SDR implementation of an OFDM transmitter and receiver

Jaime Luque Quispe*, Jackelyn Tume Ruiz*, Luis G. Meloni
Faculty of Electrical Engineering
State University of Campinas - UNICAMP
{luque, jtumer, meloni}@decom.fee.unicamp.br

***Abstract*—The new level of connectivity among machines and people promised by the forthcoming 5G mobile communication system will shape diverse application areas from healthcare and education to smart factories and autonomous driving. These applications have a variety of different sets of requirements on data rates, latency, and reliability which are expected to be supported by an enhanced version of the OFDM modulation used today by the physical layer of 4G LTE systems. The development of a new OFDM-based air interface able to flexibly support that diversity of applications is of current interest. On the other hand, software defined radio is an appealing concept to introduce flexibility in the radios as the functionality of the communication system relies on the software implemented in a programmable device. Aiming to analyze the factibility of implementing elements of new communication systems under an SDR approach, this work presents the prototyping of a SDR OFDM communication link for video transmission using the open-source tool GNU radio and the RF hardware interface Universal Software Radio Peripheral. Preliminary results shows that a SDR opens opportunities in both researching and teaching enabling the designing and testing of digital communication systems in a very practical way taking into account the real conditions of the channel and front-end impairments.**

I. INTRODUCTION

Over the past years, the development of wireless communications systems has been largely influenced by the increasing of the traffic demand. Emerging new applications of low latency, higher connection density and ubiquitous gigabit connectivity require more efficient and flexible modem and information processing algorithms thus motivating an intense research and development work at industry and academia on the physical layer (PHY) and medium access control (MAC) layer components of new communication systems, since they affects all the upper layers and overall system performance in terms of energy efficiency, spectral efficiency, achievable throughput, quality of service (QoS), etc. On the other hand, the prototyping of a communication system with ongoing research needs to be flexible such that it can be able to support new algorithms and test different use cases. Nowadays, 5G transmission and reception blocks are being formulated theoretically and analyzed by simulation, however there is a lack of implementations that allow the proof of concepts and evaluation under real channel conditions, which deteriorates the signal quality.

Software defined radio (SDR) is an appealing approach for flexible prototyping of communication systems where the functionality of the systems relies on the software implemented in a programmable device: system on chip (SOC) field programmable gate array (FPGA), digital signal processor (DSP) or general purpose processor (GPP). Thus, the reconfiguration of the system is possible, even during the execution time, by reloading a firmware or modifying the software and recompiling it. Hence, SDR opens opportunities in both researching and teaching enabling the designing and testing of digital communication systems in a very practical way taking into account not only the above-mentioned real channel conditions of the channel but also the front-end impairments.

The objective of this work is to show that the SDR technology is ready to be employed in the designing and prototyping of digital signal processing (DSP) modem algorithms to build reconfigurable communication systems. A reconfigurable platform is of particular interest since building blocks can be tailored for different applications, e.g. vehicle-to-vehicle (V2V) communications can require not only short frames to achieve low latency but also enlarged subcarrier spacing, narrowband internet of things (IoT) can require the use of a single carrier waveform to achieve sufficient coverage with low power, and custom cyclic prefix (CP) would be needed to achieve adequate service coverage in different application cell sizes. With this aim, this work presents the implementation of SDR communication link for video transmission using the GPP of a standard PC for the digital baseband functions and the universal software radio peripheral (USRP) for the RF processing. The system uses the orthogonal frequency division multiplexing (OFDM) multicarrier modulation scheme as an example case since OFDM is widely used in many current communication systems such as LTE and 802.11 and probably also in 5G new radio systems. As the software layer it was used GNU radio which is a open-source set of libraries of baseband algorithms written in C++ and python.

The remainder of this work is structured as follows: A basic model of a OFDM communication system is presented in section II. Section III presents the SDR implementation of this model and presents some performance results. Section IV presents our conclusions.

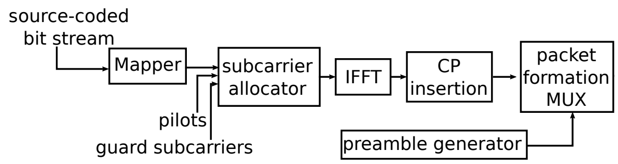

Fig. 1: Block diagram of an OFDM transmitter

II. SYSTEM MODEL

OFDM is a multicarrier modulation scheme based on the division of a serial data stream in a group of parallel data streams which are transmitted on orthogonal subcarriers. By properly configurating the OFDM waveform, when it is sent through a time-dispersive channel or frequency selective fading, the subcarriers can be affected only by a constant channel or frequency flat fading. In this kind of channels, the received signal can be distorted by intersymbol interference (ISI), which is a crosstalk between signals arrived with different delays. This problem is addressed in OFDM by inserting a copy of the last part of the symbol, also known as cyclic prefix (CP), at the beginning to absorb channel delay spreads. The CP helps to preserve the synchronization and maintain the subcarrier orthogonality, and it also introduces a cyclic convolution between the transmitted symbol and the impulse response of channel thus enabling the use of one-tap equalizers to compensate the channel distortion. A CP-OFDM signal of N subcarriers can be expressed in the time domain by

$$x[n] = \sum_{l=-\infty}^{\infty} \sum_{k=0}^{N-1} S_{k,l} p[n - lN_T] e^{j2\pi k(n-lN_T)/N} \tag{1}$$

where N, N_{CP} and N_T are the number of samples of the payload, CP and entire OFDM symbol respectively, i.e., $N_T = N_{CP} + N$.

A block diagram of a transmitter is shown in Fig.1, where a stream of bits is converted into complex data symbols by the constellation mapper and then placed on determined positions of a N-order inverse fast Fourier transform (FFT) by the subcarrier allocation block. The subcarrier allocator also places the pilots for channel estimation purposes and leaves unused some subcarrier positions (guard subcarriers) to contain the out-of-band emission (OOBE) from the active subcarriers. Then, groups of N frequency-domain samples are converted to the time domain by the inverse FFT and then a CP is appended at the beginning of each OFDM symbol. Optionally, these CP-OFDM symbols can be pulse shaped in the time-domain for spectral containment. For example, the power spectrum density (PSD) of 1 given by

$$PSD_x = T \sum_{k=0}^{N-1} E\left[|S_{k,l}|^2\right] \left| sinc\left[T\left(f - \frac{k}{T_N}\right)\right]\right|^2, \tag{2}$$

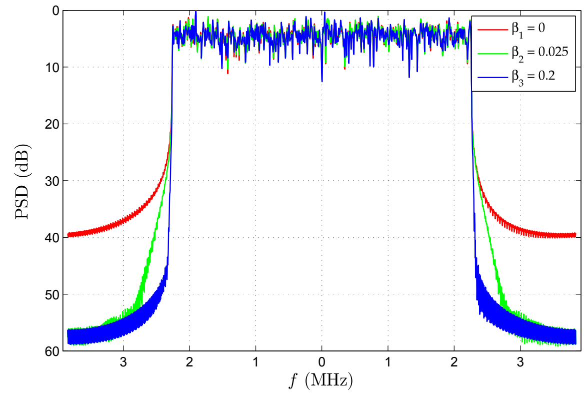

Fig. 2: Effect of pulse shaping on the reduction of the OOBE of a OFDM signal. The worst case $\beta = 0$ is included for comparison.

can be better contained by using a raised cosine (RC) pulse shaping filter and in this case, assuming the frequency response of the filter to be $P(f)$, the PSD results in

$$PSD_x(f) = \frac{1}{T} \sum_{k=0}^{N-1} E\left[|S_{k,l}|^2\right] \left| P\left(f - \frac{k}{T_N}\right)\right|^2. \tag{3}$$

In Fig. 2 is shown the reduction of the OOBE of a 5-MHz LTE waveform achieved by the application of some values of roll-off length β of the RC filter.

Additionally, in packet-based data transmissions, special preamble OFDM symbols can be prepended in the front of every frame (one frame is composed of a number of OFDM symbols). These symbols can be used for detection, time and frequency synchronization and initial channel state estimation [gnuradio]. Only for a sake of illustration, the preamble insertion is indicated at the end of the Fig. 1. At this point, the OFDM signal is ready for conversion to the RF domain and further transmission. These processes usually involve additional digital signal processing such as digital up conversion (DUC) and digital pre-correction of possible IQ imbalance of the RF components.

At the receiver side, two important process are needed before data demodulation, namely the synchronization, which detects the beginning of the OFDM symbols and estimates the frequency offset, and the channel estimation, which serves to compensate the distortions inserted by the channel. After these processes and the discarding of the CP, the time-domain samples are converted into the frequency domain and from them the payload data symbols are extracted and then equalized using the channel response previously estimated. The equalized data symbols (complex numbers) are then converted to bits and passed to a forward error correction (FEC) block to recover the original information.

A. Synchronization

Synchronization is a critical stage of the reception process. A poor synchronization accuracy leads to severe degradation of link performance thus making impossible to recover

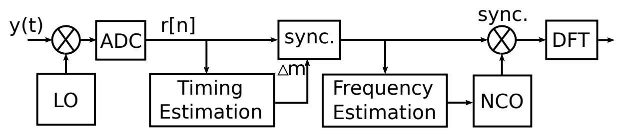

Fig. 3: Block diagram of the time and frequency OFDM synchronization stages

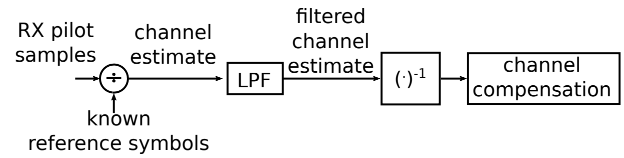

Fig. 4: Channel estimation and equalization processes

transmitted information. The synchronization aims to correct any timing and frequency deviations of the received signal respect to the transmitted one. It also includes the sampling clock synchronization. Time synchronization ensures that the receiver samples only useful OFDM payload samples thus avoiding ISI from adjacent OFDM symbols while frequency synchronization removes the frequency errors (e.g. due to Doppler effect or receiver oscillator mismatching) to guarantee the orthogonality of the subcarriers and thus avoiding ICI. A block diagram of a synchronizer is shown in Fig.3.

By assuming that the transmit carrier frequency is f_c and the received signal $y(t)$ is down-converted by a non-matched local oscillator of frequency f_{LO}, a carrier frequency error (CFO) is produced whose value is $f_o = f_c - f_{LO}$. Then, the sampled signal $r[n]$ can be expressed as

$$r[n] = y[n]e^{j2\pi f_o n T_s}, \tag{4}$$

where T_s is the sampling frequency and n is the sample index. Δm is the deviation in samples of the received signal respect to an ideal signal whose beginning is aligned with the start of sampling. Δm and f_{LO} are estimated in the timing and frequency estimation blocks respectively. Given an estimated frequency offset of $\hat{f}_o$, a numerically controlled oscillator (NCO) rotates the signal by $exp(j2\pi\hat{f}_o nT_s)$. Then, the data subcarriers can be recovered by means of a FFT for subsequent processing. In order to track online the input signal and compensate the errors as soon as possible, the synchronizer needs to respond immediately [DIEGO BARRAGAN] which requires fast signal processing algorithms. They can be classified in non data-aided and data-aided synchronization algorithms whose difference is that the first scheme exploits the redundancy of the CP to perform the time and frequency offset estimation while the second one utilizes time and frequency resources as preamble and pilot subcarriers.

B. Channel estimation and equalization

As mentioned above, the radio channel distorts the transmitted signal varying its amplitude and phase, thereby the response of the channel needs to be estimated to compensate (equalize) the errors introduced in the OFDM frame.

There are different techniques to perform channel estimation which can be classified into blind channel estimation and data-aided estimation techniques. The first scheme does not require dedicated time slots and reference subcarriers (training symbols or pilot subcarriers) for channel tracking, so the spectrum efficiency can be preserved. Similar to data-aided frequency estimation, channel estimation can employ reference information multiplexed with the useful payload. In case of pilot subcarriers, they usually use low order modulations (e.g. BPSK or QPSK), for easy recovering. The main advantage of this scheme is its simplicity, but unlike blind estimation techniques, it loses spectrum efficiency. The estimation and compensation processes using this approach are represented in Fig. 4.

III. SDR COMMUNICATION SYSTEM PROTOTYPING

This section details about an implementation of the system presented above using USRP RF interface cards and the open-source tool GNU Radio through its graphical interface GNU radio companion (GRC).

A. Hardware employed

There were utilized the following modules:

- Two standard PC with Linux operating systems and GNU radio installed. One PC acts as the transmitter and the other as the receiver.
- Two USRP boards B210, which are equipped with digital-to-analog converters (DACs) and analog-to-digital converters (ADCs), 2x2 MIMO antennas, reference clocks and RF chip transceivers with continuous frequency coverage from 70 MHz to 6 GHz and 56MHz of bandwidth. They also have 3.0 USB interface for communicating with the host PCs. This implementation used 2 GHz as the carrier frequency.

B. Description

The GRC schematic containing the building blocks of the OFDM transmitter is shown in Fig. 5 and the receiver in Fig.6.

In the OFDM transmitter, it can be seen that there are defined variables like sampling rate, Sync word 1, Sync word 2, occupied carriers, pilot symbols, etc, which will be used in the transmiter and receiver OFDM. The information comes of one TS video, indicated in the File source block, this information is sent at Stream to Tagged Stream block that will add tags at evenly spaced intervals, the output information passes through an error corrector CRC32 to then be divided into Headers bits and Payload bits, which will be multiplexed. Was used BPSK and QPSK modulations respectively. The output information will be sent to OFDM Carrier Allocator block, here is indicated the occupied carried and the Cyclic Prefixed is added.

The information will be sent using a USRP Block, where are configured the basic parameters for the transission, like sample rate, frequency, gain, etc. Was used a 2GHz frequency and 2MHz sample rate.

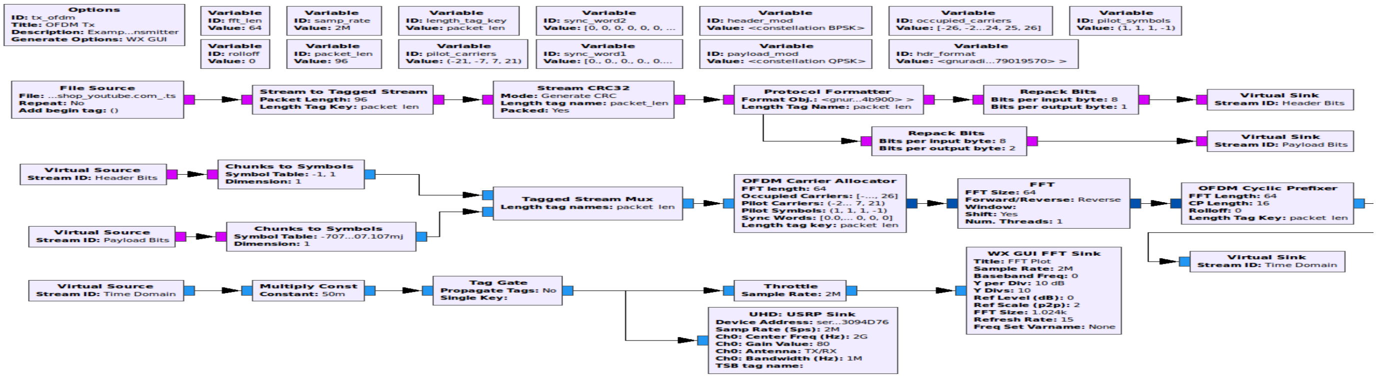

Fig. 5: Block diagram of the transmissor system

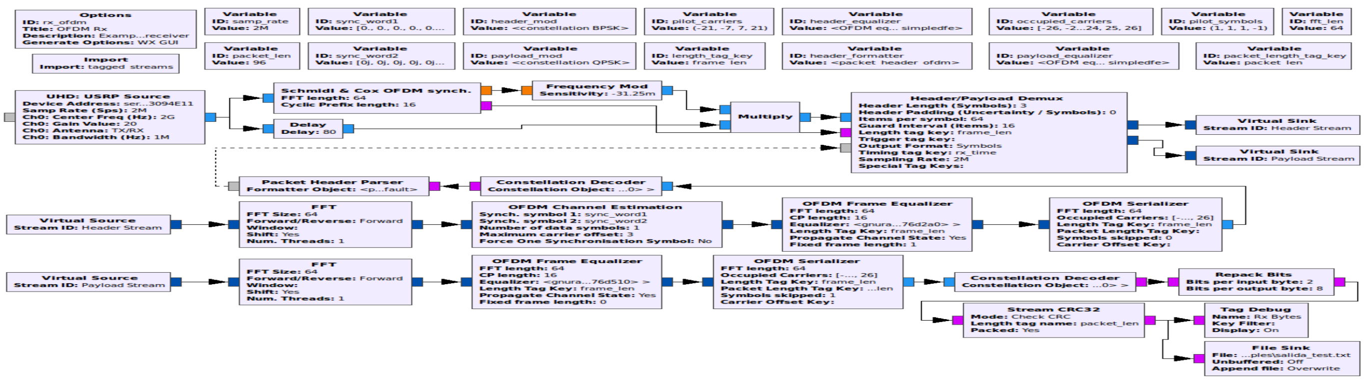

Fig. 6: Block diagram of the receptor system

In the OFDM receiver, we can visualize four important parts. First, the USRP source and the synchronizer, was used a Schmidl & Cox OFDM synchronizer to realize the frame offset correction, immediately the samples are ducted into the Header/Payload Demox block and a trigger signal is sent to indicate the begining of the frame. Then, is performed the channel estimation into the header, the FFT shifts OFDM symbols to the frequency domain and the output information is sent to the Channel Estimation block. This block carrying out the channel estimation and the frequency offset of the preamble samples OFDM. The firts two symbols are of syncronization which are used to estimated the equalization coefficient, then this symbols are discarded and the rest of symbols are sent out of the block. When the stream of OFDM symbols is sent at the OFDM Frame Equalizer, it realize the equalization in one or two dimensions for one OFDM frame, besides correct the coarse frequency offset of the carrier. The OFDM Serializer select the data symbols of the OFDM symbols and emits them as a sequency of complex values, which will be demodulated. The bits of the header then are interpreted by the Packet Header Parser block, the results of the this interpretation is feedback to the demuxer for lets it know the length of the payload and the tagged stream. The payload demodulation don't need to do the estimation channel because the channel state information is on the payload tagged stream. Finally, the information receiver is saved in a output File, we obtained the TS video sended in the transmission.

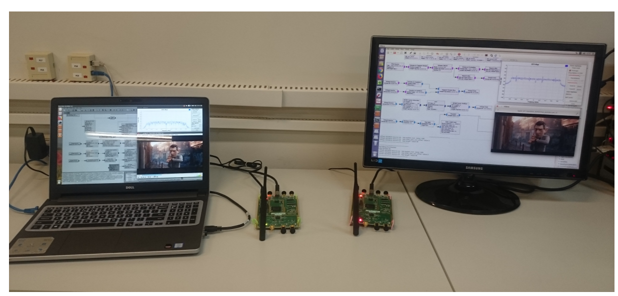

Fig. 7: Physic implementation of transmission and reception system

In the Fig. 7 can be see the hardware implemented to realize the proof of concept, booth computers have instaled the graphics interface of the GNU Radio and the are connected to their corresponding USRP. The left computer is doing the information transmission, in this case the information is a TS video, in the other hand, the right computer is doing the signal reception and can be seen that a good quality video was received as well as the OFDM spectrum.

IV. Conclusion

Increasingly, the SDR systems are introducing into radio communications world, proporcing scalables solutions easy to adapt to sundry kind of escenaries.

In the present work was implemented a example case simple of undertand in orden to serve like introduction for the construccion of more sophisticated platforms solutions aimed at development of solutions oriented to 4G, LTE or IoT systems.

This work provided a comprensive literature review of one of several SDR solutions: GNU Radio platform and USRP hardware. Based on the OFDM concepts was test out that SDR allows desing flexible and scalables systems, capable to adapt to sundry communications sistems and new applications.

Also, was review the most important concepts to consider in the blocks used in the desing of the diagram blocks of the solution such as channel model, synchronism, channel estimation which are directly related to implementation of OFDM system in hardware.

The realized example case, where was implemented a video transmission and reception system using OFDM modulation system, has been successfully, the design was did in GNU Radio and the USRP was used like a RF interface.

References

[1] J. Mitola, Software Radios, *Survey, Critical Evaluation and Future Direction*

[2] A. R. Bahai, B. Saltzberg, M. Ergen ,Multi-Carrier Digital Communications *Theory and Applications of OFDM Second Edition*

[3] https://www.ettus.com/product/details/UB210-KIT

[4] https://wiki.gnuradio.org/index.php/Main_Page

[5] M. Dillinger, K. Madani, N. Alonistioti, *Software Defined Radio, Systems and Functions*

[6] B. Farhang-Boroujeny, H. Moradi *OFDM Inspired Waveforms for 5G* , 2016