

Mini-Project #2: Basic OFDM Receiver

Due: April 6, 2021 (11:59PM EST)

(worth 20 pts total)

This project, once again, is designed to help you get a better understanding of some of the theory we have been covering in class by having you build a part of the OFDM receive chain as would be used in a wireless communication device. We can then build on this for the final projects and ultimately get to the point where you can explore security aspects of wireless systems at various layers of the protocol stack.

You will be building pieces of the OFDM receiver (LTE in this case) we covered in class in SP6 & 7 (particularly slides 43 and 44 of SP6, and slides 6 and 7 of SP7). The only piece you will not be constructing is the downconverter (that is the section of the receiver from the antenna, the modulator (mixer), and ADC block in Figure 4.3 or equivalently the “Analog[ue] receiver” block in the second figure below.

Project:

Build the following OFDM processing chain components in Matlab:

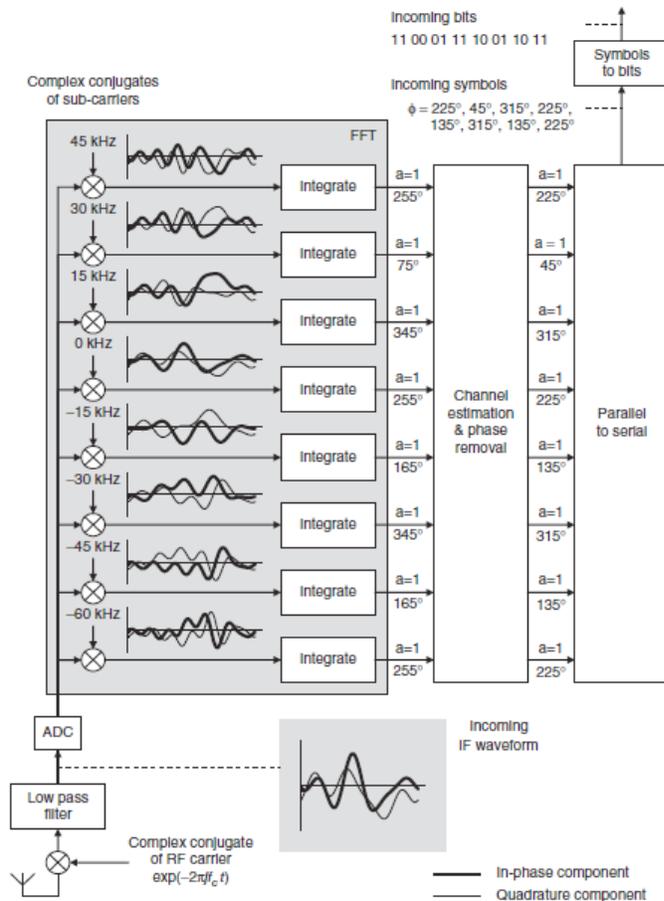


Figure 4.3 Processing steps in an OFDM receiver

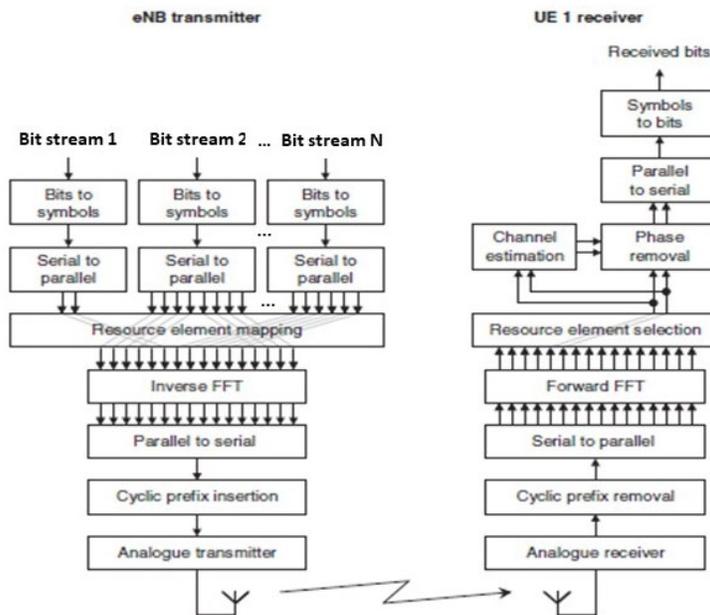


Figure 4.4 Block diagram of an OFDMA transmitter and receiver

Steps your code must perform:

1. You will be given **one subframe** of the downconverted received data (after the “Analogue receiver” in the figure above). You will first perform a cyclic prefix removal from each received OFDM symbol (i.e., ‘transmitted OFDM symbol from the last project). This is the exact opposite process you performed in the last project. However, in order to more closely simulate a more realistic LTE receiver, the first CP for a given slot in each subframe will be 5.2usec and the remaining CPs will be 4.7usec.
2. You will perform the Serial to Parallel (S/P) conversion.
3. Calculate the Forward FFT.
4. For the Resource Element Selection block: you will simulate receiving Control Channel information (put together by the MAC layer) which tells you which Resource Blocks (RBs) multiplexed within the system bandwidth are intended for your receiver. The other RBs are intended for other receivers in the range of the base station (transmitter) and are not to be decoded by your receiver. Normally this control channel information would be differentiated by the receiver (as to whether it is intended for you or not by the control element headers of the PDCCH (Physical Downlink Control CHannel) being encoded with your device’s Temporary Mobile Subscriber Identity (TMSI). However, in this case, you will be using the following scheme to determine where to look for your data by the following steps:
 - a. You will be given three numbers that represent your Resource Block allocation. In order to make this work for our class, we will be allocating the blocks differently than a real system. Your three numbers will represent an index (base 10) of where to find the data destined for you. The cell has a 20MHz total system bandwidth with 100 Resource Blocks ($180\text{kHz} * 100 = 18 \text{ MHz}$ with a 1 MHz guard band on each end).

- i. So, if your numbers were 2, 5, and 9, your information would be found in Resource Blocks (11 – 20), (41 – 50), and (81 – 90).
 - ii. That is, resource blocks are numbered 1 – 100 with the first ten being associated with index 1, second 10 being associated with the second set of ten, etc. RB1 represents the lowest part of the system bandwidth and RB100 represents the highest part of the system bandwidth.
 - iii. Each block of 10 resource blocks will be also allocated a specific modulation scheme and the scheme will not change for the entire subframe. So, for example, RB1-RB10 could be carrying QPSK modulated data and RB11-RB20 could be 32QAM. This will hold for the first slot of data in the subframe and the second slot of data in the subframe. This information would also normally be given to you on the PDCCH, however, we will be telling you the modulation associated for a given set of RBs.
 - b. You will be using these three index numbers to control your Resource Element Selection (de-mux) to pull out the desired Resource Blocks.
5. To take into account the “Propagation Channel” you will be performing a “Phase Removal” step. This information usually comes from the Channel Estimation block, but we will be providing you with a phase estimate for the channel. This is ‘extra’ phase the channel imposes on the transmitted signal, so you will need to remove this ‘extra’ phase to align the data before performing the demodulation step. As you know the channel also imposes other effects on the signal, but we will not be dealing with frequency or multipath variations. Amplitude variations will come in to play due to noise.
 6. You will then perform the Parallel to Serial (P/S) conversion.
 7. Perform the proper demodulation for your RBs (turn the modulation symbols back into bits). Perform the ‘symbols to bits’ mapping using [3GPP TS 36.211 Section 7.1.1 – 7.1.4] as your source for the definition of the given modulation schemes
 8. You will then decode the received bits into 8-bit ASCII characters (again for ease of our simulation we are only using ASCII-encoded bits) to determine the received information.
 9. You will be submitting a Matlab string that represents your received information along with the code that you wrote.

You will again be assuming a sample rate of $f_s = 30.72\text{MHz}$ and sub-carrier bandwidth of 15kHz . That is, the ‘received data’ you are given represent data sampled at a 30.72MHz sampling rate.

There will be some “noise” imposed on the ‘received signal’ so that there may be some resulting bit errors which will lead to some mistakes in our received data. As you know, noise changes the phasor associated with a given modulation symbol. This is an easy way to demonstrate the importance of CRC, FEC, and HARQ techniques in our wireless communication systems.

This isn’t as complicated as it sounds and working through the concepts will help you understand what we covered in class.

I will be posting a grading rubric. You will be submitting your code and decoded output string. We will check that against what you should have received (bit errors included) to assist in grading your project work. There will be points for your code design (meaning I don’t want it to be sloppy and/or really inefficient).

Various illustrations from class and references used in class

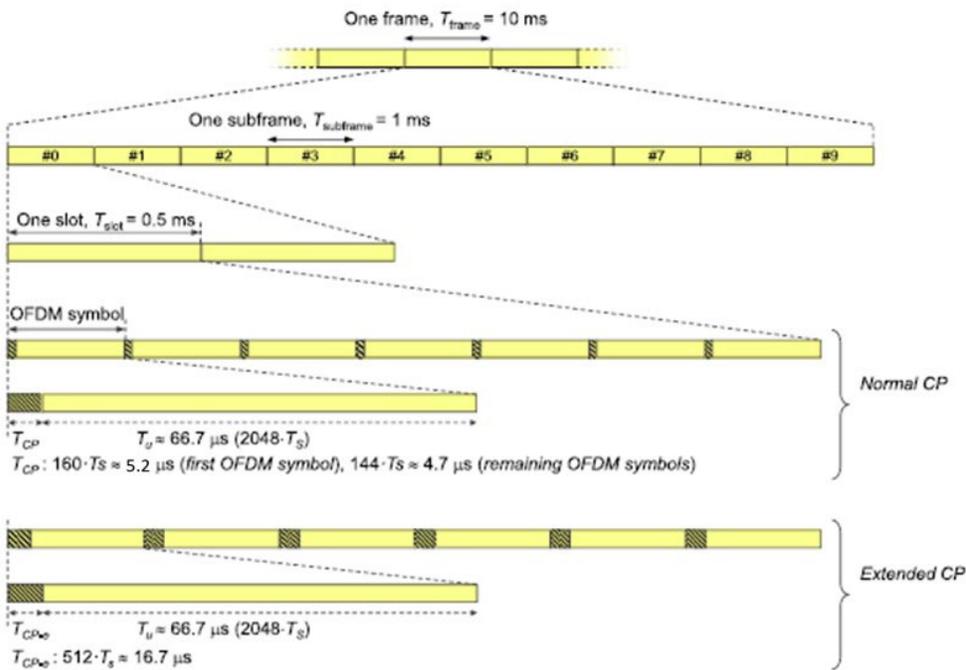
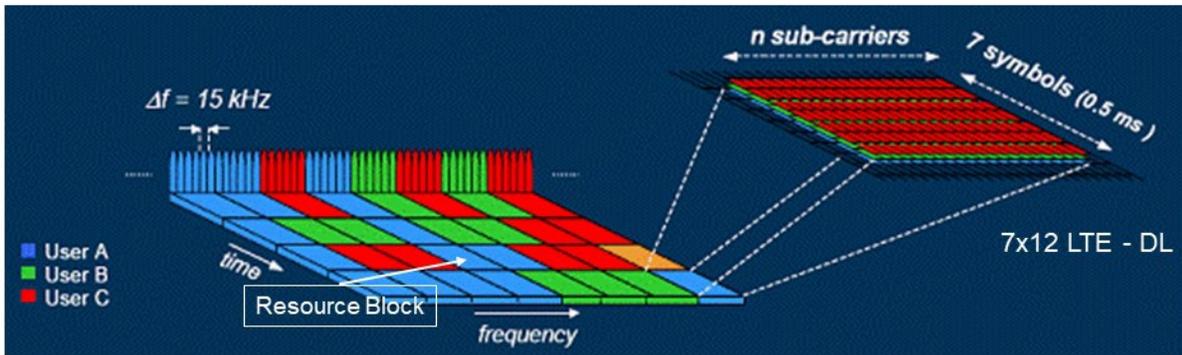
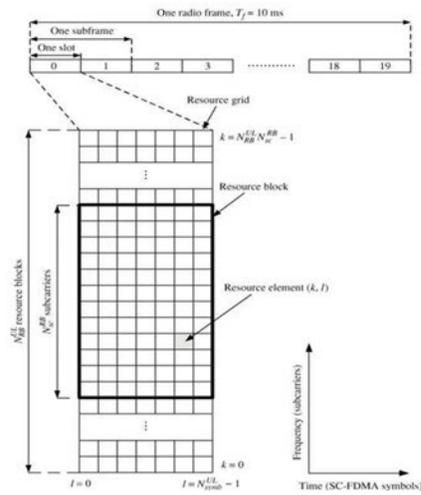


FIGURE 5.1

LTE time-domain structure.

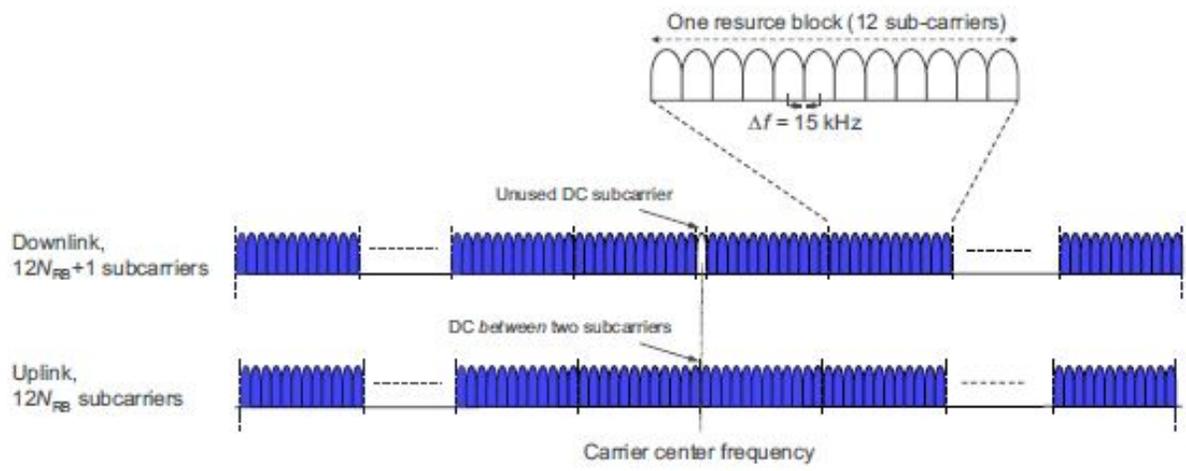


FIGURE 5.3

Frequency-domain structure for LTE.