Cooperative Localization using Ultrasound Ranging

Oliver Peters, Christopher Durand

Semester Project

Institute for Theoretical Information Technology
RWTH Aachen

Supervisor: Dr. Johannes Schmitz

June 2018

Contents

1 Introduction **5**

1.1 Cooperative Localization 5

1.2 Competitions 5

1.2.1 Eurobot Competition 6

1.2.2 Microsoft Indoor Localization Competition 6

2 System Concept **9**

2.1 Sensor Nodes and Reference Beacons 9

2.2 Distance Measurement 10

2.3 Adaptions for Microsoft Indoor Localization Competition 11

3 Acoustic Design **13**

4 Ultrasound Ranging **15**

4.1 Signal Conditioning 15

4.2 Principles of Digital Ranging 15

4.3 Performance Optimization 16

4.4 Implementation 18

5 Localization Estimation **19**

5.1 Cooperative Error Function 19

5.2 Stability Improvements 19

6 Synchronization **21**

7 Communication **23**

8 How to Replicate **25**

8.1 Microcontroller Software 25

8.2 Matlab Control GUI 25

8.3 PCB 25

8.4 Mechanics 26

Chapter 1

Introduction

The aim of the presented project s to develop a sensor system to locate mobile robots and implement a cooperative localization algorithm where multiple mobile robots support each other to improve the quality of the location estimation.

1.1 Cooperative Localization

The most widely known position sensor system is probably GPS. Satellites are used as reference nodes to locate a GPS receiver. Line of sight contact to at least four satellites is necessary to successful compute the position. Buildings obstructing your view can prevent you from measuring your position using your mobile phone. However, a person just meters away may be able to see enough satellites. If the distance between between you and the other GPS receiver is known, then you can use the additional information to calculate your position. In a truly cooperative scenario, each one of two GPS receivers can only see three satellites. Each GPS-receiver can not calculate its position alone. If the two GPS-receivers combined have contact to five or more different sattelites, exchange their measurements and measure the distance between each other, both can calculate their position when cooperating.

In the robotics competition Eurobot, the position of the reference sensor nodes (comparable to the GPS satellites) is fixed. Therefore the achievable position accuracy is dependent on the robots location. By using other robots as additional reference sensors the number ot total reference sensors can be increased and the position accuracy is no longe strongly dependent on the location of a single robot.

1.2 Competitions

The presented system was designed to be used in mobile robots for the Eurobot competition. Here a position in two dimensions is sufficient to guide the robots.
The beacon system was later adapted to fit the Microsoft Indoor Localization Competition at the IPSN 2018, where a position estimation in three dimensions is necessary.

1.2.1 Eurobot Competition

In the Eurobot[1] competition mobile autonomous robots compete for points which are awarded for completing specific tasks. These particular tasks are changed every year. What remains the same, is

- a game lasts $90\,\mathrm{s}$
- the playing area measures $3\,\mathrm{m}$ by $2\,\mathrm{m}$
- two teams are playing simultaneously on the same playing area
- your robot is not allowed to touch the other teams robot, an active opponent detection must be implemented
- to guide your robot, four sensor nodes (beacons) can be placed at predefined positions around the the playing area
- you can place a sensor node on the other teams robot and vice versa

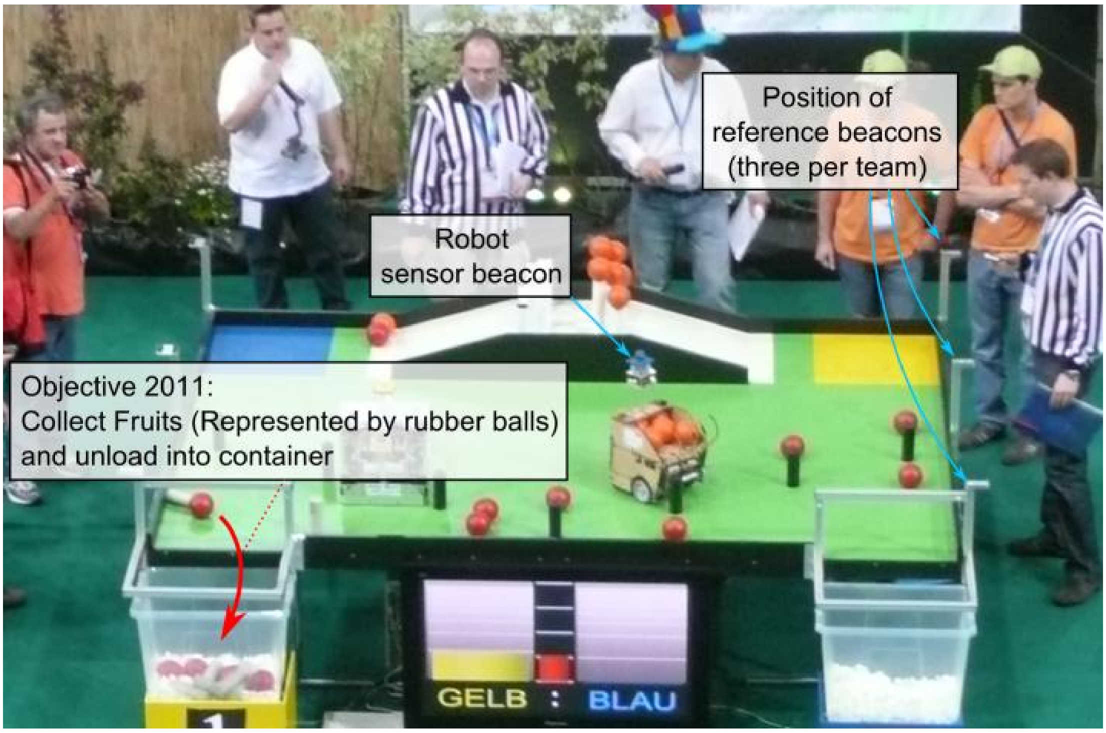

Figure 1.1: Eurobot 2011

1.2.2 Microsoft Indoor Localization Competition

In this competition the teams have to locate their mobile beacon inside a building. The space in which the beacon can be placed measures usually around $25\,\mathrm{m}$ by $25\,\mathrm{m}$ by $15\,\mathrm{m}$. You are allowed to place a specific number of reference beacons in the room. In the actual competition run, your beacon is placed on a backpack equipped with a 3D lidar which can locate itself and was the winner of a previous season. The

[1] www.eurobot.org

person carrying the backpack walks a predefined path in the building. Measurements from your beacon are compared with the lidar system. The team with the lowest average error wins. In 2018 the competition took place in the staircase of the Palácio da Bolsa in Portugal, providing an acoustically challenging environment.

Chapter 2

System Concept

This chapter describes the system concept of the localization system as designed for the eurobot.

The minimal system consists of

- one immobile master beacon
  - master clock, all other clocks in the system synchronize their clock to the master beacon
  - distributes clock reference via gold code modulated infrared bit stream
  - emits sinc-modulated ultrasound pulses
- at least two immobile slave beacons
  - synchronizes its clock to the master beacon
  - emits sinc-modulated ultrasound pulses
- up to four[1] mobile robot beacons
  - synchronizes its clock to the master beacon
  - records ultrasound and calculates distance to all ultrasound sources
  - calculates its own position
  - emits sinc-modulated ultrasound pulses for cooperative localization
  - rf communication module[2]

2.1 Sensor Nodes and Reference Beacons

The hardware of all the beacons (master, slave and robot) is based on the same design. Each beacon consists of the components displayed in figure 2.2

[1]The number of mobile robot beacons is limited by the available ultrasound channels. Currently seven channels are available. More channels are possible. Only three channels are needed if cooperation is not used, enabling two teams to use this system simultaneously. If cooperation is not used, then the mobile robot beacons are acoustically passive. Therefore, the number of mobile robots is only limited by the rf-communication.

[2]Either the rf communication or ultrasound-emitter can be omitted on the mobile robot beacon if cooperation is not used.

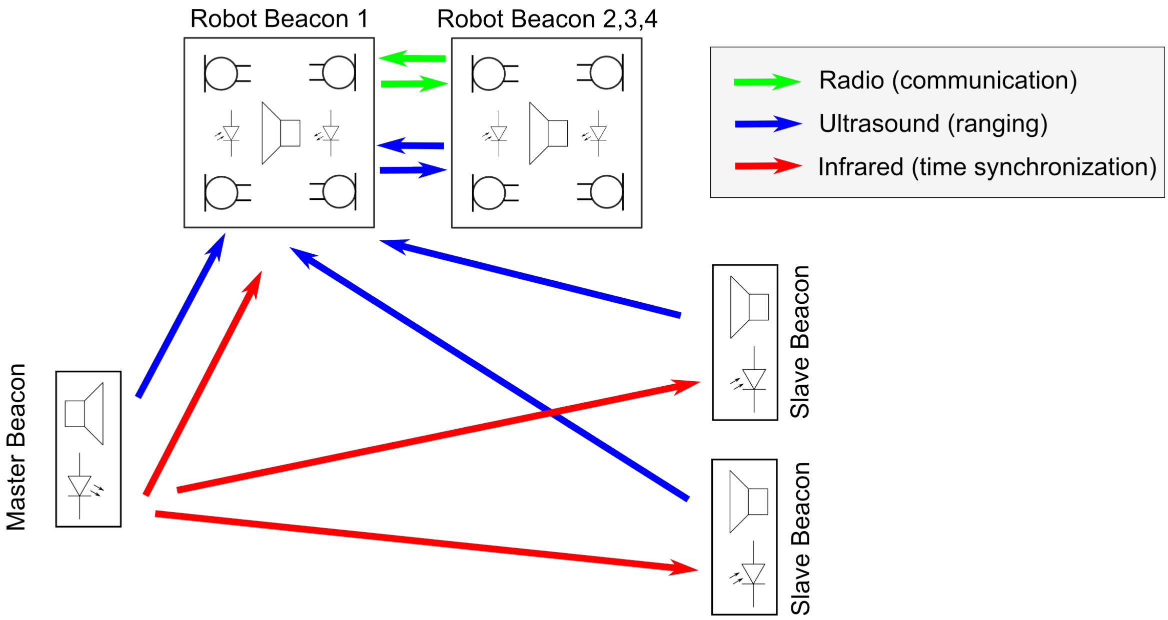

Figure 2.1: Signal paths, as observed by robot beacon 1

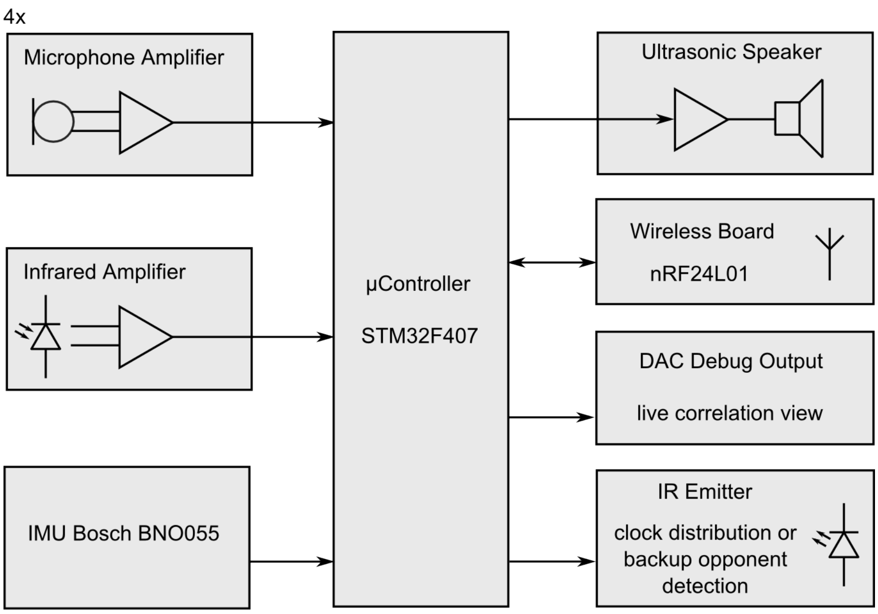

Figure 2.2: Components of every beacon

2.2 Distance Measurement

The distance from an ultrasound source to the receiving microphone is measured with the time-of-flight principle. During a mesurement, the sender emits a sinc[3] modulated pulse with is again modulated with a for this channel specific carrier frequency. If the microphone is placed directly next to the speaker, the waveforms of speaker and microphone would be identical. If the microphone is placed at a distance (d) from the speaker, the microphone signal is delayed by the time of flight (t_{tof}) induced by the speed of

[3] $sinc(x) = \frac{sin(x)}{x}$

sound (c_{sound}).

$$d = t_{tof} \cdot c_{sound} = t_{tof} \cdot 343\,\mathrm{m\,s^{-1}} \tag{2.1}$$

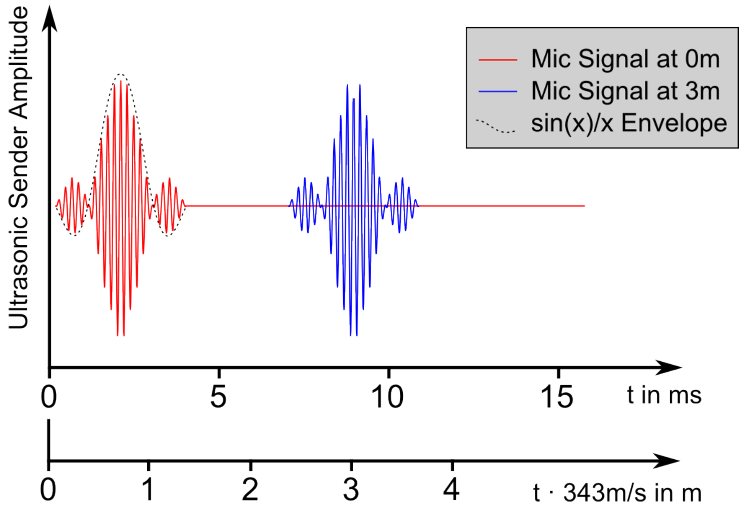

Figure 2.3: Ultrasound signal shape

The different ultrasound channels are separated in the frequency domain by changing the modulation carrier frequency. All ultrasound signals are emitted at the same time.

2.3 Adaptions for Microsoft Indoor Localization Competition

For the Indoor Localization Competition ranges up to $20\,\mathrm{m}$ must be measured. The range of the Eurobot system is $5\,\mathrm{m}$, which is achieved by sampling $2048\,\mathrm{samples}$ at $160\,\mathrm{kHz}$. The first problem was to fit the microphone samples in the memory. Because orientation is irrelevant in this competition, only one instead of four microphones were used. The second issue was the computation of the real fast fourier transformation (rfft). The highly optimized rfft functions of the cmsis dsp library were used. However, only rfft algorithms with up to $4096\,\mathrm{samples}$ are available[4]. To achieve a range of $20\,\mathrm{m}$ with $4096\,\mathrm{samples}$, the sample rate has to be lowered to $60\,\mathrm{kHz}$. This results in a bandwidth of $30\,\mathrm{kHz}$. Up to ten reference beacons were used, each sending a different ultrasound signal. The ten signals filled the spectrum from $19\,\mathrm{kHz}$ to $30\,\mathrm{kHz}$. The mobile beacon only listens and calculates the distance to all beacons. Note, that this arrangement (fixed senders and mobile receiver) does not utilize the available bandwidth very well. The used arrangement is optimal for Eurobot, where there is are more mobile self-locating beacons than fixed beacons.

[4]Limit is for 32 bit float computation. A 16 bit fixed comma implementation for $8192\,\mathrm{samples}$ is available. However, quantization noise can then be an additional issue.

Chapter 3

Acoustic Design

The audio channel comprising of speaker and microphone should have usable transmission in the frequency range from $20\,\mathrm{kHz}$ to $40\,\mathrm{kHz}$. The speakers on the fixed beacons ideally have a transmission cone with $120°$ horizontal and $45°$ vertivcal span. A lower vertical span emitts less audio power towards the ground. Waves which are reflected from the ground are the major component leading to multi-path-propagation.

The ribbon tweeter PTMini-6 from Dayton Audio can meet the requirements. The length of the membrane corresponds to minimal $\frac{60\,\mathrm{mm}*20\,\mathrm{kHz}}{343\,\mathrm{m\,s^{-1}}} = 3.5\lambda$. This gives the speaker a significant vertical directivity.

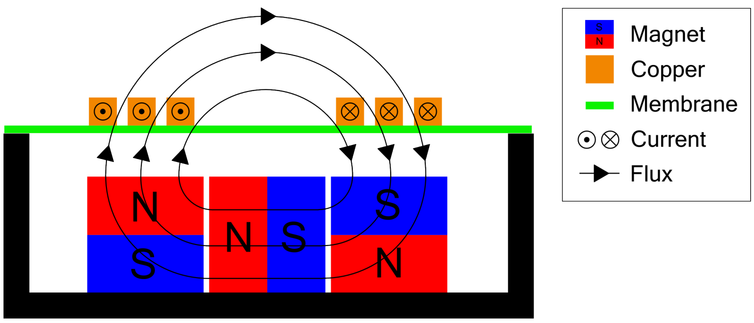

Figure 3.1: Possible implementation of a planar ribbon speaker

The microphone SPU0410LR5H-QB-7 from Knowles was recommended by the Knowles support and has proven in use. The SPU0pro410LR5H-QB-7 is a MEMS microphone in a very tiny $3\,\mathrm{mm}$ by $3\,\mathrm{mm}$ package. The small size enables the microphone to be sensitive and omnidirectional up to $80\,\mathrm{kHz}$.

Each mobile beacon has four microphones, one in each corner. An absorber between the microphones changes the directivity. With the absorber, each microphone has a line of sight to a maximum of two fixed beacons. This reduces inter channel crosstalk. Using four microphones enables the mobile beacon to estimate its orientation.

Figure 3.2: Speaker in fixed beacon

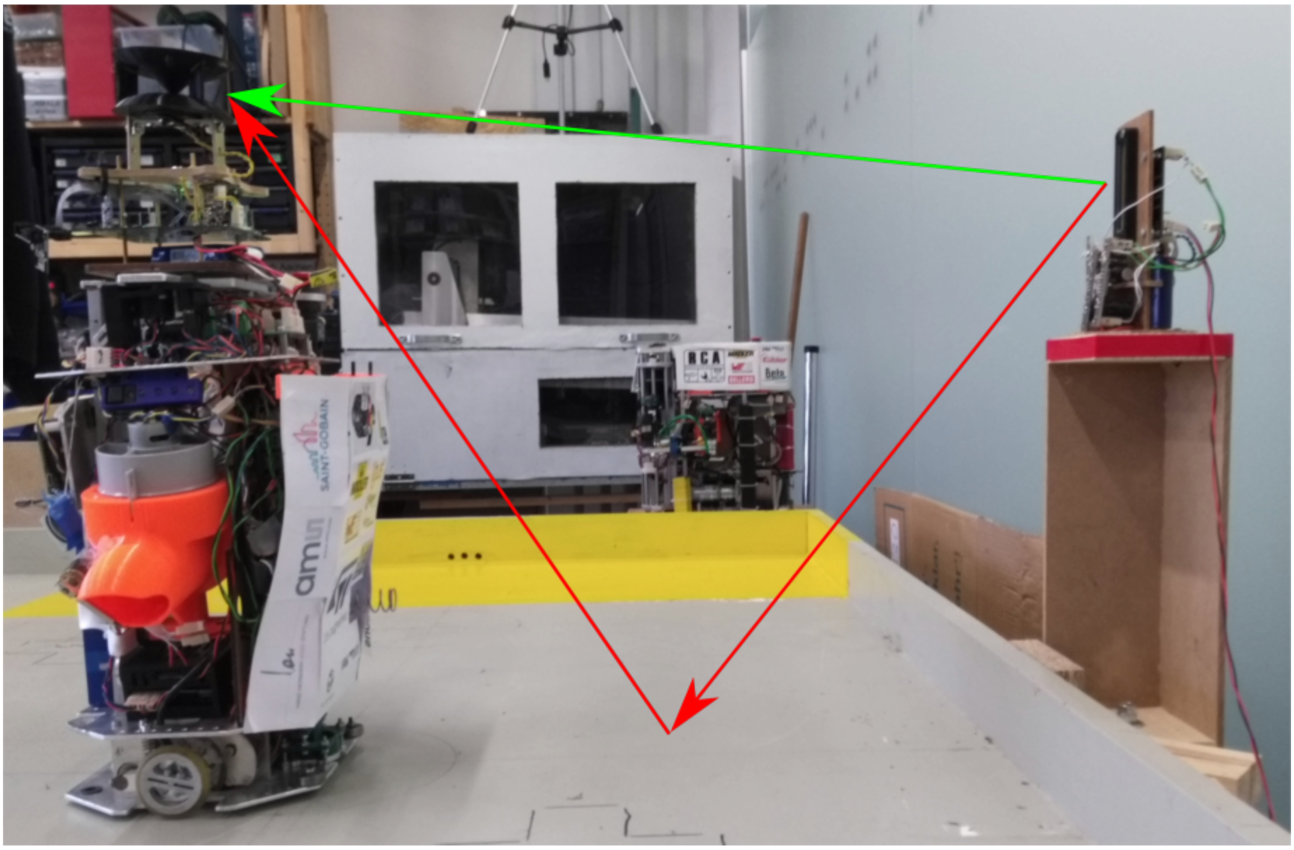

Figure 3.3: Multipath propagation due to ground reflection

Chapter 4

Ultrasound Ranging

4.1 Signal Conditioning

The used Microphone SPU0410LR5H-QB-7 has a bandwidth [1] from $100\,\mathrm{Hz}$ to $80\,\mathrm{kHz}$. The output is a single ended voltage. A first order RC-Lowpass is used to suppress the frequency range below 20kHz. The filtered signal is fed to a selectable gain amplifier. The selectable gains are 20 and unity. With the gain of 20 the complete range from $10\,\mathrm{cm}$ to $5\,\mathrm{m}$ can be covered. The unity gain can be used to prevent clipping in very loud environments.

Microphone and amplifier are located on a separate PCB including their own power supply and power supply filtering. The amplified signal is connected to the internal 12 bit ADC of the STM32F407 micrcontroller. The sample rate is $160\,\mathrm{kHz}$. In each measurement cycle, 2048 samples of every one of the four microphone channels are recorded. There are two sample buffers, one for recording and one for signal processing. Therefore, recording can start before the signal processing of the previous recording has finished.

4.2 Principles of Digital Ranging

To determine the distance between speaker and microphone, the time-shift between speaker signal and microphone signal is computed. Here, the autocorrelation is used to determine the shift in samples. The autocorrelation between two signals can be computed efficiently in the frequency domain by muliplying the respective spectra. Therefore, each microphone signal is transformed to the single sided spectrum using cmsis real fast fourier algorithm (rfft). Each microphone spectrum is then multiplied with the spectrum of each sent signal. For four microphones and seven sender, this accumulates to 21 complex multiplications of 2048 samples each. The cmsis dsp library also provides an optimized function for piecewise complex valued multiplication of two arrays (arm_cmplx_mult_cmplx_f32). After complex valued multiplication, each spectrum is transformed into the actual correlation by applying the inverse real fast fourier transformation (rifft). The position of the maximum of the correlation indicates the time shift. E. g. if the signals are shifted by 50 samples, the maximum of the correlation is also shifted to

[1] range with usable signal, not $3\,\mathrm{dB}$ bandwidth

sample nr 50. Because the sent signals are modulated, the correlation function is also oscillating. The absolute value and position of the maximum is close to the value of the next smaller local maximum. To prevent misidentifying the other maximum as the main maximum, the correlation function is demodulated, producing the envelope of the correlation function. In the envelope, the next smaller maximum is much smaller compared to the biggest maximum. This helps identifying the actual time shift.

The separation of the different ultrasound channels is a side effect of the correlation, but could also be done separately by selecting only certain frequency bins in the spectrum to zero.

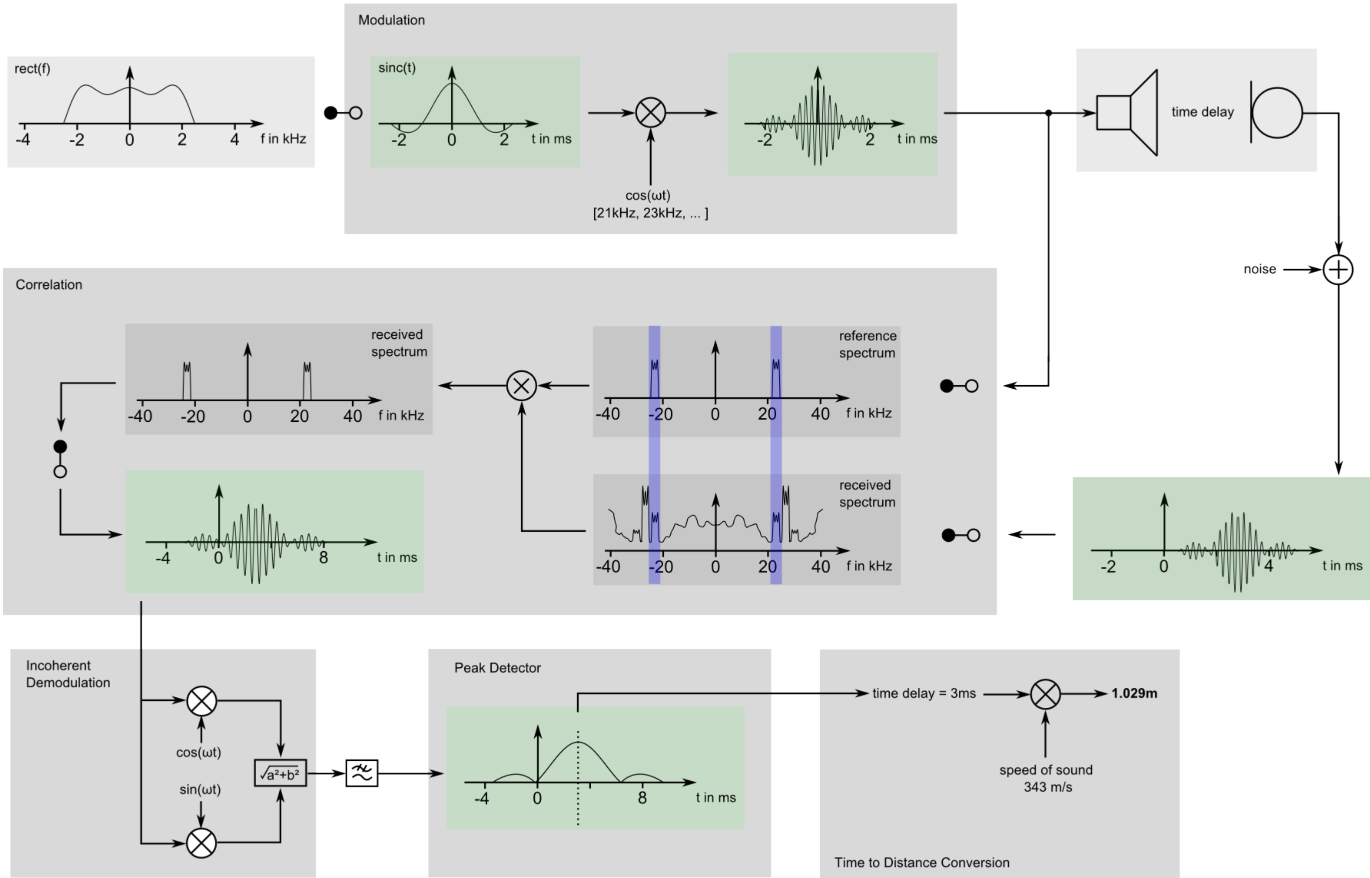

Figure 4.1: Distance measurement without performance optimizations

4.3 Performance Optimization

The most computation time is spent multiplying the spectra and computing the inverse fft. The spectrum of the reference signals is mostly zero except for a few fequency bins. Therefore, most of the computation time is spent multiplying by zero. The result of this is known beforehand and does not need to be calculated. This also means, that the valuable information is concentrated in a few frequency bins. Almost no information is lost by ignoring most of the frequency bins.

In the spectrum, only 64 complex frequency bins carry all the necessary information. Multiplying only these 64 complex frequency bins and also computing the irfft only on these bins reduces the computation time to a fraction. However, the reduced spectrum correlation is not completely equivalent to the full spectrum correlation. By extracting the important frequency bins, they are effectively shifted along the frequency axis. This represents a demodulation to an intermediate frequency which is equal to half the channels bandwidth.

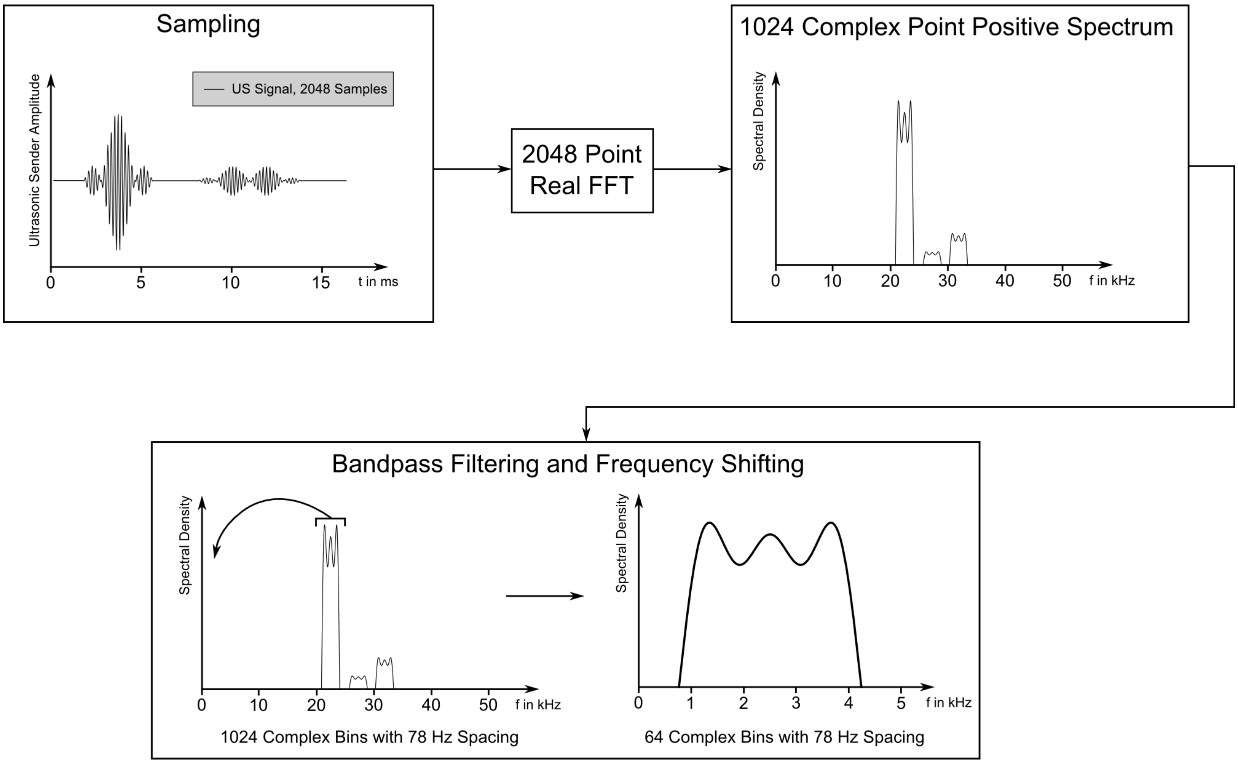

Figure 4.2: Spectrum reduction

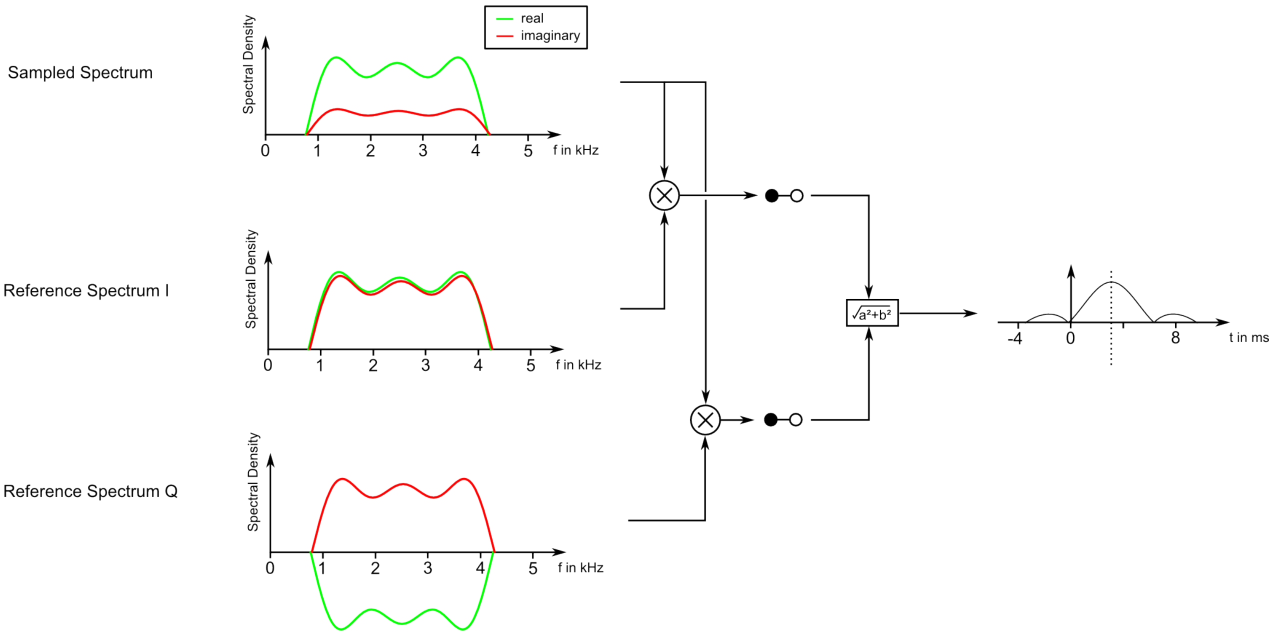

Figure 4.3: Correlation and demodulation in frequency domain on the reduced spectrum

The correlation and demodulation can be combined as shown in figure 4.3. Computing the demodulation in the frequency domain instead of the time domain has the advantage that no image frequencies are generated, omitting the need for an image frequency filter.

The demodulated correlation is a time domain function with 128 values. Only selecting the position of the maximum would result in a distance resolution of

$$d_res = c_{sound} \cdot \frac{n_{recorded}}{n_{reduced}} \cdot \frac{1}{f_{sampling}} = 343\,\mathrm{m\,s^{-1}} \cdot \frac{2048}{128} \cdot 160\,\mathrm{kHz} = 3.4\,\mathrm{cm} \tag{4.1}$$

By interpolating around the maximum, the distance resolution can be increased. In measurements, a distance measurement variation of $0.5\,\mathrm{mm}$ could be achieved.

4.4 Implementation

The described distance measurement algorithm is implemented in the `US_ranging()` function. The input of `US_ranging()` is the sample buffer
`evaluating_samples_ADC_buf[mic 0..3][sample 0..2047]`.
The output is a distance, correlation value and reflection indicator for every combination of ultrasound signal source and microphone. The respective output arrays are
`reduced_spectrum_distances[sender 0..6][mic 0..3]`,
`reduced_spectrum_max_correlation[sender 0..6][mic 0..3]` and
`reduced_spectrum_unambiguity[sender 0..6][mic 0..3]`.
The unambiguity is the ratio of maximum correlation value to the next highest local maximum correlation value. The unambiguity is an inverse measure for reflections and in-band noise.

Sometimes, the largest correlation value does not correspond to the line of sight path but to an reflection. This can happen if the line of sight path is blocked or destructive interference is reducing its amplitude. However, a property of the line of sight path is, that it is always the shortest path. Therefore, the echo suppression function `get_index_of_first_hill()` returns the position of the first local correlation maximum which is greater than the value of `maximum` times `echo_threshold_factor`.

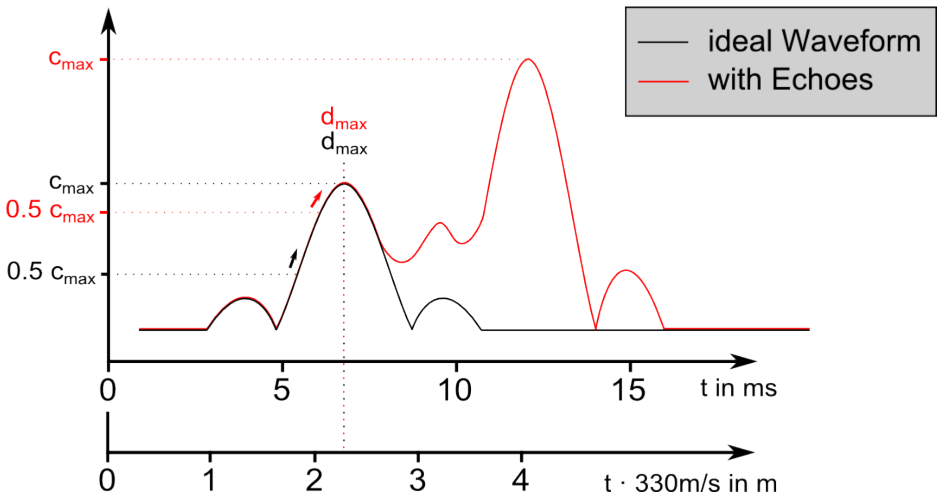

Figure 4.4: Suppression of reflections with `echo_threshold_factor = 0.5`

Chapter 5

Localization Estimation

After the ultrasound ranging algorithm has determinded the distance to all reference beacons, the localization estimator tries find the location of the robot. To simplify the process, position and orientation are handled separately. Currently, only the position estimator is implemented. Only distance measurements from microphones with a line of sight to the reference beacon are used. For every reference beacon signal, the one microphone with the smallest distance measurement while being above a certain quality threshold is selected. The distance from the robot beacons center is estimated adding a constant offset onto the line of sight distance measurements. The newton method is used to find the point where the quality-weighted-rms-error of all distances is minimized. The newton method can converge to a local minimum. It is therefore necessary to find a good starting point for the iterative process. One possible starting point is the old calculated position. However, if the the iterative process diverges one time, it is possible to get stuck. From the list of possible starting points, the point with the lowest output of the error function is selected as the actual starting point. Other possible starting points are all the crossing points of the circles around the reference beacons with the radius of the distance measurement.

The newton method is iterated until either the maximum number of iterations (`max_iterations`) is reached or the distance from the previous iteration is below $0.1\,\mathrm{mm}$.

5.1 Cooperative Error Function

To adapt the iterative process to the concept of cooperative localization, only the error-function has to be adapted. The algorithm of the newton method is not changed. The cooperative error function `float32_t multilateration_error(...)` returns the squared weighted-rms-distance-error for an arbitrary number of reference beacons. It does not matter if the reference beacons are master, slave or mobile robot beacons as long as their position is known. It is much more likely that the mobile robot beacons have a bad distance reading or a wrong location. This has to be considered when calculating the weights of each distance measurement.

5.2 Stability Improvements

Bad distance readings can produce a very shallow map of the error function over the playing area. This can lead to a diverging behaviour of the newton method. For every iteration, the newton method calculates the orientation and distance for the next step. The algorithm may diverge ,if the distance estimate is too large. On the other hand, if the distance estimate is too small, a lot of iterative steps are necessary. To find the optimal distance in each newton iteration, the distance is optimized using a bisection algorithm with up to eight iterations.

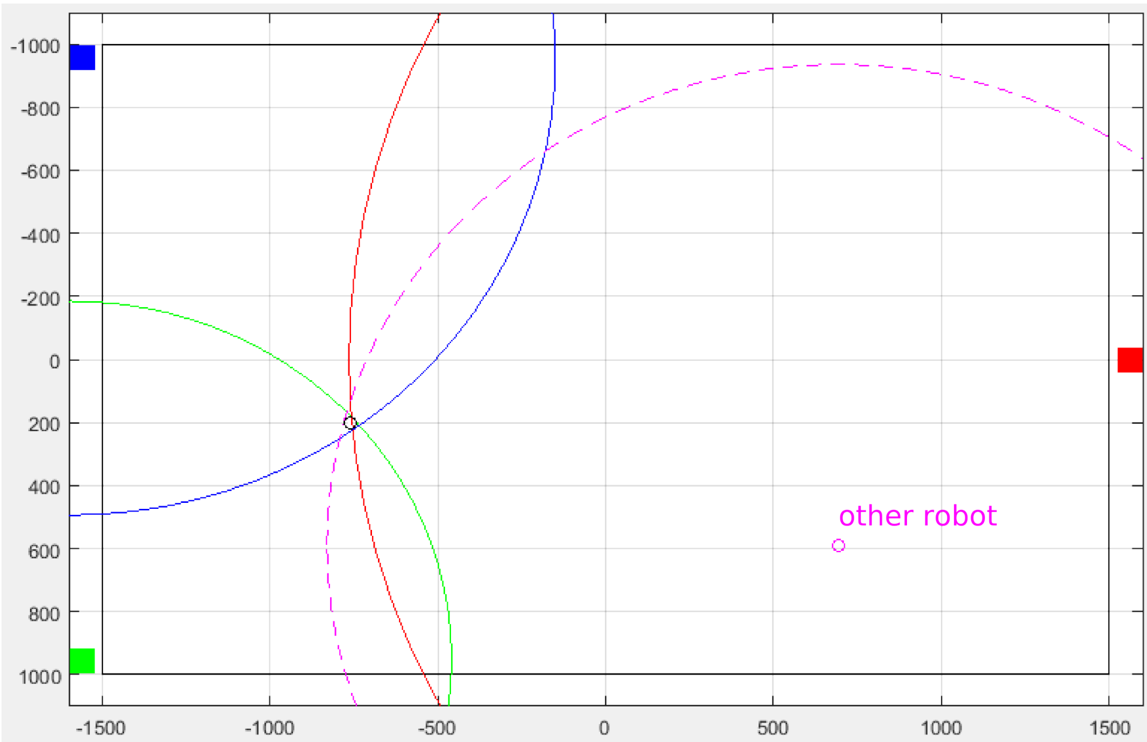

Figure 5.1: The position estimation minimizes the weighted RMS error of all distance measurements. red, green, blue boxes are fixed beacons. black circle: estimated beacon position

Chapter 6

Synchronization

To calculate the distance from the correlation, the mobile beacon must be synchronized to the fixed beacons. Infrared light is used to synchronize all beacons to a master beacon. The IR-LEDs on the master beacon are modulated with a gold code sequence. The gold sequence is known to the receiver. The receiving beacon therefore correlates the received IR signal to the sent IR sequence. The position and amplitude of the correlation maximum within the correlation corresponds to the synchronization offset of the clocks and strength of the IR signal. Knowing the clock difference, the mobile beacon inserts a corresponding delay before starting the next IR and US reception cycle. In the next cycle both clocks should run synchronous. The speed offset of the local oscillators is constantly corrected. The correction rate is measured while an IR signal is present. If the IR signal is no longer present, the estimated correction rate keeps the clocks synchronized for about 30s.

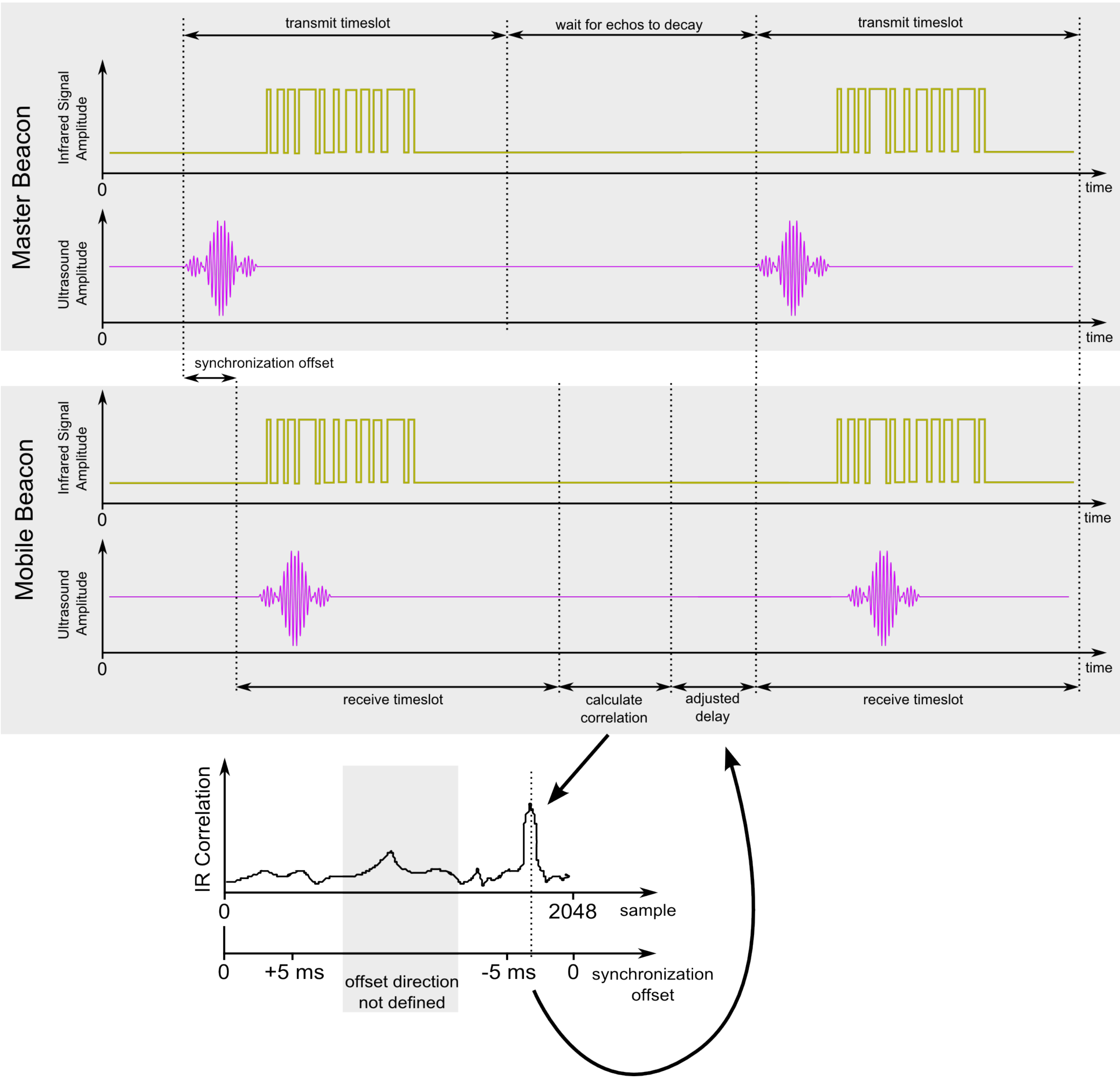

Figure 6.1: Time synchronization between master beacon and mobile beacon

Chapter 7

Communication

Each mobile beacons measures and calculates its own position. To avoud collissions between two robots they have to know the position of the other robot. This exchange of position information is also vital for the cooperative localization approach. Each beacon offers two communication interfaces, CAN and SPI. In the current implementation the SPI interface is used to communicate with a NRF24L01 2.4 GHz wireless module. After each ultrasound cycle each mobile beacon broadcasts its position, the measured distances and signal qualities.

Chapter 8

How to Replicate

All available files can be found under https://github.com/54600B/ultrasonic-beacon.

8.1 Microcontroller Software

Use Atollic True Studio to open the project file

`/software/beacon-uC/US_Beacon_V3/.project`

8.2 Matlab Control GUI

A very basic and undocumented control gui can be found under

`/software/pc-gui/matlab_playground_gui.m`

8.3 PCB

All beacons, fixed master beacon (1x), fixed slave beacons (2x) and mobile robot beacons use the same platform

`/electronics/baseboard_v3rf/us_beacon_base_v3rf.brd (Eagle)`

Depending on the function of the board, not all parts have to be populated. The design has not been replicated yet by an external party. Therefore, there may be still some errors in component values or ordering numbers. Please make sure to understand and test each subcircuit separately.

The mobile beacons each have four microphone preamplifiers

`/electronics/knowles_mic_v2/knowles_mic_v2.brd`

The microphone preamplifiers can be soldered directly to the base board. The microphones have a solder iron unfriendly package and are therefore glued to the amplifier and connected with enamelled wire in dead bug style.

Each fixed beacon has an additional audio power amplifier

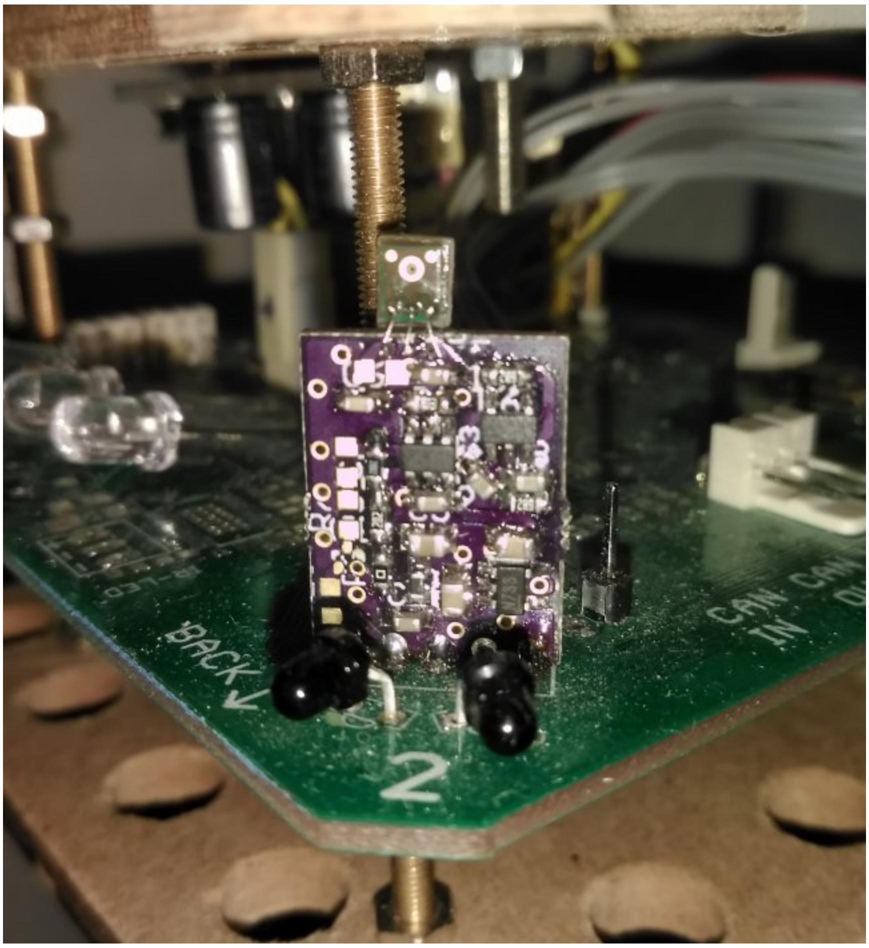

Figure 8.1: Arrangement of microphone, preamplifier and beacon base on the mobile beacon

`/electronics/amplifier/amplifier_v3.brd`

whose input is connected to the DAC output of the baseboard. If cooperative localization is used, the mobile beacons have an amplifier as well.

8.4 Mechanics

No digital design files of the mechanical design are currently available.