Essential Modules

CRC (Cyclic Redundancy Check)

CRC is a standard process to check Error in the transmission, in the communication system. The CRC process is divided into 2 parts.

- CRC Attach
- CRC check / CRC removal

- Consider, the data to be transmitted 1010111...1
- We use a particular sequence of bits (aka) CRC polynomial
  For example : 11001 (5 bits)
  to perform modulo-2 binary division with the data.

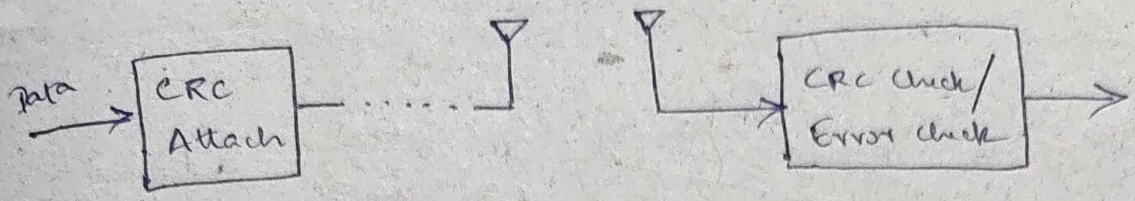
```
11001 ) 1010111....1
        ⊕ 11001
          ------
            ....
            ------
              1001  (4 bits) ← Remainder
```

∵ the CRC polynomial is of 5 bits, the Remainder will be of maximum 4 bits.

- This Remainder has to be attached with the data to be transmitted.

  1010111...11001

Now, this becomes the complete set of data to be transmitted.



- The CRC polynomial is known at both the transmitter and receiver side. So, at the receiver side, we use the same CRC polynomial to check for Error. How?

```
11001 ) 1010111 ... 11001
        ⊕ 11001
          .....
                  ------------
                     Remainder
```

- If the received data exactly matches with the transmitted data, then the Remainder = 0.

  Otherwise, the Remainder = Non-Zero value.

- So, at the receiver side, we can find out whether the received sequence is Error free or it contains Error. This operation does not tell which bit is wrong (or) how many bits are wrong.

- It may also happen that, out of 10,000 received bits, there is only one bit that is wrong. Even in this case, the remainder is Non-Zero value.

⊙ In 5G, following CRCs are being used.

CRC 6 , CRC 11 , CRC 16 , CRC 24 A
CRC 24 B
CRC 24 C

Degree of the polynomial /
Length of the Remainder bits.

For example: CRC 6 → 11,00001

Channel Coding

Channel coding is used to correct the errors caused by the wireless channel. Channel coding comprises 2 blocks.

① Channel Encoder

The information bits that we want to transmit is fed into the channel Encoder.

Channel Encoder will add redundancy to the information bits. With the different techniques used by different channel encoders, the information bits are transformed into different sequence, which became the final bits that are going to be transmitted.

② Channel Decoder

The corresponding block at the receiver side is the channel Decoder.

channel Decoder knows exactly how the information being transformed by the channel Encoder. It uses that information and the redundancies, to retrieve the information bits that were transmitted.

Channel Coding Algorithms in 5G NR

① LDPC (Low Density Parity Check Codes)
Used in PDSCH, PUSCH

② Polar codes
Used in PDCCH, PUCCH (While transmitting >12 bits), PBCH

③ Reed-Muller codes
Used in PUCCH (While transmitting 3 to 11 bits)

④ Repetition codes
Used in PUCCH (while transmitting 1 bit)

⑤ Simplex codes
Used in PUCCH (While transmitting 2 bits)

LDPC codes :

- First discovered in 1963
- But being neglected for almost 35 years due to very high computational complexity. (Encoding & Decoding complexity is high)
- One advantage is that, it supports Iterative Decoding. (ii) Output of one decoder is fed to the subsequent decoder. The benefit is that, more the iterations, better the decoding. But the disadvantage is that, it requires more time. Example, if one decoder takes 100 μs, using 10 of them will take 1 ms. So, there is Trade-off b/w No. of iterations and time.
- Another Advantage of LDPC codes is that, High data rates with low complexity, which other decoders do not support normally. (ii) at high data rates, LDPC codes are executed with low complexity.
- LDPC performs better than Turbo (Turbo codes were used in LTE shared channels).
- Another advantage of LDPC is that, it supports Incremental Redundancy that is being used in HARQ operations and Soft combining. This is a very required feature for the shared channels (PDSCH & PUSCH)
- Another advantage of LDPC is that, it can be divided into Small blocks. In this way, we can have high parallelism in Hardware implementation.
- One of the requirement of shared channel is that, we need to have wide range of block lengths & coding rates. For example, in PDSCH and PUSCH, we might have from 16 to 4 lakh bits. This is the wide range of block lengths. LDPC runs from 16 to 8448 bits. If the bit

size increases beyond 8448, then we divide into multiple LDPC blocks.

Also, in 5G, we have coding rate, as low as 1/5 and as high as 8/9.

- High Encoding Complexity & Low Decoding complexity.

Polar Codes :

- Polar codes are being used in Control and Broadcast channels (PUCCH, PDCCH, PBCH). These channels carry very less number of information bits, and requires very high reliability and robustness. Hence Polar codes suits better as the block size is small.
- One of the disadvantage of Polar code is that, it doesn't support HARQ process, and redundancy is not supported. This could be the reason why it cannot be used for shared channels.

Plots : Inference

Fig. 3.

For Block length of 64 bits and Coding Rate of 1/3, we see that Polar is better than LDPC. (or) almost 1.5 dB better

Fig. 4.

For larger block size, Eg. 1024 bits, there is no much difference b/w Polar and LDPC.

This is the reason why Polar codes can be used for control channels (with short block lengths) and LDPC can be used for shared channels.

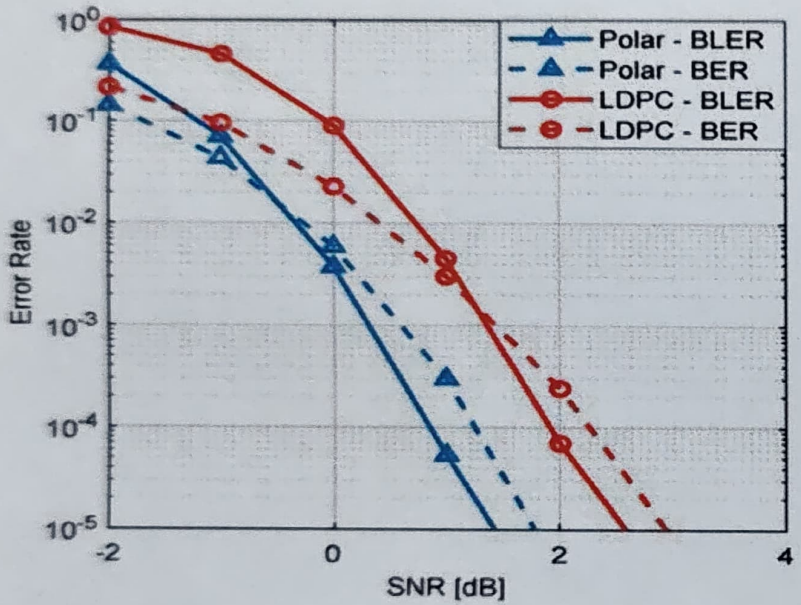


Fig.3. BER and BLER Performance of channel codes for K = 64 bits and R=1/3.

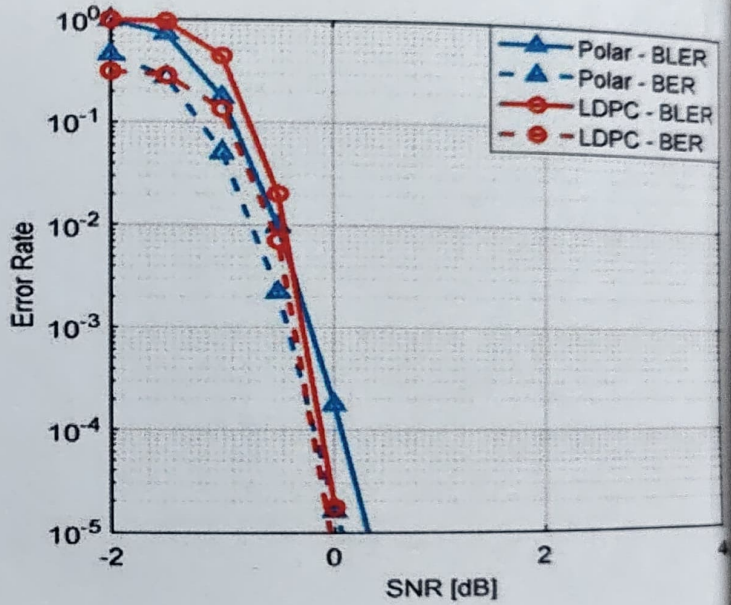


Fig.7. BER and BLER Performance of channel codes for K = 1024 bits and R=1/3.

Reed Muller codes :-

Reed Muller codes are used for PUCCH, when the control bits are between 3 – 11 bits.

Reed Muller codes and Polar codes are very much similar.

Advantage of RM code is that, it performs better than Polar codes (for small Block length), as we use the MAP decoder. For Large block length, Polar code performs better.

Note :

- When the block length increases, performance of Polar code and LDPC code increases.
- But, if the block length is small, both LDPC and Polar codes performs poor, when compared to RM code.

This is the reason behind, why we use RM code for PUCCH when the information bits are less.

Rate Matching.

Rate Matching is the module that comes after channel Encoding. at the Transmitter side. (i). the output of the channel encoder is fed to Rate Matching module.

⊙ Why we need Rate Matching? What exactly this module do?

consider, the Output of channel encoder is 200 bits. And these 200 bits needs to be transmitted over the resources that are available.

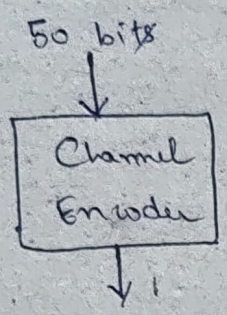
time (OFDM symbol)
frequency (subcarriers)
space (Layers)

So, the resources that are available, may not exactly match with the number of channel encoding output bits. Say, the available resources may carry 400 bits, or in other case, it may carry just 150 bits.

So, Mapping this 200 bits to these resources that are available, is taken care by Rate Matching module.

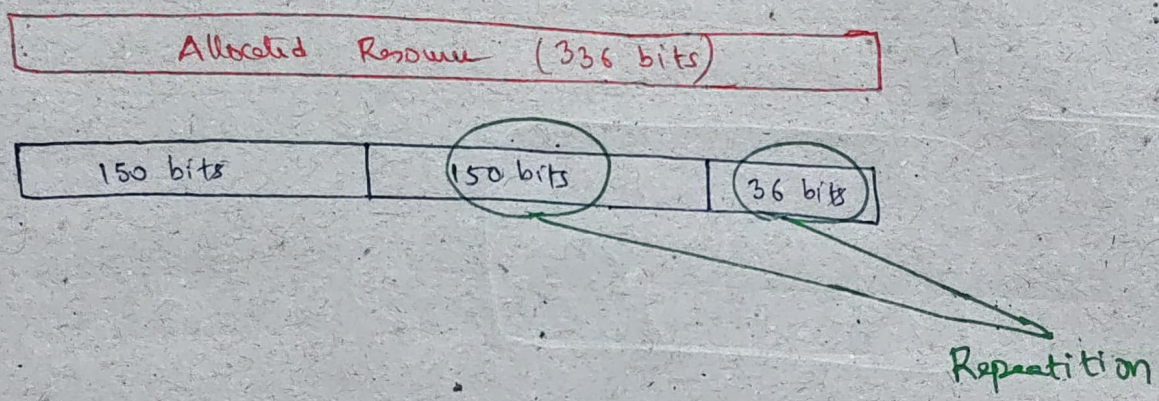
Rate Matching module also gives us the desired / requested code rate, that is configured in the System.

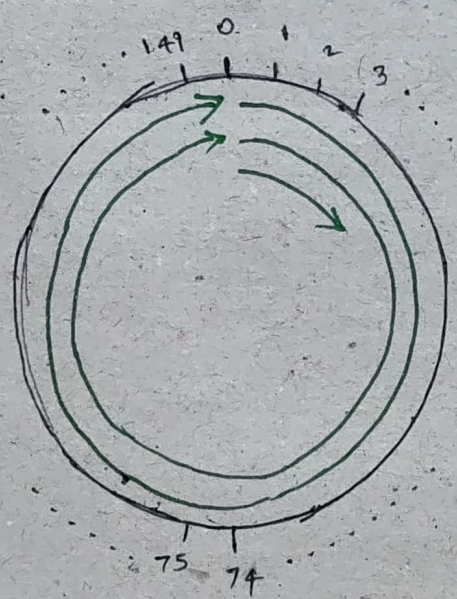
Example 1:



(Base Code Rate = $\frac{1}{3}$)

150 bits (No. of bits to be transmitted)





Circular Buffer

Repetition occurs when the allocated Resource is more than the Number of bits to be transmitted.

$$\text{Effective Code Rate} = \frac{\text{No. of bits input to channel Encoder}}{\text{No. of bits Output from Rate Matching}}$$

$$= \frac{50}{336} = 0.15$$

Example 2

150 bits

↓

Channel Encoder (Base code Rate $= \frac{1}{3}$)

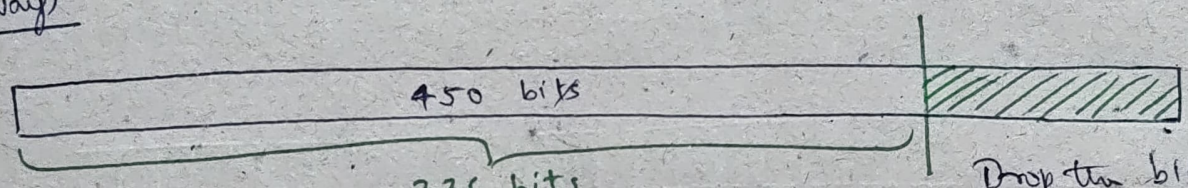
↓

450 bits (No. of bits to be transmitted)

Allocated Resource (336 bits)

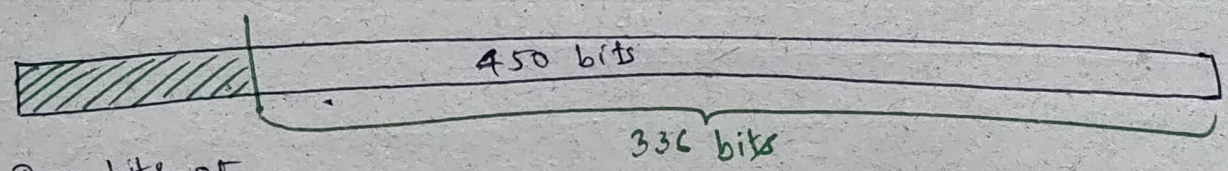
2 ways

(i)

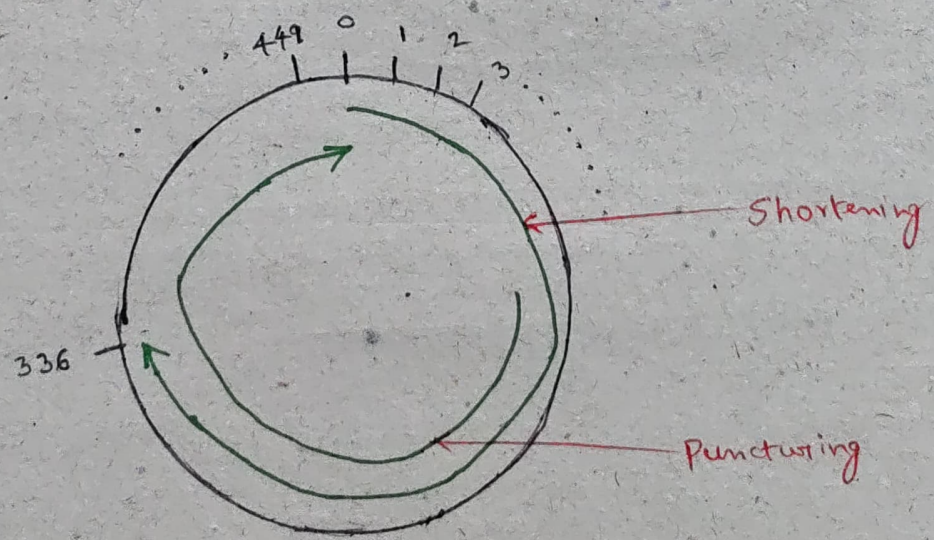


Drop the bits at the end.
(Shortening)

(ii)



Drop bits at the beginning.
(Puncturing)



Circular Buffer

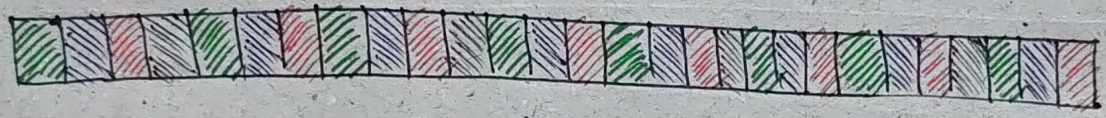
Effective Code Rate $= \frac{150}{336} = 0.45$

So, this is how Rate Matching helps matching the allocated resources (or) desired code rate.

Another operation, which Rate Matching does is, Interleaving.

Lets assume the Rate Matching output bits sequence (336 bits) as below.

Instead of transmitting just like above, we actually interleave the bits and transmit something like below.



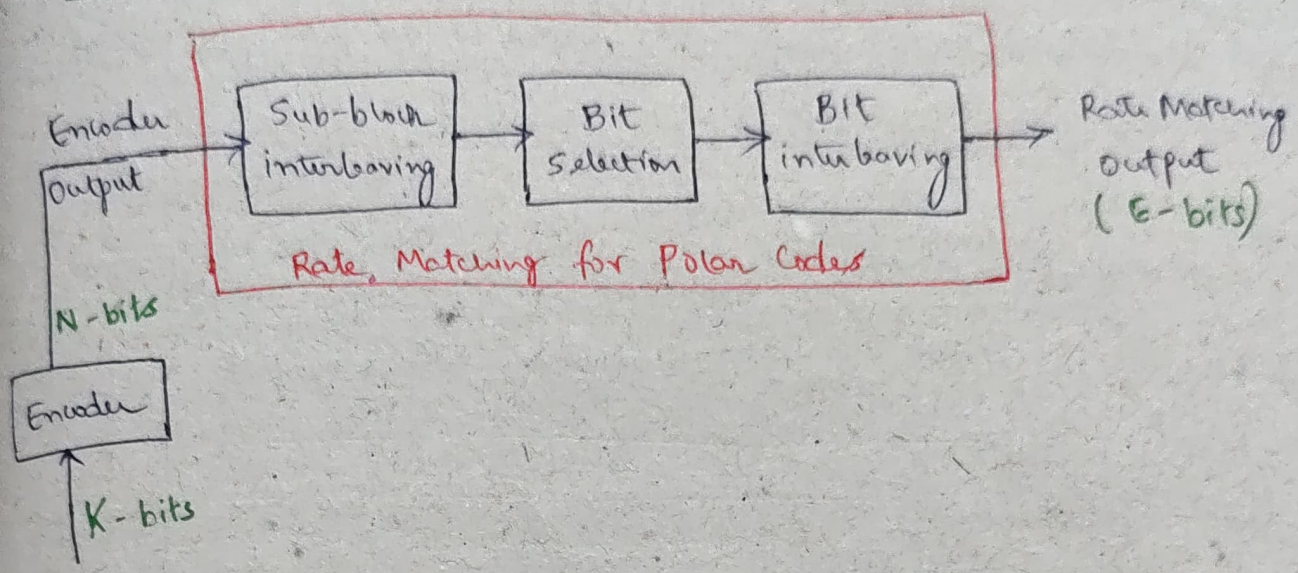
This pattern may change, for different Rate Matching functions or different channels. But the concept remains the same.

The benefit of transmitting in Interleaved fashion is that, if there is a burst error, because of the channel conditions, then transmitting without interleaving costs loss of continuous bits. Whereas, while transmitting in interleaved fashion, if burst error happens, here we lose continuous bits, but they are not actually continuous.

So, Interleaving helps the channel decoder in decoding bits correctly.

Rate Matching module is different for Polar channel encoders and LDPC channel encoders.

Rate Matching for Polar Codes



(i) Sub-block interleaving

The sub-block interleaving pattern $P(i)$ is defined for $0 \le i \le 31$. (In Table 5.4.1.1-1 of 38.212)

So, we take a block of 32 bits, apply the interleaving. And then take next 32 bits, apply the same, and so on...

This is how the sub-block interleaving is done.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

↓

0	1	2	4	3	5	6	7	8	16	9	17	10	18	11	19	12	20	13	21	14	22	15	23	24	25	26	28	27	29	30	31

This sub-block interleaving mainly increases the coding performance (ie) Better BER

(ii) Bit Selection

Puncturing (We dont transmit initial bits)

						6	7	8	16	9	17	10	18	11	19	12	20	13	21	14	22	15	23	24	25	26	28	27	29	30	31

0	1	2	4	3	5	6	7	8	16	9	17	10	18	11	19	12	20	13	21	14	22	15	23	24	25

Shortening (we dont transmit last bits)

0	1	2	4	3	5	6	7	8	16	9	17	10	18	11	19	12	20	13	21	14	22	15	23	24	25	26	28	27	29	30	31	0	1	2	4	3	5	6	...

Repeat until N == E

5.4.1.1 Sub-block interleaving

The bits input to the sub-block interleaver are the coded bits $d_0, d_1, d_2, \ldots, d_{N-1}$. The coded bits $d_0, d_1, d_2, \ldots, d_{N-1}$ are divided into 32 sub-blocks. The bits output from the sub-block interleaver are denoted as $y_0, y_1, y_2, \ldots, y_{N-1}$, generated as follows:

for $n = 0$ to $N-1$

$i = \lfloor 32n / N \rfloor$;

$J(n) = P(i) \times (N/32) + \mathrm{mod}(n, N/32)$;

$y_n = d_{J(n)}$;

where the sub-block interleaver pattern $P(i)$ is given by Table 5.4.1.1-1.

Table 5.4.1.1-1: Sub-block interleaver pattern $P(i)$

i	$P(i)$	i	$P(i)$	i	$P(i)$	i	$P(i)$	i	$P(i)$	i	$P(i)$	i	$P(i)$	i	$P(i)$
0	0	4	3	8	8	12	10	16	12	20	14	24	24	28	27
1	1	5	5	9	16	13	18	17	20	21	22	25	25	29	29
2	2	6	6	10	9	14	11	18	13	22	15	26	26	30	30
3	4	7	7	11	17	15	19	19	21	23	23	27	28	31	31

```
if E ≥ N   -- repetition
    for k = 0 to E − 1
        e_k = y_mod(k,N) ;
    end for
else
    if K / E ≤ 7/16   -- puncturing
        for k = 0 to E − 1
            e_k = y_k+N−E ;
        end for
    else   -- shortening
        for k = 0 to E − 1
            e_k = y_k ;
        end for
    end if
end if
```

(iii) Bit Interleaving (aka) Channel Interleaver

This Bit Interleaving is not defined for all the channels. (ie) If $I_{BIL} = 1$, then we do interleaving. Otherwise, there is no interleaving.

So, for channels like PBCH & PDCCH, there is no interleaving. For other channels like PUCCH and PUSCH, there is interleaving.

So, basically this bit interleaving improves the coding performance for Higher Order modulation. That's why it is not being used in downlink.

This Bit Interleaving can be explained using a triangular Matrix. Assume 40 bits are output from ~~Bit Selection~~ Rate Matching module. Assume $T = 9$.

1	2	3	4	5	6	7	8	9
10	11	12	13	14	15	16	17	
18	19	20	21	22	23	24		
25	26	27	28	29	30			
31	32	33	34	35				
36	37	38	39					
40								

The input to the Bit interleaver (Bit Selection output) is placed in a triangular Matrix as shown above, and is read in column wise. So, the final output would be

1	10	18	25	31	36	40	2	11	19	26	32	37	3	12	.....

5.4.1.3 Interleaving of coded bits

The bit sequence $e_0, e_1, e_2, \ldots, e_{E-1}$ is interleaved into bit sequence $f_0, f_1, f_2, \ldots, f_{E-1}$, as follows:

```
If I_BIL = 1
    Denote T as the smallest integer such that T(T+1)/2 ≥ E;
    k = 0;
    for i = 0 to T-1
        for j = 0 to T-1-i
            if k < E
                v_{i,j} = e_k;
            else
                v_{i,j} =< NULL >;
            end if
            k = k+1;
        end for
    end for
    k = 0;
    for j = 0 to T-1
        for i = 0 to T-1-j
            if v_{i,j} ≠< NULL >
                f_k = v_{i,j};
                k = k+1
            end if
        end for
    end for
else
    for i = 0 to E-1
        f_i = e_i;
    end for
end if
```

The value of E is no larger than 8192.

Rate Matching for LDPC Codes

WKT, LDPC is used for Shared channels (PDSCH/PUSCH).

Topics

① Limited Buffer Rate Matching (LBRM) & Full Buffer Rate Matching (FBRM)

② Redundancy Versions (RV)

③ Bit Interleaver

① LBRM / FBRM

Why LBRM / FBRM ?

WKT, the Encoder output is written in a Circular Buffer and produce desired number of bits that are actually going to be transmitted.

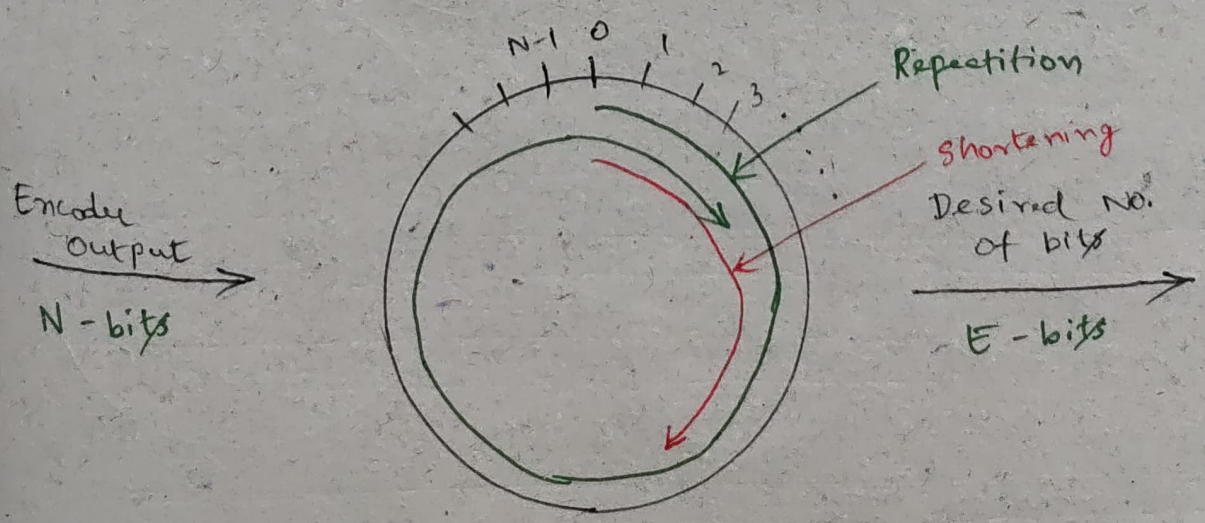
And, this whole process is done in 3 ways.

- Shortening
- Repetition
- Puncturing

Shortening, Repetition — Used in LDPC

Puncturing — Used in Polar

Note: Puncturing is not used in LDPC



Circular Buffer

In Limited Buffer Rate Matching (LBRM), instead of having buffer of size N, we use a buffer of lesser size.

By limiting the buffer size, the memory requirement as well as the implementation complexity got reduced. And hence, decoding latency at the receiver side also got reduced.

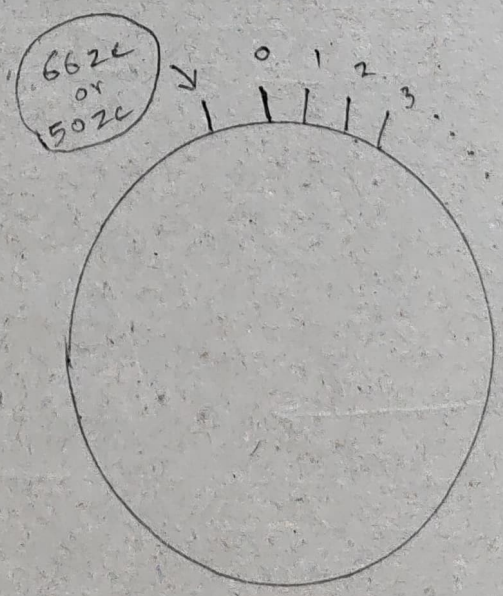
LBRM	$\xrightarrow{\text{used in}}$	PDSCH (mandatory), PUSCH (optional)
FBRM	$\rightarrow$	PUSCH

Note: In PDSCH, we always use LBRM, because the decoding part is done at the UE side. By limiting the buffer size, the memory requirement at the UE side got reduced.

How LBRM impacts the performance?

WKT, for LDPC encoder/decoder, it has the following parameters.

- Base code Rate
- Base Graphs (BG1 & BG2)
- Z_c



For Base Graph 1,

* Input size = 22 Z_c
* Output size = 66 Z_c
* Code Rate = $\frac{1}{3}$

For Base Graph 2,

* Input size = 10 Z_c
* Output size = 50 Z_c
* Code Rate = $\frac{1}{5}$

$Q_M = 6$

L_M \ P_M	32	66	107	135	162	217	273
1	27656	57376	92200	116792	139376	188576	237776
2	55304	114776	184424	233608	278776	376896	475584
3	83976	172176	278776	352440	417976	557416	704904
4	110632	229576	368872	467240	557416	753816	950984

$Q_M = 8$

L_M \ P_M	32	66	107	135	162	217	273
1	36896	75792	122976	155776	188576	250056	319784
2	73776	151608	245976	311368	376896	500136	638984
3	110632	229576	368872	467240	557416	753816	950984
4	147576	303240	491800	622760	753816	999576	1277992

So, ideally, the Buffer size would be either $50Z_c$ or $66Z_c$ based on what is the Base Graph of LDPC.

By limiting the size (less than $50Z_c$ or $66Z_c$), if the code rate is very high, that means we are actually transmitting lesser number of bits than the actual buffer size (Shortening). Here, the buffer size doesn't impact.

(ii) For Higer code Rate, performance doesn't change.

If the code rate is low, that means we are actually transmitting more number of bits than the actual buffer size (Repeatition). Here, the performance will get impacted.

So, the formula to calculate the buffer size (fixed in Specification) impact only the large code blocks. It depends on the following parameters

- Modulation Order (Q_M)
- Layers (L_M)
- PRB size (P_m)

Eg. 1 If PRB size, $P_m = 25$

Layers, $L_m = 2$

Modulation Order, $Q_M = 6$

then the Buffer size would be 55304.

Eg. 2 If PRB size, $P_m = 250$

Layers, $L_m = 3$

Modulation Order, $Q_M = 8$

then the Buffer size would be 950984.

So, this Buffer size is fixed in case of LBRM. But in case of FBRM, we have to use the full buffer (i.e) $66Z_c$ or $50Z_c$.

② Redundancy versions (RV)

WKT, a circular buffer is of size N.

Based on the Code Rate (or) desired bits, we pick as many bits as we transmit, from the circular buffer.

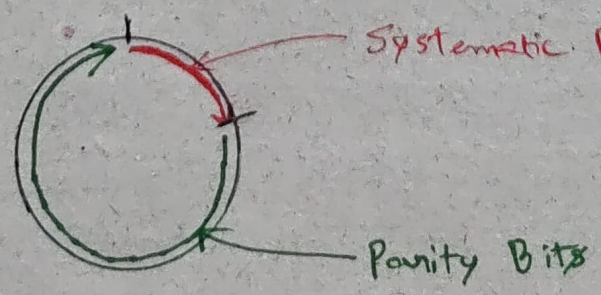
We dont always start picking the bits from 0th bit. We can pick the bits anywhere from the circular buffer. This particular concept is known as Redundancy version (RV).

Redundancy Version (RV) tells us, "what is the starting point, to pick the bits from the circular buffer".

Another concept related to LDPC is that, the output of LDPC has

- Systematic bits and
- Parity bits.

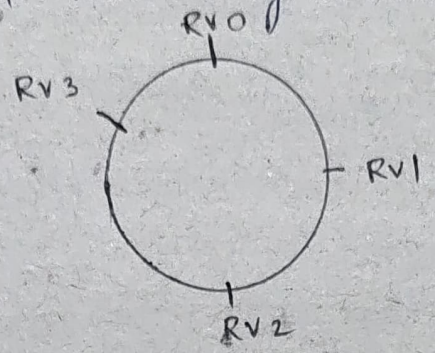
And, we write the systematic bits followed by parity bits, in the circular buffer.



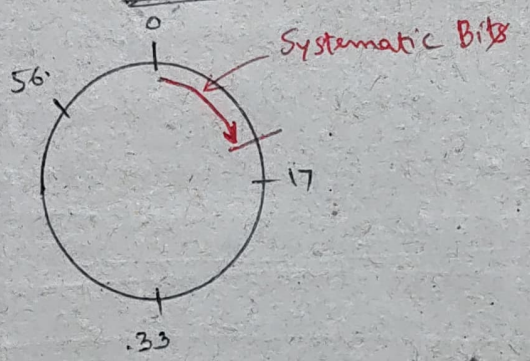
able 5.4.2.1-2: Starting position of different redundancy versions, k_0

rv_{id}	k_0	
	LDPC base graph 1	LDPC base graph 2
0	0	0
1	$\left\lfloor \frac{17N_{cb}}{66Z_c} \right\rfloor Z_c$	$\left\lfloor \frac{13N_{cb}}{50Z_c} \right\rfloor Z_c$
2	$\left\lfloor \frac{33N_{cb}}{66Z_c} \right\rfloor Z_c$	$\left\lfloor \frac{25N_{cb}}{50Z_c} \right\rfloor Z_c$
3	$\left\lfloor \frac{56N_{cb}}{66Z_c} \right\rfloor Z_c$	$\left\lfloor \frac{43N_{cb}}{50Z_c} \right\rfloor Z_c$

There are 4 redundancy versions defined in Specification.



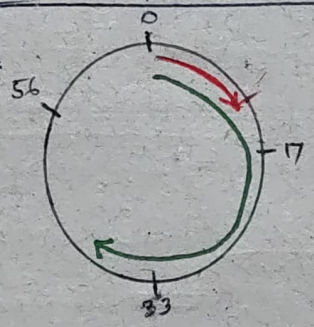
For LDPC BG1, the circular buffer can be represented as



Input Size = 22 Z_c

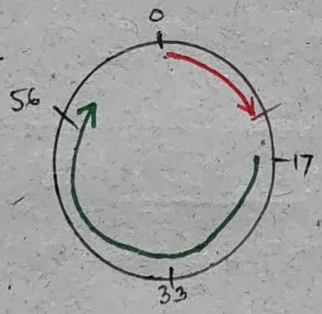
Output Size = 66 Z_c

RV0



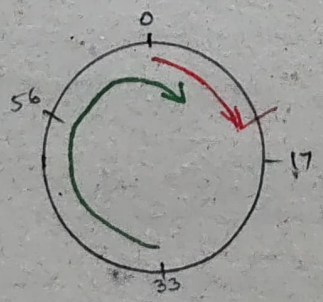
If Redundancy version is 0, start picking from Bit 0 and transmit the desired number of bits.

RV1



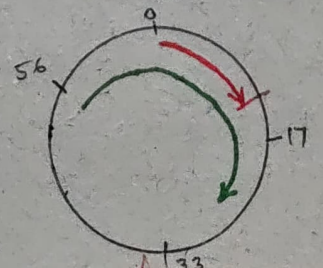
If Redundancy version is 1, start picking from Bit 17 and transmit the desired number of bits.

RV2



If Redundancy version is 2, start picking from Bit 33 and transmit the desired number of bits.

RV3



If Redundancy version is 3, start picking from Bit 56 and transmit the desired number of bits.

Notice 1 :

The Systematic bits are transmitted only in Redundancy versions 0 and 3.

In other Redundancy versions 1 and 2, the Systematic bits are not actually transmitted, until unless the code rate is very low.

If the code rate is really low, then full buffer is actually transmitted and also do repetition.

But if the code rate is high, the Systematic bits are not actually transmitted. So there is a concept called Self Decodable Redundancy versions.

What is Self Decodable Redundancy version ?

At the Receiver side, we take the transmitted bits (say RV0), put those in a circular buffer, and give it to the LDPC decoder, try to decode the information.

Now, since the Systematic bits are part of RV0, we dont actually require any other ~~bits~~ information bits to correctly decode the information bits.

So, RV0 is self sufficient, to retrieve the actually transmitted information bits. We need not combine RV0 with any other next transmission bits in order to decode the transmitted information. Hence it is called as Self Decodable Redundancy versions.

Actually, at the receiver side, this is a HARQ concept, where, in the first transmission, we receive some information bits. If it fails, then the transmitter will transmit next set of information bits. Then we'll combine both of them. With the combined information, we try to decode the information bits.

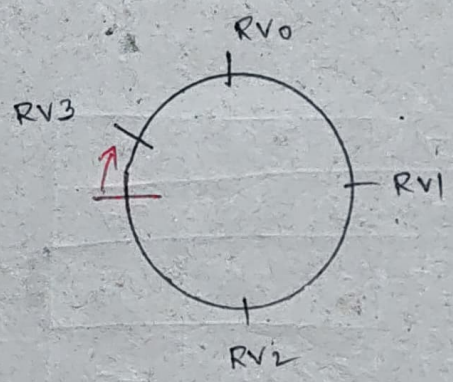
Here, RV0 is self decodable, because we can decode the information without combining it.

Similarly, RV3 is also self decodable, because this also captures the Systematic bits and also some parity bits.

Whereas RV1 and RV2 are not self decodable.

Notice 2:

Also notice that, the redundancy versions 0, 1, 2 and 3 are not uniformly distributed.



(ii) RV 3 would have been $\left\lfloor \frac{50\, N_{cb}}{66\, Z_c} \right\rfloor Z_c$ for BG1 and $\left\lfloor \frac{35\, N_{cb}}{50\, Z_c} \right\rfloor Z_c$ for BG2 (approx). Instead, RV3 is placed at $\left\lfloor \frac{56\, N_{cb}}{66\, Z_c} \right\rfloor Z_c$ and $\left\lfloor \frac{43\, N_{cb}}{50\, Z_c} \right\rfloor Z_c$ for BG1 and BG2 respectively.

The reason is that, it captures more Systematic bits, makes it more of a Self decodable Redundancy version, and helps in the HARQ (without having to combine with other bits).

---

③ Bit Interleaving

WKT, Bit Interleaving module helps decoders when there is burst error. So, the probability of successful decoding increases when Bit Interleaving is done.

Bit Interleaving in LDPC is done according to the modulation order (Q_m). For different Q_m, the Interleaving is different.

The Equivalent analogy for Specification defined formula is given below.

1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40

Assuming there are 40 bits after Rate Matching. The 40 bits are mapped in Row wise (as shown above) and read them column wise.

1	11	21	31	2	12	22	32	3	13	23	33	4	14	24	34	5	...

Note: The number of rows is decided by the modulation order. Here, in this example, the modulation order is assumed to be 4. (i.e) 16 QAM. (Mapping 4 bits to One Symbol)

Assume, at a particular instant, the channel went bad.

(i) a particular subcarrier (or) a particular symbol went bad.

With Bit-Interleaving module, we may loose Bits at positions 6, 16, 26, 36 instead of contiguous bits at positions 6, 7, 8, 9.

Hence, Bit Interleaving ensures, we dont look contiguous bits, which is very much advantageous in the decoding part.

5.4.2.2 Bit interleaving

The bit sequence $e_0, e_1, e_2, \ldots, e_{E-1}$ is interleaved to bit sequence $f_0, f_1, f_2, \ldots, f_{E-1}$, according to the following, where the value of Q_m is the modulation order.

for $j = 0$ to $E/Q_m - 1$

  for $i = 0$ to $Q_m - 1$

    $f_{i+jQ_m} = e_{i \cdot E/Q_m + j}$;

  end for

end for

5.2.1 Pseudo-random sequence generation

Generic pseudo-random sequences are defined by a length-31 Gold sequence. The output sequence $c(n)$ of length M_{PN}, where $n = 0,1,\ldots,M_{PN}-1$, is defined by

$$c(n) = (x_1(n+N_C) + x_2(n+N_C)) \bmod 2$$
$$x_1(n+31) = (x_1(n+3) + x_1(n)) \bmod 2$$
$$x_2(n+31) = (x_2(n+3) + x_2(n+2) + x_2(n+1) + x_2(n)) \bmod 2$$

where $N_C = 1600$ and the first m-sequence $x_1(n)$ shall be initialized with $x_1(0) = 1, x_1(n) = 0, n = 1,2,\ldots,30$. The initialization of the second m-sequence, $x_2(n)$, is denoted by $c_{init} = \sum_{i=0}^{30} x_2(i) \cdot 2^i$ with the value depending on the application of the sequence.

PN Sequence (aka) Gold Sequence:

PN Sequences are heavily used in 5G (in scrambling, in DMRS generation, and more)

PN Sequence is generated by XOR-ing two m-sequences.

$$\boxed{\text{PN Sequence} = \text{m-Seq}^{(1)} \oplus \text{m-Seq}^{(2)}}$$

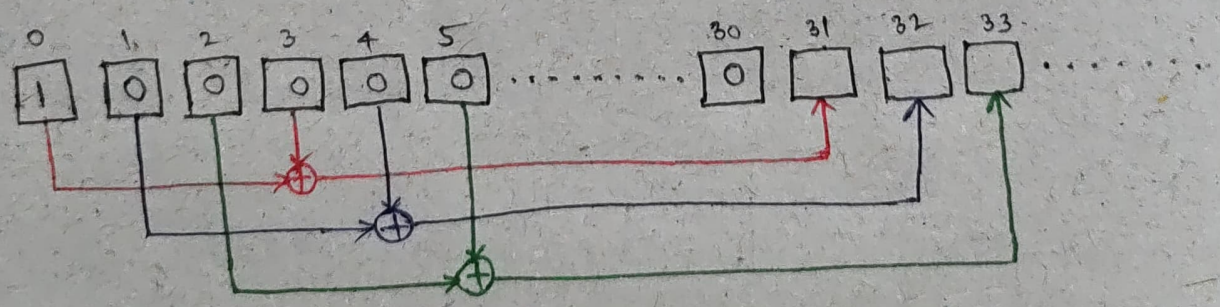
The property of m-sequence is, it has high autocorrelation, which is good. But the problem is, it has high cross correlation also.

But, we need sequence that has "low cross correlation". So, for that purpose, we do not use m-sequence directly. Instead, we generate two different m-sequences and do XOR operation.

How do we generate these m-sequences is as follows.

1st m-Sequence

- Initialize 0th bit as 1
- Initialize 1st bit to 30th bit with 0's
- XOR 0th bit with 3rd bit to generate 31st bit
  XOR 1st bit with 4th bit to generate 32nd bit
  and so on....

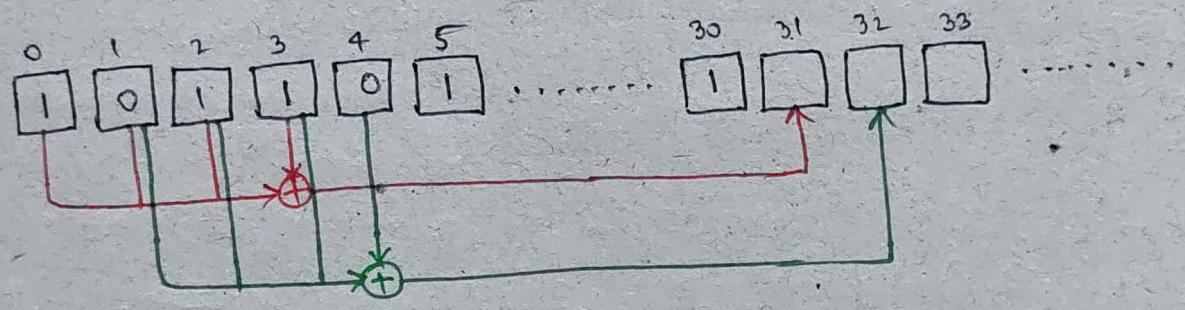


- This way, we continue generating the sequence.
- The length of the sequence depends on how many samples we need.

2nd m-sequence

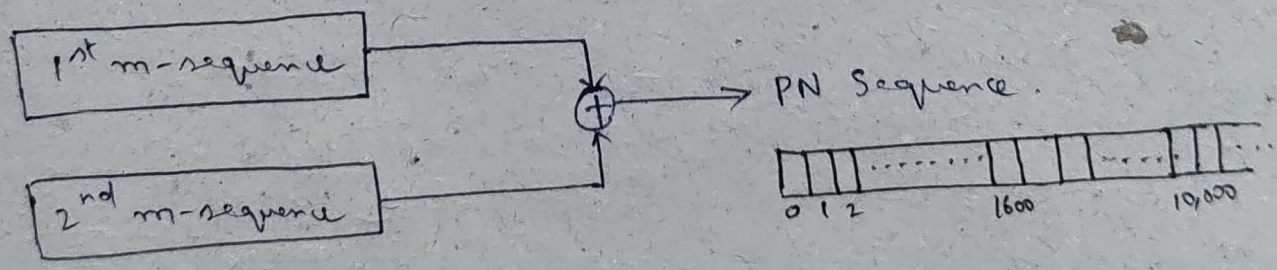
- The initialization of this sequence depends on different use cases. (ie) C_{init} value is different for, say different DMRS, different signals, different channels for scrambling. So, it depends on the use cases.
- Lets initialize here randomly.
- XOR bits at 0, 1, 2, 3 to form 31st bit
- XOR bits at 1, 2, 3, 4 to form 32nd bit

and so on...



- This way, we continue generating 2nd m-sequence.

PN Sequence



- We start reading the PN sequence from 1600 onwards. This is the point from which we start using the output of the PN sequence. This is because, we assume that, after 1600, the generated sequence is sufficiently random enough.

Zad-off Chu Sequence (aka) Low PAPR Sequence.

- Very famous sequence, heavily used in communication systems.
- In 5G also, this sequence is heavily used (mostly in UL)
- Zad-off chu sequence is defined as

$$\boxed{x_q(m) = e^{\frac{-j\pi q m(m+1)}{N_{zc}}}}$$

where, $N_{zc} = 3/5/7/\ldots$ (typically Prime number)

- It has a couple of very good properties.

(i) Constant amplitude ($\because$ it is an exponential term. No matter what the value of q and m, the amplitude remains 1)

(ii) Autocorrelation of this sequence is zero.

Consider $N_{zc} = 5$ (length of sequence $x_q(m)$)

$q = 1$

$$\Rightarrow x_1(m) = e^{\frac{-j\pi m(m+1)}{5}}$$

$$x_1(0) = e^0 = 1$$

$$x_1(1) = e^{\frac{-j\pi(1)(2)}{5}} = e^{-j\frac{2\pi}{5}}$$

$$x_1(2) = e^{\frac{-j\pi(2)(3)}{5}} = e^{-j\frac{6\pi}{5}}$$

$$x_1(3) = e^{\frac{-j\pi(3)(4)}{5}} = e^{-j\frac{12\pi}{5}} = e^{-j\frac{2\pi}{5}}$$

$$x_1(4) = e^{\frac{-j\pi(4)(5)}{5}} = e^{-j4\pi} = 1$$

So, the length 5 Z_c sequence is

$$x_1(m) = \left[1 \quad e^{-j\frac{2\pi}{5}} \quad e^{-j\frac{6\pi}{5}} \quad e^{-j\frac{2\pi}{5}} \quad 1\right]$$

By finding the cyclic shift on this sequence and calculate the auto correlation

(ii) $x_1^*(m) \cdot x_1^*(m+2) = 0$

5.2.2 Low-PAPR sequence generation

The low-PAPR sequence $r_{u,v}^{(\alpha,\delta)}(n)$ is defined by a cyclic shift α of a base sequence $\bar{r}_{u,v}(n)$ according to

$$r_{u,v}^{(\alpha,\delta)}(n) = e^{j\alpha n}\bar{r}_{u,v}(n), \quad 0 \le n < M_{ZC}$$

where $M_{ZC} = mN_{sc}^{RB}/2^{\delta}$ is the length of the sequence. Multiple sequences are defined from a single base sequence through different values of α and δ.

Base sequences $\bar{r}_{u,v}(n)$ are divided into groups, where $u \in \{0,1,\ldots,29\}$ is the group number and v is the base sequence number within the group, such that each group contains one base sequence ($v = 0$) of each length $M_{ZC} = mN_{sc}^{RB}/2^{\delta}$, $1/2 \le m/2^{\delta} \le 5$ and two base sequences ($v = 0,1$) of each length $M_{ZC} = mN_{sc}^{RB}/2^{\delta}$, $6 \le m/2^{\delta}$. The definition of the base sequence $\bar{r}_{u,v}(0),\ldots,\bar{r}_{u,v}(M_{ZC}-1)$ depends on the sequence length M_{ZC}.

5.2.2.1 Base sequences of length 36 or larger

For $M_{ZC} \ge 3N_{sc}^{RB}$, the base sequence $\bar{r}_{u,v}(0),\ldots,\bar{r}_{u,v}(M_{ZC}-1)$ is given by

$$\bar{r}_{u,v}(n) = x_q(n \bmod N_{ZC})$$

$$x_q(m) = e^{-j\frac{\pi q m(m+1)}{N_{ZC}}}$$

where

$$q = \lfloor \bar{q} + 1/2 \rfloor + v \cdot (-1)^{\lfloor 2\bar{q} \rfloor}$$

$$\bar{q} = N_{ZC} \cdot (u+1)/31$$

The length N_{ZC} is given by the largest prime number such that $N_{ZC} < M_{ZC}$.

5.2.2.2 Base sequences of length less than 36

For $M_{ZC} \in \{6,12,18,24\}$ the base sequence is given by

$$\bar{r}_{u,v}(n) = e^{j\varphi(n)\pi/4}, \quad 0 \le n \le M_{ZC}-1$$

where the value of $\varphi(n)$ is given by Tables 5.2.2.2-1 to 5.2.2.2-4.

For $M_{ZC} = 30$, the base sequence $\bar{r}_{u,v}(0),\ldots,\bar{r}_{u,v}(M_{ZC}-1)$ is given by

$$\bar{r}_{u,v}(n) = e^{-j\frac{\pi(u+1)(n+1)(n+2)}{31}}, \quad 0 \le n \le M_{ZC}-1$$

(iii) Cross correlation of this sequence is always constant

(ie) Cross correlation $= \frac{1}{\sqrt{N_{zc}}}$

For the above example, the sequence length $N_{zc} = 5$

Hence the cross correlation of that sequence $\Big\} = \frac{1}{\sqrt{5}}$

(iv) DFT of the Z_c sequence is also a Z_c sequence.

Again, this property is heavily used in OFDM.

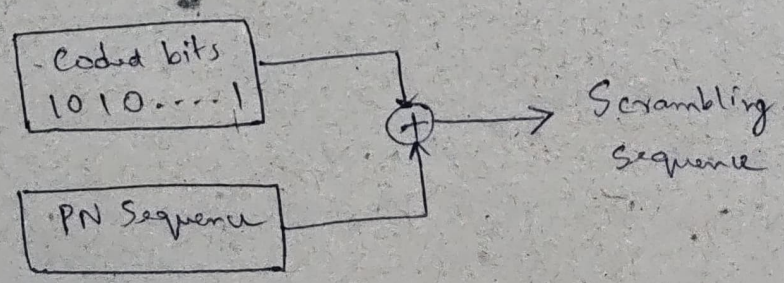
(ie) We don't need to take DFT (or) FFT to convert the sequence back to Frequency domain. Instead we can directly generate the sequence in Freq. domain.

Scrambling

- Scrambling is a process that randomize the information bits that is being transmitted.
- Scrambling is done at the transmitter side. And the reverse of it (Descrambling) is done at the reciver side.
- The purpose of doing the Scrambling process is to make sure that the signals from neighboring cells and different UEs acts as Noise to the signal that we are transmitting.
- So basically this Scrambling process reduces the inter-cell interference.
- The Second advantage is that, it ensures the full utilization of the processing gain provided by the channel code.
- So, this Scrambling process is kind of last module that changes the sequence of bits, before the transmission of the signal.

Example: Consider the coded bits 1010....1. The Scrambling process is nothing but XOR operation of this coded bits with the PN sequence.

-



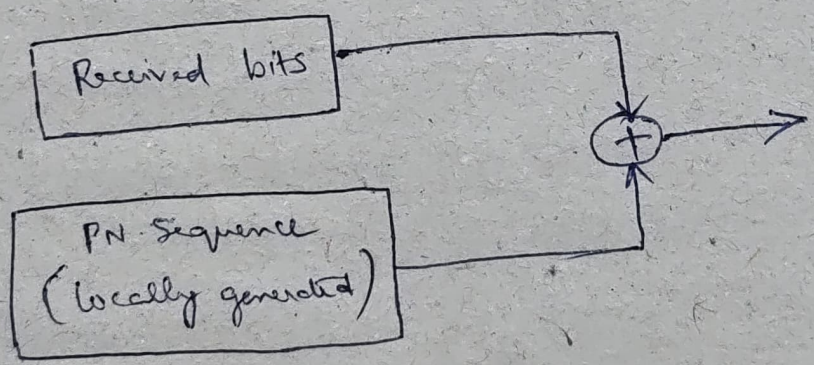
The PN sequence is fed with different inputs which are function of cell ID, RNTI, N_{ID}, ~~XXXXXXXX~~ Codeword, ... These functions are explained in channel specific topics.

If the input to the PN sequence is a function of cell Id, the PN sequence will be different for different cells.

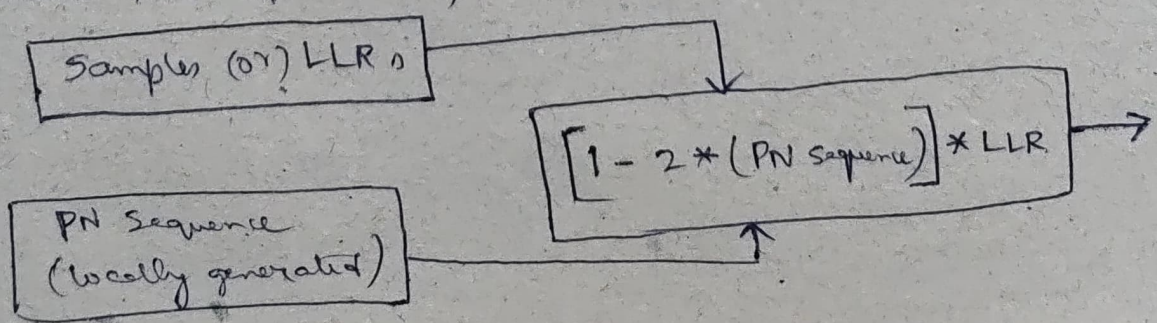
If the input to the PN sequence is a function of RNTI, the PN sequence will be different for different messages.

(ii) for each cell (or) message, we generate Unique sequence. and this sequence is used to do XOR operation with the coded bits. that we are transmitting.

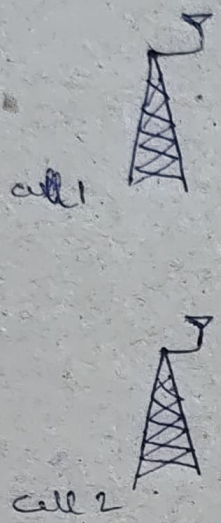
At the Receiver side, if the input to the Descrambling module is Bits, then the same process is executed at the receiver. (ii)



If the input to the Descrambling module is not Bits, but is Samples (or) LLRs (Log Likelihood Ratio), which can be Integers (or) Float numbers, then instead of doing direct XOR operation, we do



Advantage of Scrambling :



Consider there are 2 different cells.

- cell 1
- cell 2 (Neighboring cell)

Assume Both the cells are time synchronized. That means, the slot boundaries (or) the signal transmissions happens right at the same time in both the cells. (i.e) Both cells are transmitting some broadcast information at the same time, using the same resources.

In such scenario, it might happen that; both the signals interfere, and the UE may not be able to decode properly. (i.e) UE may decode cell 2 signal when it actually wanted to decode cell 1 signal.

In this case, Scrambling would be helpful.

Modulation Mapper

There are actually two things done in the Modulation of 5G, in the signal transmission of baseband.

(i) Mapping the bits to the Complex Samples

(ii) Modulate the OFDM Waveform with this Complex Samples

Following are the different modulation mappers that are defined and being used in 5G, in different channels/Signals.

① BPSK & $\frac{\pi}{2}$ BPSK	→	1	→	$a+ib$
② QPSK	→	2	→	$a+ib$
③ 16-QAM	→	4	→	$a+ib$
④ 64-QAM	→	6	→	$a+ib$
⑤ 256-QAM	→	8	→	$a+ib$

No. of bits mapped to a complex sample.

Throughput ↓ Error Probability ↓

So, higher the modulation order, the more the number of bits mapped to a complex sample, and hence higher the throughput.

So, 256-QAM has 8 times more throughput compared to BPSK:

Disadvantage:

If we move higher in the modulation order, then the Error probability also increases for the given scenario.

5.1 Modulation mapper

The modulation mapper takes binary digits, 0 or 1, as input and produces complex-valued modulation symbols as output.

5.1.1 π/2-BPSK

In case of $\pi/2$-BPSK modulation, bit $b(i)$ is mapped to complex-valued modulation symbol $d(i)$ according to

$$d(i) = \frac{e^{j\frac{\pi}{2}(i \bmod 2)}}{\sqrt{2}}\left[(1-2b(i)) + j(1-2b(i))\right]$$

5.1.2 BPSK

In case of BPSK modulation, bit $b(i)$ is mapped to complex-valued modulation symbol $d(i)$ according to

$$d(i) = \frac{1}{\sqrt{2}}\left[(1-2b(i)) + j(1-2b(i))\right]$$

5.1.3 QPSK

In case of QPSK modulation, pairs of bits, $b(2i), b(2i+1)$, are mapped to complex-valued modulation symbols $d(i)$ according to

$$d(i) = \frac{1}{\sqrt{2}}\left[(1-2b(2i)) + j(1-2b(2i+1))\right]$$

5.1.4 16QAM

In case of 16QAM modulation, quadruplets of bits, $b(4i), b(4i+1), b(4i+2), b(4i+3)$, are mapped to complex-valued modulation symbols $d(i)$ according to

$$d(i) = \frac{1}{\sqrt{10}}\left\{(1-2b(4i))\left[2-(1-2b(4i+2))\right] + j(1-2b(4i+1))\left[2-(1-2b(4i+3))\right]\right\}$$

5.1.5 64QAM

In case of 64QAM modulation, hextuplets of bits, $b(6i), b(6i+1), b(6i+2), b(6i+3), b(6i+4), b(6i+5)$, are mapped to complex-valued modulation symbols $d(i)$ according to

$$d(i) = \frac{1}{\sqrt{42}}\left\{(1-2b(6i))\left[4-(1-2b(6i+2))\left[2-(1-2b(6i+4))\right]\right] + j(1-2b(6i+1))\left[4-(1-2b(6i+3))\left[2-(1-2b(6i+5))\right]\right]\right\}$$

5.1.6 256QAM

In case of 256QAM modulation, octuplets of bits, $b(8i), b(8i+1), b(8i+2), b(8i+3), b(8i+4), b(8i+5), b(8i+6), b(8i+7)$, are mapped to complex-valued modulation symbols $d(i)$ according to

$$d(i) = \frac{1}{\sqrt{170}}\left\{(1-2b(8i))\left[8-(1-2b(8i+2))\left[4-(1-2b(8i+4))\left[2-(1-2b(8i+6))\right]\right]\right]\right.$$
$$\left. + j(1-2b(8i+1))\left[8-(1-2b(8i+3))\left[4-(1-2b(8i+5))\left[2-(1-2b(8i+7))\right]\right]\right]\right\}$$

So, if the channel quality is really bad between the transmitter and receiver, then it is preferred to use lower modulation order. And, if the channel quality is good, then it is preferred to use higher modulation order. So, there is a trade-off.

If we push high throughput in the transmission by increasing the modulation order, the chances that the bits will get decoded decreases. So, based on the scenario, we pick the right modulation order.

Consider $b = \underset{b(0)}{0}\ \underset{b(1)}{0}\ \underset{b(2)}{1}\ \underset{b(3)}{1}$

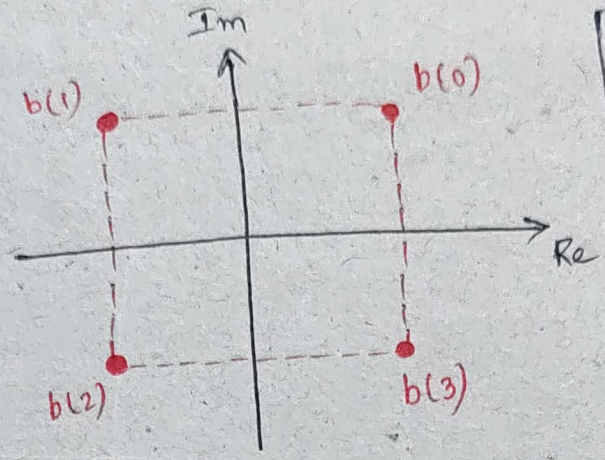
$\frac{\pi}{2}$ BPSK $\Rightarrow d(i) = \dfrac{e^{j\frac{\pi}{2}(i \bmod 2)}}{\sqrt{2}}\left[(1-2b(i)) + j(1-2b(i))\right]$

$$\Rightarrow d(0) = \frac{e^{j\frac{\pi}{2}(0)}}{\sqrt{2}}[1+j] = \frac{1}{\sqrt{2}}[1+j]$$

$$d(1) = \frac{e^{j\frac{\pi}{2}(1)}}{\sqrt{2}}[1+j] = \frac{j}{\sqrt{2}}[1+j] = \frac{1}{\sqrt{2}}[-1+j]$$

$$d(2) = \frac{e^{j\frac{\pi}{2}(0)}}{\sqrt{2}}[-1-j] = \frac{1}{\sqrt{2}}[-1-j]$$

$$d(3) = \frac{e^{j\frac{\pi}{2}(1)}}{\sqrt{2}}[-1-j] = \frac{j}{\sqrt{2}}[-1-j] = \frac{1}{\sqrt{2}}[1-j]$$



Note: The constellation looks similar to QPSK as there are 4 points. But is not equivalent to QPSK, coz here 1 bit is mapped to the complex-valued modulation symbol $d(i)$

Notice that, even though $b(0)$ and $b(1)$ (same bits) are transmitted, we are not actually transmitting same samples (at different quadrants).

Similarly, even though $b(2)$ and $b(3)$ (same bits) are transmitted, they are mapped to different samples.

So there is always $\frac{\pi}{2}$ (or) 90° phase shift exists. This is what makes it different from BPSK.

$\underline{\underline{\frac{\pi}{2}\text{ BPSK}}} \Rightarrow d(i) = e^{j\frac{\pi}{2}(i \bmod 2)} \cdot \frac{1}{\sqrt{2}}\left[(1-2b(i)) + j(1-2b(i))\right]$

$\underline{\underline{\text{BPSK}}} \Rightarrow d(i) = \frac{1}{\sqrt{2}}\left[(1-2b(i)) + j(1-2b(i))\right]$

From the above 2 equations, the only difference is that, for all odd samples, there is a $\frac{\pi}{2}$ phase shift.

This $\frac{\pi}{2}$ phase shift in BPSK reduces the PAPR in transmission, and we mostly use it in the Uplink.

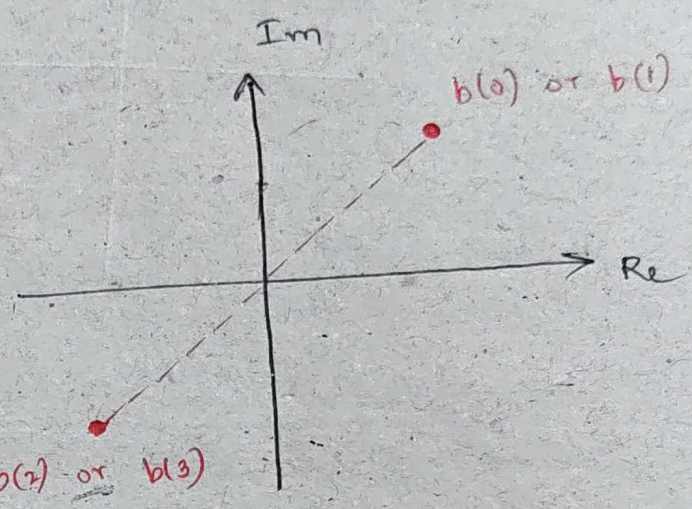
BPSK $\Rightarrow$ $d(i) = \frac{1}{\sqrt{2}}\left[\left(1-2b(i)\right) + j\left(1-2b(i)\right)\right]$

$\Rightarrow d(0) = \frac{1}{\sqrt{2}}[1+j]$

$d(1) = \frac{1}{\sqrt{2}}[1+j]$

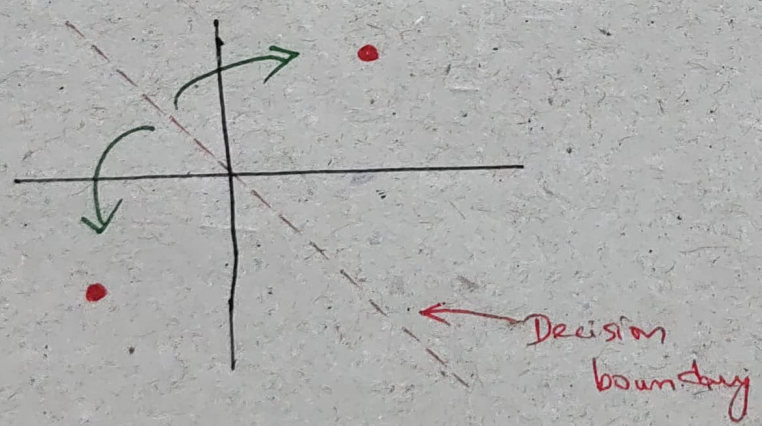
$d(2) = \frac{1}{\sqrt{2}}[-1-j]$

$d(3) = \frac{1}{\sqrt{2}}[-1-j]$



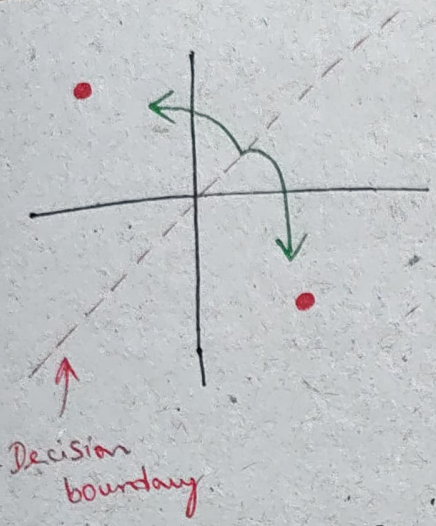
Demodulation of $\frac{\pi}{2}$ BPSK

We noticed that, at <u>even index</u>, $\frac{\pi}{2}$ BPSK is same as BPSK. So, at even index, following are the samples that we transmit (one of them).



(For Even samples)

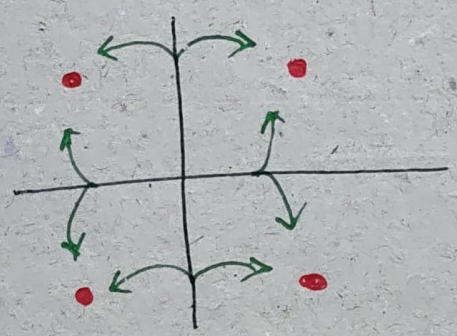
- If the transmitted sample is towards left side of decision boundary, then we decide that bit 1 is transmitted.
- If the transmitted sample is towards right side of decision boundary, then we decide that bit 0 is transmitted.



(For odd samples)

- If the transmitted sample is towards the left side of the decision boundary, then we decide that bit 0 is transmitted.
- If the transmitted sample is towards the right side of the decision boundary, then we decide that bit 1 is transmitted.

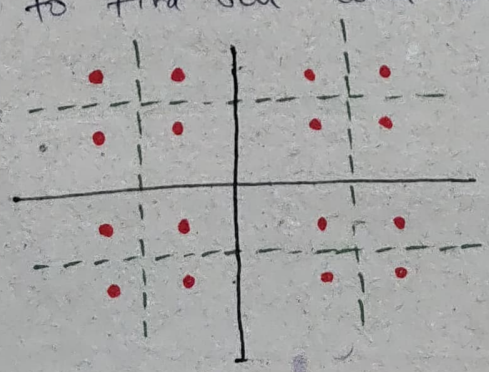
Similar decoding is done for QPSK.



- For QPSK, it is bit simpler.
- Here we directly use the axis instead of Decision boundary, to demodulate the bits transmitted.

Same analogy goes for 16-QAM, 64-QAM and 256-QAM, where the demodulation becomes more complex because there comes more decision boundaries.

(i.e) we have to demodulate more bits from a single complex valued sample. So we need to draw more decision boundaries to find out what bits are transmitted.



OFDM (Orthogonal Frequency Division Multiplexing)

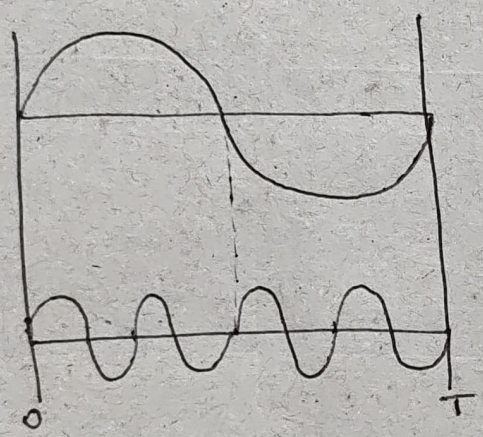
It is a modulation technique that is being used in 4G and 5G.

Consider there are N complex modulated symbols.

$$a_0, a_1, a_2, a_3, \ldots, a_{N-1}$$

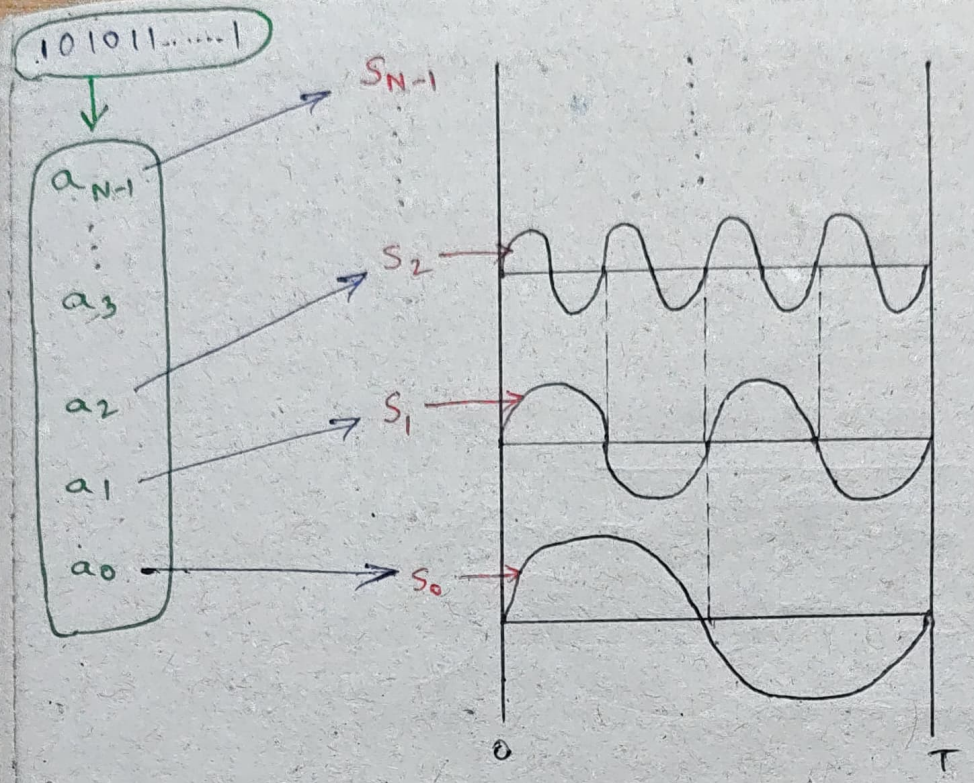
We discussed in the previous topic that, the bits are mapped to the complex valued symbols.

In OFDM, the frequencies / subcarriers are Orthogonal.



Here, these two subcarriers are Orthogonal in time 0 to T. The subcarriers need not be Sine, it could also be Cosine also.

We take similar orthogonal signals, and we modulate these signals using the complex valued symbols that we have generated from the bits.



S_0 has frequency $f_0 = \frac{1}{T}$

S_1 has frequency $f_1 = 2 f_0 = \frac{2}{T}$

S_2 has frequency $f_2 = 4 f_0 = \frac{4}{T}$

$\vdots$

S_0 is modulated with symbol a_0

S_1 is modulated with symbol a_1

S_2 is modulated with symbol a_2

$\vdots$

Now, $a_0, a_1, a_2, \ldots, a_{N-1}$ each complex valued symbol has some phase and amplitude. With this phase and amplitude, $S_0, S_1, S_2, \ldots S_{N-1}$ is modulated respectively.

Now, the modulated symbol of S_0 can be written as

$$y_0(t) = a_0 \, e^{j2\pi f_0 t}$$

The modulated symbol of S_1

$$y_1(t) = a_1 \, e^{j2\pi f_1 t}$$

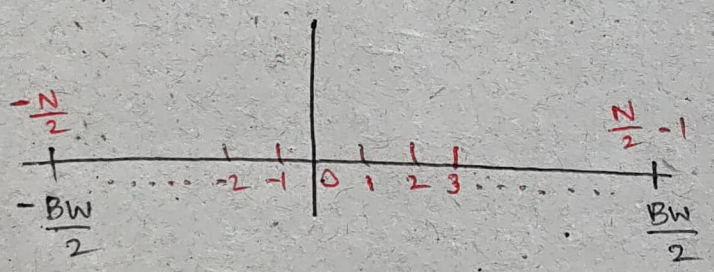
And so on...

Overall,

$$y(t) = \sum_{K=0}^{N-1} a_K \, e^{j2\pi \underbrace{\left(K - \frac{N}{2}\right)} f_0 t}$$

HOW ?

Lets consider the example below.



In baseband, we write from $-\frac{BW}{2}$ to $\frac{BW}{2}$. In this bandwidth, we take N number of subcarriers from $-\frac{N}{2}$ to $\frac{N}{2}-1$. So, we could have written the formula as

$$y(t) = \sum_{K=-\frac{N}{2}}^{\frac{N}{2}-1} a_K \, e^{j2\pi K f_0 t}$$

But, since the index of the complex valued symbol starts from 0 to N-1 (i.e) $a_0, a_1, \ldots, a_{N-1}$, it is rewritten as below, by compensating the K.

$$\boxed{y(t) = \sum_{K=0}^{N-1} a_K \, e^{j2\pi \left(K - \frac{N}{2}\right) f_0 t}}$$

where,

$N \rightarrow$ Total No. of Subcarriers

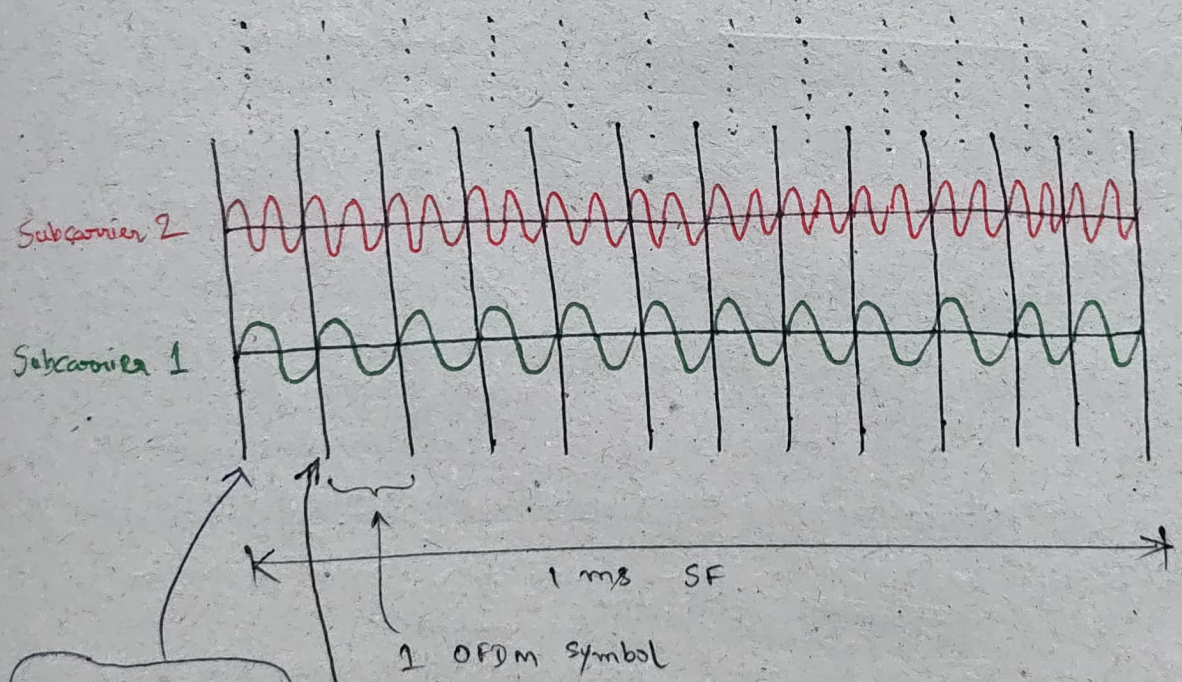
$K \rightarrow$ Subcarrier index

$a_K \rightarrow$ complex valued symbol

$f_0 \rightarrow$ Subcarrier spacing.

$t \rightarrow$ Time index.

In 5G, the refrence for time t is <u>Start of the Subframe</u>. Following figure shows 1 ms SF with many Symbols.



The Refrence for time t is Start of the SF

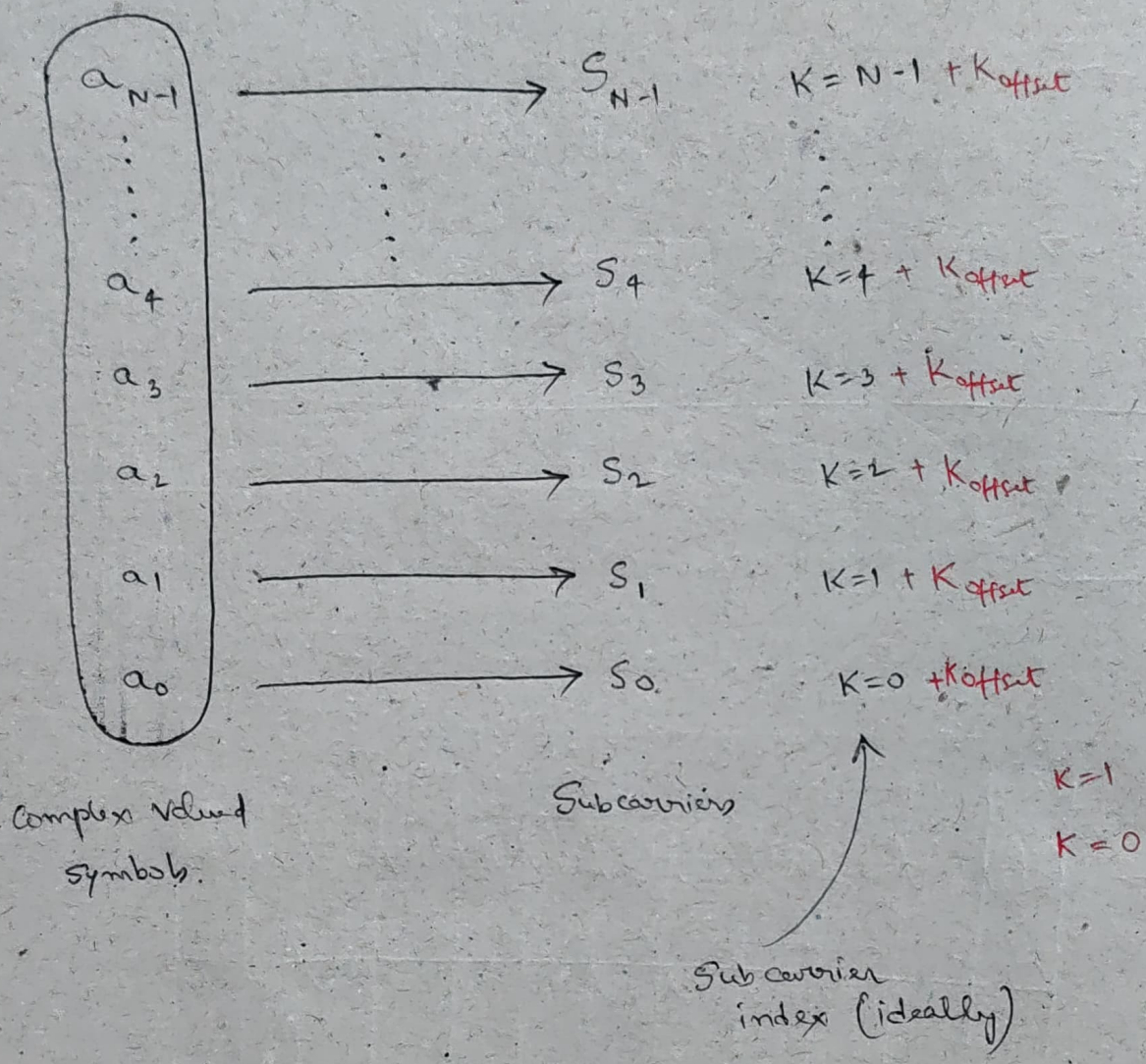
So, for each OFDM symbol after t, we need to compensate the start time.

So, for second OFDM symbol, $t = t - t_{start}$
where t_{start} is the offset from SF start.

So, we can rewrite $y(t)$ as

$$\boxed{y(t) = \sum_{K=0}^{N-1} a_K \, e^{j2\pi\left(K-\frac{N}{2}\right) f_0 \left(t - t_{start}\right)}}$$

Another thing to consider in modulation is that as follows.



The subcarrier index may not start from $K = 0$ ideally. There can be an offset in the subcarrier index as well.

So, the complex valued symbol a_0 may not be mapped to $K = 0$. It might be mapped to a different subcarrier. So to consider this offset, the formula is updated as follows.

$$\boxed{y(t) = \sum_{K=0}^{N-1} a_K \, e^{j2\pi\left(K + K_0 - \frac{N}{2}\right) f_0 \left(t - t_{start}\right)}}$$

Here, a_K is mapped to the subcarrier at index $K + K_0 - \frac{N}{2}$.

The time-continuous signal $s_l^{(p,\mu)}(t)$ on antenna port p and subcarrier spacing configuration μ for OFDM symbol $l \in \{0,1,\ldots,N_{\text{slot}}^{\text{subframe},\mu} N_{\text{symb}}^{\text{slot}} - 1\}$ in a subframe for any physical channel or signal except PRACH is defined by

$$s_l^{(p,\mu)}(t) = \sum_{k=0}^{N_{\text{grid}}^{\text{size},\mu} N_{\text{sc}}^{\text{RB}} - 1} a_{k,l}^{(p,\mu)} \cdot e^{j2\pi\left(k + k_0^{\mu} - N_{\text{grid},x}^{\text{size},\mu} N_{\text{sc}}^{\text{RB}}/2\right)\Delta f\left(t - N_{\text{CP},l}^{\mu} T_c - t_{\text{start},l}^{\mu}\right)}$$

$$k_0^{\mu} = \left(N_{\text{grid},x}^{\text{start},\mu} + N_{\text{grid},x}^{\text{size},\mu}/2\right) N_{\text{sc}}^{\text{RB}} - \left(N_{\text{grid},x}^{\text{start},\mu_0} + N_{\text{grid},x}^{\text{size},\mu_0}/2\right) N_{\text{sc}}^{\text{RB}} 2^{\mu_0 - \mu}$$

where $t_{\text{start},l}^{\mu} \le t < t_{\text{start},l}^{\mu} + \left(N_{\text{u}}^{\mu} + N_{\text{CP},l}^{\mu}\right) T_c$ is the time within the subframe,

$$N_{\text{u}}^{\mu} = 2048\kappa \cdot 2^{-\mu}$$

$$N_{\text{CP},l}^{\mu} = \begin{cases} 512\kappa \cdot 2^{-\mu} & \text{extended cyclic prefix} \\ 144\kappa \cdot 2^{-\mu} + 16\kappa & \text{normal cyclic prefix, } l = 0 \text{ or } l = 7 \cdot 2^{\mu} \\ 144\kappa \cdot 2^{-\mu} & \text{normal cyclic prefix, } l \ne 0 \text{ and } l \ne 7 \cdot 2^{\mu} \end{cases}$$

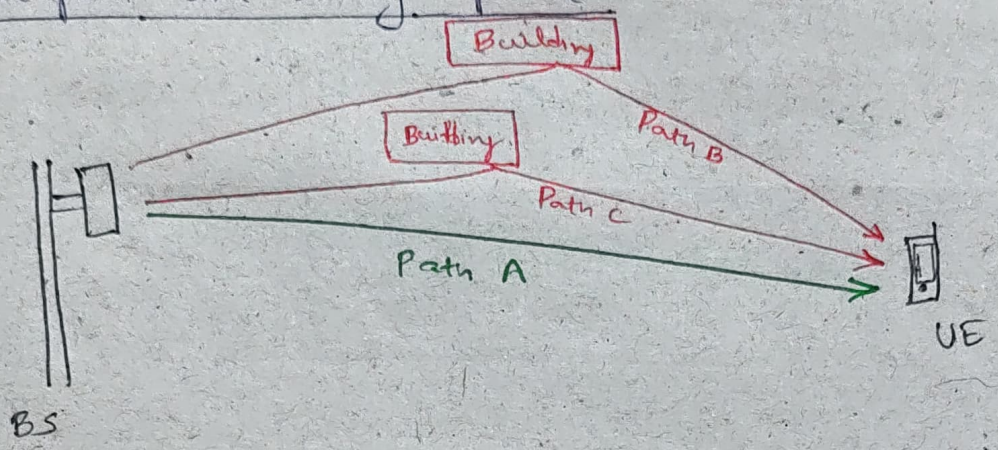
Δf is given by clause 4.2, μ is the subcarrier spacing configuration, and μ_0 is the largest μ value among the subcarrier spacing configurations provided to the UE for this carrier. The starting position of OFDM symbol l for subcarrier spacing configuration μ in a subframe is given by

$$t_{\text{start},l}^{\mu} = \begin{cases} 0 & l = 0 \\ t_{\text{start},l-1}^{\mu} + \left(N_{\text{u}}^{\mu} + N_{\text{CP},l-1}^{\mu}\right) \cdot T_c & \text{otherwise} \end{cases}$$

Cyclic Prefix (CP)

- What is CP, Why do we use it, and What are the different CP that are there in 5G.

Multipath and Delay Spread



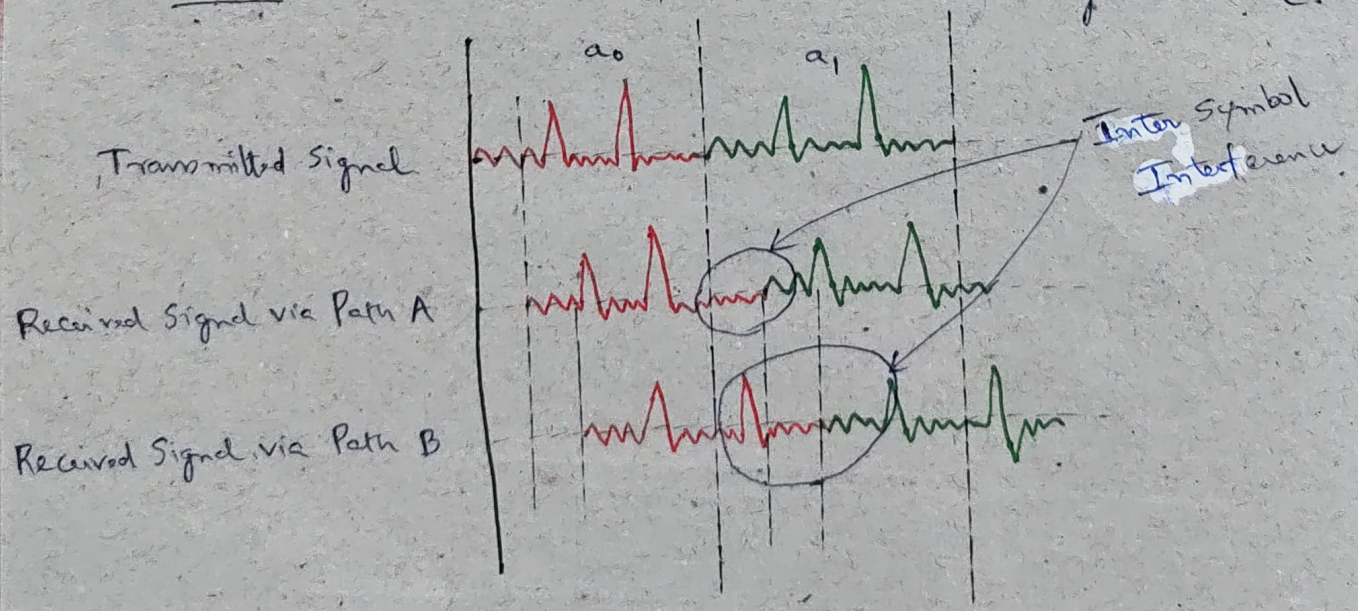
Let's assume, these three paths are the only paths, having significant power to consider.

Path A — Shortest path

Path B — Longest path

The Signal received from Path A might have the good amount of power compared to Path B and Path C.

The Signal received from Path B might have the least amount of Power, coz it travelled longer distance.



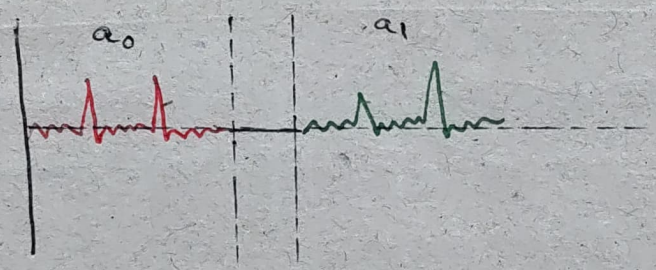
From the figure, it can be observed that, in both the received signals via Path A and Path B, the symbol carrying a_0 overlaps in the time window of symbol carrying a_1.

This is because of, two multipaths Path A and Path B, having different Travel time.

This kind of overlapping is known as Inter Symbol Interference. (ISI)

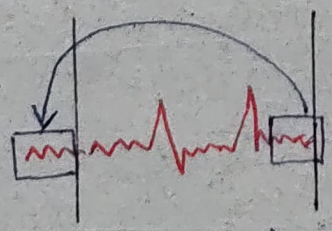
And, the difference between the first arriving path and the last arriving path is called Delay Spread.

Because of this ISI, it becomes difficult to decode the signal, as we loose orthogonality. So, to compensate this, we can delay the second OFDM symbol while transmitting.



That means, between a_0 and a_1, do not transmit anything. And this delay should be more than the delay spread of the channel, so that there wont be any overlap between a_0 and a_1 at the receiver side. So, we get rid of ISI.

But, sending zeros between a_0 and a_1 has other disadvantages. So, instead we add a portion of the signal a_0 at the end to its beginning.



This addition of last part at the beginning is known as Cyclic Prefix.

Addition of CP not only compensates ISI but also compensates Inter-Carrier-Interference (ICI). How?

At the receiver side, the transmitted signal is convolved with Channel Impulse response.

At the receiver side, the Time Domain receivers are very complicated as it has to work on each delay spread component.

So, we convert the received signal into Frequency Domain by using DFT or FFT.

Fourier Transform property		
Circular convolution in Time	$\xrightarrow{\text{FFT}}$	Multiplication in Frequency

So, in order to get the circular convolution in time, we need to make sure that this CP has longer the length of the Delay Spread.

And, this CP addition of the last part at the start, also makes sure that, in the discrete domain, it becomes circular convolution in Time domain. By taking DFT/FFT, we convert the signal into Frequency Domain, to get back the transmitted signal.

So, this is one of the biggest advantage of CP that, it not only compensates ISI, but also ICI.

Note:

If the length of the CP is less than Delay Spread, then there will be ICI.

$$N_u^{\mu} = 2048\kappa \cdot 2^{-\mu}$$

$$N_{CP,l}^{\mu} = \begin{cases} 512\kappa \cdot 2^{-\mu} & \text{extended cyclic prefix} \\ 144\kappa \cdot 2^{-\mu} + 16\kappa & \text{normal cyclic prefix, } l = 0 \text{ or } l = 7 \cdot 2^{\mu} \\ 144\kappa \cdot 2^{-\mu} & \text{normal cyclic prefix, } l \neq 0 \text{ and } l \neq 7 \cdot 2^{\mu} \end{cases}$$

$\kappa \rightarrow$ Kappa $= 64 \left(\frac{T_s}{T_c}\right)$

Assume $\mu = 1$ (ie) 30 KHz, Bandwidth = 100 MHz

(i) For Normal CP, where $l \neq 0$ and $l \neq 7 \times 2^{\mu}$

$$N_{CP,l}^{\mu} = 144\,\kappa \times 2^{-\mu} \quad \text{(in terms of } T_c\text{)}$$

$$= \frac{144 \times 64}{2^1 \times (480\text{ KHz} \times 4096)} = \underline{\underline{2.343\ \mu s}}$$

(ii) For Normal CP, where $l = 0$ or $l = 7 \times 2^{\mu}$

$$N_{CP,l}^{\mu} = 144\,\kappa \times 2^{-\mu} + 16\,\kappa$$

$$= \frac{144 \times 64}{2^1 \times (480\text{ KHz} \times 4096)} + 16\left(\frac{64}{480\text{ KHz} \times 4096}\right)$$

$$= \underline{\underline{2.864\ \mu s}}$$

Lets calculate, how much the delay spread, the signal can tolerate, given the cyclic Prefix.

The distance difference between Path A and Path B is

$2.343\mu s \times 3 \times 10^8 \approx 703$ m. | $l \neq 0$, $l \neq 7 \times 2^{\mu}$

$2.864\mu s \times 3 \times 10^8 \approx 859$ m | $l = 0$ (or) $l = 7 \times 2^{\mu}$

For different Numerology (μ), the CP time / Delay Spread is different.

For example: PTO

Assume $\mu = 3$ (ii) 120 KHz, Bandwidth = 100 MHz

(i) For Normal CP, where $l \neq 0$ and $l \neq 7 \times 2^{\mu}$

$$N^{\mu}_{CP,l} = 144 K \times 2^{-\mu}$$

$$= \frac{144 \times 64}{2^3 \times (480 \text{ KHz} \times 4096)} = \underline{\underline{0.59 \ \mu s}}$$

(ii) For Normal CP, where $l = 0$ or $l = 7 \times 2^{\mu}$

$$N^{\mu}_{CP,l} = 144 K \times 2^{-\mu} + 16 K$$

$$= \frac{144 \times 64}{2^3 \times (480 \text{ KHz} \times 4096)} + 16 \left(\frac{64}{480 \text{ KHz} \times 4096} \right)$$

$$= \underline{\underline{1.11 \ \mu s}}$$

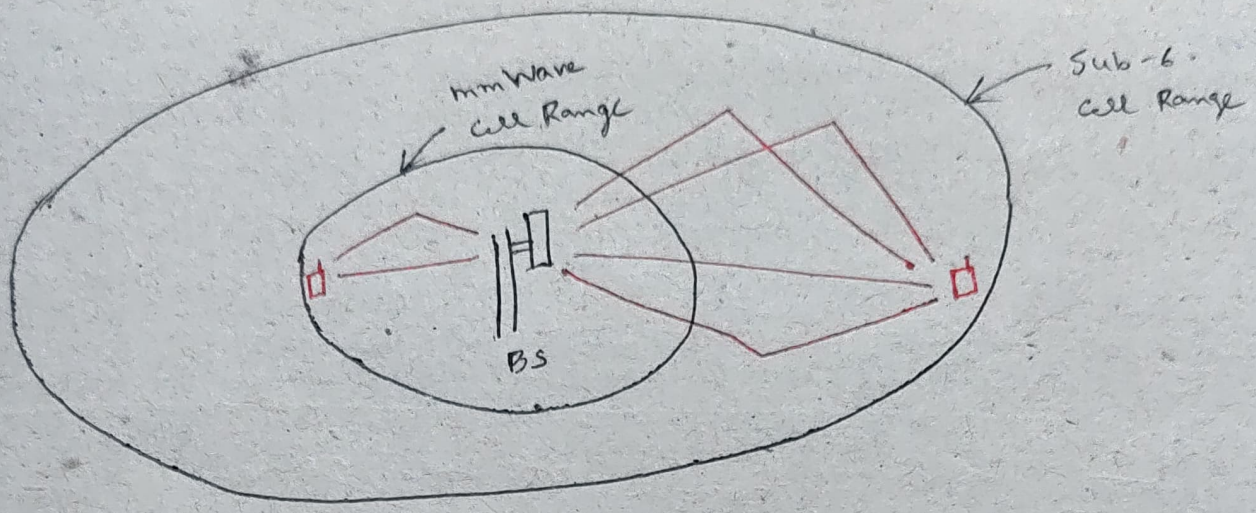
The <u>distance difference</u> between Path A and Path B is

$0.59 \ \mu s \times 3 \times 10^8 \approx 177$ m | $l \neq 0$ and $l \neq 7 \times 2^{\mu}$

$1.11 \ \mu s \times 3 \times 10^8 \approx 333$ m | $l = 0$ or $l = 7 \times 2^{\mu}$

Note that, we get lower distance difference (177 m) for 120 KHz subcarrier spacing. And higher distance difference (703 m) for 30 KHz subcarrier spacing.

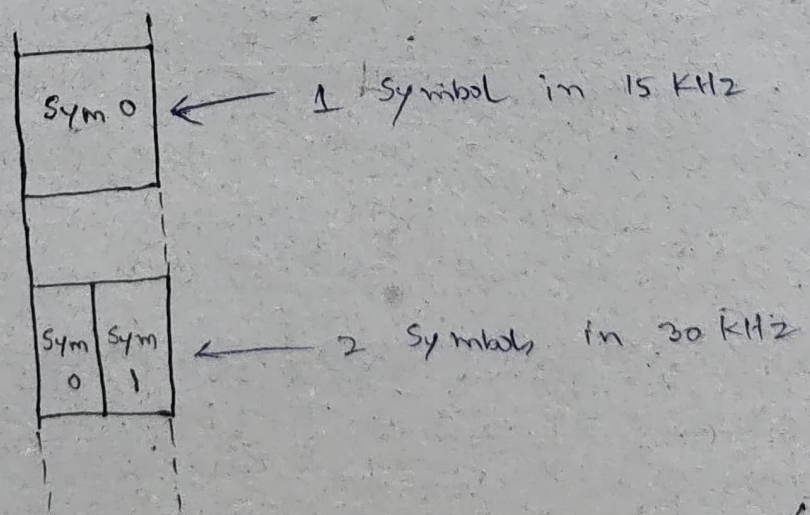
The reason is that, higher subcarrier spacings are used for higher carrier frequencies (mm Wave), where the coverage / cell range is less. Hence the Delay Spread / the distance difference is less.



We can also notice that, for some OFDM symbols the CP size is more than the other OFDM symbols. This is because

(i)
- To align the OFDM symbols within the SF boundaries.
- Also, we want to align that, in 0.5 ms, there should be integer multiple of OFDM symbols

(ii) - Two different numerologies boundaries are also aligned.



This can be possible, by adding extra samples/ extra time length, for some of the OFDM symbols.

We also have Extended CP, which have higher length.

$$N^{\mu}_{CP,l} = 512 \times 2^{-\mu}$$

In 5G, this extended CP is only supported for 60 KHz subcarrier spacing, not for all the subcarrier spacing.

Note: If we have two different CP lengths for each subcarrier spacing, then we can blindly say it as Normal CP.

$\mu = 2$ (ie) 60 KHz, Bandwidth = 100 MHz

$$N^{\mu}_{CP,l} = 512 K \times 2^{-\mu} \quad \text{(Extended CP)}$$

$$= \frac{512 \times 64}{2^2 \times (480 \text{ KHz} \times 4096)} = 4.17 \ \mu s$$

The distance difference between Path A and Path B is

$$4.17 \ \mu s \times 3 \times 10^8 \approx 1251 \text{ m}$$

The Extended CP is equivalent to 25 % overhead, where a Normal CP overhead is 7 % only.

So, By using Extended CP, we get good compensation of ISI and ICI, only at higher SNR conditions.

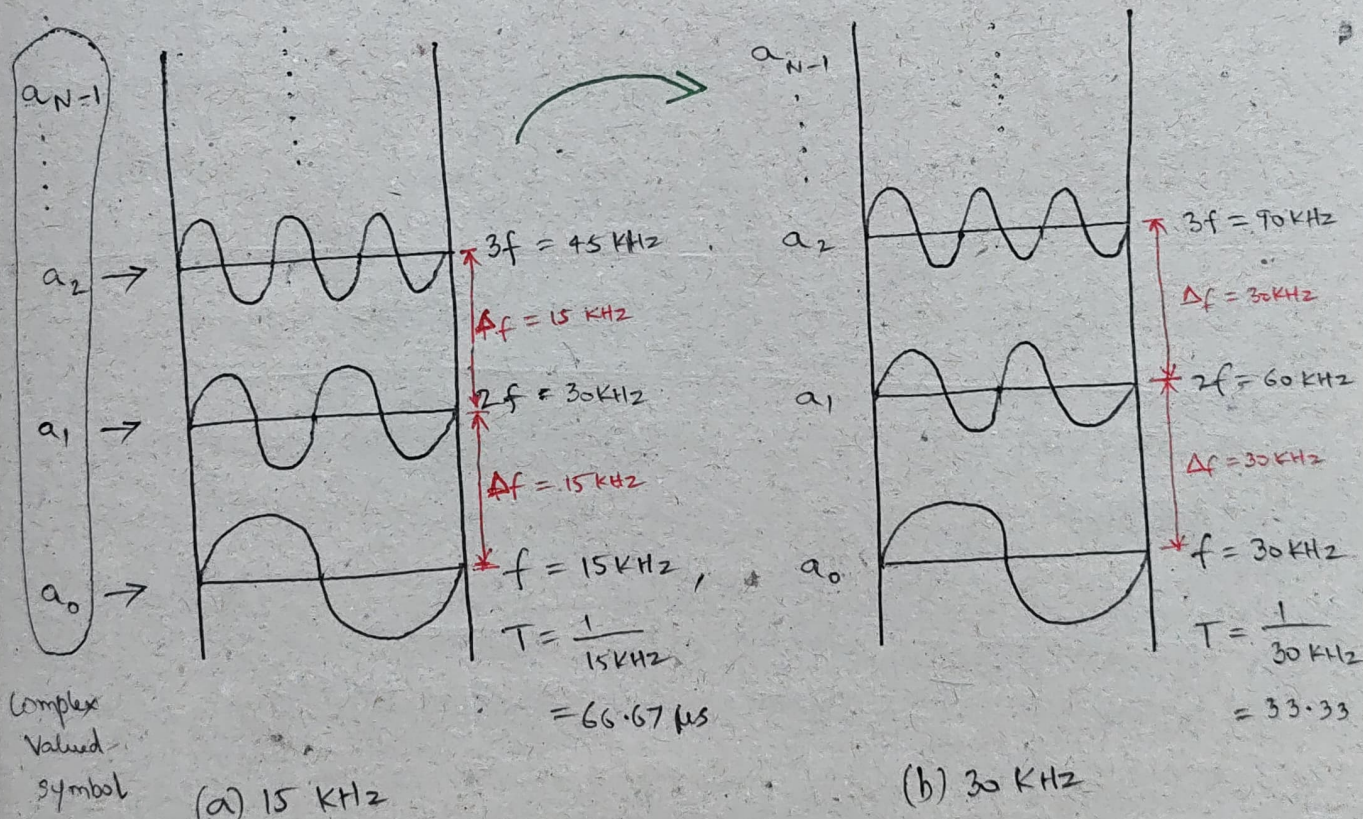
By using Extended CP at lower SNR conditions, we dont get a good advantage as we loose much in the overhead. Because of this additional overhead, even the effective bandwidth also get reduced.

This is the reason, we dont unnecessarily use Extended CP (for other subcarrier spacing than 60 KHz).

Numerology

In 5G, following are the numerologies defined.

μ	Subcarrier Spacing
0	15 KHz
1	30 KHz
2	60 KHz
3	120 KHz
4	240 KHz



If we move from 15 KHz to 30 KHz, the spacing between the subcarriers <u>increases</u> and the symbol time T <u>decreases</u> by half.

For the maximum bandwidth = 50 MHz, we use 15 KHz subcarrier spacing.

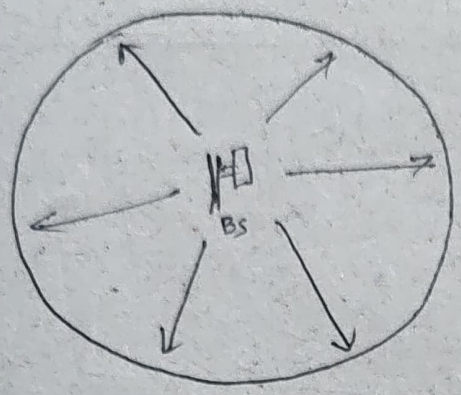
But, when we increase the maximum bandwidth = 100MHz if we continue to use 15 KHz subcarrier spacing, what happens is that, the number of subcarriers increases. And if the number of subcarriers increases, the FFT size also increases.

And since, in 5G, we limited the FFT size to 4096 (coz, beyond that, the system becomes more complicated).

So, for maximum bandwidth = 100 MHz, we do not use 15 KHz subcarrier spacing. Instead, we use 30 KHz subcarrier spacing as the minimum Subcarrier Spacing. Of course, we can use more than 30 KHz Subcarrier Spacing.

What else happens when we use different Subcarrier Spacing is follows.

- Lower SCS: 15 KHz, 30 KHz, 60 KHz, Higher SCS: 120 KHz, 240 KHz.
- Lower SCS are used for Large cell size
  Higher SCS are used for Small cell size
- If the cell radius is high, then CP is high, as the Delay Spread is higher.
  If the cell radius is low, then CP is low, as the Delay Spread is lower.



(a) Large cell

- Cyclic Prefix ↑
- Delay Spread ↑

(b) Small cell

- Cyclic Prefix ↓
- Delay Spread ↓

- For Lower SCS, Symbol duration is High
  For Higher SCS, Symbol duration is Low.

- So, when we move to higher SCS, Symbol duration reduces. So, higher SCS can be used for <u>Low latency applications</u>. (URLLC type of applications)

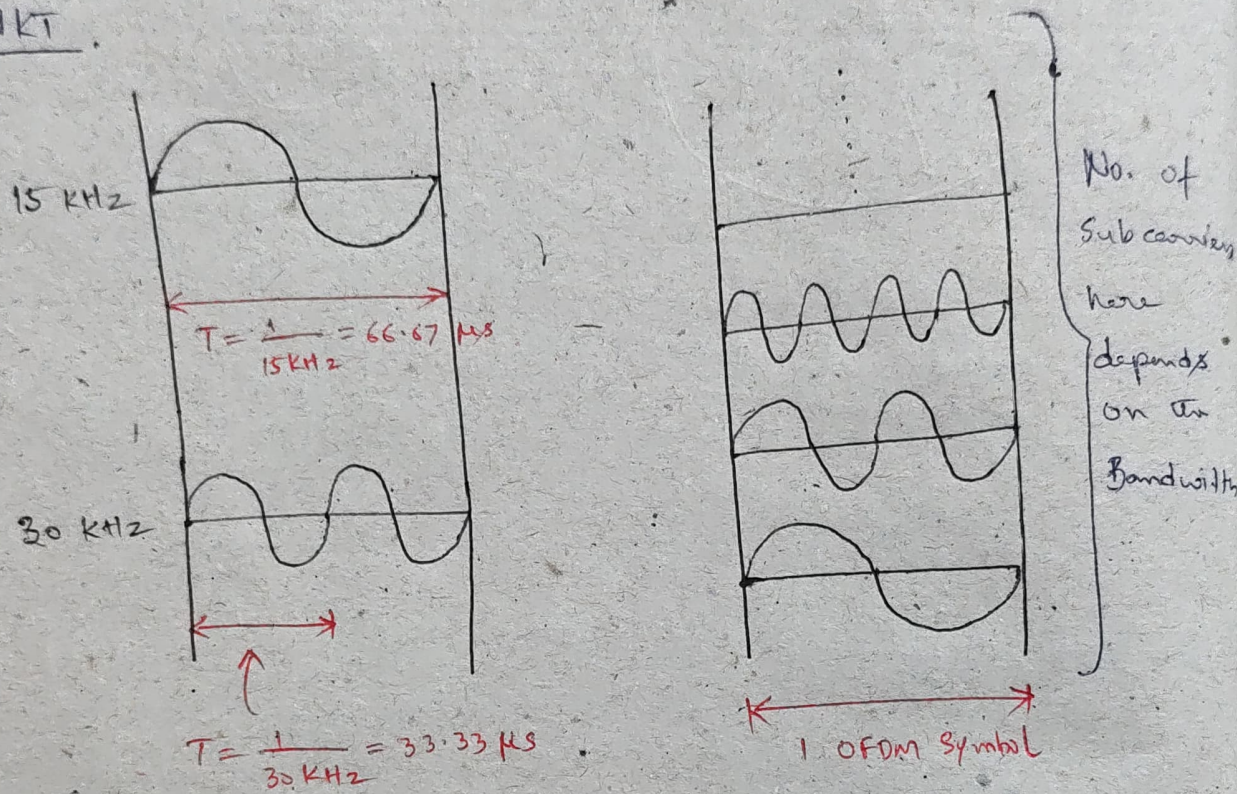
- Also, in specifications, according to Radio frequencies, Subcarrier spacings are differentiated. (ie)
  15 KHz, 30 KHz, 60 KHz → FR1 (410 MHz to 7.125 GHz)
  60 KHz, 120 KHz, 240 KHz → FR2 (24.25 GHz to 52.6 GHz)

This is how the numerologies are defined. And, the Maximum bandwidth that can be used for each numerology / Subcarrier spacing also increases as we increase the SCS, since we limit the FFT size for all SCS to 4096.

Radio Frames, Slot, Subframe and Symbols

Lets understand the time axis of 5G.

— WKT.



- Larger the Bandwidth, larger is the number of Subcarriers.

---

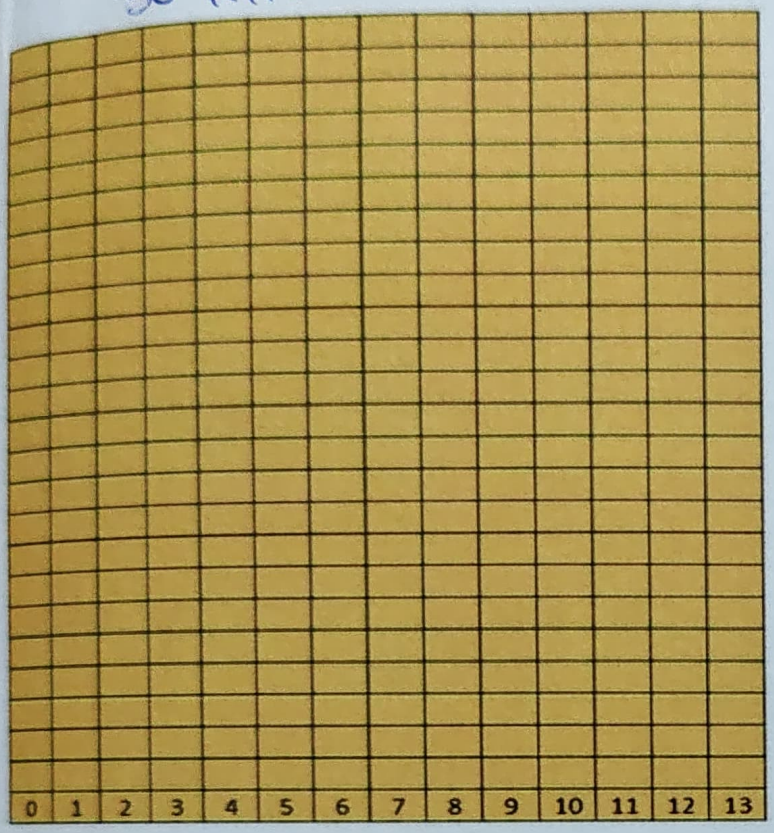
1 Slot = 14 OFDM Symbols (Normal CP) — Mainly used
12 OFDM Symbols (Extended CP) — Rarely used

- The time duration of 1 slot depends on the time duration of single OFDM Symbol + CP

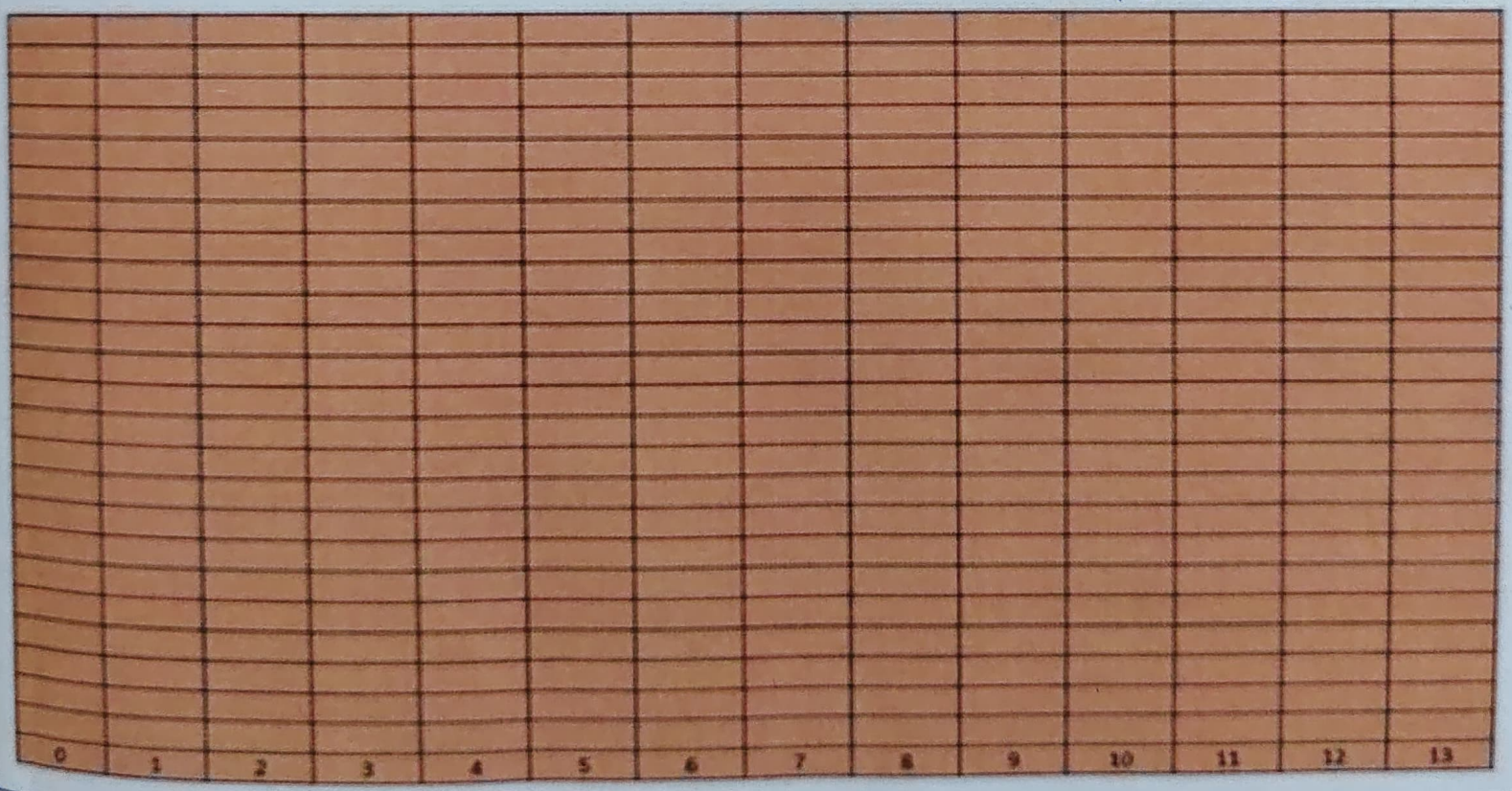
As we change the Numerology / Subcarrier spacing, the Symbol time T changes. As the Symbol time changes, the Slot duration also changes.

For 15 KHz, 1 Slot duration = 1 ms
30 KHz, 1 Slot duration = 0.5 ms

30 KHz



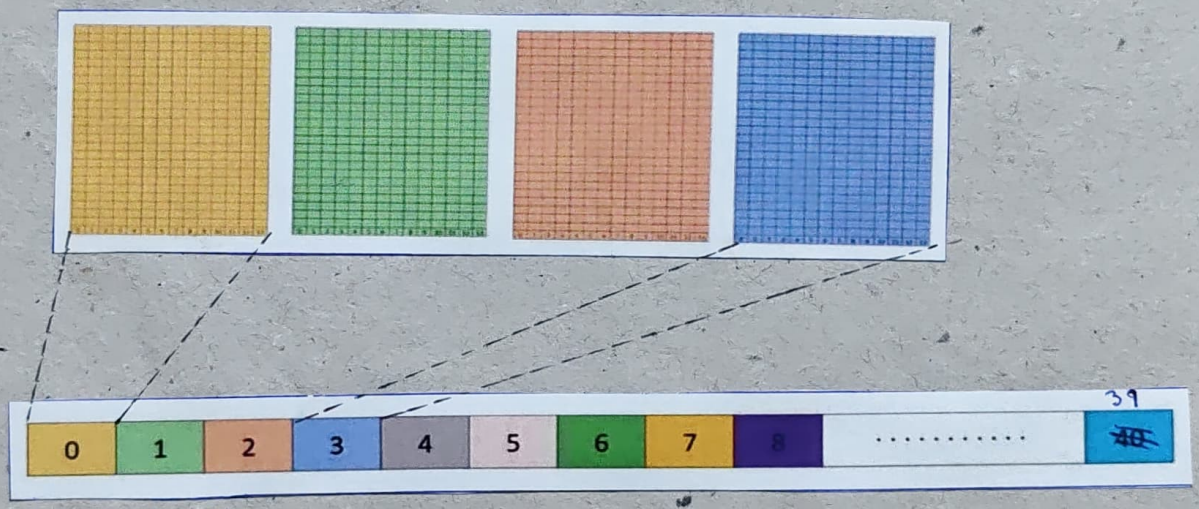
15 KHz



Inference from the figure:

- Smaller the SCS, Larger is the 1 symbol duration
  Larger the SCS, Smaller is the 1 symbol duration
- Smaller the SCS (15 KHz), Larger is the slot duration (1 ms)
  Larger the SCS (30 KHz), Smaller is the slot duration (0.5 ms)
- So, in the time of 1 slot of 15 KHz, we have 2 slots of 30 KHz.
- As per slot definition, we have 14 OFDM symbols/slot.
  Based on the numerology, the time will change.

Now, we put all these slots together, to form Radio Frame.



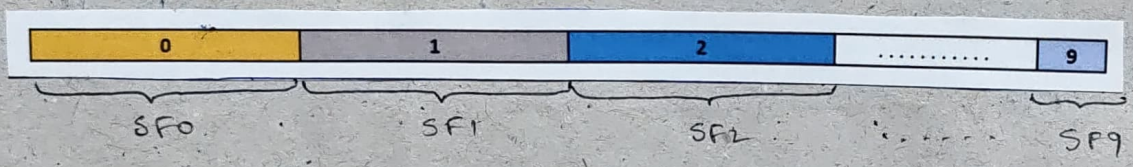
- 1 Radio Frame is always 10 ms long.
- The Number of Slots in a Radio Frame will vary. (Depends on the slot size)
- For 15 KHz, 1 slot duration = 1 ms. So 10 slots in a RF.
  For 30 KHz, 1 slot duration = 0.5 ms. So 20 slots in a RF
  For 60 KHz, 1 slot duration = 0.25 ms. So, 40 slots in a RF
  For 120 KHz, 1 slot duration = 0.125 ms. So, 80 slots in a RF
  For 240 KHz, 1 slot duration = 0.0625 ms. So 160 slots in a RF.
- The numbering of the slots is from 0 to the (total number of slots - 1).

- We can also say, the No. of Slots in a Frame is

$$\boxed{10 * 2^{\mu}}$$

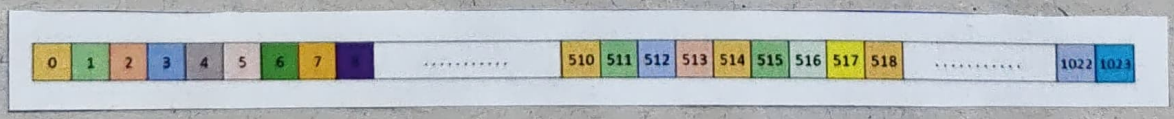
- <u>Sub Frame</u>

  - 1 SF is always 1 ms long.
    Therefore, 1 RF is always 10 SF.
  - The definition of SF is not mostly used in 5G.
    It was heavily used in 4G.
  - Most frequently used definitions in 5G are OFDM symbols, Slots and Frames (RF).
  - Most predominantly, Slot definition is used in 5G.



  - The Number of Slots in a SubFrame will vary.
    (Depends on the Slot size)
  - For 15 KHz, 1 slot duration = 1ms. So, 1 Slot in a SF.
    For 30 KHz, 1 slot duration = 0.5ms. So, 2 Slots in a SF.
    For 60 KHz, 1 Slot duration = 0.25 ms. So, 4 Slots in a SF.
    For 120 KHz, 1 Slot duration = 0.125ms. So, 8 Slots in a SF.
    For 240 KHz, 1 slot duration = 0.0625 ms. So, 16 Slots in a SF.

- <u>Radio Frame indexing</u>



  - The RF index starts from 0 to 1023. After that, the index repeats.
  - The RF index repeats every 10.24 seconds
    (∵ Each RF is of 10 ms long)

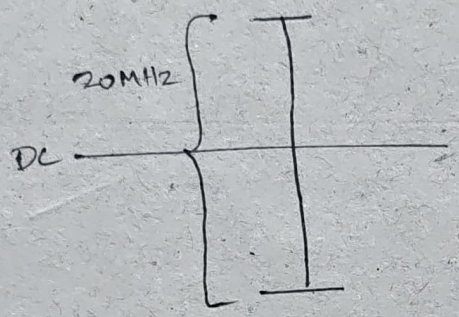
- This is the largest unit that we come across in 5G.

So, this is all about the Time dimensioning in 5G.

Resource Grid, Point A and Common Resource Block

Lets understand the Frequency axis of 5G.

As we know, in LTE, the reference was at center of the subcarriers.



In LTE, the maximum bandwidth was 20MHz. And the center (DC carrier) was used as a Reference.

In LTE, the design was very simple and it had very less flexibility.

But in 5G, we have different numerologies, and it is designed with great flexibility. For example. If the BS supports large bandwidth (say 100 MHz), the UE need not support same amount of bandwidth. UEs supporting 20 MHz bandwidth can still connect to the same BS and do the data transfer.

Now, keeping all these in mind, having the center of the Bandwidth as Reference (in 5G) doesn't make sense. So, in 5G, there is a new center defined which is called Point A. (Reference point A).

As we know, we have multiple numerologies (15 KHz, 30 KHz, 60 KHz, 120 KHz and 240 KHz) in 5G. So, having a single notion is important. Keeping that in mind, common resource blocks are defined. (CRB).

Common Resource Blocks (CRBs) are kind of <u>virtual</u>. They are used, only, to define the offset in the grid.

CRBs are defined from Point A. (i.e) the first subcarrier of CRB coincides with Point A.

For each numerology, there is a different Common Resource Block (CRB). Figure shows 15 KHz, 30 KHz and 15 KHz. CRBs are numbered from Point A as shown in figure. And The colored part is defined as <u>Resource Grid</u>.

The size of the Resource Grid is the actual bandwidth where the data transfer happens.

The offset from the Resource Grid to the Point A, is known as <u>offset to carrier</u>.

So, the Resource Grid is defined for each different numerology and for each different antenna port. (ii) In figure, for 15 KHz subcarrier spacing, there is one Resource Grid. For 30 KHz SCS, there is a different Resource Grid. And for a different antenna port, there is a different Resource Grid.

So, for each numerology, and for each antenna port, there is a different Grid. And the Grid defines, where the data transfer happens.

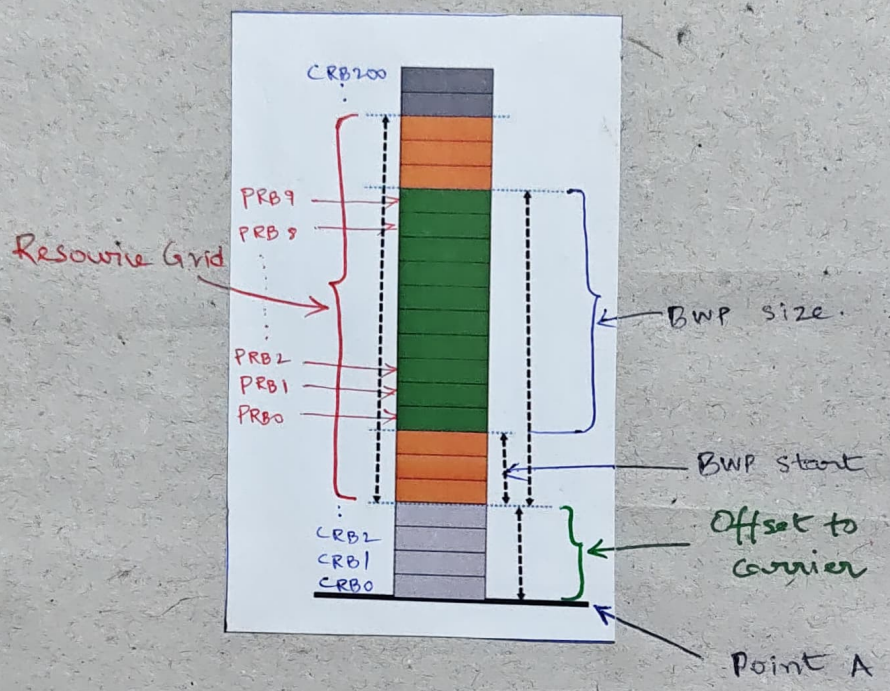
In case of SSB, (ii) the Resource Grid where SSB is transmitted, the offset from the Resource Grid to the Point A is known as offset to Point A.

Also, we can have multiple SCS transmitted at the same time, as shown in figure.

The offset to Point A for FR1 is 15 KHz. And for FR2, it is 60 KHz.

Bandwidth Part (BWP)

BWP is a very powerful concept that comes with lots of complexity, though the definition of BWP is very simple.

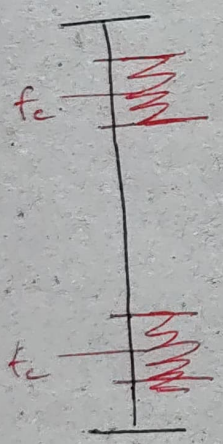


- Resource Grid can be of any Numerology.
- From Point A To start of the Resource Grid } → Offset to carrier
- From the start of Resource Grid To where the BWP starts } → BWP start
- As the name says, $\boxed{\text{BWP}}$ is the part of the bandwidth where the whole data is being transmitted or received.
- The CRB numbering is from Point A
  (i.e) CRB0, CRB1, CRB2, ... CRB 200 (say)
- The PRB numbering is inside the BWP. (i.e) from start of BWP.
  (i.e) PRB 0, PRB1, ... until the BWP size.
  This part of the bandwidth is where all the communication between the gNB and UE happens.

⊙ So, UE is not expected to receive data outside BWP, until unless UE is doing any measurement. So, UE is always decoding the data or transmitting the data inside the BWP.

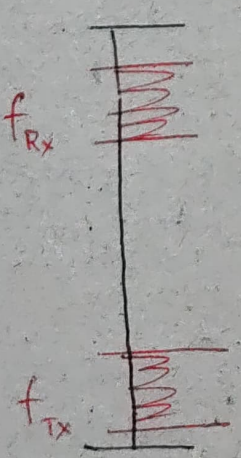
The UE may have many Bandwidth parts. In total, 4 BWPs can be configured to a UE. Out of 4, there will be only one BWP which is active. So, UE will be connected to the active one, but it can anytime switch to any of these 4 BWPs that are signalled from the gNB.

Now, this 4 BWPs configured are "per carrier and per data direction". That means,



Lets say there are 2 carriers with center frequencies as shown in figure. Now, in each carrier, UE can be configured with 4 BWPs. Inside each carrier, out of 4, only one BWP will be active.

In case of FDD / TDD, it goes in the same way as 2 carriers that we just saw above.

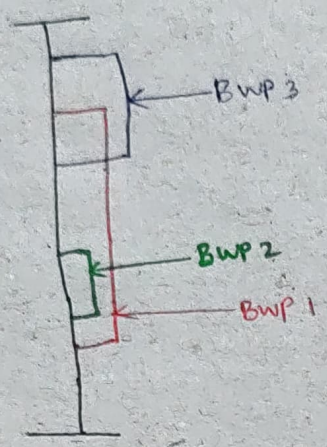


Here also, in each data direction, we can configure 4 BWPs for the UE, with one as active BWP.

In TDD also, we can have only one active BWP for UL and DL.

(a) FDD case

The BWPs can overlap. That means,

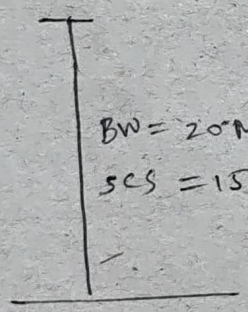


Lets say, BWP 1 has 100 PRBs
BWP 2 has 20 PRBs
BWP 3 has 40 PRBs

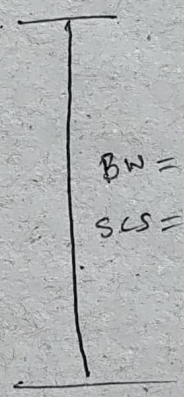
They can overlap with each other as shown in figure. And the BWPs are specific to the UE.

Motivation behind BWS Concept.

Consider the following in 4G and 5G Radio configurations.



(a) 4G



(b) 5G

"The power consumed by the UE connected to 5G, not doing any data transfer, and is only monitoring the PDCCH and decoding DCI"

is same as

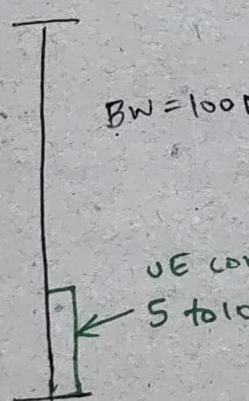
"The power consumed by the UE connected to 4G, doing data transfer at 150 Mbps".

This comparison says that, even though the UE is not doing any data transfer, as it monitors a large bandwidth and continuously monitoring the PDCCH, the radio is always ON. So, the power consumption is very high and it drains the battery really fast.

To overcome this issue, what we can do is, the UE can be allocated small bandwidth (say 15 MHz) and we change the bandwidth from time to time, from one location to another (within 100 MHz band), based on the application.

Few applications include

- Video gaming — Requires 4 PRBs
- Voice Over NR (VoNR) — Requires 4 PRBs
- Video Over NR (ViNR) — Requires 10 PRBs



For these applications, a UE might require only 5 to 10 PRBs.

If BWP concept is not being used, even though for these applications UE requires 5 to 10 PRBs, it has to monitor the entire bandwidth of 100 MHz for PDCCH, which leads to very high UE power consumption.

Whereas, if BWP concept is used, we can restrict both the above control and data operations to a smaller bandwidth. (i.e) within 5 to 10 PRBs.

This saves more than 50% of power compared to a full band case. (i.e) Without BWP, if we are able to do

video call for 1 hour, now with the reduced bandwidth and with the bandwidth switching concepts, we can do the video call for 2 hours, with the same battery (power consumption).

Figure from MediaTek white paper.

Scenario 1 : 200 MHz Wideband

Scenario 2 : 20 MHz Narrowband

Scenario 3 : 200 MHz Adaptive (Alternatively switch b/w WB and NB using BWP)

We see that, the power consumption is reduced in 20 MHz NB compared to 200 MHz WB; as we increase the data rate in the video call from 1 Mbps to 35 Mbps.

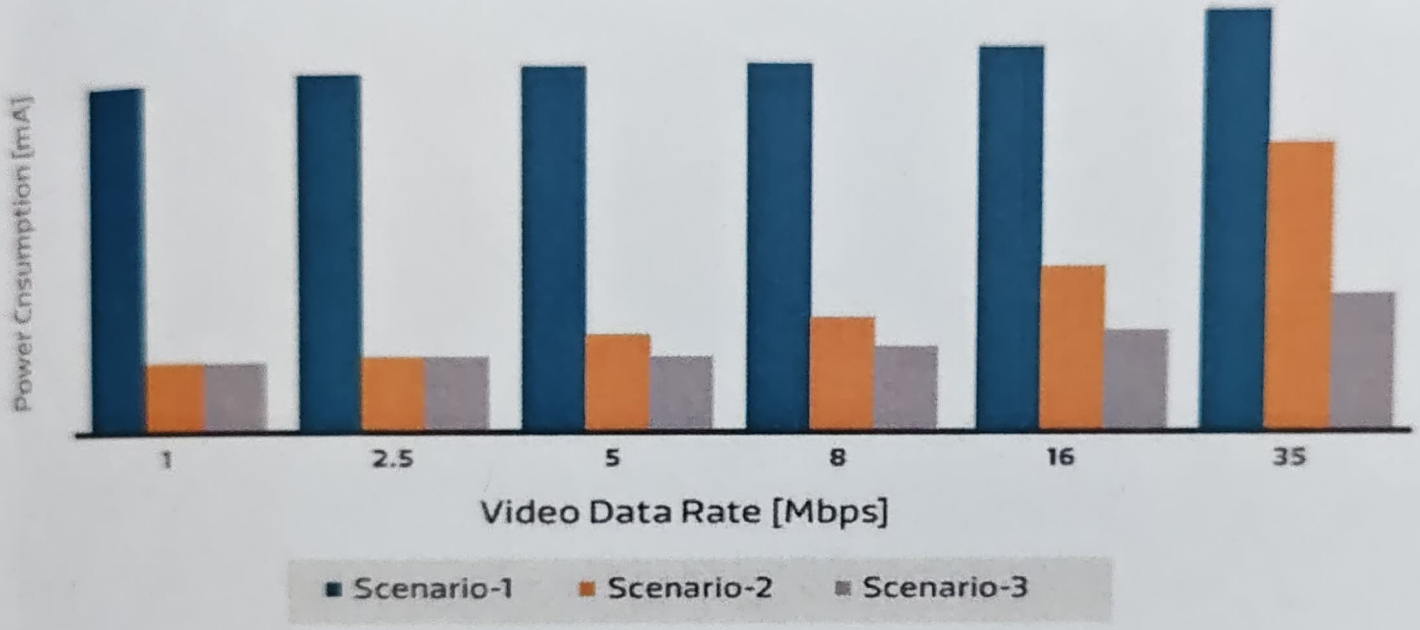
Also, in scenario 3, switching between NB and WB modes, saves even more power than NB allocation.

So, this is the major advantage of using BWP in UE side, to save the Battery life.

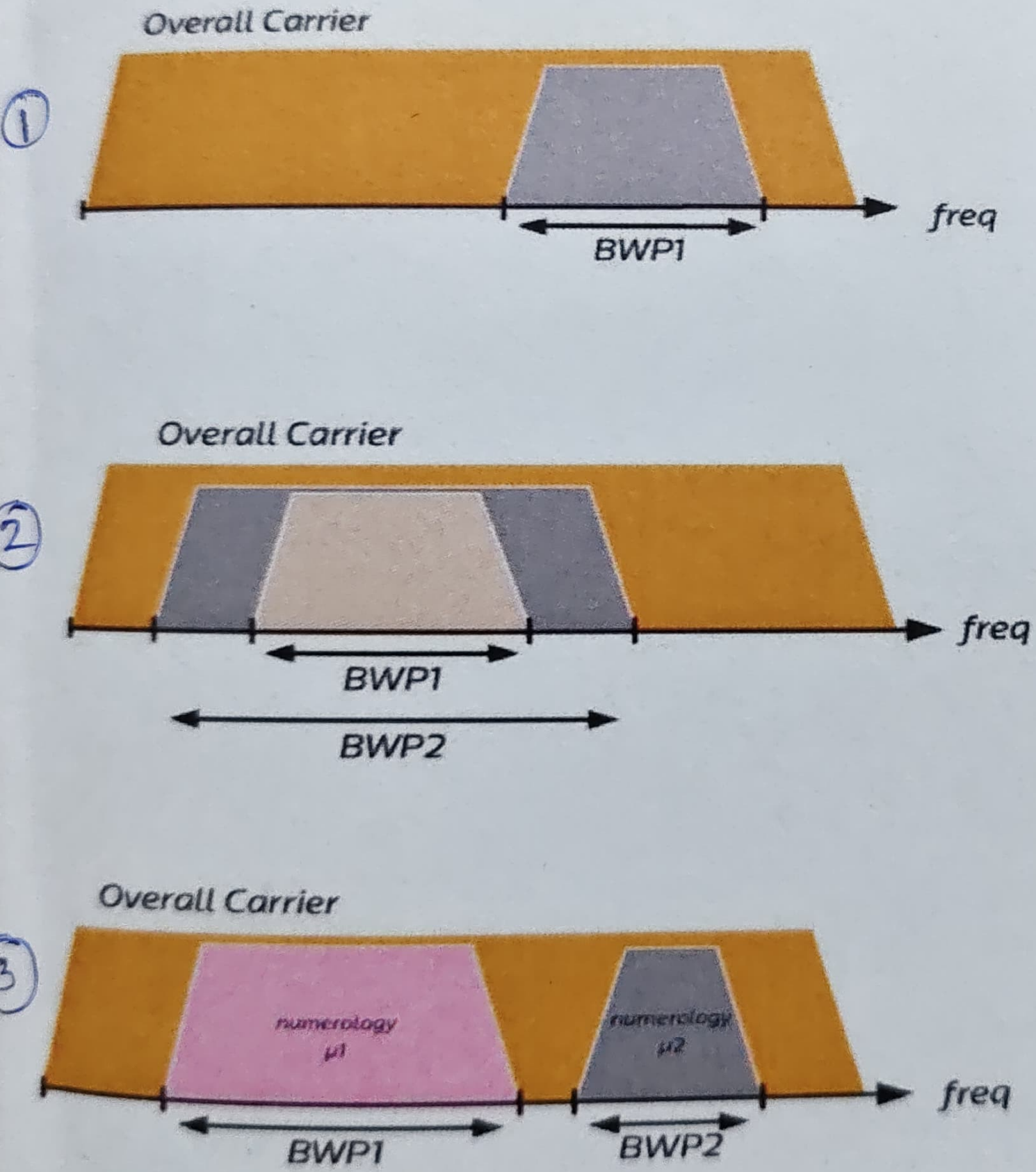
Let us see few other scenarios.

① Lets consider the overall carrier bandwidth (say 100 MHz). This is the case when the UE does not support Full carrier bandwidth. Here, the UE is capable of transmitting / receiving 50 MHz (say) (BWP1). So it operates on a reduced bandwidth.

② In this case, the UE might be capable of doing data transfer over Full Carrier bandwidth, but it doesn't have to transmit / receive on the full band. Based on the application, it can switch from one BWP to another. Here BWP2 is wide and BWP1 is shorter. And the UE switches between these two, based on the application requirement.



- Scenario-1 (200MHz WB): UE always operates in wideband mode during ON period.
- Scenario-2 (20MHz NB): UE always operates in narrowband mode during ON period.
- Scenario-3 (200MHz Adaptive): UE alternates between narrowband and wideband modes.


Overall Carrier
1
freq
BWP1
Overall Carrier
2
freq
BWP1
BWP2
Overall Carrier
3
numerology
μ1
numerology
μ2
freq
BWP1
BWP2

③ Lets say, UE has to transmit / receive burst of data over a short period of time. In that case, UE can switch from one numerology to another (say from 15 KHz to 30 KHz, do the data transfer, and then come back to 15 KHz).

So, In these ways, we can have different use cases for low latency scenarios, we can switch back and forth from a higher numerology to a lower numerology, we can have Machine Type communication (MTC), we can have a less data transfer for a particular device that does not support higher bandwidth (for device with reduced capability).

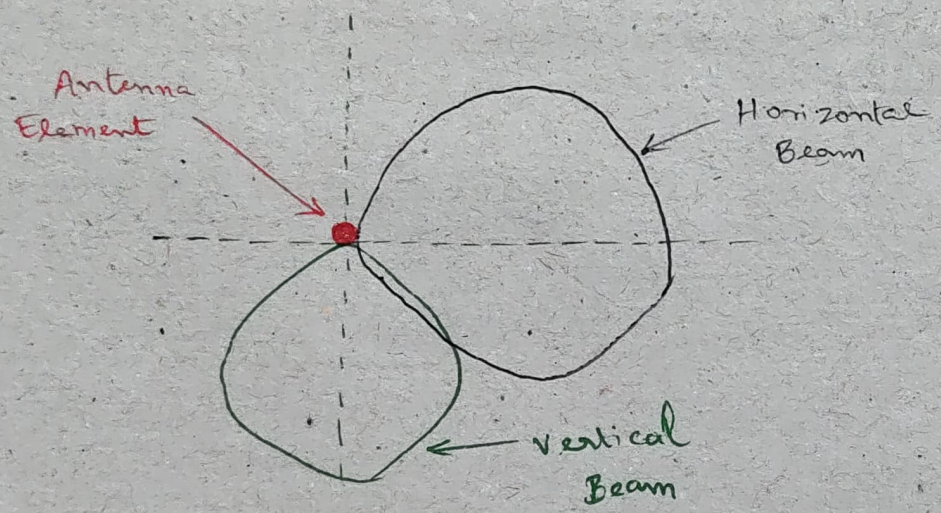
Antennas, Antenna Array and Beams

Antenna Element :

Any device that converts Electric signals to Electromagnetic waves, is an Antenna element.

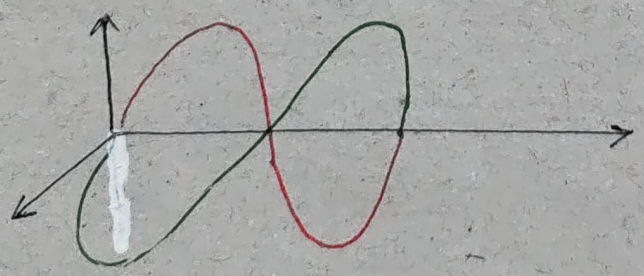
Basically something that radiates Electromagnetic waves of some configured frequency.

The Antenna Element radiates like a light bulb. So, it will radiate, in all direction, forming a three dimensional beam, both horizontally and vertically.



Antenna Polarization :

As we know, the Electric and Magnetic field are orthogonal to each other.



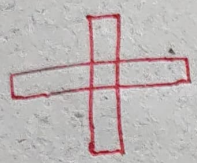
It depends on how we place the Antenna. (ie) whether the antenna is placed horizontally or vertically.

→ Vertically placed

→ Horizontally placed

In the above two cases, the electric field will be orthogonal to each other, and this property can be utilized to place two antennas together.

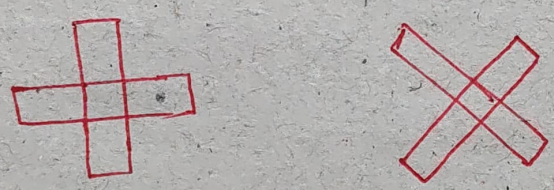
This is called Dual Polarization.



We can have two antennas placed together in a small space, which gives more gain.

There are two different ways of placing the antennas together compared to the ground level.

① ②



Ground level

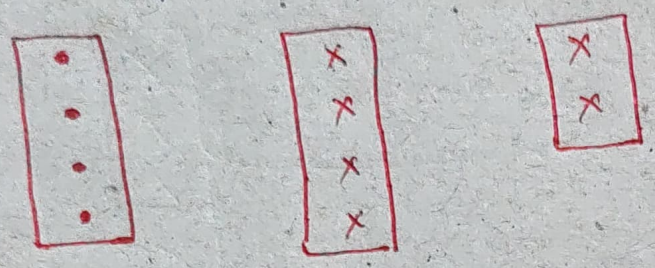
① One antenna is perpendicular to the ground level
Another antenna is parallel to the ground level

② Both the antennas are symmetrical to the ground level

Comparing ① and ②, ② gives uniform coverage and less cross-polarization interference compared to ①, as ② gives symmetrical performance. So, we always place dual polarized antennas as in ②. Also, in literatures, antennas are represented as X. (Basically there are 2 antennas, dual polarized)

Antenna Array :

Antenna Array is nothing but, a group of Antennas placed together.



What happens when 2 antennas placed together ?

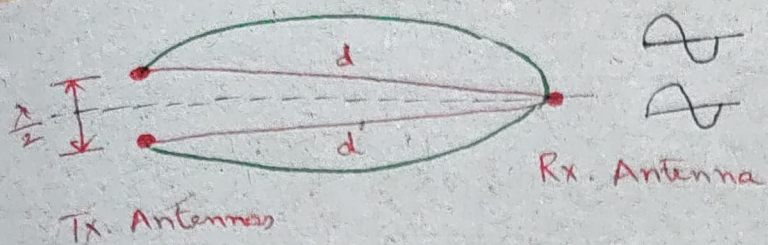
(a) Single Antenna (Wide Beam)

(b) Two Antennas (Narrow Beam)

By placing 2 antennas together, the beam becomes narrower and is more directional. So approximately, 3 decibels of directional gain will be there (beam shrinks 3 dB in the direction which antennas are placed).

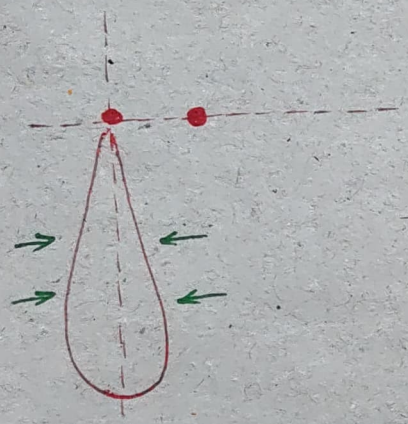
If Antennas are placed in vertical direction, the beam becomes narrower in vertical direction. This is because how the two different EM waves adds up.

Example :

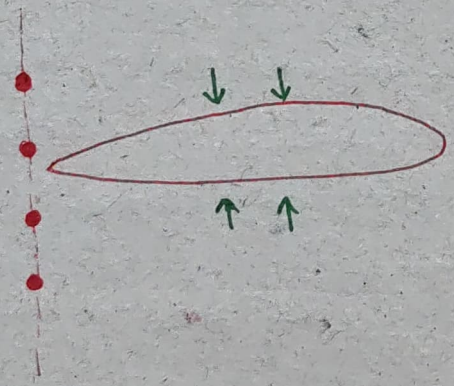


Two waves coming from Tx antennas (separated by distance $\frac{\lambda}{2}$) reaches the Rx. antenna, travels same distance d. So, they will be constructively added at the Receiver. Hence, in the vertical direction, there will be <u>more Energy distributed</u>.

However, the beam in horizontal direction is still be a wide beam. This is because, the two antennas are placed vertically.



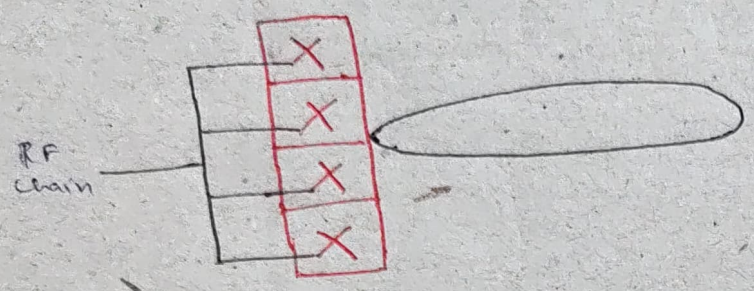
If two Antennas are placed horizontally, the beam narrower in horizontal direction. However, it is wider in vertical direction.



So, more the number of Antenna elements, the beam will become more narrower and more directional.

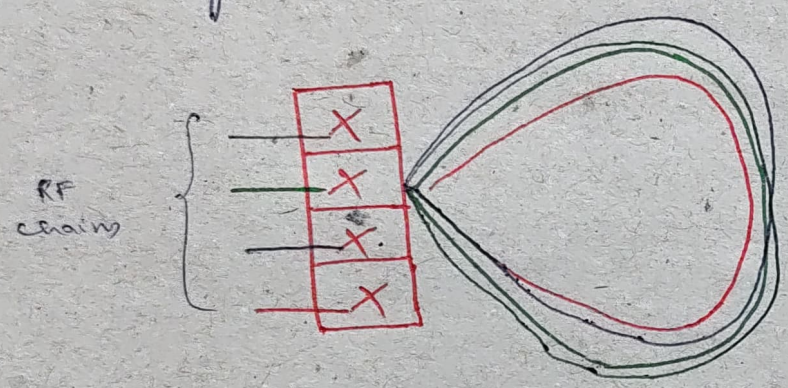
Different combinations of Antenna Elements Connection :

① Feeding same data to all Antenna Elements.



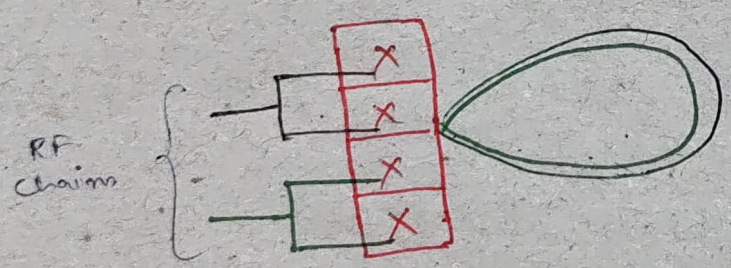
Here, the beam will be more narrower and more directional.

② Feeding different data to each Antenna Element.



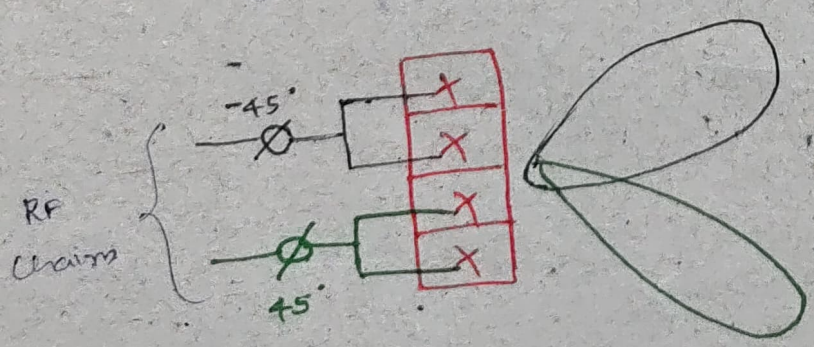
Here, the beam from each Antenna element would be wider.

③ Feeding same data to fewer Antenna Elements.



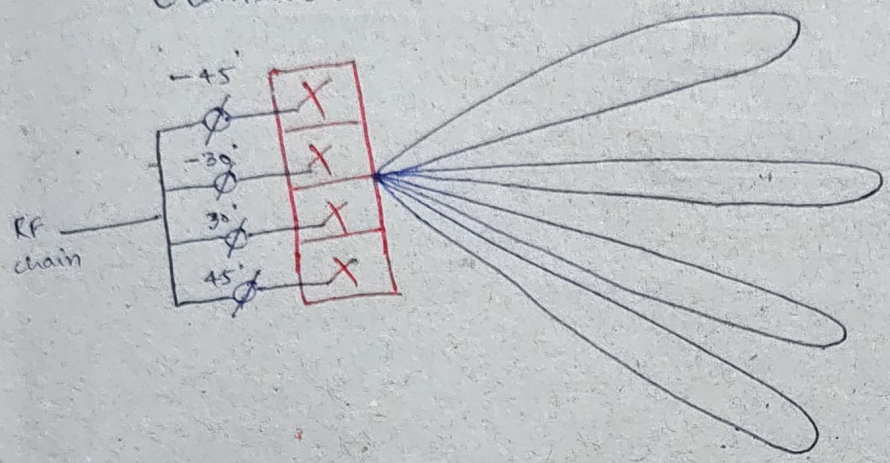
Here, the beam is neither too narrow nor wide. Somewhere in the mid range.

④ Feeding phase shifted data to group of Antenna Elements.



Here, based on the phase shift, we can direct each beam, covering the desired area.

⑤ Feeding same data (but phase shifted) to all Antenna Elements.



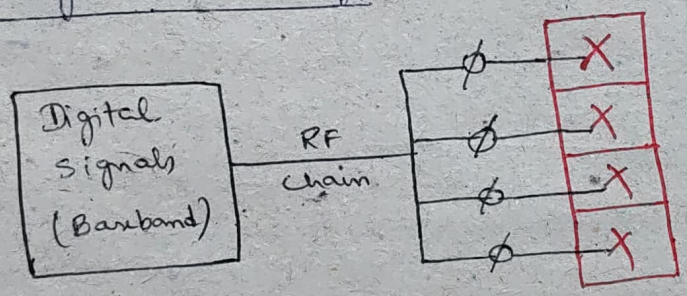
The way of giving feedback is of 3 types.

(i) Digital Beamforming

(ii) Analog Beamforming

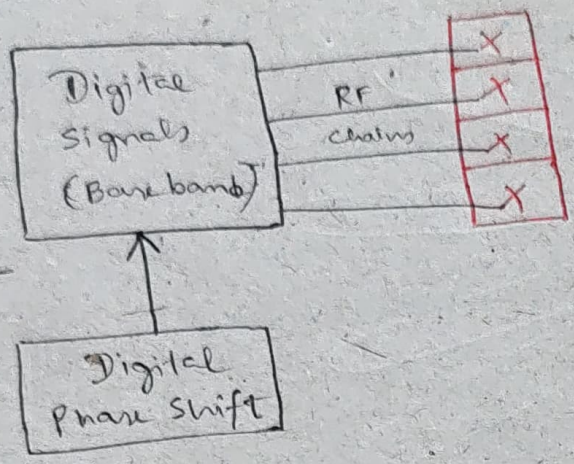
(iii) Hybrid

(i) Analog Beamforming



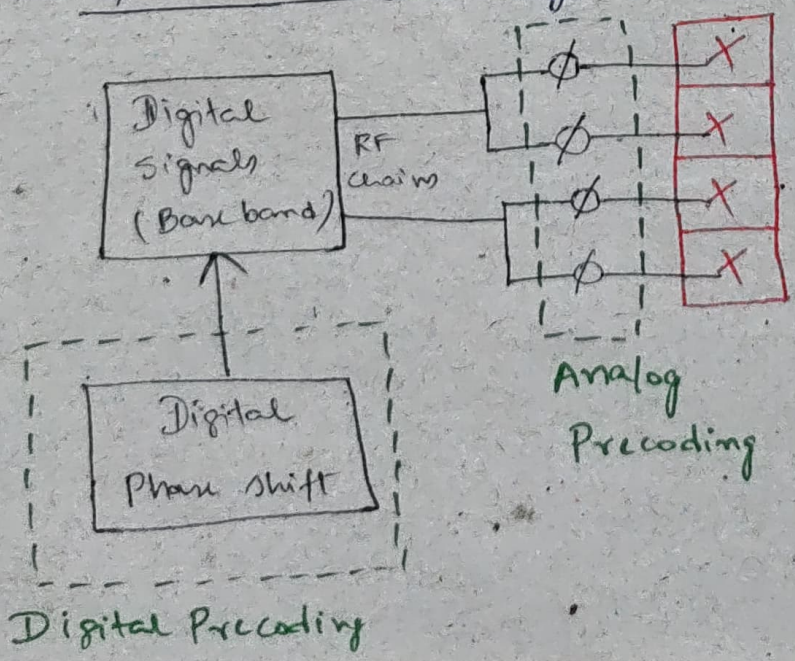
- Here there are less number of RF chains
- Low cost Solution
- We cannot have more number of data streams, each going to different Antennas.
- So, we can't steer the beam however we want.

(ii) Digital Beam-forming

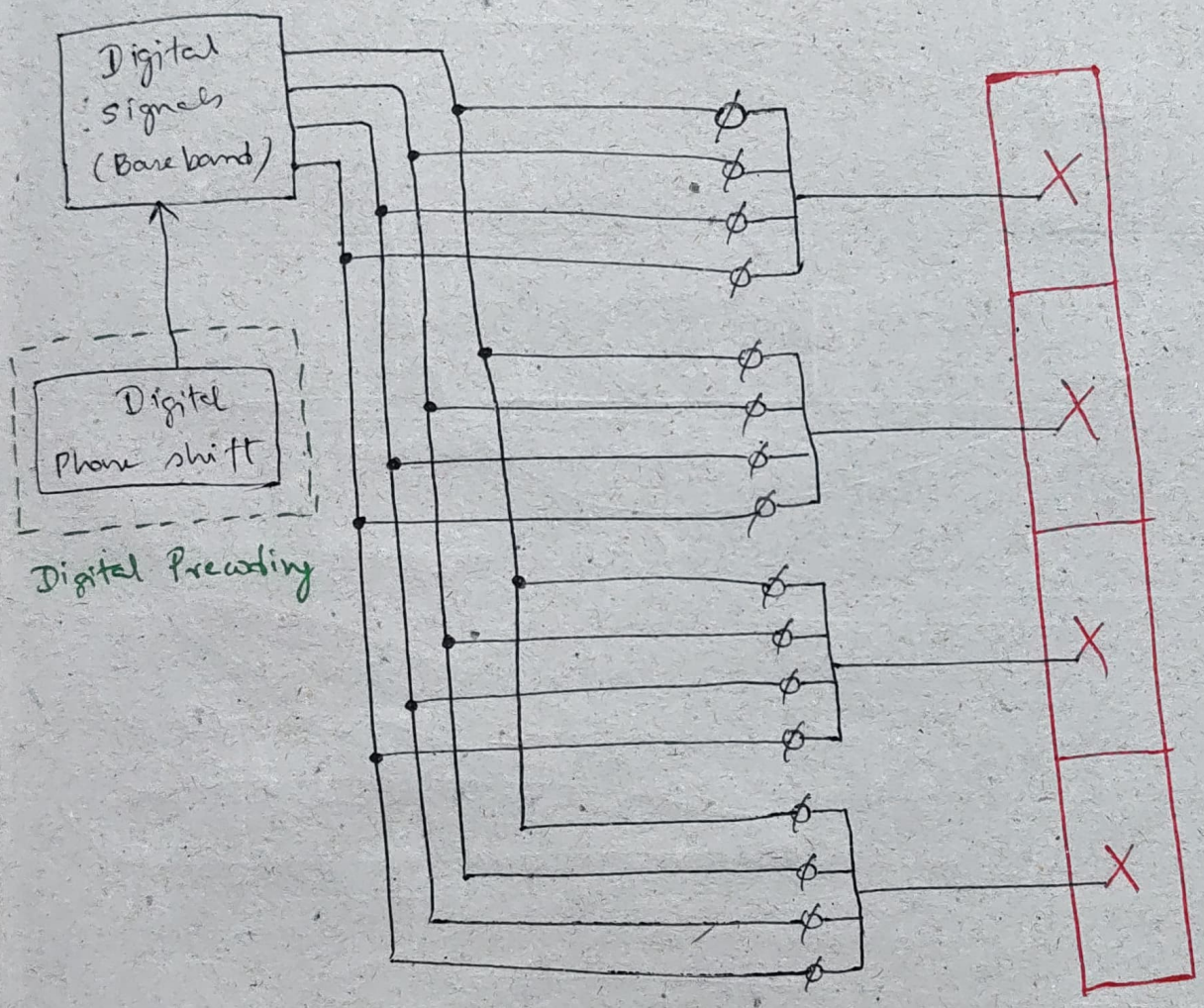


- Here there are multiple RF chains
- The phase shift is given digitally
- This way, we can independently control each antenna element.
- We can have desired direction
- We can have multiple streams
- As there are multiple RF chains, this is a costly solution. Think of mmWave, where we have 64 or 256 antenna elements, and Having a dedicated RF chain for each of them is a very costly solution.

(iii) Hybrid Beam-forming.



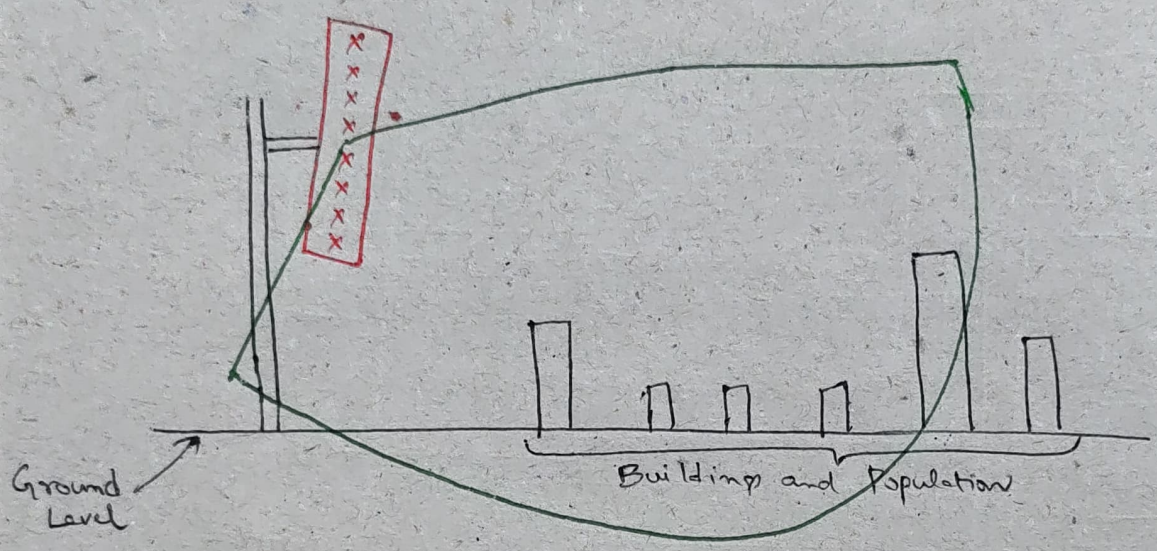
- Here, there are multiple RF chains.
- The phase shift is controlled both in Analog as well as digital.
- A complicated way, that we also have practically as well is shown below.



- Here, each RF chain is connected to each of the Antenna elements.

Figure shows the practical 5G Antennas that are installed in the field. It contains 8 RF ports. Hence termed as 8T8R. Solution.

Why the antenna elements placed vertically ?



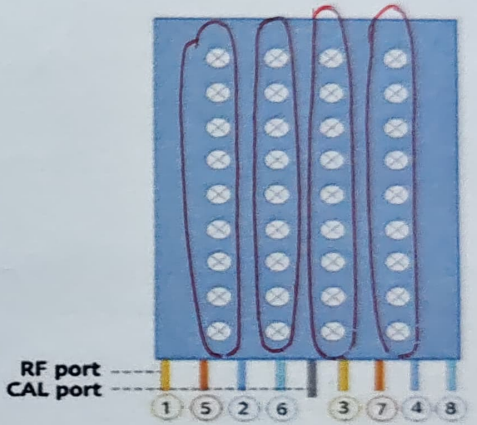
The Antenna elements are tilted downside, so that the orientation is towards ground side (not upwards)

We mostly want the Antennas to give coverage lets say 120°. But vertically, we mostly want it to be directional, pointing towards the ground. We dont want the antenna to radiate on top.

Antenna Elements Arrangement :

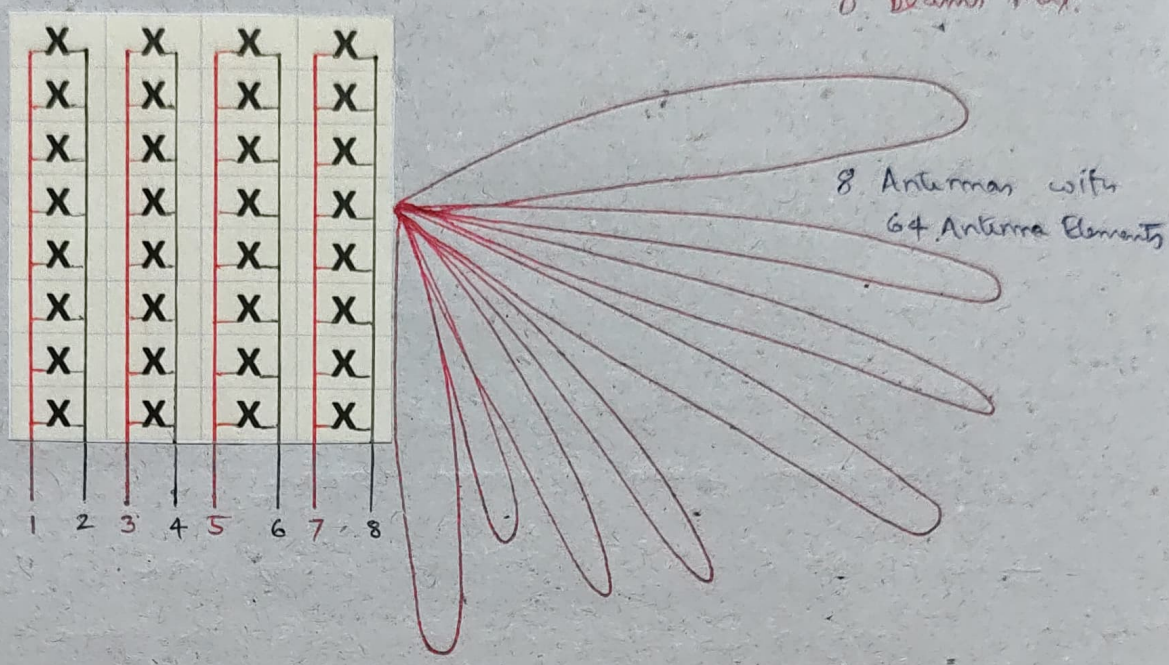
X	X	X	X
X	X	X	X
X	X	X	X
X	X	X	X
X	X	X	X
X	X	X	X
X	X	X	X
X	X	X	X

8 (Rows) x 4 (columns) x 2 (Dual polarized) ⟹ 64 Antenna Elements.


RF port
CAL port
1
5
2
6
3
7
4
8

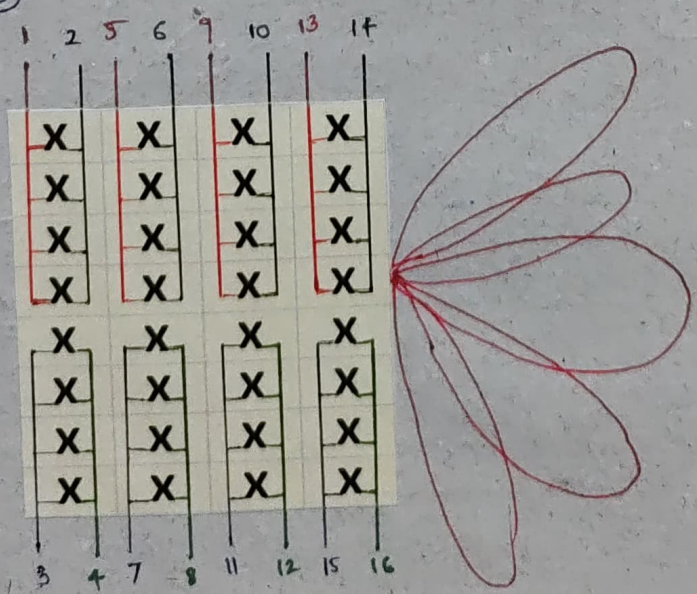
Case ①

8 beams Max.



- Here, each column is connected with different inputs.
- This kind of arrangement is called 8T8R System
- Since, the Antennas are arranged vertically, it will have vertical narrow beams.
- And since there are 8 vertical Antennas, so there'll be 8 vertical beams.
- Based on the phase shift / Analog precoding / Digital precoding, the direction of the beam changes.
- All the 8 vertical beams are horizontally wide.

Case ②

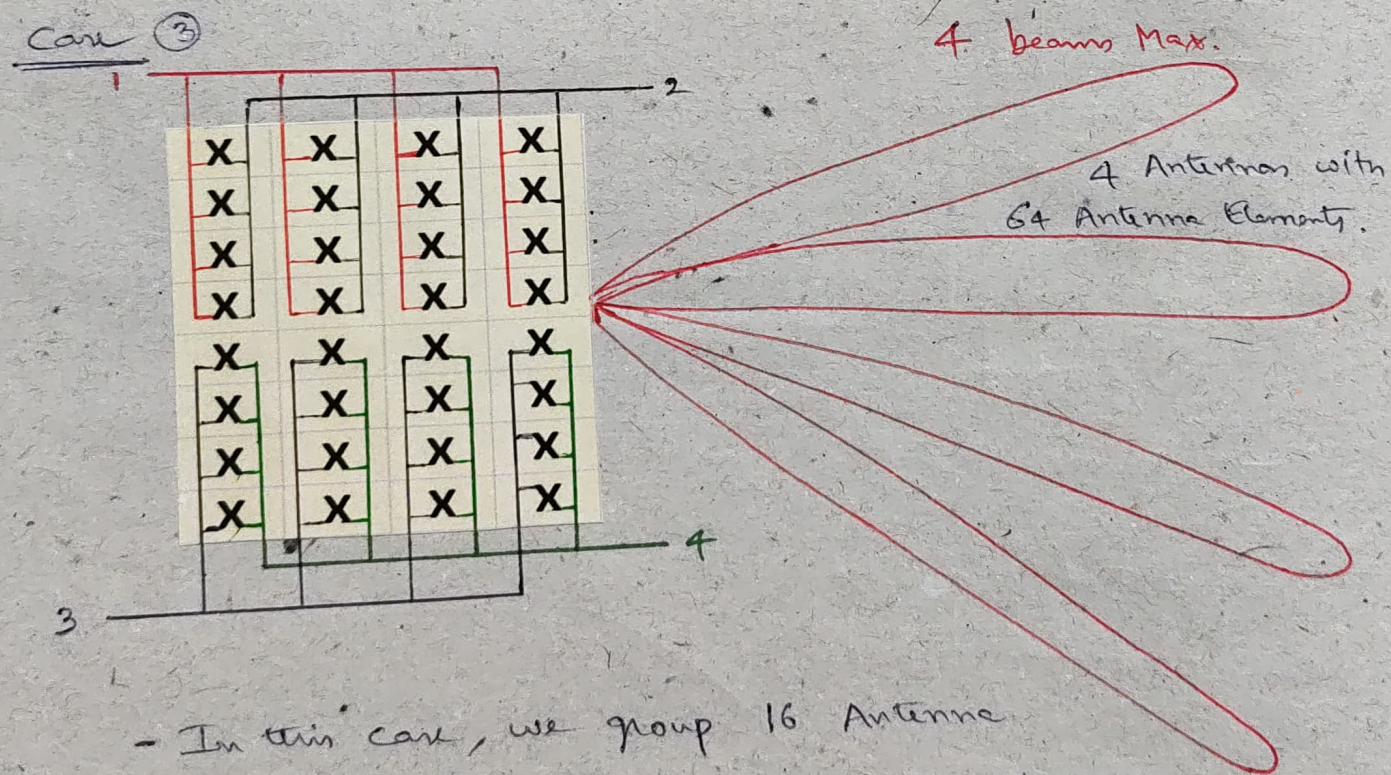


16 beams Max

16 Antennas with
64 Antenna Elements

- In this case, we group 4 Antennas Elements
- This kind of arrangement is called 16T16R system.
- Since, the Antennas are placed vertically, it will have comparatively wider beams than case ①.
- And since there are 16 vertical Antennas, so there'll be 16 vertical beams.
- And the direction of the beams will depend on the phase given to each input.
- And still the 16 vertical beams are horizontally wider. say, all the beams will cover the same 120° horizontally.

Case ③



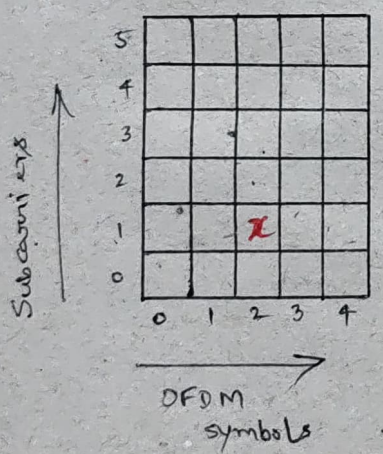
- In this case, we group 16 Antenna Elements from 4 Rows and 4 columns.
- This kind of arrangement is called 4T4R system.
- Since, the Antennas are placed both vertically and horizontally, the beam will be narrower both horizontally and vertically.
- In this way, we can have beams focused on particular UEs at different locations.

DMRS, Channel Estimation and Equalization

- DMRS → Demodulation Reference Signal
- DMRS do not carry any useful information.
- These signals are used to estimate the channel at the receiver side.
- Most of the Physical layer channels carries DMRS.

PBCH	PUSCH
PDCCH	PUCCH
PDSCH	

- These DMRS signals are basically generated either using
  - M-sequence (or)
  - Zad-off chu sequence.



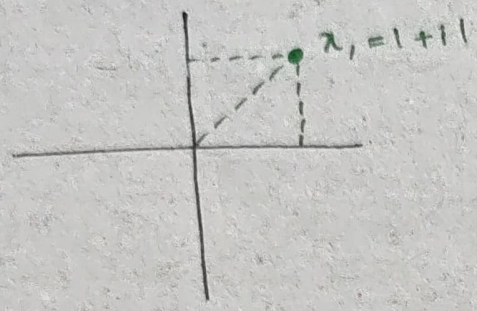
Consider, signal x was transmitted at OFDM symbol 2 and subcarrier 1.

Once signal x is transmitted, it will go through many reflections and multipaths.

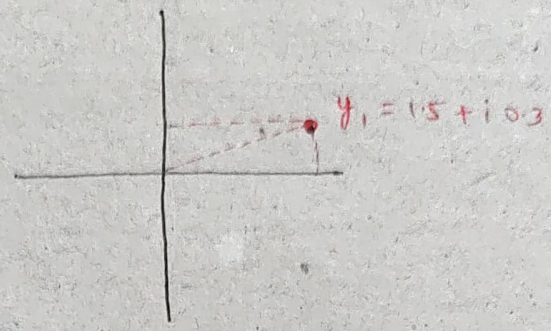
Hence, the received signal y is a combination of all the signals received over these multipaths with reflections. So the received signal won't be exactly similar to the transmitted signal.

(ie) If the transmitted signal has some phase and amplitude, the received signal may have different phase and amplitude.

Example:



(i) Transmitted Signal



(ii) Received Signal

Consider Signal $x_1 = 1 + i1$ is transmitted.

- Amplitude $= \sqrt{1^2 + 1^2} = \sqrt{2} = 1.414$
- Phase $= \tan^{-1}(1) = 45^\circ$

And Signal $y_1 = 1.5 + i\,0.3$ is received

- Amplitude $= \sqrt{1.5^2 + 0.3^2} = 1.53$
- Phase $= \tan^{-1}\left(\frac{0.3}{1.5}\right) = 11.3^\circ$

Hence, the received signal is changed, by

Amplitude $\Rightarrow \frac{1.53}{1.414} = 1.082$

Phase $\Rightarrow 45^\circ - 11.3^\circ = 33.7^\circ$

Change in Amplitude	$= 1.082$
change in Phase	$= 33.7^\circ$

We cannot recover the transmitted signal x_1 from the received signal y_1, until unless we find the changes as above. So, in order to estimate the channel, the receiver needs to know what exactly was transmitted.

If we dont know what was transmitted, we cannot find out how much change was done due to channel.

To estimate the channel (Phase and Amplitude change), the transmitter will have to transmit some additional information which is pre-known at the Receiver. (ie) Both x_1 and y_1 are known at the receiver, so that the receiver can find out the phase and Amplitude change.

The Signals that are pre-known at the Reciver and Transmitter, to estimate the channel, are known as DMRS.

Each Physical Layer channel have their DMRS signals, which are pre-known at the receiver.

One way of estimating the channel is follows.

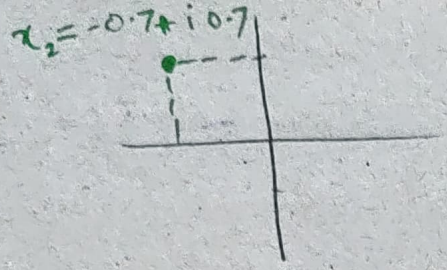
$$H = \frac{y_1}{x_1} = \frac{1.5 + i\,0.3}{1 + i1} = 0.9 - 0.6i$$

$$\boxed{\begin{aligned} \text{Amplitude of } H &= \sqrt{(0.9)^2 + (-0.6)^2} = 1.082 \\ \text{Phase of } H &= \tan^{-1}\left(\frac{0.6}{0.9}\right) = 33.7^\circ \end{aligned}}$$

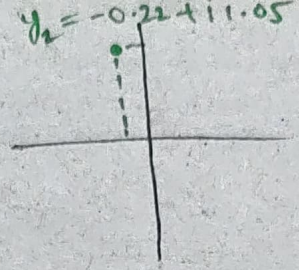
This is known as channel estimation, and is important in every physical layer receiver design. There are many algorithms to estimate the channel.

<u>DMRS Estimation Example</u>:

Consider, the DMRS signal $x_2 = -0.7 + i\,0.7$ is transmitted. Assume, this DMRS signal experiences the channel $H = 0.9 - 0.6i$. The received signal is $y_2 = -0.22 + i\,1.05$.



(i) Transmitted Signal



(ii) Received Signal

Now, at the Receiver side, with the knowledge of H, x_2 can be easily calculated.

$$\hat{x}_2 = \frac{y_2}{H} = \frac{-0.22 + i\,1.05}{0.9 - i\,0.6} = -0.71 + i\,0.69$$

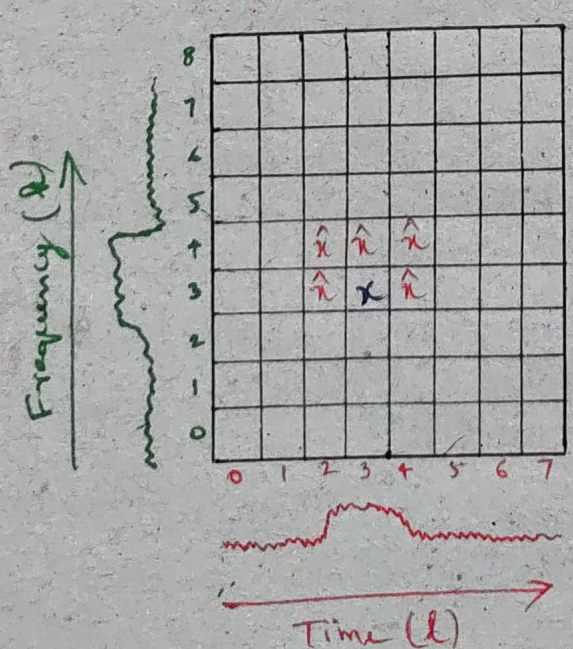
where,

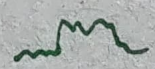
$$x_2 \doteq -0.7 + i\,0.7.$$

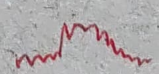
<u>Note</u>: $\hat{x}_2$ may not be exactly same as x_2, as the channel is impacting the signal x_2.

So, we may not correctly estimate DMRS signal, but a very close estimation.

Placement of DMRS signal in Resource Grid.




- Channel Variation across frequency


- Channel variation across time

- More the Multipath, more the channel variation in frequency.
- Faster we move (train, car), faster would be the channel variation in time. (or) UE is stationary, and the environment changes faster, here also the channel varies in time.
- Assume, DMRS signal x is transmitted at (3,3). But, the estimation could point out at (3,2) or (3,4) or (4,2) or (4,3) or (4,4), due to the channel variations in time and frequency. So, we need to transmit DMRS in few other places as well.
- So, the placement of DMRS signal depends on how fast the channel is changing, across the frequency and time.
- The DMRS signals are placed differently at different channels, especially in data channels (PDSCH & PUSCH).