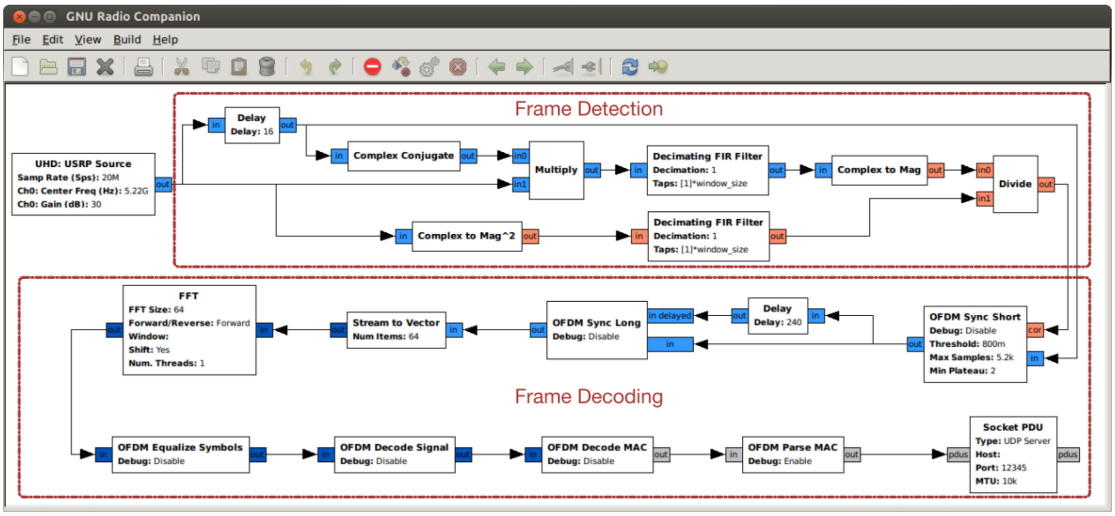
Acording to the design parâmetros:



We can visualize the Bandwidth for the Orthogonal Frequency Division Multiplexing (OFDM) receiver implemented in GNU Radio that supports both WiFi with a bandwidth of 20 MHz and IEEE 802.11p DSRC with a bandwidth of 10 MHz. It supports both WiFi with a bandwidth of 20 MHz and IEEE 802.11p DSRC with a bandwidth of 10 MHz.

There 12 unused sub-carries. Theses are of 2 types: DC(Center Subcarriar) and edge subcarries

Center Sub-carrier is not used, the center subcarrier in 802.11a is intentionally left unused because it corresponds to DC (0 Hz) in the baseband signal. In practical hardware, imperfections in the transmter and receiver, such as mixer leakage or small DC biases, can introduce a DC offset, which appears as unwanted energy right at 0Hz.

If that subcarrier were used carry data, the DC offset could easily easily distort or completely mask the transmitted symbol, making reli iable demodutation impossibo

This has 2 benefits:

1- It prevent hardware DC bias from distorting a data tone
2- It creates a notch (a visible hole) at the center of the spectrum

Edg subcarriers: the outermost 11 subcarriers at both end of the spectrum are set to zero

Frame Detection: The first task in the receive chain is to actually detect the start of an OFDM frame. Each IEEE 802.11a/g/p frame starts with a short preamble sequence, which consists of a pattern that spans 16 samples and repeats ten times. The employed frame detection algorithm has been introduced in [3]. It is based on the autocorrelation3 of the short training sequence.Following [3, Algorithm 1], we exploit this cyclic property and calculate the autocorrelation value a of the incoming sample stream s with lag 16 by summing up the autocorrelation coefficients over an adjustable window Nwin

$$\mathrm{a}[n] = \sum_{k=0}^{N_{\mathrm{win}}-1} \mathrm{s}[n+k]\,\bar{\mathrm{s}}[n+k+16].$$

The summation over the window (which finally results in the calculation of a moving average) acts as a low-pass filter. We experimented with different window sizes and found 48 to work well. Due to the cyclic property of the short training sequence, the autocorrelation is high at the start of an IEEE 802.11a/g/p frame. In order to be independent of the absolute level of incoming samples, we normalize the autocorrelation with the average power p and calculate the autocorrelation coefficient c as

$$\mathrm{p}[n] = \sum_{k=0}^{N_{\mathrm{win}}-1} \mathrm{s}[n+k]\,\bar{\mathrm{s}}[n+k];$$

$$\mathrm{c}[n] = \frac{|\mathrm{a}[n]|}{\mathrm{p}[n]}.$$