EE6323: Wireless System Design

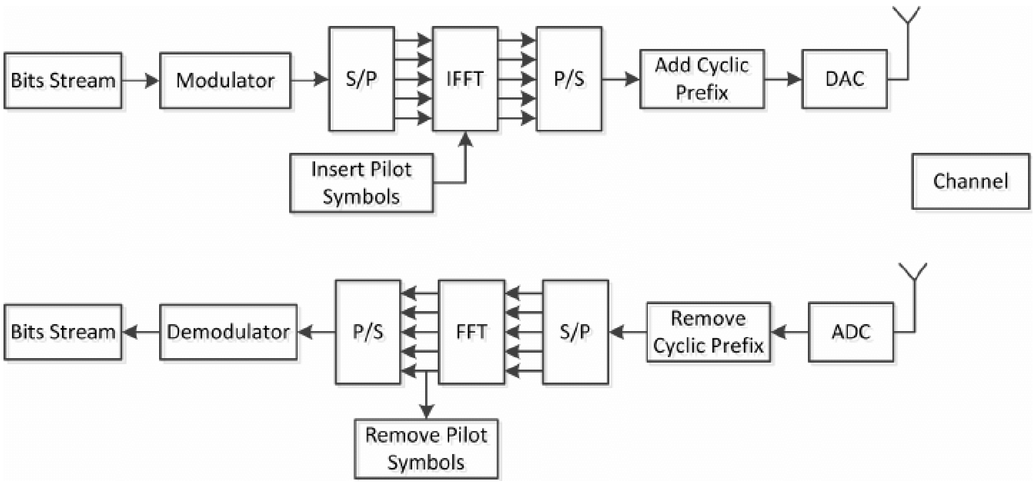
Final project report

Project members:

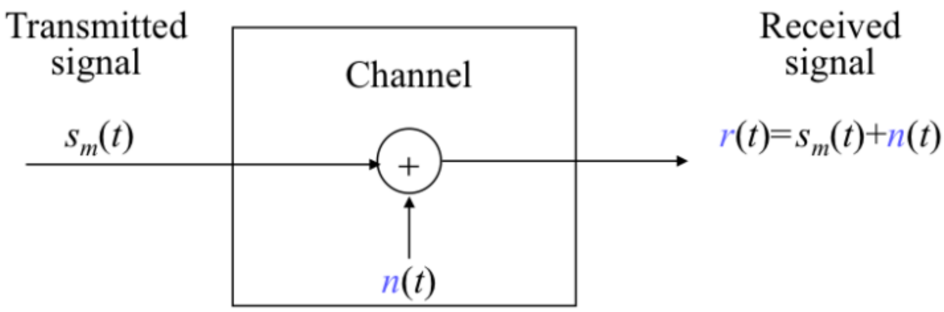
- Baseband transmitter: K R Srinivas (EE18B136)
- RF transmitter: Hari Prakash P (EE20S055)
- RF receiver: J Antonson (EE19B025)
- Baseband receiver: Ruban Vishnu Pandian V (EE19B138)

• Baseband transceiver design:

The designed baseband model is an ideal baseband OFDM transceiver which is based on the figure given below:



The RF module is abstracted as an ideal AWGN channel for simulation purposes.



1. Baseband transmitter:

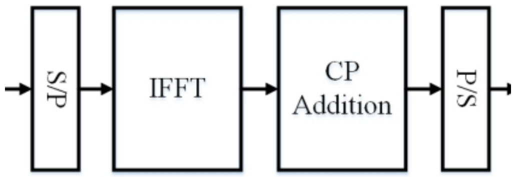
The transmitter consists of four major blocks:

- Modulator
- IFFT
- Cyclic prefix addition
- DAC

Modulator: Information bits are grouped and mapped to appropriate complex symbols based on the constellation used.

IFFT: This is the key module used in OFDM waveform-based communication. The serial input stream of information symbols is parallelized and sent through the IFFT block. The output of this block is the data to be transmitted in time domain.

CP addition: Another essential module in this transmitter is CP addition. Last CP data points are taken and placed before the OFDM symbol. This is done so that channel equalization becomes a simple multiplicative operation in the frequency domain.



DAC: Finally, the discrete-time data points are embedded on a continuous-time transmit pulse (usually a rectangular or a raised cosine pulse) and sent to the RF system.

2. Baseband receiver:

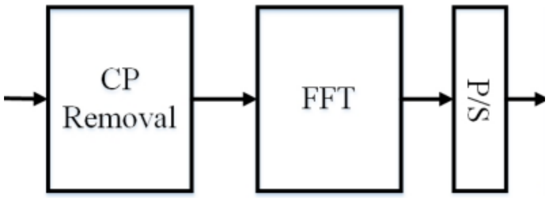
The receiver consists of four major blocks:

- Demodulator
- FFT
- Cyclic prefix removal
- ADC
- Channel estimation and equalization

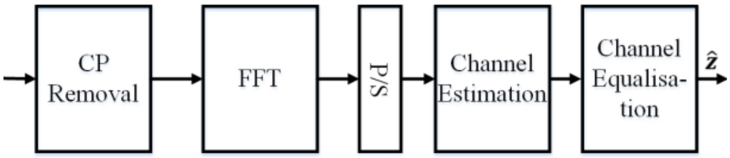
ADC: The signal received from the Rx antenna is converted to baseband through the RF module of the receiver. It is then sampled and quantized by the ADC.

CP removal: The serial Rx signal is parallelized and sent through the CP remover which removes the first CP data points (Inverse of the CP addition module of Tx).

FFT: Output of the CP remover block is sent through the FFT block. The data in time is once again converted back to frequency domain so that the information symbols can be obtained (Inverse of the IFFT block of Tx).



Channel estimation and equalization: To estimate the channel taps, pilot data is transmitted. Based on the received pilot data, channel is estimated. These channel tap estimates are used to equalize the actual Rx data and hence, obtain the actual data symbols.



Demodulator: Finally, information symbols are decoded back to information bits.

3. Parameters used in design:

- Number of slots = 10
- Number of central PRBs = 273
  (Hence, no. of central subcarriers = 273*12 = 3276)
- FFT length = 4096
- CP length = 288
- Constellation: 16-QAM

4. Channel model, estimation, and equalization:

The baseband channel is assumed to be an AWGN channel with unknown attenuation. To find this channel tap, estimation using **Block pilots** is done (Pilots are sent through all subcarriers at the same time).

Mathematically, the channel model is given as:

$$y[n] = (h * x)[n] + w[n]$$

where $y[n]$ is the received signal, $x[n]$ is the transmitted signal, $h[n]$ denotes the filter taps and $w[n]$ denotes the AWGN noise. '$*$' denotes convolution.

For every slot (14 OFDM symbols), the first symbol is assumed to be the pilot. After taking FFT, in frequency domain, the model will be:

$$Y[k] = H[k]X[k] + W[k]$$

Let us assume the pilot symbols are $X_{pilot}[k]$ and the corresponding received data be $Y_{pilot}[k]$. The channel estimates of $H[k]$ is given as:

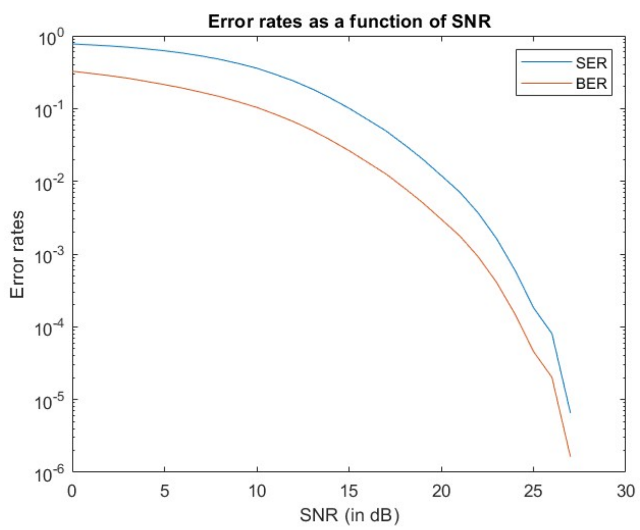
$$H_{est}[k] = \frac{Y_{pilot}[k]}{X_{pilot}[k]}$$

Now when the actual data is received, the channel is equalized as:

$$X_{decoded}[k] = \frac{Y[k]}{H_{est}[k]}$$

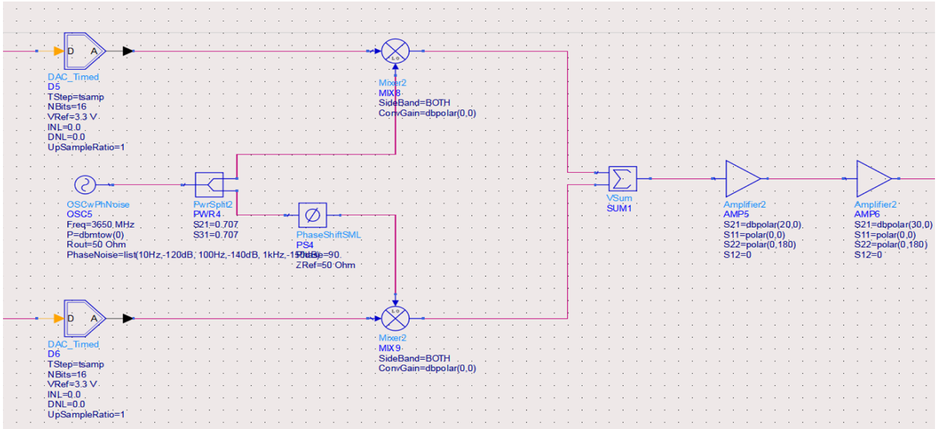
5. SER/BER waterfall curves:

To test the validity of the MATLAB baseband codes, the baseband channel is assumed to be a channel with power gain 100 and AWGN noise. For SNRs varying from 1 to 40, the SER/BER curves are given below:



- **RF Design Report**

1. Transmitter Architecture:



Zin of all components = 50 ohms

Zout of all components = 50 ohms

Digital to Analog Converter:

- Reference voltage level – 5 V
- Bits - 16
- Sampling rate - 122.88 MHz

The 32-bit baseband data is fed to two 16-bit DAC for in-phase and quadrature phase separately. The reference voltage of 3.3v is fixed to get -10 dBm of output power.

Amplifier:

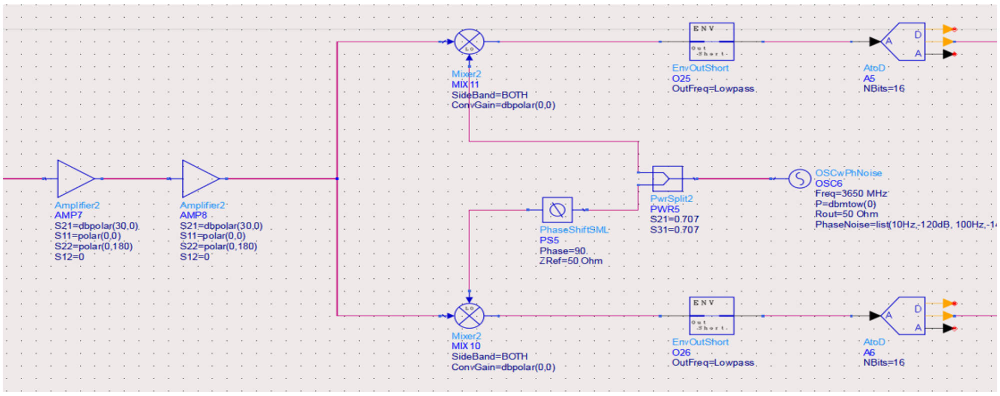
- Amplification stages - 2
- Total Gain - 50 dB

The transmitter gain of 50 dB is fixed to amplify -10 dBm output power of DAC to 10W radiated power (+40 dBm). The total gain is achieved by 2-stage amplification to realize the effects of real-time Power amplifier and driver amplifier.

Mixer:

- LO frequency - 3650 MHz
- Sideband - Lower

2. Receiver Architecture:



Zin of all components = 50 ohms

Zout of all components = 50 ohms

Amplifier:

- Amplification stages - 2
- Total Gain - 60 dB

The receiver gain of 60 dB is fixed to achieve the sensitivity of -94 dBm such that ADC is provided within -35 dBm of power. The total gain is achieved by 2-stage amplification to realize the effects of real-time Low noise amplifiers.

Analog to Digital Converter:

- Reference voltage level – 5 V
- Bits - 16
- Sampling rate - 122.88 MHz

Mixer:

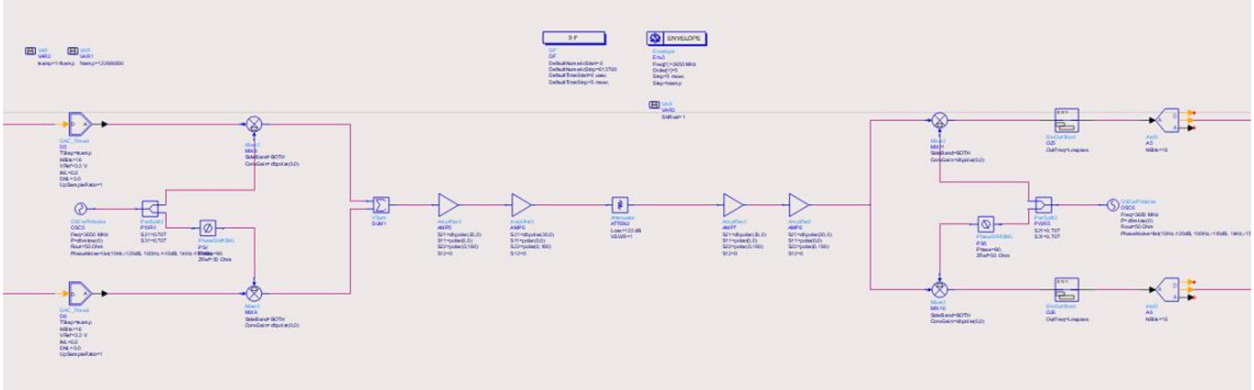
- LO frequency - 3650 MHz
- Sideband - Lower

3. Co-simulation:

Envelope simulation was performed.

Data was read from BaseBand Tx matlab code.

Data output stored as a file to be used by BaseBand Rx matlab code.



Total RF Gain (Transmitter Gain + Receiver Gain) = 110 dB

Channel Attenuation= 122 dB

- ## Instructions for performing the simulations:

1. Unzip the zip file and store all the files in a common directory.
2. First run the MATLAB file **Baseband_Tx.m** to generate the Tx signal data. They will be available in form of text files named as **Tx_I_data.txt** and **Tx_Q_data.txt**.
3. Once these files are generated, transfer them to the data folder of the ADS workspace folder **test3_wrk**. Now run the ADS file named **tx_rx_without_filters**. The Tx signal data will be passed through the RF system and the Rx signal data will be obtained in files named as **Rx_I_data.tim** and **Rx_Q_data.tim**.
4. Transfer those files to the original folder where the MATLAB codes are present.
5. Finally, run the MATLAB file **Baseband_Rx.m** to obtain the SER and BER.