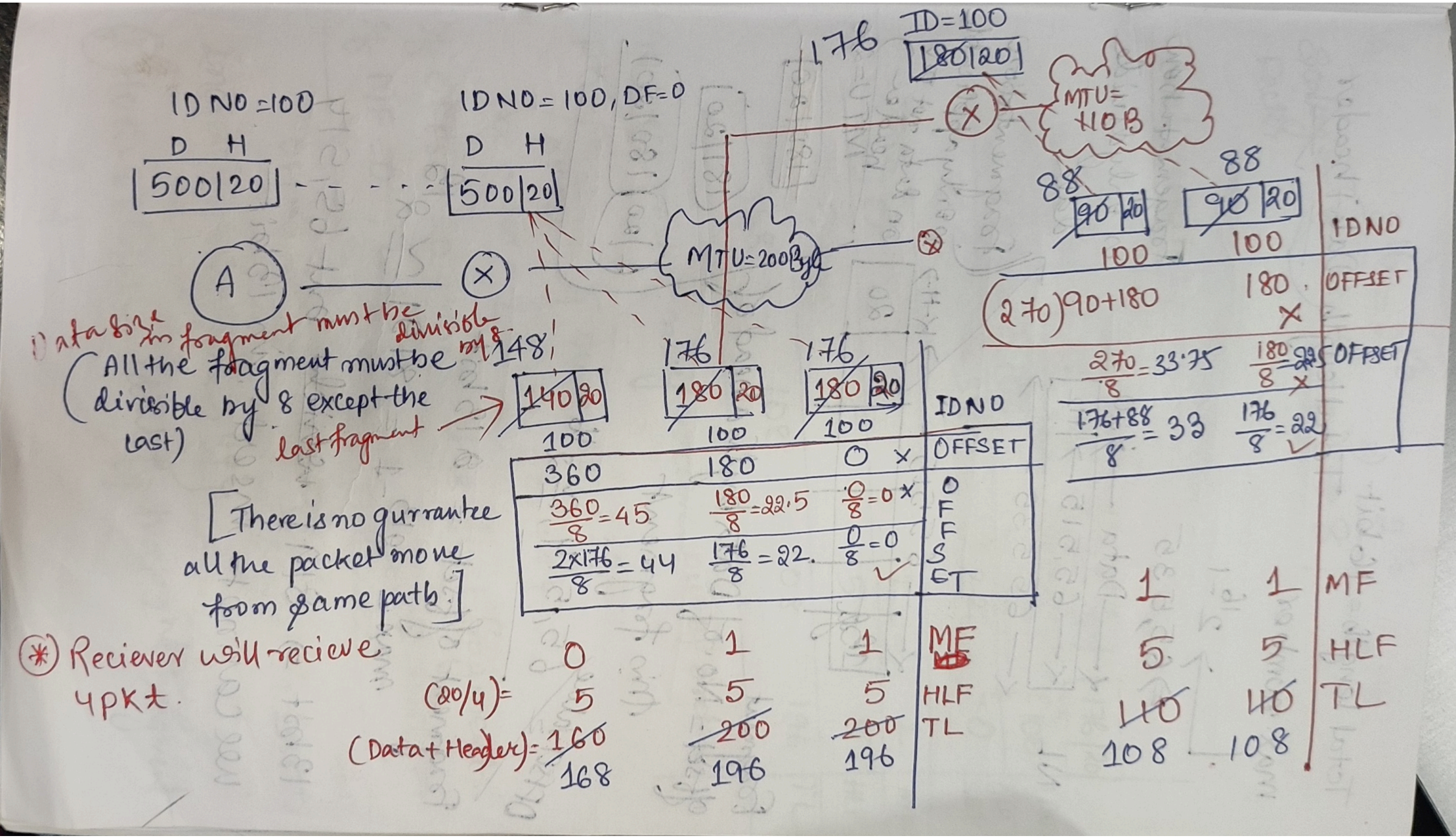

AL
msg
TL
H1
segment
NL
H2
datagrame
Frame
DLL
T1
H3
Segmentation → TL
Fragmentation → NL
Framing → DLL


148
160
20
176
180
20
MF=1
176
180
20
MF=1
148
140
20
MF=0
88
90
20
MF=1
88
90
20
MF=1
176
180
20
MF=1


ID NO = 100
D H
500 20
ID NO = 100, DF = 0
D H
500 20
A
MTU = 200 Byte
MTU = 110 B
ID = 100
180 20
176
88
90 20
90 20
100
100
IDNO
OFFSET
(270) 90+180
180
×
270/8 = 33.75
180/8 = 22.5
OFFSET
176+88/8 = 33
176/8 = 22
1) data size in fragment must be divisible by 8
(All the fragment must be divisible by 8 except the last)
last fragment
148
140 20
180 20
180 20
100
100
100
IDNO
OFFSET
360
180
0 ×
360/8 = 45
180/8 = 22.5
0/8 = 0 ×
2×176/8 = 44
176/8 = 22.
0/8 = 0
0
F
F
S
LT
[There is no gurrantee all the packet move from same path.]
* Reciever will recieve 4 pkt.
0
1
1
MF
(20/4) = 5
5
5
HLF
(Data + Header) = 160
200
200
TL
168
196
196
1
1
MF
5
5
HLF
140
140
TL
108
108

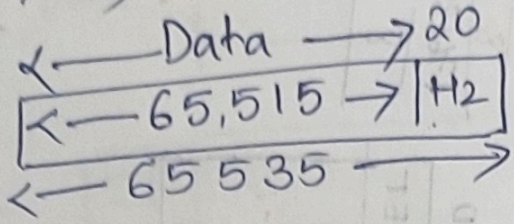
Total length = 16 bit

Total length = Data + header

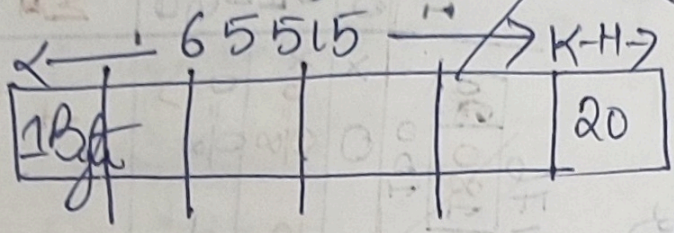
max number

$\rightarrow 2^{16}-1$

$\Rightarrow 65,535$



Fragmentation is only divide data



Fragmentation is only applicable on data not on header

M.T.U = 200

~~180/20~~

181/20

1/20 180/20

Fragment offset = No. of Data bytes ahead of this fragment

OFFSET value $\leq 65,514 \cong 2^{16}$

Fragment offset = 13 bit

$max = 2^{13}-1 = 8191 \cong 2^{13}$

13 bit ⇒ 8191 ~~which~~ but 65,514

we cannot insert in 13 bit.

$$\frac{2^{16}}{x} = 2^{13} \Rightarrow \frac{2^{16}}{8} = 2^{13}.$$

Scaling Factor ← x (S.F) → 8

Final results:

148 | 20	88 | 20	88 | 20	176 | 20	IDNO
100	100	100	100	OFF SET
44	33	22	0	
0	1	1	1	MF
5	5	5	5	HLF
168	108	108	196	TL

All these packet may recieve out of order pkt.

OFFSET = {44, 0, 33, 22}

Inc. order = {0, 22, 33, 44}.

If OFFSET = 0, then it is first fragment.

If MF = 0, then it is last fragment.

OFFSET = {44, 0, 33, ~~88~~ 22}

suppose 22 lost

increasing order = {0, 33, 44}

(*) How reciever identify that 2nd fragment lost.

| 176. | 20 | → 1st Fragment

$\frac{176}{8} = 22$ → 2nd Fragment
OFFSET.

Reciever then say that "22" is lost.

| 88 | 20 | | 176 | 20 |

$\frac{176+88}{8} = 33 \Rightarrow$ 3rd Fragment
OFFSET

| 88 | 20 | | 88 | 20 | | 176 | 20 |

$\frac{176+88+88}{4} = 44 \Rightarrow$ 4th Fragment
OFFSET.

Reassemble Algorithm

If each fragment follow a different path and arrives out of Order, the final destination host can reassemble the original datagram from the fragment recieved by using the following strategy:-

1. Identify the fragment OFFSET = 0 and it is first fragment.

2. Identify the fragment with MF = 0 and it is the last fragment.

3. Divide the data length of the first fragment by 8. The second fragment has an Offset value equal to that result. ($\frac{176}{8} = 22 \Rightarrow$ 2nd Fragment OFFSET)

4. Divide the data length of the first and Second fragment by 8. The third fragment has an offset value equal that result ($\frac{176+88}{8} = 33$

5. Repeat this process as many times as possible to cover all fragments.

Note (If fragment loss, reciever discard all packet then ask for all ~~pkt.~~ ~~data~~ plet.)