COMMNET

PS-2

Quasi 5G NR PHY layer modelling

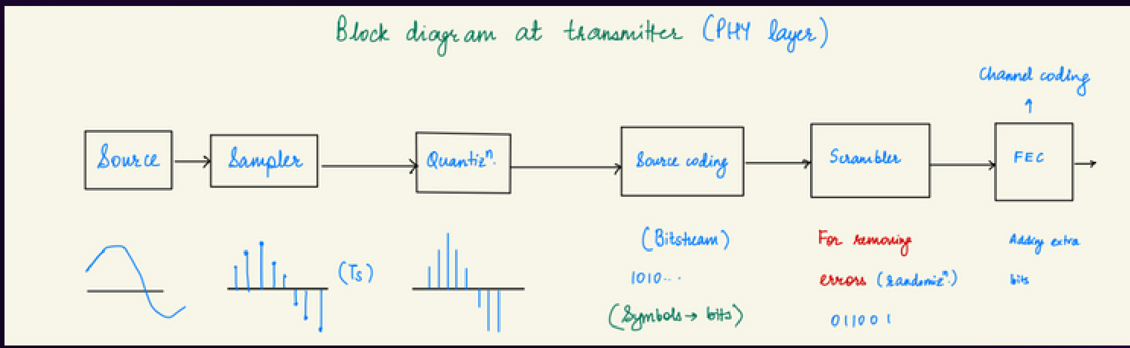
5G is the next major phase of mobile telecommunications standards. The scope of 5G will ultimately range from mobile broadband services to next-generation automobiles and connected devices.

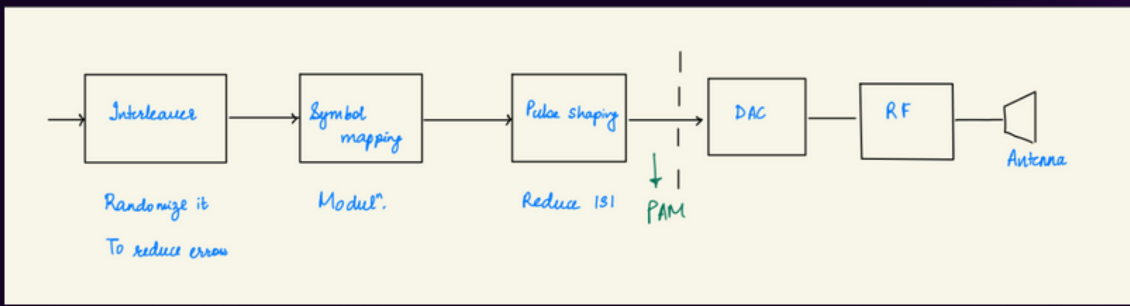
Two major trends are behind the race to 5G: the explosive growth in demand for wireless broadband that can carry video and other content-rich services, and the Internet of Things (IoT), where large numbers of smart devices communicate over the Internet. To achieve these objectives, 5G will provide **extreme broadband speed, ultralow latency, and ultrareliable web connectivity**.

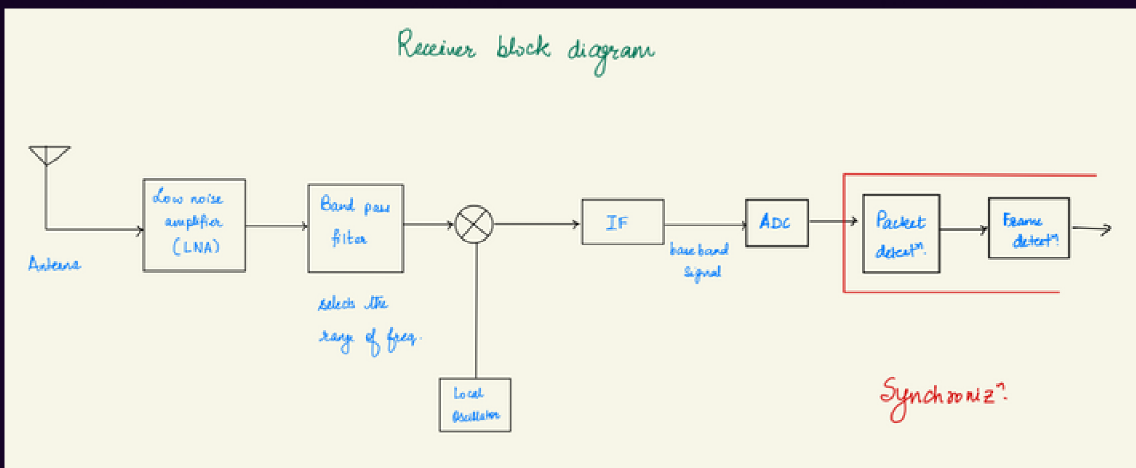
Key 5G parameters	
Latency in the air link	<1 ms
Latency end-to-end (device to core)	<10 ms
Connection density	100x vs. current 4G LTE
Area capacity density	1 (Tbit/s)/km2
Peak throughput (downlink) per connection	10 Gbit/s
Energy efficiency	>90% improvement over LTE

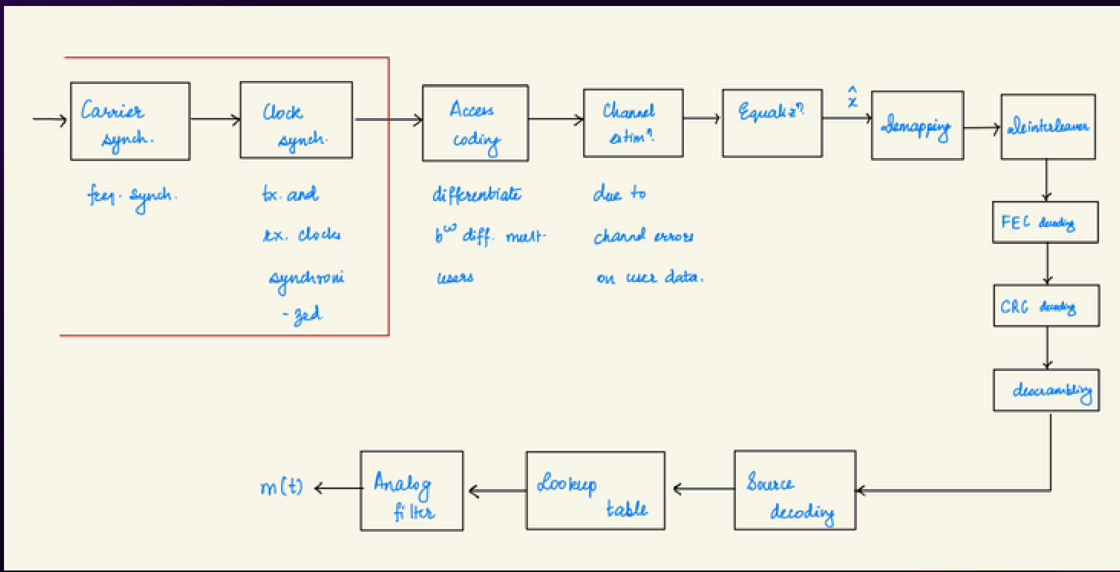
In this PS, we will design and model a quasi 5G NR system, incorporating some important concepts from 5G that mimic real world communication systems

Physical layer is the lowest layer of a communication system that deals with the bit level transformations and error corrections.









Given above is the block diagram of a real world transmitter and receiver signal chain in the physical layer.

In this PS, you are required to implement only few of these blocks. There will be some bonus blocks that will be optional but will carry higher marks.

Transmitter

- **Source, sampling and quantization**
  - Implement these blocks as a random digital message.
  - Create floating point values as sampled data
  - 1 frame should consist of 20*10^3 bits
  - Quantize it by taking the floor value

Transmitter

- **Source Encoder**
  - Convert integers to 2's complement binary numbers
  - Ensure that you take sufficient bits to accommodate all values
  - Also ensure that you take minimum possible bits needed for communication

- **Forward Error Correction (FEC)**
  - Introduce redundancy in the code by writing each bit 5 times over.
  - This redundancy will help us in detecting and correcting bit errors that occur in the channel

- **Modulation/ Mapping**
  - Introduce QAM - 64 modulation scheme
  - Plot the signal constellation

- **Pulse shaping**
  - Implement raised cosine filter
  - Filter the data stream

Channel

- **Rayleigh Fading channel**
  - Implement 5 path channel with different delays and path gains
- **AWGN**
  - Keep SNR as 6dB
- **Timing Offset**
  - Introduce a random (unknown) timing offset
- **Frequency Offset (Bonus)**
  - Introduce a random (unknown) frequency offset

Receiver

- **Carrier synchronization (Bonus)**
  - Corrects the frequency offset
  - Implement if frequency offset is introduced
  - Implement using standard MATLAB functions only
- **Timing synchronization**
  - Corrects the timing offset
  - Implement using standard MATLAB functions only
- **Inverse pulse shaping**
  - Also known as matched filter
  - Take care of the group delay
  - Take care of the filter gain
- **Channel equalization (Option 1)**
  - The channel parameters is used to create the Rayleigh Fading channel in the Channel part
  - Do channel equalization using zero-forcing method
  - That is divide the signal by the transfer function

Receiver

- **Channel estimation and equalization (Bonus-Option 2)**
  - Do channel estimation using MATLAB standard functions
  - Do channel equalization using MMSE

- **Channel decoding**
  - Do inverse repetitive coding
  - That is create a sub-frame of 5 bits
  - Replace the majorly occurring bits among those 5 as the decoded bit
  - Logic is even if bit errors are introduced, then the system will be robust and only a minority of the ensemble of 5 bits will be corrupted

- **Demodulation/ Demapping**
  - Demodulate using qamdemod function

- **The output of this block is the received data**

- **Calculate the bit errors**

Evaluation criteria -

- All the blocks (non-optional) carry 5 marks each
- All the bonus blocks carry 7 marks in addition

Important point -

- After each block in the receiver chain do a constellation plot
- Try to implement using standard MATLAB functions
- Also, implement the entire system using MATLAB

Have fun !!!

CONTACT COORDINATORS FOR MORE QUERIES