Wireless Monsters - Wireless Link Project

Sveinn Finnsson, Pavel Gueorguiev, Nikolaos Kollatos, Tomas Pettersson, Ezhil Arasan Vinayagam

***Abstract*—Given a set of criteria we introduce and calculate a link budget to meet those criteria. We design and implement a wireless link at RF, matching the criteria of the link budget. We introduce the hardware and software components necessary to achieve successful transmission along with our implementation of those components. We conclude that the system we designed and implemented works as expected, but has room for future improvements.**

I. Introduction

The aim of this project is to design and implement a functioning wireless link capable of transmitting simplex over a distance of 100 m in the "Canyon". This project consists of both a hardware and software part and those two parts are each split into a transmitter and receiver part. In this report we start by evaluating link budget constraints and hardware design in section II. In section III we discuss the software implementation of the transmitter and receiver before presenting our results in section IV and providing concluding remarks in section V.

II. Hardware Design

A. Basic System Parameters

The basic figures used for initial design of the system are shown in the list below:

$$\begin{cases} f_c = 2.4GHz \\ B = 500kHz \\ R = 100m \\ 16-QAM modulation \end{cases} \tag{1}$$

B. Receiver Architecture and Transmitter design

Receiver architecture is chosen to be heterodyne, which means that an IF (intermediate frequency) is used for up- and down-conversion and in order to simplify the hardware design the USPRs do the IQ-mixing and go down to the baseband signal. The hardware is limited to only the strictly necessary components, with the addition of a pre-selector band bass filter on the receiver side.

Please refer to Figures 1 and 2.

C. Link Budget

Path loss calculations are based on Friis formula for free space propagation modified to account for reflections, since Canyon is assumed to be a highly reflective environment. Note most of the equations and the propagation loss constant are gathered from the book "David M. Pozar, Microwave and RF Design of Wireless Systems. Amhest MA, 2000.". The exponent is set to 2.2. The propagation losses are extracted from Friis formula in order to estimate said loss. The minus sign is just to correct for loss being equal to negative gain.

$$L_{Friis} = -10\log\frac{\lambda^2}{(4\pi)^2 R^{2.2}} \tag{2}$$

Final expression for minimum required signal level at the receiver antenna is found according to (3), where $\left(\frac{S_0}{N_0}\right)_{min} = 20dB$ is required for 16-QAM modulation at $BER = 10^{-6}$.

$$S_{i,min} = 10\log kT_0 + 10\log B + F\big[dB\big] + \left(\frac{S_0}{N_0}\right)_{min} \tag{3}$$

If we assume $T_A = T_o$ than we can find the minimum allowed signal at the input of the receiver according to the aforementioned equation giving $S_{i,min} = -92.8[dBm]$. Note that the noise figure F was assumed to be 10dB. As well the output of the USRP at the transmitter is measured by the oscilloscope to be -3dBm. The minimum detectable input at the receiver side is also measured by the oscilloscope and it is: -90dBm. These along with $S_{i,min}$, the frequency band selected, the assumed bandwidth of our signal, and a suggested carrier frequency of the URSP (IF frequency for all practical purposes) allows us to select the components. After which we will go back and check our assumptions on the noise figure, correct for and reselect components if required. The link budget results are shown in table I. The chosen components based on the link budget calculations are seen in table II

USRP output (transmitter)	$P_{USRP} = -3dBm$
Gain (transmitter)	$G_t = 10dB$
Transmitted power (entering the antenna)	$P_t = 7dBm$
Path Loss	$L_{path} = 84dB$
Received Power (input of receiver)	$P_r = -71dBm$
Receiver's Noise Figure	$F_{rec} = 4.6dB$
System's Noise Temperature (receiver and antenna)	$T_{sys} = 836.4K$
Noise Power at the output of receiver	$N_{out} = -84dBm$
Signal Power after receiver (input of USRP)	$P_{out} = -47.2dBm$
Gain (receiver)	$G_r = 23.8dB$
Minimum SNR for 16-QAM	$SNR_{min} = 20dB$

TABLE I
Table for link budget calculations

When components are selected, we calculate the cascaded noise figure in the receiver using equation 4 and checking if our receiver is smaller or bigger than the assumed $F_{rec} = 10dB$

$$F = F_1 + \frac{F_2 - 1}{G_1} + \frac{F_3 - 1}{G_1 G_2} + ... + \frac{F_6 - 1}{G_1 G_2 G_3 G_4 G_5 G_6} \tag{4}$$

This yields a total noise figure of $F = 4.26$. The total gain is also determined to be 9.66dB. From this $T_e = 836[K]$ is calculated. For $T_A = T_p$ we assume that the background temperature is equal to the physical temperature of the antenna, such that:

Component	Name	Gain/Loss	Noise Figure/Noise temp.
Upconversion mixer	HMC272AMS8E	$L_{mixer_1} = 9dB$	$F_{mixer_1} = 9dB$
PA	HMC414MS8G	$G_{PA} = 20dB$	$F_{PA} = 7dB$
Antennas	VERT2450	$G_{antenna} = 3dB$	$T_{antenna} = 300K$
Pre-selector BPF	2450BP15E0100E	$L_{Pre-selector} = 2dB$	$F_{Pre-selector} = 2dB$
LNA	MGA-86563	$G_{LNA} = 21.8dB$	$F_{LNA} = 1.6dB$
Downconversion mixer	HMC272AMS8E	$L_{mixer_2} = 9dB$	$F_{mixer_2} = 9dB$
LPF	MEM2012S50R0T001	$L_{LPF} = 1dB$	$F_{LPF} = 1dB$
LO and IF Amplifiers	HMC311ST89E	$GIF = 15dB$	$F_{IF} = 4.5dB$

TABLE II
TABLE FOR THE CHOSEN COMPONENTS

$$N_o = k(T_A + T_e)BG \tag{5}$$

Giving a noise value of $N_0 = -84dBm$. As well $S_i = -90dBm$, so our initial calculation was of $S_{i,min}$ was in the right ballpark. Next we can backpropagate our signal power using Friis equation and adding a margin for error of 20dB, and taking into account our antenna gains (3dBi each). We now have the minimum transmit power before the antenna $P_{t,min} = 5.2dBm$. We can now verify that the initial assumptions are satisfied, and the components are able to output the required gain for the P_t and $P_{i,USRP}$. For a detailed calculation please refer to section E. Design Verification.

D. Receiver and Transmitter Topologies

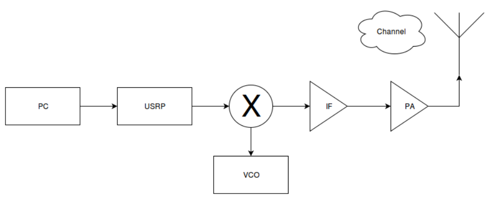

Fig. 1. Transmitter topology

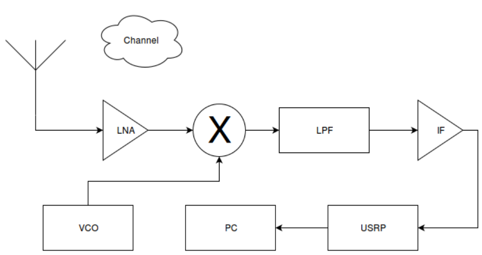

Fig. 2. Receiver topology

E. Design Verification

Beginning at the USRP, at the transmitter, we will move through each component in the transmitter and add up the gains, then onto the channel, receiver and ending with the USRP. Note connectors were not displayed in the transmitter and receiver diagrams for brevity.

Measured power at the output of the USRP.

$$P_{out} = -3[dBm] \tag{6}$$

Let P_{Tx} be the power before the transmitter antenna, let $L_.$ be a loss of the respective component and let $G_.$ be the gain of the respective component.

$$\begin{aligned} P_{Tx} = P_{out} - L_{connector} - L_{mixer} + G_{PA} - L_{connector} \\ P_{Tx} = -3 - 0.5 - 9 + 20 - 0.5 \\ \Rightarrow P_{Tx} = 7[dBm] \end{aligned} \tag{7}$$

Now we go over the transmitter antenna followed by the channel, which is modelled by Friis path loss equation at 100 metres, and we also incorporate the receiver antenna gain. Lambda in our case is 0.125m for the 2.4 GHz frequency we use, $P_{tx_linear} = 5.01[mW]$, $G_{rx_lin} = G_{tx_lin} = 2$ and the path loss coefficient is purposely overestimated to be 2.2, a coefficient experimentally determined for a retail store.

$$\begin{aligned} P_{Rx} = \frac{G_{tx}G_{rx}\lambda^2}{(4\pi)^2 R^{2.2}} \\ P_{Rx} = 7.89E - 11[W] \\ P_{Rx} = -71[dBm] \end{aligned} \tag{8}$$

Let $P_{USRP,i}$ be the input power at the USRP.

$$\begin{aligned} P_{USRP,i} = P_{Rx} - L_{connector} + G_{LNA} - L_{mixer} - L_{LPF} \\ + G_{IF} - L_{connector} \\ P_{USRP,i} = -71 - 0.5 + 21.8 - 9 - 1 + 15 - 0.5 \\ \Rightarrow P_{USRP,i} = -47.2[dBm] \end{aligned} \tag{9}$$

The approximate measured sensitivity at the input of the USRP was around $S_{i,min} = -90[dBm]$. The noise level at the receiver is calculated as follows.

$$\begin{aligned} N_o = k(T_A + T_e)BG_{Rx,lin} \\ N_o = k(836.4)(500000)(302) \\ N_o = 1.73E - 12[W] \\ \Rightarrow N_o = -84[dBm] \end{aligned} \tag{10}$$

We see that our received signal at the USRP is both above the measured sensitivity of the USRP and the calculated noise level, by the required amount for our modulation scheme. This concludes our link budget verification.

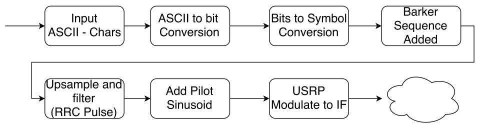

Fig. 3. Transmitter software diagram

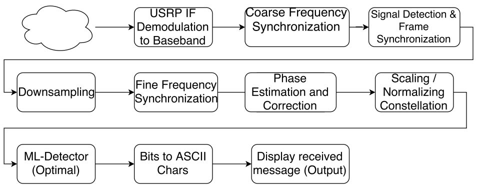

Fig. 4. Receiver software diagram

F. Components chosen

Low pass filter was purchased from Digikey. Model name: MEM2012S50R0T001. [4]

Bandpass filter: Originally purchased from Farnell, the bandpass filter was removed after the assembly of the chip. This was due to the faulty connection that was made in the soldering oven, and the decision was made to be removed all together, gaining approximately two decibels of gain. Model name: 2450BP15E0100E. [5]

III. Software

The software for the transmitter and receiver was developed over several iterations and tested and simulated in Matlab for quick testing before being ported to LabView for further testing and final implementation. This worked well as Matlab and LabView's MathScript have similar syntax and share a vast amount of functions and function names. Thorough explanation of the software implementations can be found in the sub sections below.

A. Labview

Our software was implemented in LabView due to LabView's support for the USRPs. The software made use of the built in USRP blocks for controlling the USRPs, but used mathscript blocks for other parts of the implementation.The mathscript blocks runs matlab code, but with a few discrepancies compared to how Matlab runs it.

B. Transmitter

The software on the transmission side is relatively straight forward, a block diagram of the transmitter can be seen in figure 3. The transmitter starts by accepting ASCII characters and converting them to a bit-stream. After the bit conversion the bit-stream is mapped to symbols, matching the chosen modulation. The transmitter and receiver implementation offers 4-QAM (QPSK) and 16-QAM modulation. A 13-symbol Barker sequence [1] is repeated twice, a total of 26 symbols, and added to the front of the message. The Barker sequence is created by using the symbol in the upper right corner of the constellation as representation of +1 and the symbol in the lower left corner as representation of -1. This Barker sequence is used for signal detection and frame synchronization on the receiver side and is chosen due to its properties of high correlation with itself and low correlation with noise. Next the symbols are up-sampled and filtered using a Root-Raised Cosine filter [2] with a roll-off factor of $\alpha = 0.8$ such that we have a transmittable signal. Having filtered the signal a sinusoid with a known offset is added to the signal, this will be used for coarse frequency synchronization at the receiver side. The signal is then sent to the USRP which mixes the signal and moves it to an intermediate frequency and transmits it to our hardware.

Due to time constraints and ease of implementation we choose to use fixed length messages where each 4-QAM (QPSK) message was of 800 symbols length and 16-QAM messages of 760 symbols length plus a 26 symbol addition of preambles. The reason a 16-QAM message has a shorter message length of 760 symbols compared to 4-QAM is that the 16-QAM implementation needs 40 preamble symbols for fine frequency correction and phase correction at the receiver side. If the input message from a user did not make use of all symbols, the message padded with random bits to achive the correct message length.

C. Receiver

A block-diagram of the software design for the receiver can be found in figure 4. At the receiver side the USRP moves the signal down from our intermediate frequency to baseband. In order for us to be able to detect the signal we have to start by doing a coarse frequency synchronization, with that done we remove the pilot sinusoid by running the received signal through a low-pass filter, we can then correlate the received signal with a our known Barker preamble. The correlation will have a distinct peak where the Barker sequence correlates, this peak is then used to detect the signal and find the correct sampling point of the frame. Next we employ our fine frequency synchronization that corrects the remaining frequency offset. With the frequency offset taken care of, some phase offset, so the next step is phase correction. With the phase correction done we need to scale the received signal to match the reference constellation before sending it to our Maximum-Likelihood detector. After the ML-detector the bits are converted to ASCII-characters before being displayed as a message on the receiver side. The reception is now complete. In the following sub chapters we expand on the concepts used above.

D. Coarse Frequency Synchronization

The coarse frequency synchronization is used to move the signal to within the limits that the fine frequency synchronization can handle, but also within the limits of the Barker sequence detection such that a frame can be found. The coarse frequency synchronization FFTs the signal and finds the peak of the pilot sinusoid that was added at the transmitter side and calculates the distance of it form its correct location. The coarse frequency synchronization is limited by the resolution

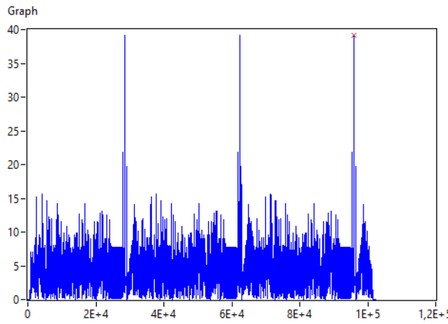

Fig. 5. Correlation of signal to Barker sequence

of the FFT, in our case $F_s/N = 4\times10^6/2^{20} \approx 3.84$ Hz, where F_s is the sampling frequency used and N is the number of FFT points.

E. Signal Detection and Frame Synchronization

The signal detection and frame synchronization is vital for further processing at the receiver side. Both are done at the same time by correlating the known Barker sequence added at the transmitter side to the received signal at the receiver. As can be seen in figure 5 a successful correlation results in a distinct peak where the signal correlates. If the peak is above a certain threshold we can conclude that we have received a signal and we continue. Another property of the correlation peak is that it is located exactly where the Barker sequence of the received signal and our reference Barker sequence intersect. Knowing this and the length of the used Barker sequence we can calculate the start of the frame and the optimal sampling point from the index of the peak. The optimal sampling point is then used for downsampling.

F. Fine frequency Synchronization

Due to the inadequate resolution of the coarse frequency synchronization we must add a fine frequency synchronization. As we are doing our processing ”offline” we can make use of the property that all points of a 4-QAM constellation, or in the case of 16-QAM the outermost corner points, when raised to the power of 4 map to a single point on the unit circle. This stems from the fact that the corner points can be written as $\left[e^{\pi/4}, e^{3\pi/4}, e^{-3\pi/4}, e^{-\pi/4}\right]$, when raised to the power of four we get $\left[e^{4\pi/4}, e^{12\pi/4}, e^{-12\pi/4}, e^{-4\pi/4}\right] = e^{pi}$. However, if there is a frequency offset, this offset will cause the points to drift away from e^{pi} when raised to the power of four. By calculating the phase drift from the ideal point, un-wraping the phase and doing a linear fit, finding the best line through the points and dividing the slope of that line by 4, we get the frequency offset per sample. The main benefit of using this method for 4-QAM is that we can use all the received symbols minimizing the effects of noise on the frequency correction. For 16-QAM we only use the Barker sequence and preamble of 40 symbols.

G. Phase Estimation and Correction

The phase estimation and correction is used after the fine frequency correction to rotate the constellation such that it matches our reference constellation. The phase estimation is done in a similar way as the fine frequency offset estimation above. We calculate the phase offset of the known points in the Barker sequence. By doing a linear fit through these point we should get a straight line with a slope of 0 and an intersection point on the Y-axis. The value of this intersection point is our phase offset estimation used for phase correction.

H. Scaling of received symbols

In order for our Maximum-Likelihood detector to work, the received symbols must be scaled to match those of our reference constellation. For a 4-QAM message, we take the average amplitude of all received symbols and divide our received symbols by that value. For 16-QAM we take the average amplitude of the Barker sequence and the 40 symbols in the pre-amble, we then divide all received symbols by that number, resulting in a constellation that matches the reference one.

I. Maximum-likelihood decoder

We use a maximum-likelihood decoder[3] to decode our symbols into bits. This decoder is used as it is an optimal decoder for our case where we have no extra information of the signal received. There are several ways to implement an ML decoder, but ours was implemented such that for each received symbol, we took the absolute value of the received symbol value minus the value of every single point in the reference constellation. The point with the lowest value is then the point closes to our received symbol and used for decoding.

IV. Results

The link was designed to have a high enough SNR for 16-QAM to work. The hardware group encountered unforeseen difficulties as the circuitry did not supply the expected amount of gain, but improved the hardware to fulfill the link budget. The hardware provided gain enough for 16-QAM on the link demonstration, while simultaneously keeping the noise low enough.

A. Hardware performance

The top level hardware process was: link budget, component selection, component mapping, circuit organization, assembly and troubleshooting. There was issues with each.

1) Problems: Some of the parameters required in our link budget were tricky to get, like the USRP sensitivity and USRP output power. When we assumed to have satisfactory values the link budget was carried out and components were selected. The circuit layout had significant design errors and/or misprints.

B. Mistakes, Fallout, and Perseverance

- Missing Power: Receiver and transmitter were both missing a copious amount of power. The faults were in incomplete solder joints making capacitive connections. The solution was to seek and resolder suspicious joints.
- Missing Vias: The missing vias occurred either due to design fault during folding or a manufacturing fault. The fix was to drill the board, feeding a conductive wire and soldering both sides - connecting the top to the ground layer. This was especially problematic underneath the power amplifier. The solder joint created a raised mound that needed to be carefully flattened by a dremel.
- RF Grounding Issue: Touching of ground plates caused variations of signal level, indicative of an RF ground issue. Manual via holes were created to ensure a solid RF ground to the ground plane.
- Oscillations: Oscillations in the receiver were observed on the oscilloscope during testing of the unit. An inductor of higher value was placed to increase the impedance of the RF signal, and prevent RF from leaking into the amplifier biasing circuits.
- Power Amplifier Heating: The power amplifier was overheating during the operation of the transmitter. After failure it needed to be replaced. The biasing resistors were increased in value to provide less voltage to the amplifier. As well as a heat sink was installed below the bed of the amplifier. After replacement the unit behaved according to specification with temperature never reaching above 60 degrees centigrade (before it was climbing above 85). . . .

1) Fatal errors:

- Mismatched and Right Angle Lines: the transmitter had design flaws with mismatched RF lines, and low impedance DC lines. Nothing was done about the mismatch, a consequence was the possibility of reduced gain. Transmitter surprisingly didn't have any oscillation or grounding issues.

2) Regrets and Future improvements: If we were in the first day of this course the circuit design would be modified. The components chosen were satisfactory but the circuit layout left something to be desired. First it appears from literature that curved lines perform better than ones at right angles for non-digital waveforms. There are certain RF rules of thumb that we later discovered that should have been followed. A lot of space was actually wasted by our relatively simple hardware, less than 50 squared cm could have been used.

Even though there were many mistakes in our design the final product turned out extremely successful. With that in mind there are certain things that we would not change. These include the components we used and the overall simplicity of the hardware setup we used. From the stand point of a more productive link (more data, faster transfer, ...) the demand is on the software to be able to process the information in real time. For the current hardware setup a transfer of about 0.4M bits/second was shown to be possible. With introduction of 64-QAM this can be further increased. Increases in reception distance would eventually require more transmit and receive gain, a few more hardware chips and antennas could be used for MIMO setup and again software can be implemented for optimal combining (MRC for example).

C. Software performance

The software was built to handle both QPSK and 16-QAM, both were demonstrated on the link demonstration at full distance without LoS. During the testing we used a symbol rate of $R_s = 100000 sym/s$ and a roll-off factor of $\alpha = 0.8$ for our root-raised cosine filter. We made use of an intermediate frequency at 4 MHz and had a sampling frequency of $f_s = 20$ MHz. For these parameters the system performed flawlessly during the demonstration. The eyediagram from the demonstration can be seen in figure IV-D3 and the constellation plot can be seen in 7. From these two figures we see that our hardware performs very well and achieves a good SNR even at full distance. The eyediagram in figure IV-D3 also shows us that we manage to find the optimal sampling point, resulting in a wide open "eye". Furthermore the frequency and phase correction do their part and the received constellation matches the ideal reference constellation, resulting in successful decoding of the received message.

D. Software limitations and future improvements

After several iterations the software performed well, however it still contains some limitations and has a lot of room for future improvements.

1) Pilot sinusoid detection: Detection of pilot sinusoid is not very robust. For example if there is a large peak in the transmitted message, that peak can fool the software into thinking that it is the sinusoid, resulting in a failed coarse frequency synchronization. This could be improved by either increasing the amplitude of the sinusoid to consistently be higher than that of the signal. Another possible solution would be to check if there is any power in the frequencies surrounding the peak, as the sinusoid peak should have a fixed frequency offset from the main signal, little to no power should be in the surrounding frequencies. Also, one could exploit the fact that the sinusoid is of known frequency, and hence the peak search for the pilot sinusoid could be limited to some region of the spectrum.

2) Fixed message length: Messages sent and received are all of fixed message length, this results in a lot of wasted power during transmission. The fixed message length is however easier to implement and due to time constraints we choose to make use of it. For a variable message length a end of message sequence would have to be added to each message.

3) Modulation: Our software implementation only supports 4-QAM and 16-QAM. Having demonstrated our link it became clear that our link probably could have handled a higher order modulation. A future improvement would therefore be to add 64-QAM to the software.

V. Conclusion

The desired task was to design a simplex link at RF. Both hardware (HW) and software (SW) parts were to be created

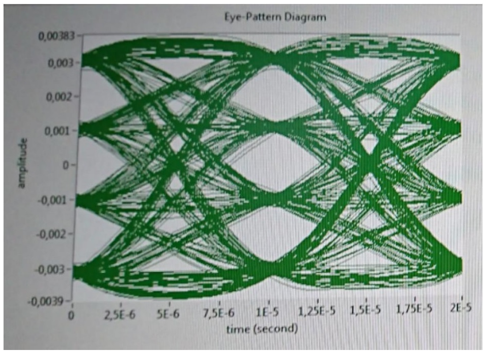

Fig. 6. Eye diagram for 16-QAM at long distance, no LoS.

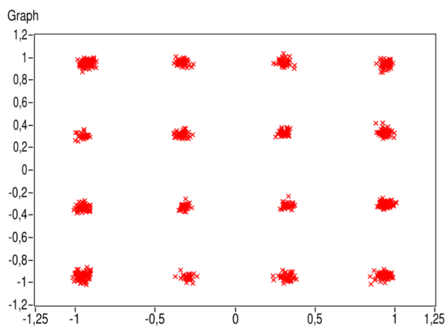

Fig. 7. Constellation diagram for 16-QAM at long distance, no LoS.

from scratch, which means that link budget calculations, circuit layouts, codes for implementing modulations and data mapping were needed to be carried out. Although various difficulties were faced at almost all stages of design, the link worked well at the end.

Link budget calculations were a bit optimistic and corresponded to ideal cases. In reality, the achieved gain of Tx and Rx were different from the expected ones but that did not seem to create any problems.

Software part was a bit tricky because of the nature of LabVIEW and the fact that software group was not very familiar with it. However, it worked more than fine thanks to the persistence and hard work of software group.

In conclusion, when designing such systems it is always necessary to add some margin in order to account for additional losses or misprints in circuits. The two groups cooperated well for carrying out this task, learned a lot from the whole process and the results payed off the effort. Finally, there is always the chance of improvement and many things can be done to achieve higher modulation schemes and transfer various types of data.

Contributions to Report

Sveinn Finnsson

Software section and the subsections (A,B,C,D,E,F,G,H,I) relating to the Software section. Software part of the Results section. Part of the Introduction.

Nikolaos Kollatos

Proof-reading, corrections at some parts, part of Hardware section, Conclusions section

Tomas Pettersson

Early report draft, Software section,Parts of Hardware design, Parts of Results, proof reading.

Ezhil Arasan Vinayagam

Initial abstract draft, correction in few parts of Hardware design, double check overall report

Pavel Gueorguiev

Hardware flow charts, hardware sections D and E. Results sections A, B.

References

[1] [Wikipedia] (January 2017). Barker Code [Online]. Available: https://en.wikipedia.org/wiki/Barker_code

[2] [Wikipedia] (January 2017). Root-Raised-Cosine filter [Online]. Available: https://en.wikipedia.org/wiki/Root-raised-cosine_filter

[3] [Wikipedia] (January 2017). Maximum Likelihood Decoding [Online]. Available: https://en.wikipedia.org/wiki/Decoding_methods#Maximum_likelihood_decoding
[4] [Digikey] (January 2017). Low Pass Filter - MEM2012S50R0T001. Available: http://www.digikey.se/product-detail/en/tdk-corporation/MEM2012S50R0T001/445-6201-1-ND/2465528
[5] [Farnell] (January 2017). Band Pass Filter - 2450BP15E0100E. Available: http://se.farnell.com/johanson-technology/2450bp15e0100e/filter-bandpass-2-4ghz/dp/2148528