A radio communication system for voice transmission using LabVIEW and USRP2 in the ISM band at 2.4 GHz

Børge Myran
NTNU
borgemy@stud.ntnu.no

Baptiste Robin David Veuillez-Mainard
NTNU
baptiste.veuillez–mainard@telecom-bretagne.eu

Stian Fossan
NTNU
stianfos@stud.ntnu.no

***Abstract*—A software designed radio system for broadcasting speech using USRP2 and LabView. The system should work in the ISM band, 2,4GHz and should not interfer with the wireless lan. Our system broadcast speech up to 3kHz, uses frequency of 2.4495GHz to avoid interfering with WLAN, and a bandwitdh of 50kHz.**

1. Introduction

1.1. Project description

The aim of this project is to design and implement a radio communication system for voice transmission in the ISM band at 2.4GHz and/or 5.8GHz using USRP2 from Ettus Research and LabView from National Instruments. The project description was given by:

- The system shall not affect NTNU's WLAN communication systems
- Range at least covering the room with the equipment
- Broadcasting of speech (one way communication)
- Use LabView to program the radio TX and RX
- Use USPR2 hardware for the radio TX and RX
  - One USRP unit for transmission (TX)
  - Another USRP unit for reception (RX)

Any other specification was not given, and was decided by ourself.

1.2. Specification

The specification we decided for our system design: Sound quality was decided to be low, enough for speech broadcast to allow low complexity of the system, and easier implementation. Delay was mostly ignored since it is not an important parameter in this broadcasting system, but we tried to stay below 5seconds. The power of the system we tried to keep as low as possible to get a stable working system. Our goal was to have a stable system with good interference tolerance.

- Bit depth: 8bit
- Samling frequency: 6kHz
- Frequency: 2.4495GHz
- Bandwidth: 200kHz
- Bitrate: $48\mathrm{kbs}^{-1}$

- Sound quality: low
- Transmission range: 10m
- Delay: 5second max
- Complexity: low
- Power: good
- Interference tolerance: good

1.3. Software defined radio

A radio communication system in which components such as modulators, demodulators, mixers, filters, detectors, etc. are implemented in software. Attempts to push the software as close to the antenna as possible. Digital communications and signal processing are at the heart of SDR.

1.4. LabView

The LabVIEW Communications System Design Suite 1.1 from National Instruments is a graphical programming language which uses icons instead of text to create programs. Programs created using LabVIEW are called VIs, Virtual Instruments, VI consists of two sections; front panel which is the user-interface and block diagram, which is the graphical source code.

1.5. USRP2

USRP from Ettus Research, a National Instrument company. Universal Software Radio Peripheral is a programmable hardware that allows computers to function as software radios. USRP performs high speed signal processing operations such as: ADC / DAC, digital up/down conversion and filtering. Higher level signal processing such as coding/decoding, modulation/demodulation takes place in the host processor.

2. Methods

2.1. System overview

How the system is built up is shown in figure 1, on the transmitter side the sound is taken from the microphone, converted into digital 8bits per sample, then bit correction coding is used on the data before the data is added with guardbits and syncbits to form a packet, then the packet is modulated and sendt to the USRP2 for broadcasting. On the recieverside the signal is read from the USRP2, then demodulated, syncronized to know where the data in the packets are located. Then the data is decoded and sent to the soundcard to be played. The frequency chosen for the system is 2.4495GHz to avoid interfering with the WLAN as seen in figure 2, then we are in the guardband between channel 6 and 11.

2.2. Link budget

Equation 1 is used to create the link budget, each of the parameters is described in table 1. Calculations for the link budget taken from equation 1 is listed in table 2. The link budget tells us if we have enough power to broadcast and recieve the data we want to in our system. According to our calculations taken in all the parameteres we have for our system, we should have a margin of 39,2dB.

$$\frac{E_b}{N_0} = P_t G_t \left(\frac{\lambda}{4\pi d}\right)^2 \frac{1}{k} \frac{G_r}{T_r} \frac{1}{R} G_k \frac{1}{L_0} \tag{1}$$

Parameter	*Description*
E_b	Energy per bit
N_0	Noise power density
P_t	Transmitted power
G_t	TX antenna gain
λ	wavelength
d	distance TX-RX
k	Boltzmann's constant
G_r	RX antenna gain
T_r	RX noise temperature
R	Data rate
G_k	Coding gain
L_0	All other losses

TABLE 1. LINK BUDGET-EQUATION1 DESCRIPTION

pic/system2-eps-converted-to.pdf

Figure 1. System structure

2.3. Sound

Sound is sampled from the soundcard at a sampling rate of 6kHz, and each sample from the soundcard has a resolution of 16bits. The samples are then added with a positive correction value of 0.4 and multiplied with a factor of 100 to avoid having negative bit values in the system that can cause errors. Each sample is then compressed with µ-law to 8bit resolution.

Transmitter	
Transmit power	15dBm
Antenna gain	3dBi
Antenna diameter	0,154m
Near field	$\approx$ 0,39m
Carrier frequency	2.4495GHz
EIRP	18dB
Propagation	
Distance	10m
Lightspeed	299792458$\mathrm{m\,s^{-1}}$
Wavelength	0,122364269m
Free space loss	-60,18dB
Indoor loss [3]p.284	-68,78dB
Reciever	
Noise temperature 627,1K	27,97dB K
Reciever G/T	-24,97dB/K
Boltzmann's constant	228,59dB W/K/Hz
G/Tk	203,8dB Hz/W
Recieved C/N_0	152,84dB Hz
Signal	
Datarate 48kbs^{-1}	-46,81dB
E_b/N_0	106,03dB
Requirements	
BER requirements for 10^{-4}	12dB
Coding gain	0dB
Margin	**96,43**dB

TABLE 2. LINK BUDGET

2.4. µ-law

µ-law encoding is used because speech has a wide dynamic range. It ensures higher bit depth resolution for low amplitudes and lower bit depth resolution for higher amplitudes, which is good for speech since it contains more low level amplitudes than high level amplitudes. [5]

2.5. LDPC

Low-density parity-check (LDPC) code is a linear error correcting code, a method of transmitting a message over a noisy transmission channel. [8]

2.6. Interleaving

Interleaving is frequently used in digital communication and storage systems to improve the performance of forward error correcting codes. Errors typically occur in bursts rather than independently. If the number of errors within a code word exceeds the error-correcting code's capability, it fails to recover the original code word. Interleaving ameliorates this problem by shuffling source symbols across several code words, thereby creating a more uniform distribution of errors. [9]

2.7. Modulation

When transmitting speech through a wireless channel, the digital speech signal needs to be modulated to an analog carrier. A general periodic signal have 3 degrees of freedom to which data can be modulated; frequency, amplitude and phase. In the world of digital modulation, a designer can choose from Amplitude Shift Keying (ASK), Frequency Shift Keying (FSK) or Phase Shift Keying (PSK). PSK and ASK can also be combined to create Quadrature Amplitude Modulation (QAM). Here, keying indicates a discrete number of states for the modulated signal. For this project, the PSK scheme was chosen. PSK requires less bandwidth than FSK, the constant power level consumes less dynamic power compared to ASK. PSK also allows for more efficient (Class-C) Power Amplifiers compared to ASK. The same applies to QAM. The simplest form of ASK, PSK and FSK is its binary form, which allows only for 2 states, yielding 1 bit per symbol. Comparing the *Complementary Gauss Error Function*, P_e of the three schemes yield results as shown in table 2.7.

Modulation Scheme	*Error Function,* P_e
BPSK	$\frac{1}{2}erfc(\sqrt{\frac{E_b}{N_0}})$
BASK	$\frac{1}{2}erfc(\sqrt{\frac{E_b}{4N_0}})$
BFSK	$\frac{1}{2}erfc(\sqrt{\frac{E_b}{2N_0}})$

Examening the results, it's readily seen that for a given SNR level, the BPSK yields the lowest error. Examening BPSK, it has 2 possible descrete values depending on the signal phase. However, in most digital modulation schemes it's common to use two orthogonal carriers, the In-Phase, I component and the Quadrature, Q component. Thus by using the BPSK scheme for both carriers, twice the data rate per Hz can be achieved with the same error probability. This is called Quadrature PSK and was the used modulation scheme for this project, also known as 4QAM, having 2 bits per symbol. Increasing the amount of symbols per bit makes the system more sensitive to phase changes, adding requirements for good antennas, amplifiers and other hardware. Thus for this project, QPSK gives a sufficient data rate to transfer speech, and no higher modulation scheme was needed. Though for a high performance system, higher order QAM should be used to improve the data rate pr Hz.

2.8. Padding

2.9. Interchannel Interference

pic/wlanchan.png

Figure 2. WLAN Channels [4]

2.9.1. Subsubsection Heading Here. Subsubsection text here. Subsubsection text here. Subsubsection text here. Subsubsection text here.

2.10. Packet structure

The packet struckture is shown in figure 3 it consists of 40 guard bits in the front and end of each package, these are used to initiate the filter on the reciever side and to seperate between packets, 32 syncbits is used for location of the databits in each package, each package has a total av 128 samples with 8bit in each sound sample giving a total of 1024 databits. Giving a total packagesize of 1136bits.

pic/packet-eps-converted-to.pdf

Figure 3. Packet structure

- Packetsize: 1136bits
  - 40 guardbits in each end
  - 32 sync bits
  - 1024 databits

2.11. Interleaving and Error correction

The 1024 bits of sound samples in each packet can be encoded using Low Density Parity Check and interleaved before being put in the packet. The interleaving should always be used when the LDPC coding is enabled, as it makes for a smoother distribution of the errors which allows the error correction to actually work. Errors typically happen in burst, with most packets being error-free, and sometimes one packet being entirely wrong. Without interleaving, LDPC code would be useless both on the error-free packets, which do not require correction, and on the wrong packets, which have to many errors to be corrected. Interleaving distribute the errors of one wrong packet on many LDPC codewords, so that each one can be corrected.

The interleaved packets have 10 additional bits to index them and be able to de-interleave, thus up to $2^{10} = 1024$ packets can be interleaved at a time. However this increases the delay, as there are only 500 packets per second, so it is not recommended to maximize it. The number of packets interleaved should be adjusted to achieve number of error per packet under the threshold of the LDPC code used. LDPC codes typically have a waterfall effect : it corrects errors almost perfectly under a threshold error-rate, but above this threshold performances degrades very quickly. This threshold is higher if the number of parity checks is increased (lower code rate), but in counterpart the length of the coded message increases, as well as the time necessary to decode.

Measurements hint toward around 200 errors per packet of length 1024+2*40+12 (?) LDPC codes with code rate of 1/2 can correct around 50 errors out of 1024 bits, (? verify the values) In this case we would want to interleave at least 200/50=4 packets to get under error rate threshold. If the receiver has limited computing power, a higher code rate can be used. In this case the threshold will be lower, and the a higher number of packets should be interleaved to get under it, increasing the delay.

LDPC coding works by computing the parity of given subsets of the message bits, and to add this as redundancy in the codeword. To decode, the parity of those subsets are computed against and compared to the redundancy bits. If it does not match, errors occurred during the transmission. Each bit is taken into account in several parity check, and if the majority of those are wrong, it is flipped. This process is supposed to be iterated to achieve a good decoding. The exact code used is determined by a pair of binary matrix, one for encoding (generator G) and one for decoding (check H). The check matrix is semi-randomly generated to verify certain properties, such as the number of bits per check and the number of checks per bit. It is also made to be sparse, ie filled with mostly zeros and few ones. This allows a faster decoding, which is the bottleneck in term of complexity. The generator matrix is calculated from the check matrix. It is possible to optimize the generator matrix for faster encoding, but it wasn't necessary in this case.

3. Results

Parameter	*Description*
Bandwidth	50kHz
P_r @ 2m	-44dBm
P_r across the room	-64dBm
SNR @ 2m	43dB
SNR across the room	27dB
Datarate	48kbs^{-1}

TABLE 3. RESULTS

We have measured a datarate of 48kbps, and we can see that this gives us a bandwitdh of 50kHz, which is linked with the theoretical bandwitdh of a QPSK system, shown in fig. 6.27 at p.302 [1]

pic/noise_notransmit.PNG

Figure 4. NSpectrum without any transmitter turned on

pic/noise_transmit_2m_20dbgain.PNG

Figure 5. Spectrum with transmitter 2m away

pic/noise_transmit_room20dbgain.PNG

Figure 6. Spectrum with transmitter on across room

pic/psk.png

Figure 7. BER with PSK [6]

4. Discussion

With an USRP turned on for a long time, there seems to be something with the internal buffer that creates difficulties to retrieve a sync, so it needs to be turned off. With the ITU indoor model, calculations shows that the loss in dB gives us a theoretical range of about 21km and then the link budget margin is 0.

5. Conclusion

We managed to fulfill the specification in the given task to transfer speech over the length of the room.

References

[1] Theodore S. Rappaport *Wireless Communications, Principles and practice*, 2nd ed. Upper Saddle River, USA: Prentice Hall PTR,

2015.

[2] David M. Pozar *Microwave and RF Design of Wireless Systems*, USA, John Wiley & Sons, Inc. 2000.

[3] Simon Saunders, Alejandro Aragón Zavala *Antennas and Propagation for Wireless Communication Systems*, 2nd ed. USA, John Wiley & Sons, Inc. 2007.

[4] Wikipedia IEEE802.11 *https://en.wikipedia.org/wiki/IEEE_802.11*

[5] Wikipedia µ-law algorithm *https://en.wikipedia.org/wiki/M-law_algorithm*

[6] Wikipedia Phase Shift Keying *https://en.wikipedia.org/wiki/Phase-shift_keying*

[7] Lecture material TTT4145 *https://innsida.ntnu.no*

[8] Wikipedia Low-density parity-check code *https://en.wikipedia.org/wiki/Low-density_parity-check_code*

[9] Wikipedia Forward error correction *https://en.wikipedia.org/wiki/Forward_error_correction#Interleaving*