Demonstration of a USRP-based Communication Network for Internet-of-Things (IOT) Application

Sohaib Yaqoob Chaudhry
National University of Sciences and Technology
Islamabad, Pakistan
syaqoob@cae.nust.edu.pk

Jamal Haider
National University of Sciences and Technology
Islamabad, Pakistan
jrasool@cae.nust.edu.pk

***Abstract*—In this paper, an SDR-based single-carrier, simplex, wireless communication system is presented for Internet-of-Things (IOT) applications. The research work includes the design of both hardware and software such that the system can communicate reliably over a distance of 100 meters. The hardware component of this work includes the complete design and fabrication of the transmitter and receiver front-end PCBs. On the other hand, the software aspect which consists of signal processing is implemented on a software-defined radio where various impairment correction techniques are implemented to mitigate challenges in the wireless channel such as carrier frequency offset, phase offset, and timing offset. Furthermore, an equalization algorithm is implemented to compensate for errors that arise due to the phenomena of multipath fading. To demonstrate the efficacy of the proposed communication system in an indoor environment, an extensive validation test scheme was devised and performed. The proposed communication system demonstrates an optimum spectral efficiency of 3.2 bps/Hz at a bit rate of 26.66 Mbit/s.**

I. Introduction

The Internet of Things (IoT) has emerged as a transformative force, interconnecting devices and enabling a seamless exchange of information in ways that redefine the capabilities of modern technology. In the era of smart cities, industrial automation, and interconnected ecosystems, the importance of IoT applications cannot be overstated. Central to the success of IoT, however, is the ability to handle vast amounts of data efficiently and utilize the scarce radio frequency spectrum judiciously. These challenges can be addressed by harnessing the power of Software-Defined Radios (SDR) with a custom-made software and hardware solution, aimed at achieving high data rates and spectral efficiency for IoT applications. Software-Defined Radio (SDR) stands at the forefront of revolutionizing traditional communication paradigms, offering a dynamic and adaptable solution to the evolving needs of wireless communication systems [1]. Unlike conventional radio systems, SDR enables the separation of hardware functionality from software control, providing a flexible platform for real-time reconfiguration of communication parameters. The programmability and agility of SDR make it a key enabler in overcoming the limitations posed by fixed communication schemes.

To achieve higher data rates and spectral efficiency within IoT applications, this article introduces a novel approach that combines the versatility of SDR with a custom-designed software and hardware solution. By tailoring both the hardware components and the software algorithms to the specific demands of IoT communication, a synergistic system is created that maximizes performance. The custom solution enhances the adaptability of SDR, allowing it to intelligently adapt to the unique communication needs of IoT devices. This comprehensive integration enables the network to dynamically allocate resources, optimize modulation schemes, and intelligently navigate the spectrum landscape, resulting in superior data rates and spectral efficiency.

In the subsequent sections, a detailed architecture of the custom SDR solution, the intricacies of the developed software, and the hardware designed for implementation, that collectively contribute to the achievement of desired performance, is presented. The paper presents a complete single-carrier, simplex, wireless communication system designed to reliably communicate at a distance of 100 meters. This includes the design of the receiver and transmitter PCB's, and signal processing implemented on a software defined radio (SDR), the USRP N210 from Ettus research. Through experimental results and case studies, the effectiveness of this approach and its potential to reshape the landscape of IoT communication, setting new benchmarks for efficiency and adaptability in the ever-expanding realm of connected devices, is emphasized.

The paper is structured as follows. Firstly, a detailed discussion of the software design is presented. It includes the complete architecture comprising of impairment corrections and equalization, their implementation, and how the issues of multipath fading is modeled and inverted via the equalizer. Next, the hardware part is discussed which includes the design of both transmitter and receiver PCBs used in the testing and validation process. Finally, the results from real link testing conducted is presented and discussed.

II. Software Design

The optimized software plays a pivotal role in the success of any communication system, facilitating seamless signal processing and robust error correction mechanisms. This section delves into the details of software design, comprehensively elaborating on the implementation of impairment corrections for carrier-frequency offset (CFO), phase offset (PO), and

timing offset (TO) in the communication link due to low-cost hardware and multipath transmission channels. Figure 2 presents a detailed block diagram illustrating the software architecture for both the transmitter and the receiver of the communication system.

A. Transmitter Software

In Figure 2 the software architecture of the transmitter is depicted which shows a conventional process, including mapping the concerned data bits to Quadrature Amplitude Modulation (QAM) symbols. Afterwards, the frame is constructed followed by pulse shaping and up-sampling. Subsequently, the signal is sent to the Universal Software Radio Peripheral (USRP) for transmission. The transmitted frame structure is illustrated in Figure 1. As evident from the figure, the frame structure consists of different types of symbols i.e., the initial coarse synchronization symbols, repeated twice in order to facilitate coarse timing and coarse CFO estimation. These coarse synchronization symbols are followed by 512 fine synchronization symbols, which are also referred to as pilot symbols. Furthermore, these pilot symbols are then followed by $N_d = 4096 - 512 = 3584$ actual data symbols. The complete frame consists of 4096 symbols, excluding the coarse synchronization symbols. The system is single-carrier using a root-raised cosine (RRC) pulse shaping filter with a fixed roll-off factor of $\beta = 0.25$ [3].

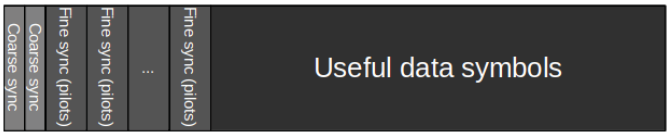


Fig. 1. Frame structure

The Ettus USRP has a sampling rate of 100 MS/s [8], thus for a fixed interpolation factor of N_{int}, sampling rate f_s can be expressed as:

$$f_s = \frac{100}{N_{\text{int}}} \text{ MS/s} \tag{1}$$

which is the sampling rate of the complex baseband signal before interpolation. Pulse shaping, i.e., the number of samples per symbol, requires up-sampling with a factor N_{ups}. Then, due to the excess bandwidth of the RRC pulse [3], the occupied bandwidth will be calculated as:

$$s_{\text{bw}} = (1+\beta)\frac{f_s}{N_{\text{ups}}} = (1+\beta)\frac{100}{N_{\text{int}}N_{\text{ups}}} \text{ MHz.} \tag{2}$$

Hence, from (2) it is evident that bandwidth can be increased via two methods i.e., decreasing either N_{int} or N_{ups} or both.

B. Receiver Software

The efficacy of the receiver software profoundly influences the overall performance of a link, dictating the system's ability to discern and interpret transmitted data accurately. In the landscape of wireless communication, the receiver software stands as the linchpin, transforming electromagnetic signals into actionable intelligence and ensuring the reliability and integrity of the transmitted message. In the subsequent sections, the receiver software is elaborated extensively with sections covering the impairment corrections and equalization implementation.

C. Impairment corrections

To achieve precise reception of transmitted signals, various impairment correction techniques are employed which are summarized in the following steps. The accuracy and robustness of these steps were verified through simulations for various levels of SNR, CFO, PO, TO, and fading:

1) The USRP produces a complex baseband signal, by Digital Down Conversion (DDC) of an Intermediate Frequency (IF) signal.
2) Coarse Frequency Offset (CFO) estimation and correction by measuring the DC offset in the spectrum is performed. For this purpose, a DC component was deliberately introduced in the coarse sync symbols at the beginning of the frame, by repetition of the same symbols.
3) Subsequently, signal detection and coarse timing offset estimation are performed by offsetting, conjugating, and multiplying the received baseband signal. Since coarse sync symbols are repeated twice, hence, the multiplication produced a "plateau" over their duration in the regions where they align which is further correlated with a rectangular window of the same size, producing a peak, that is used for signal detection as well as a coarse symbol timing offset (TO) estimation. This procedure is similar to the well-known timing estimation done for OFDM by Schmidl and Cox [7] and is illustrated in Figure 3.
4) Using the fine sync symbols, matched filtering is carried out followed by fine TO estimation. This needs to be done before downsampling since the coarse timing may result in residual TO. Correlating an *upsampled* version of the fine sync symbols with the received baseband signal *before* downsampling, will produce a very high peak since the zeros in the upsampled version will essentially kill the received signal when it's only off by one symbol. This method is however quite sensitive to CFO and PO which is why coarse CFO (and potentially PO) correction is performed before this step. The resulting fine timing estimation from one of the link tests can be seen in figure 11a.
5) In the next step the coarse sync symbols are downsampled and discarded.
6) Using the fine sync symbols, fine CFO estimation is performed. Specifically, the received fine sync symbols are extracted, and the difference in phase between them and the actual known fine sync symbols as a function of symbol frame-index is computed. This in turn produces measurements with an (ideally) clear linear trend that may fit into a first-degree polynomial, where the slope will be the residual CFO and the interception be the PO.

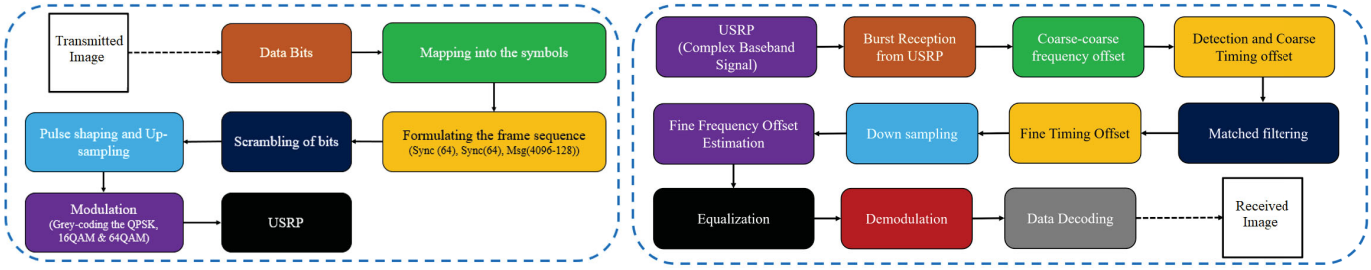


Fig. 2. Block diagram of the complete software of transmitter and receiver

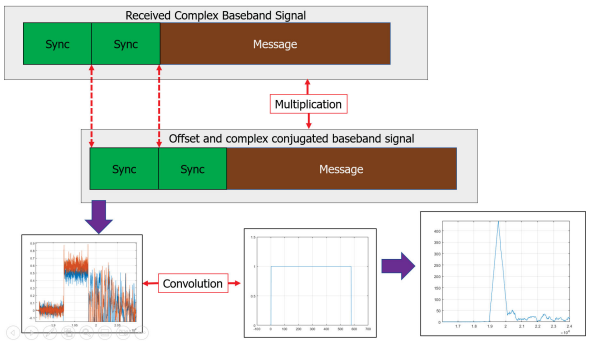
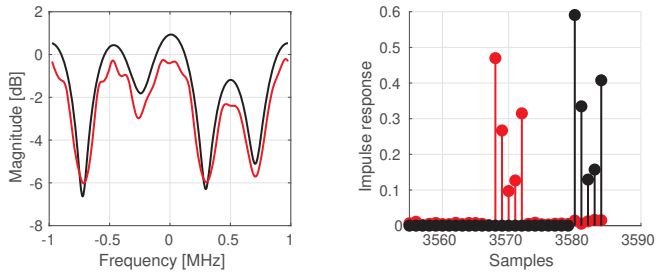


Fig. 3. Conceptual block diagram for detection and coarse timing offset estimation



(a) (b)

Fig. 4. An example of the estimation of the wireless channel. In (a) the magnitude of the frequency response, and in (b) the magnitude of the impulse response. The black curve is the simulated channel, and red is the estimation. The true impulse response is purposely delayed so as to not clutter the image.

The resulting fine CFO estimation from one of the link tests can be seen in figure 11b.

7) In the last step equalization was performed to invert the wireless channel which gave us the final constellation diagram.

D. Equalization Algorithm

In order to achieve higher spectral efficiency an equalization algorithm of the real transmission channel was implemented. It is one of the many algorithms used for increasing the transmission speed in the given frequency bandwidth while maintaining a low BER (Bit Error Rate) value. Assuming a constant wireless channel over the complete transmission duration since the transmitter and receiver are stationary, the wireless channel can be modeled as a simple linear FIR filter [5]. Thus, in our case, the channel is modeled as an L-length FIR filter and denoted as $\{h_\ell\}_{\ell=0}^{L-1}$. Let s_k and r_k be a transmitted and received symbol, for $k = 1, ..., N_d$, respectively. Then

$$r_k = \sum_{\ell=0}^{L-1} h_\ell s_{k-\ell} + n_k, \tag{3}$$

where n_k is AWGN.

While modeling the wireless channel the *delay spread* and *coherence bandwidth* are the parameters of interest. Delay spread is a measure of the time delay between the 'first' and 'last' multipath signal at the receiver, and the coherence bandwidth can be approximated as the inverse of delay spread [5]. These parameters play a vital role in deciding the symbol time and, consequently the occupied signal bandwidth. If the symbol time is smaller than the delay spread then the signal will experience ISI (Inter Symbol Interference) at the receiver, which is also known as frequency-selective fading [5]. To utilize the high bandwidth, measures are taken to avoid the ISI in (3). In this case, there is no ISI, and the filter in (3) reduces to a simple Dirac delta with the AWGN channel.

The goal is thus to estimate $\{h_\ell\}_{\ell=0}^{L-1}$, by using the fine sync symbols, i.e. pilot symbols p_k for $k = 1, ..., N_p$.

Let $f : \mathbb{C}^L \longrightarrow \mathbb{R}$ be the mapping such that $f(h) = \sum_{n=1}^{N_p} f_n(h)$ with f_n defined as

$$f_n = |(r_p * h)[n] - s_p[n]|^2 \quad \forall\, n = 1, ..., N_p, \tag{4}$$

where $*$ is linear convolution, s_p is the vector of transmitted pilots and r_p the received versions of those. The goal is to find the optimal inverse filter $h_{\text{inv}} \in \mathbb{C}^L$ that minimizes the norm.

Now, similar to the approach taken in the derivation of the Wiener filter and the RLS algorithm in [6], it must be noted that for any fixed n it can be written as:

$$(r_p * h)[n] = \sum_{\ell=0}^{L-1} h[\ell] r_p[n-\ell] = h^T r_p^{(n)}, \tag{5}$$

with the vector $r_p^{(n)} \in \mathbb{C}^L$

$$r_p^{(n)} := \{r_p[n], r_p[n-1], ..., r_p[n-L+1]\}$$

which can be calculated easily in MATLAB for each n.

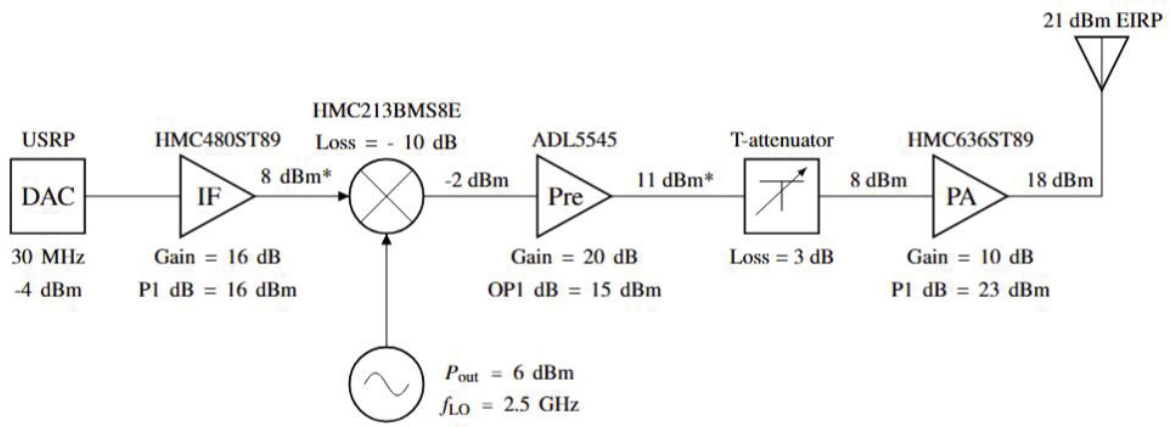


Fig. 5. The transmitter block diagram, with the components and values from the link budget. The powers indicated with * is in backoff state.

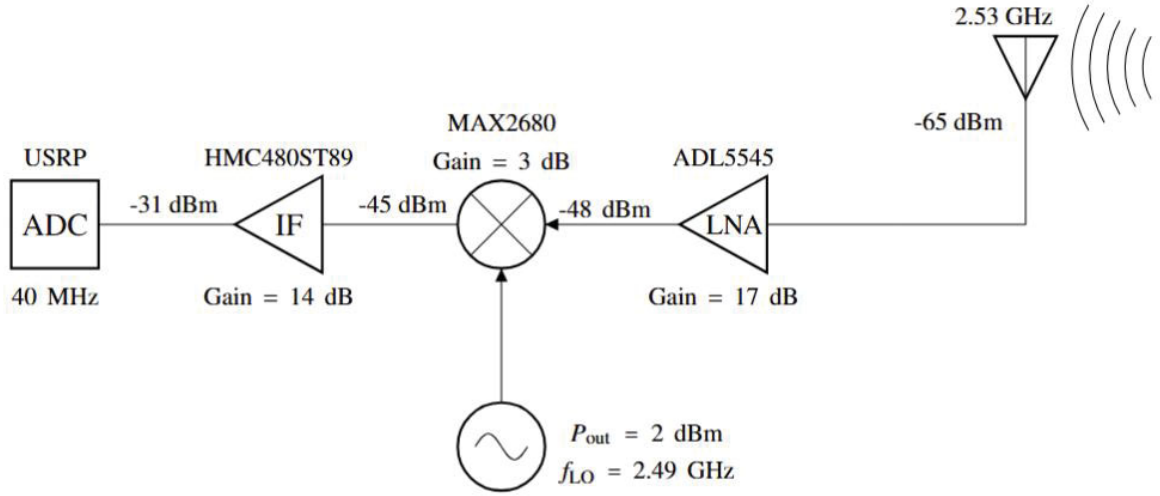


Fig. 6. The transmitter block diagram, with the components and values from the link budget. The powers indicated with * is in backoff state.

Standard algorithms for estimating the inverse filter in the literature include the RLS and LMS algorithms [6]. In this research work, a gradient descent (GD) algorithm is implemented, from a numerical perspective. In [4], mathematical results are given, showing that optimization routines like GD can be generalized to complex functions with appropriate definitions of the gradient.

In further detail, [4] argues that $\partial_h |h|^2 = \partial_h h^H h = h^H$ and shows that the gradient should be evaluated as $\nabla_h = (\partial_h)^H$ and that this indeed points in the direction of steepest ascent/descent. In this work, following above-mentioned rules, it yields:

$$\partial_h f_n = \partial_h \left\{ \left| h^T r_p^{(n)} - s_p[n] \right|^2 \right\} = \left(h^T r_p^{(n)} - s_p \right)^H r_p^{(n)} \quad (6)$$

and

$$\nabla_h f = \sum_{n=1}^{N_p} \nabla_h f_n = \sum_{n=1}^{N_p} \left(\left(h^T r_p^{(n)} - s_p[n] \right)^H r_p^{(n)} \right)^H . \quad (7)$$

This is an iterative algorithm, and a necessary number of iterations are required to be carried out,

$$h_{m+1} \longleftarrow h_m - \lambda \nabla_h f(h_m) \quad (8)$$

where m is the current iteration and λ is some step size.

The GD algorithm estimates the transmission channel accurately in simulations, as shown in figures 4a and 4b, where a Rician fading channel [5] is simulated using Matlab. Upon careful analysis, it can be stated that the estimated channel is usually a few dBs more than the actual channel, and that is due to scaling. Similarly, there also exists inaccuracies in phase but that does not affect the equalized constellation and the same was validated in simulations and the real testing.

III. Verification and Validation

In order to demonstrate the performance of the communication link in the LOS/BLOS 100m link with high spectral efficiency, an indoor hardware setup is assembled. Ettus N210 SDR is used for baseband signal generation/reception along with custom hardware in conjunction with a COTS local oscillator source for RF conditioning and up/down conversion at both the transmitter and receiver side. The omni-directional monopole Wifi antennas were used for signal transmission and reception. Custom RF Front End is designed to fulfill the link budget requirements thus starting with the required signal-to-noise ratio at the receiver keeping in consideration the USRP output power as well as antenna gain. Thus, using BER vs E_b/N_0 curves from [5], for a modulation scheme of 16-QAM with the BER of 10^{-5} and 2 MHz bandwidth, the required E_b/N_0 is computed around 13.5 dB. The spectral efficiency of 16-QAM (assuming the symbol time $T_s = 1/B$ with B the bandwidth) is found to be $10\log_{10}(4) \approx 6$ dB, which leads to the required SNR of approximately 19.5 dB. Ettus USRP N210 generates limited output power and an IF around 50MHz. Therefore, for up/down conversion (to/from 2.4GHz, a typical frequency for IOT applications) and to fulfill the link budget requirements, customized RF front-end PCBs for both transmitter and receiver were designed and fabricated.

IV. Hardware design

The block diagrams of the transmitter and receiver are shown in Figures 5 and 6. This section includes details about the hardware design, EM simulations, fabrication, and assembly of the transmitter and receiver Front-End PCBs.

A. Transmitter design

The transmitter front-end was designed using Keysight ADS software in microstrip technology using an FR-4 substrate. The transmitter up converts the IF signal from the USRP using an external LO source signal. The transmitter includes multiple components including an IF amplifier, a passive mixer for up-conversion, and two RF amplifiers with a 3 dB T-attenuator between them to avoid oscillations due to active components. The transmitter setup utilizes a VERT-2450 Wifi antenna to radiate the signal.

The transmitter design consists of the application circuits of the individual components with the remaining area covered in the ground via holes for heat dissipation generated by the active components. Additionally, two DC source connections are kept on the board to provide the required supply voltages to the devices. The fabricated transmitter PCB is illustrated in Figure 7. An LO amplifier along with the bypass path was also designed to amplify the external LO signal (if required). However, in this testing, we used an external LO source with sufficient amplitude and hence the bypass path was used. The antenna utilized in the link testing was categorized using a VNA for matching and its gain was computed (with over-the-air testing) to be around 3 dB which leads to a 21 dBm of

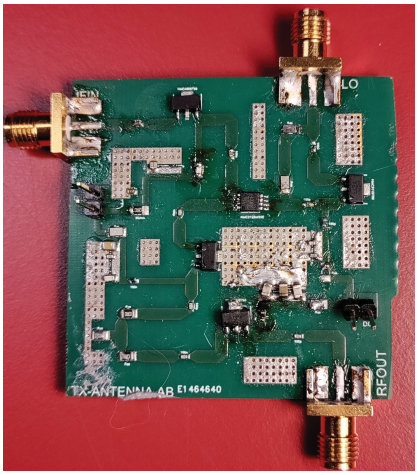


Fig. 7. Fabricated transmitter PCB used in link testing

Effective Isotropically Radiated Power (EIRP). To avoid non-linearities in the mixer and power amplifier, the transmitter is operated in the 3dB backoff region to ensure linear operation and low inter-modulation products.

After the transmitter fabrication and components stuffing, its performance was evaluated using a signal generator and spectrum analyzer. The overall output power of the transmitter was calculated to be around 20 dBm.

B. Receiver design

Similar to the transmitter, the receiver front-end PCB was designed. After the fabrication, the components were stuffed according to the design requirements, and lab testing was carried out for performance validation using PMEs. The block diagram of the receiver is depicted in figure 6. Initially, perceived design included two Low-Noise Amplifiers, one IF amplifier, and one LO amplifier to achieve the desired gain with a few dBs additional margin in the SNR requirements. Additionally, during the design phase, multiple bypass paths were also incorporated into the PCB layout. Unlike the transmitter, an active mixer MAX2680 is added to the receiver design to fulfill the gain and SNR requirements according to link-budget computations. The overall gain accomplished via receiver front-end PCB was around 34 dB The finalized and fabricated receiver hardware is shown in Figure 8.

V. Results

In this section, the overall results from the link demonstration are presented and discussed. To increase the directivity of the antenna and subsequently, the SNR of the receiver, a crude reflector made out of tin foil was placed behind the receiving antenna. To establish the efficacy of the designed link, the classic image of Lena was transmitted, obtained from the image processing literature [2]. The image was converted to gray-scale and scaled to 40×40 pixels, i.e. 12800 bits, to fit in one frame structure. The default parameters used

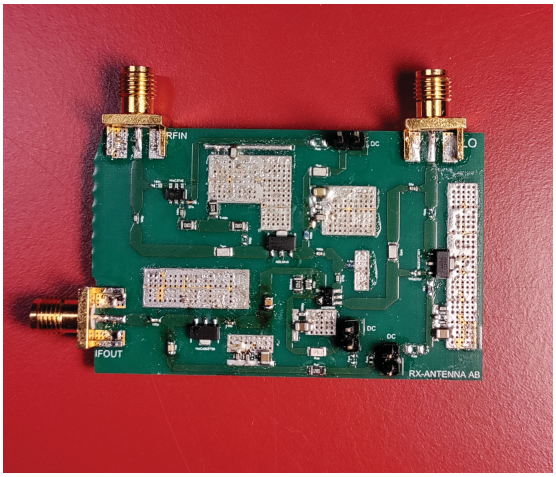
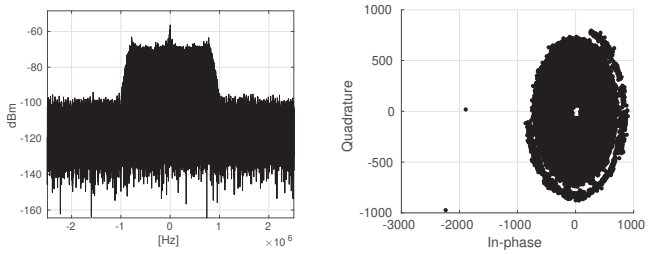


Fig. 8. Fabricated receiver PCB used in link testing.



(a) Power spectrum. (b) Without corrections.

Fig. 9. Results from using 16 QAM, 2 MHz bandwidth, at distance 100 meter.

during the testing are summarized in Table I. The 'base-case' bandwidth is chosen to be $s_{\text{bw}} \approx 1.95$ MHz, corresponding to $N_{\text{int}} = N_{\text{ups}} = 8$, since the delay spread indoors is normally around 50 ns and hence a coherence bandwidth of $B_{\text{coh}} = 20$ MHz [5]. We thus choose $s_{\text{bw}} \approx 2$ MHz << 20 MHz, as a 'base-case' to avoid ISI due to frequency-selective fading. Multiple tests were performed, and below, three of these tests are presented and elaborated. In Figure 10, the constellation diagrams after corrections, with and without equalization, are shown.

TABLE I. DEFAULT PARAMETERS USED.

Parameter	Value
Sample rate of USRP	100 MHz
Interpolation factor of USRP, N_{int}	8
IF frequency for DUC in tx USRP	30 MHz
IF frequency for DDC in rx USRP	40 MHz
Roll off factor β of RRC pulse	0.25
Upsampling factor (no. of samples per symbol), N_{ups}	8
Number of symbols, N_{sym}, in each tx frame	4096
Number of symbols allocated as pilots/fine sync, N_{p}	512
Number of coarse sync symbols	2×64
Number of actual data symbols, N_d	3584
Modulation for data symbols, M_d	16QAM
Modulation for pilot symbols, M_p	16QAM
Signal bandwidth, s_{bw}	1.95 MHz
Number of taps L in inverse channel filter	32

A. 100 meters, 16 QAM, and 2 MHz bandwidth

The default parameters provided in Table I were used in this test. The transmitter and receiver were placed 100 meters apart in a corridor. The chosen corridor was slightly curved so there was no immediate LOS. The received power spectrum is displayed in Figure 9a, where it can be seen that the SNR is roughly 32 dB. In Figure 9b, the received constellation before any corrections are displayed for comparison, and in Figures 10a and 10d the constellation diagram before and after equalization are displayed. Finally, in Figure 12 the estimation of the wireless channel is displayed. Here, it is seen that the impulse response resembles a Dirac delta and hence there was negligible multi-path effect. This is also evident from the constellation diagram before and after equalization. The BER in this transmission was found to be 0, and it was achieved consistently during multiple transmissions.

B. 100 meters, 64 QAM and 2 MHz bandwidth

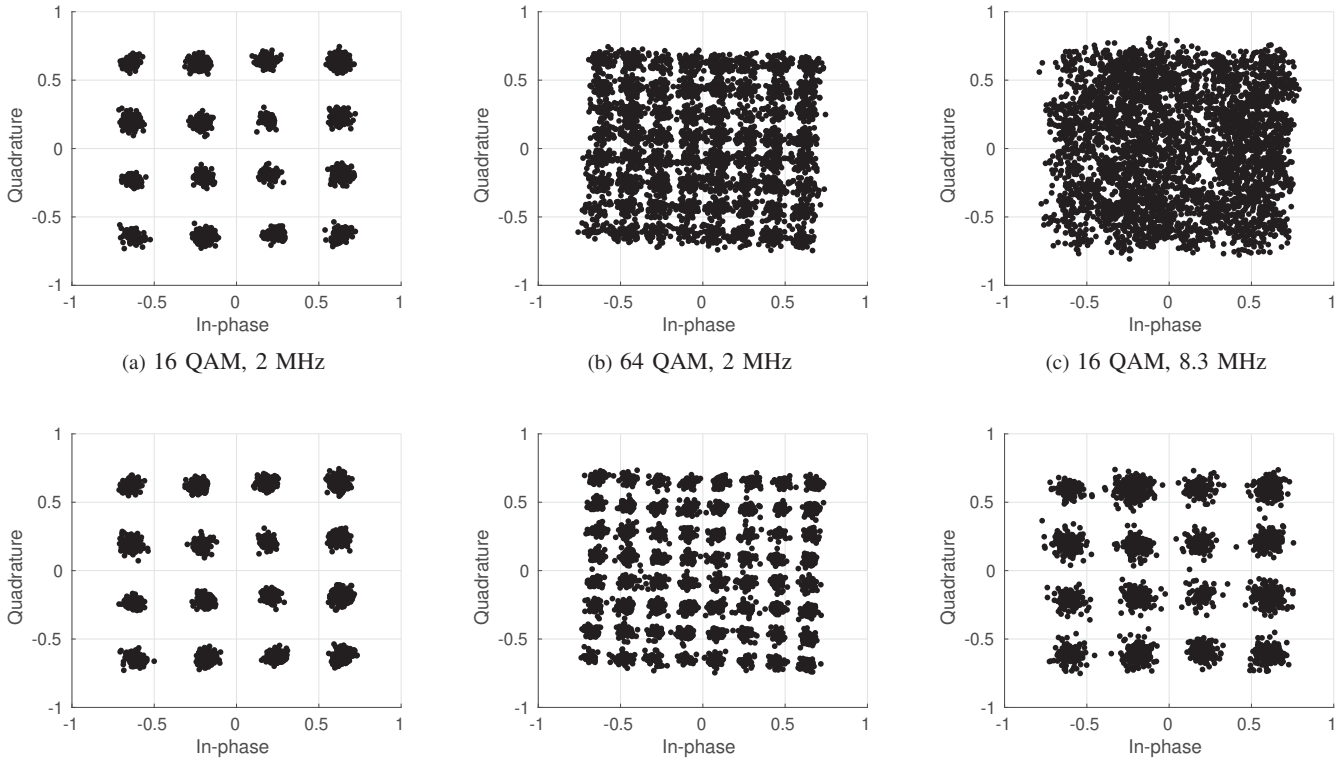
Here the same setup is used as before but the modulation order is increased to 64 QAM. In Figures 10b and 10e, the constellation diagrams before and after equalization are presented and in Figure 11 the results of the impairment corrections are illustrated.. The BER turns out to be approximately equal to 0.0047. The impact of the equalization algorithm is more appreciable in this scenario as it significantly improves the constellation diagram. In this case, the bandwidth was 2 MHz and the reason the equalizer makes such improvements for this bandwidth is due to the delay spread of the corridor not being 50 ns (as initially assumed), since it was a very wide and open area with a high ceiling. Hence, the delayed signals travel farther before reaching the receiver, making the delay spread, most likely, larger than 50 ns, and thus decreasing the coherence bandwidth. Therefore, 2 MHz might be relatively wideband, in that respect, and therefore more fading was observed due to randomness.

C. 100 meters, 16 QAM and 8.3 MHz bandwidth

For this scenario, the modulation order is reduced to 16 QAM and $N_{\text{int}} = 5$, and $N_{\text{ups}} = 3$, are used which leads to the bandwidth of approximately 8.3 MHz. Now by observing significant fading in the Figures 10c and 10f, it can be concluded that the equalizer is essential in removing the ISI. This can be further correlated from Figure 12 where the impulse response shows some significant taps around the main tap.

VI. Conclusion

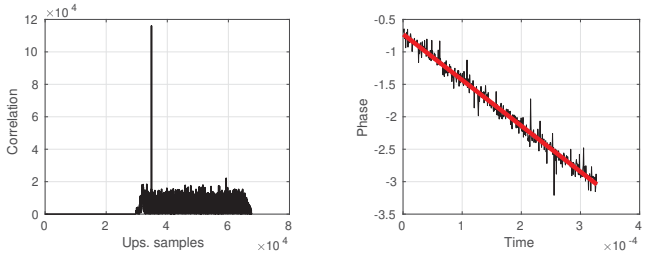
A single-carrier, simplex, wireless communication system is designed and demonstrated in this research work. The research work encompasses the complete design and development of the hardware and software. For the hardware part, it includes the detailed designs and fabrication of the transmitter and receiver front-end PCBs. For the software



(a) 16 QAM, 2 MHz (b) 64 QAM, 2 MHz (c) 16 QAM, 8.3 MHz

(d) 16 QAM, 2 MHz (e) 64 QAM, 2 MHz (f) 16 QAM, 8.3 MHz

Fig. 10. The resulting constellation diagrams after corrections. The top and bottom row is before and after equalization, respectively.



(a) Fine timing estimation. (b) Fine CFO estimation.

Fig. 11. The fine timing and CFO estimation for 64 QAM, 2 MHz bandwidth, at 100 meters.

part, the overall signal processing scheme is implemented in MATLAB and the signal is received via Ettus USRP N210. Multiple impairment correction techniques are implemented to mitigate hardware / wireless channel issues such as carrier frequency offset, phase offset, and timing offset. Additionally, to achieve high spectral efficiency and to mitigate multipath fading effects, an equalization algorithm is also implemented. The bit-error rates (BER) achieved for the cases of 16 and 64 QAM with 2 MHz bandwidth were 0 and 0.00477, respectively. The SNR achieved in both cases was 32 dB.

For capability demonstration, multiple tests were performed on the system to evaluate its performance. The optimum results were achieved with 16 QAM modulation, 8.3MHz bandwidth having a BER of 0.00086. In this case, the data rate can be calculated as:

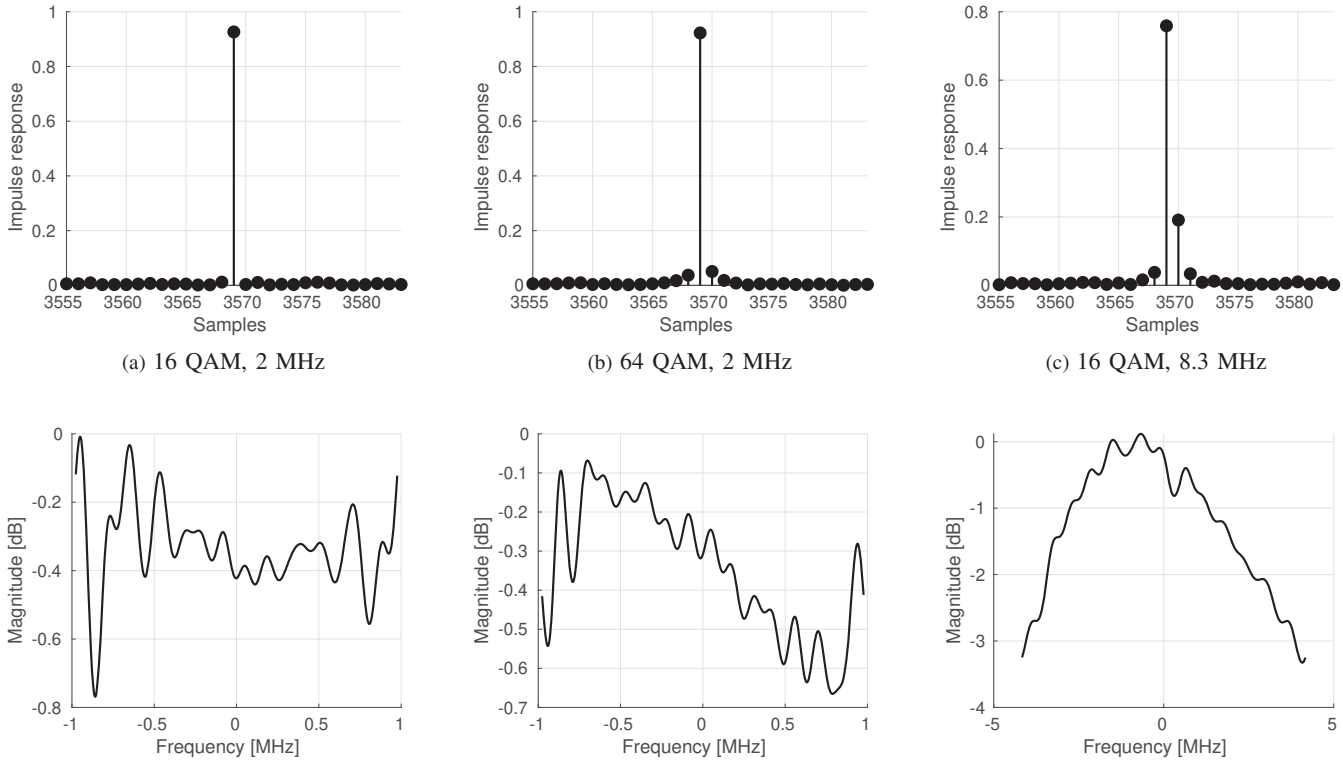
$$R_s = \frac{1}{T_s} = \frac{s_{\text{bw}}}{1+\beta} = \frac{8.33\text{MHz}}{1.25} \approx 6.66 \times 10^6 \text{ Symbols/s} \quad (9)$$

and we therefore have achieved, with uncoded 16-QAM, a bitrate of

$$R_b = 4R_s \approx 26.66 \text{ Mbit/s}. \quad (10)$$

with a spectral efficiency of 3.2 bps/Hz.

In the optimal case, the fine sync symbols were disabled. However, these symbols reduce the spectral efficiency by a factor of $512/4096 = 0.1250$, or 0.1562 including the coarse sync symbols. Thus, in this scenario, the bit rate becomes approximately 22.50 Mbit/s. Spectral efficiency can be further enhanced by optimizing the number of pilots, and reusing the coarse sync symbols as fine sync symbols, instead of discarding them. To conclude, this work demonstrates the complete software and hardware of a communication system designed for IOT systems. The system provides a satisfactory performance with an acceptable BER, and a relatively high BW, which validates the performance of the equalization algorithm, and the customized hardware.



(a) 16 QAM, 2 MHz (b) 64 QAM, 2 MHz (c) 16 QAM, 8.3 MHz

(d) 16 QAM, 2 MHz (e) 64 QAM, 2 MHz (f) 16 QAM, 8.3 MHz

Fig. 12. The estimation of the impulse and frequency response of the wireless channel for the different cases

REFERENCES

[1] Zhe Huang, Weidong Wang and Yinghai Zhang, "Design and implementation of cognitive radio hardware platform based on USRP," IET International Conference on Communication Technology and Application (ICCTA 2011), Beijing, 2011, pp. 160-164, doi: 10.1049/cp.2011.0651. keywords: cognitive radio;USRP;GNU Radio;reserved channel.

[2] Rafael C. Gonzalez and Richard E. Woods. Digital image processing. Pearson, 2018. ISBN: 1292223049.

[3] Anderson John B. Digital Transmission Engineering. 2nd ed. IEEE Series on Digital & Mobile Communication. Wiley-IEEE Press, 2005. ISBN: 9780471694649.

[4] Ken Kreutz-Delgado. The Complex Gradient Operator and the CR-Calculus. 2009. DOI: 10.48550/ARXIV.0906. 4835. URL: https://arxiv.org/abs/0906.4835.

[5] Andrea Goldsmith. Wireless Communications. Cambridge University Press, 2005. ISBN: 9780521837163.

[6] Bernard Mulgrew, Peter Grant, and John Thompson. Digital Signal Processing, Concepts and Applications, 2'nd edition. Palgrave Macmillian, 2005. ISBN: 978-0-333-96356-2.

[7] T.M. Schmidl and D.C. Cox. "Robust frequency and timing synchronization for OFDM". In: IEEE Transactions on Communications 45.12 (1997), pp. 1613–1621. DOI:10.1109/26.650240.

[8] USRP Hardware Driver and USRP Manual. Accessed: 22'th December, 2022. URL: https : / / files . ettus . com / manual/page usrp2.html.