POLITECNICO DI TORINO

Master's Degree in Electronic Engineering

Master's Degree Thesis

Capacitive Sensor Front-end Using Carrier Demodulation

Supervisors

Prof. Mihai LAZARESCU

Prof. Luciano LAVAGNO

Candidate

Chenjie CAO

June 20, 2020

Summary

Nowadays, indoor human localization plays an important role in security system, smart house and health-care monitoring. It can be used to optimize the way to control the devices which require the locational information of a person. There are many technologies for indoor human localization have been developed in past decades. The application is usually based on WI-FI, infrared, RFID, bluetooth, ultrasonic sensors. Every technology has its own advantages and limitations.

Capacitive sensors are considered for indoor localization in this project due to their low cost, low power consumption and effective sensing. The applications based on capacitive sensors can be contactless and wear-free. It is desirable that the sensing ranges are much larger (10-20x) than the diameter of the sensing plate.

The purpose of this research project is to design a front-end interface for a long range, low cost indoor human localization with capacitive sensors. The existing design uses a carrier demodulation technique in which the phase and amplitude of the signal affected by the presence of the human body. The basic idea is to generate the carrier with sine wave of 10 kHz using the PWM block of ATmega328P microcontroller. The sine wave carrier passes a RC filter which is formed by the plate sensor and an adjustable resistor. The plate capacitance is affected by the distance between the person and the sensor plate. The change of the capacitance affects the phase and amplitude of the signal at the output of the RC filter. We convert both the RC sine wave signal and the original one into square waves using comparators, and compare their phases with an XOR gate. We filter the demodulated output and measure it using the built-in ADC of the microcontroller.

My contribution was to test the circuit, characterize the new front-end performance and design the PCB board of an improved version. For testing part, first I tested the sensitivity in a setting in which a person stand in front of the plate sensor at different ranges and was recorded the value from the microcontroller on a server through an XBEE radio module. Then, I tested the stability by running the circuit for hours in two ways. One was testing with a fixed capacitance

capacitor instead of the plate sensor, and the other was testing with the plate sensor while no person stands in front of it. By performing these tests, a fourth order butterworth BP filter part was found to contribute drift noise, so it was removed.

Other problems were the large drift over time and the noise. Thus I did resoldering some parts of the circuit in a nicer way, changing some appropriate resistance or capacitance to several filters, and also replacing some components. Due to the change of some components, the power supply needed to be redesigned. For the analog part, the virtual ground needed to be precisely at the half of the analog voltage, thus a dual-output with Vref and Vref/2 outputs component is considered. For the digital part, a switching voltage regulator is needed. And to reduce the board size, a dual-out regulator is considered. After some tests for different voltage regulators, MAX6072 and LTC3256 were introduced, one for analog voltage and virtual ground, the other one for digital voltage and XBEE (3.3V). But cause of the components limitation, the needed power supply is about 5.5 V, so that an extra external boost regulator from 5 V was to be designed as a voltage adaptor.

In the end, I designed the PCB of the new front-end circuit. The basic rule was to design different parts of the circuit depending on whether they would affect each other, and to make the board size as small as possible. It was to reduce the noise from crosstalk. First I designed the power part to make sure that the power traces connect the components in short and in a tree shape. Then to design the XBEE and microcontroller parts with the power filtering parts close to the microcontroller. For the analog part, the different functional parts were designed with ground capacitors close to the power pins of the ICs, and all components were placed as close as possible. At the end, the design passed the DRC check.

Acknowledgements

First of all,I would like to express my great gratitude to Prof. Mihai Lazarescu and Prof. Luciano Lavagno, for giving me the guidance and full support for my thesis. I would also like to thanks Osama Bin Tariq and Junnan Shan for sharing their knowledge and assistance whenever I faced difficulties.

Contents

List of Figures

Chapter 1

Introduction

1.1 Indoor human localization

Nowadays technology especially the smart phones and other wireless devices expanding a huge influence of our daily life. In particular the ""Smart Home", it provides us a convenient living environment. The need for indoor human localization is obvious in smart homes and assisted living environments. Reliable location information is vital for different location-based services that make living easier and safer. For example, one service that uses location information is the automatic adjustment of lighting, heating, or home electronic devices based on a user's physical location [1].

The positioning system should be extremely easy to use. In practice, the whole system should also be invisible to the user, as users might find even small sensors or actuators too unattractive if they are visible. [1]. Another concern is the privacy of people, unauthorized person should not be able to receive the localization information of the user. To be acceptable for home use, a positioning system needs to be passive, device-free, unobtrusive, and tag-less. Moreover, the power consumption of the system should be low for a long-term running and low maintenance.

According to the requirements, there are several types of technologies, each of them having its own advantages and limitations.

1. **Pressure Sensors** are the most traditional positioning system technology[1]. The pressure-sensitive sensors are installed in a group of load cells structure under the floor surface. They can identify a person corresponding to his unique weight and gait. The limitation is due to the high cost of their installation and maintenance. Also the system requires a lot of space which may not available to some situation.

2. **Ultrasound** mainly relies on ToF measurements of ultrasound signals (>20 kHz) and the sound velocity to calculate the distance between a transmitter and a receiver node [2]. Reflection of sound waves can be received and measured in time and frequency with precise calculation of position of the person. However, the ultrasound system requires numerous sensors to cover the house space. Moreover, it may cause some other side effects such as their possible harm for hearing and pets when it being an active system.

3. **Image Recognition** is to equip a set of sensors (cameras, Infrared thermal cameras) which remain at a fixed and known position in the location. This system is highly accurate and able to determine not only the position of person but his speed. Also it can identify the person. However, it has the drawback that it is not very well-suitable neither for privacy nor for power consumption. And it always requires a line of sight between the user and the camera.

4. **RFID** technology is primarily intended for transferring and storing data using electromagnetic transmission from a transmitter to any Radio Frequency (RF) compatible circuit [3].

Here we focus on **capacitive sensors** which are very suitable for mid-range positioning and tracking, they have low power consumption and low installation and maintenance cost. Also they are tag-less and no privacy problems. My task is to optimize and tune the capacitive sensor in terms of signal to noise ratio, power consumption, stability and maximum achievable detection range. And finally to design the sensor PCB.

1.2 Capacitive Sensing

Capacitive sensing is widely used on smart phones, tablets, and laptops from touchscreens and touchpads to the capacitive "buttons" as consumer electronics devices and commercial equipment. The use of capacitive sensing is common in human computer interaction research.

As shown in Figure 1.1, due to the capacitance between the human bodies and other objects, the information can be derived by measuring the level of the capacitance coupling that exist between conductive objects including people. We can consider the human body acts like a conductive plate of the parallel plate capacitor, with air as dielectric. Between the two conductive plates separated by a

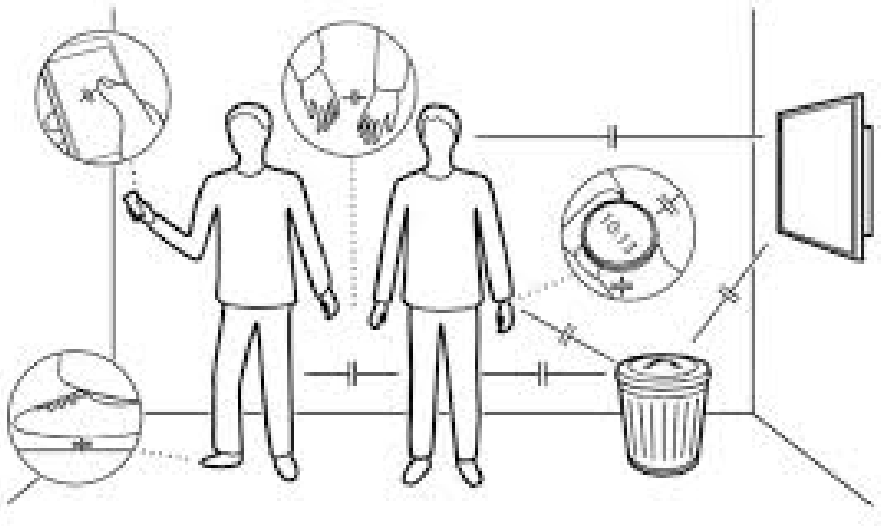

Figure 1.1: Capacitive coupling in a room between different masses from [4].

dielectric, a voltage **V** developed. From theory of capacitance, we know that

$$Q = C \cdot V \tag{1.1}$$

Where C is capacitance and Q is total charge held by plates. As mentioned in [5], the capacitance depending on its size, the distance to other objects, and the dielectric properties of the objects and the dielectric between them. When A»d,

$$Q = \frac{\varepsilon_0 \cdot k \cdot A}{d} \tag{1.2}$$

where C is the total capacitance formed between the two plates (in Farad), k is the relative dielectric permittivity of the material between plates (k = 1 in case of free space), 0 is the absolute dielectric permittivity of free space (8.854×10^{-12} F/m), and d is the distance between the capacitor plates (in meters). For our case, it is not valid because d»A for long range capacitive sensors. An approximated formula for the sensor is:

$$Q \cong \frac{k \cdot A}{d^{2 \sim 3}} \tag{1.3}$$

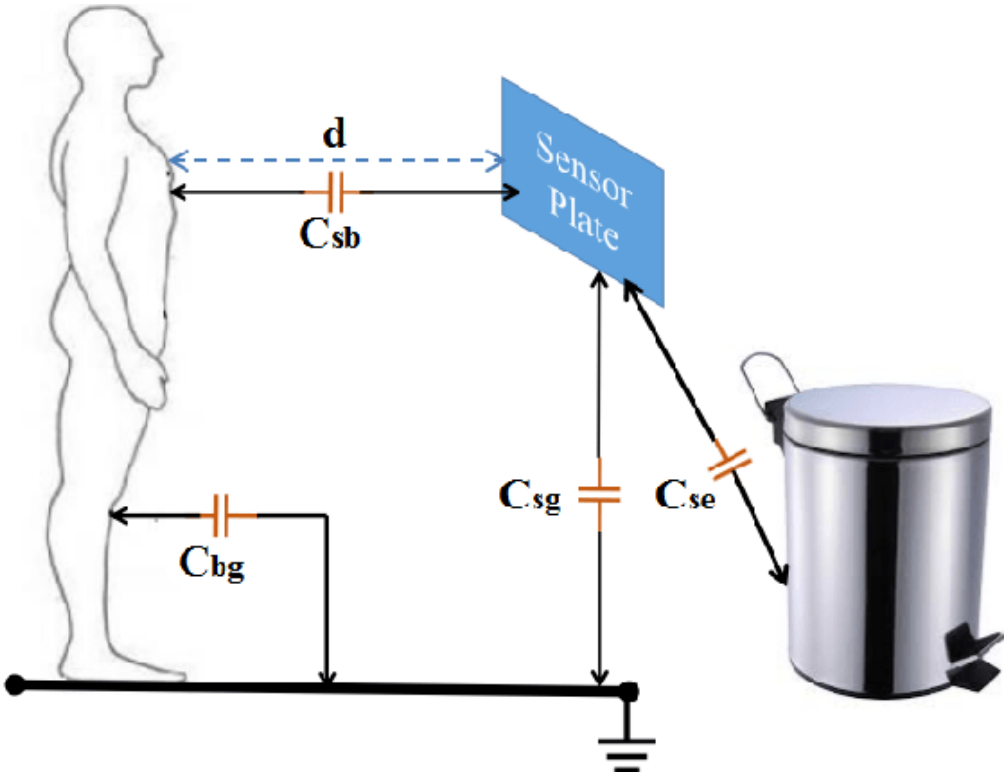

Figure 1.2: Capacitive coupling between sensor and objects from [6].

Referring to Figure 1.2 from [6], the capacitance is due to the four types of capacitances as follows:

1. **C**sb: capacitance between human body and sensor plate
2. **C**sg: capacitance between sensor plate and ground
3. **C**bg: capacitance between human body and ground
4. **C**se: capacitance between sensor plate and environment

The capacitance is also affected from environment humidity and temperature. In this case, **C**sb is function depend on the distance between the sensor and the human body which is the measurement interested.

1.3 Working modes for capacitive sensing

Considering the system shown in Figure 1.3, there are mainly three types of capacitive sensing working modes. Two of them involves the utilization of two sensors,

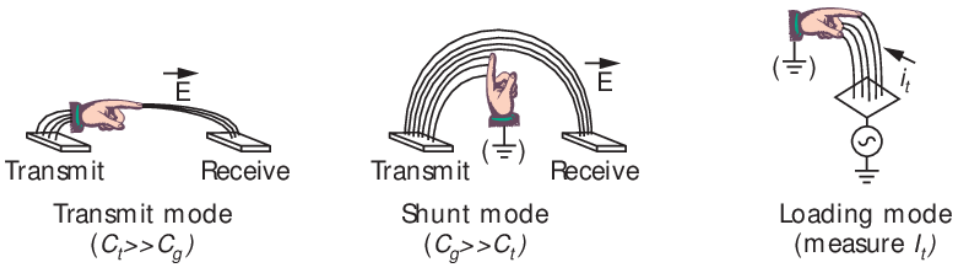

Figure 1.3: Capacitive sensing working modes from [7].

one for transmitting and the other one for receiving.

Transmit mode allows the human body to become part of the transmitter, with the body close to transmitter. This mode is suitable for close distance between the transmitter and the body, due to the significance of the capacitance between body and plate compared to the body ground capacitance.

Shunt mode happens when the body not very close to the plate, and the body ground capacitance prevails. In this situation, the signal received is decreased with the close of body.

The two methods are very powerful for the amount of information. For example, the transmit mode is adept at following the movement when the user is contact to the sensor. But according to the previous discussion, this would not be a simple and tag-less method.

Loading mode is simple because it needs only a single plate. It permits to measure the induced current in the plate. Both plate and body are referred to a common ground, that is a shared potential [7].

1.4 Previous research

There are many previous research done about human localization with capacitve sensing using different techniques. Some of them are [8], [9], [10]. Their work were to perform different types of methods and experiments to obtain the data which would be used to draw some conclusion.

The main idea about the circuit design is based on the measurement of signal amplitude and phase difference which is caused by the change in capacitance of a capacitive sensor. A copper plate of 16 cm×16 cm was chosen as a sensor. It has a

good sensitivity of 200 cm with a low noise and good resolution. With the sensor mounted on each wall of a room of 3 m×3 m, human in this room was accurately localized.

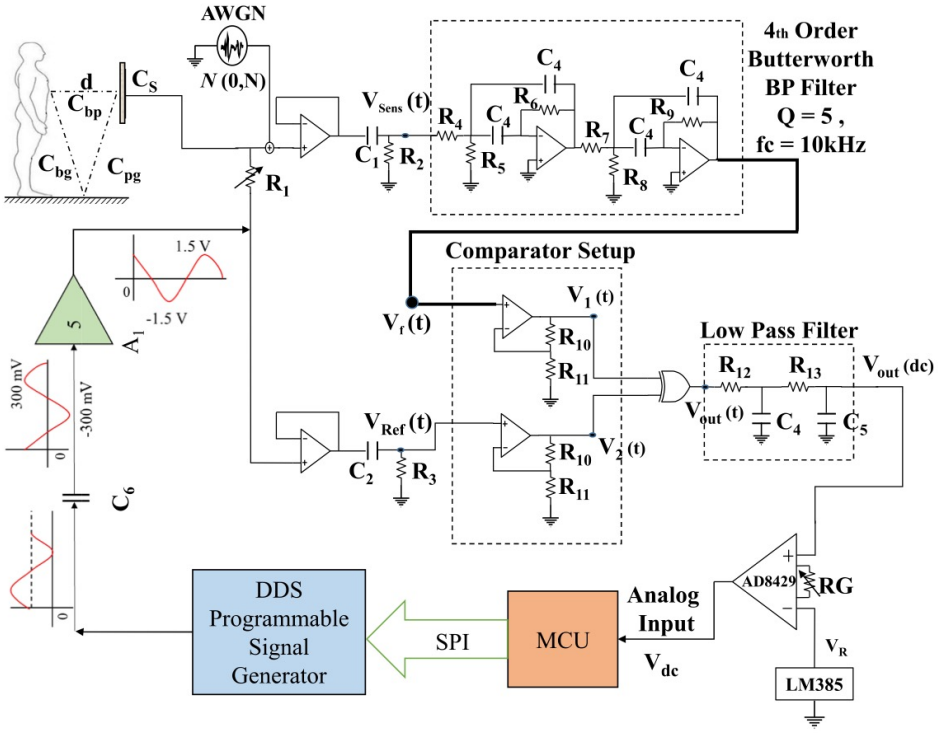

Figure 1.4: Circuit design of front-end interface from [11].

As shown in Figure 1.4, the change in capacitance of the sensor is caused by human coupling. The board using DDS programmable signal generator to generate a sine wave of 10kHz. The sine signal is fed to two branches. Both of them have RC low pass filters but a sensor plate is used as a capacitor on one branch, which is shown as C_s in Figure 1.4. So that the sine signal passing through the RC filter is affected in terms of amplitude and phase. The high-pass filters that follow the reader circuits, C_1R_2 and C_2R_3, eliminate the very low frequency noise and DC offsets from the buffered signals. Then sine signal of the upper branch is passed through a high Q-factor band-pass filters to remove most of the noise from the signal. The signals of the two front-ends paths, V_{ref}(t) and V_f(t)(see Fig 1.4), are then passed through a hysteresis comparator to prepare them for the phase discriminator circuit that is implemented using a XOR logic gate. The signals

generated from the comparators are then passed through an XOR gate which generate V_{out}(t). In order to obtain the DC component of the phase discriminator output, a 2nd-order low-pass filter is introduced which passes only the DC term. The DC value is measured by an internal ADC of ATmega328P microcontroller which is also used to control signal generator with SPI interface.

From [11], it explains the mathmatical model. The two signals from the same signal source:

$$V_f(t) = Asin(2\pi ft + \theta), \tag{1.4}$$

with A = 3 V_{pp} and f = 10 kHz.

We model the noise captured by the sensor transducer, as an additive white Gaussian noise generator (AWGN) at the input of the reader circuit on the top path. So that at the output of the top path reader circuit we have:

$$V_{Sens}(t) = Bsin(2\pi ft + \theta + \phi) + N(0, N), \tag{1.5}$$

while at the output of the bottom path read circuit we have:

$$V_{Ref}(t) = Asin(2\pi ft + \theta) \tag{1.6}$$

where the amplitude is

$$B = \frac{1}{\sqrt{1 + (2\pi fC_sR_1)^2}} \tag{1.7}$$

and the phase shift is

$$\phi = -arctan(2\pi fC_sR_1) \tag{1.8}$$

The maximum phase shift sensitivity of an RC filter to variations of the filter capacitance is at the cut-off frequency. Hence adjust the value of R_1 in order to tune the R_1C_s cut-off frequency to 10 kHz, when nobody is present in sensor range. At the cut-off frequency we have

$$B = \frac{A}{\sqrt{2}} \tag{1.9}$$

and

$$\phi = -\frac{\pi}{4}. \tag{1.10}$$

So we have the formula for the R_1C_s at cut-off frequency:

$$V_{Sens}(t) = \frac{A}{\sqrt{2}} sin(2\pi ft + \theta - \frac{\pi}{4}) + N(0, N). \tag{1.11}$$

To redece the noise, a 4th order butterworth bandpass filter with a quality factor Q = 5 is implemented. Its gain is about 1.4, to compensate for the attenuation of R_1C_s filter:

$$V_f(t) = 1.4\frac{A}{\sqrt{2}} sin(2\pi ft + \theta - \frac{\pi}{4}). \tag{1.12}$$

The comparator hysteresis is small, the upper threshold voltage, V_{TH}, and lower threshold voltage, V_{TL}, are:

$$V_{TH} = V_{CC}(\frac{R_{11}}{R_{11} + R_{10}}), \tag{1.13}$$

$$V_{TL} = V_{EE}(\frac{R_{11}}{R_{11} + R_{10}}). \tag{1.14}$$

With V_{CC} =5 V, V_{EE} =0 V, R_{10} = 100 kΩ and R_{11} = 1 kΩ, we get $V_{TH} \simeq 50$ mV, V_{TL} = 0 V.

The fundamental frequency of both V_1(t) and V_2(t) is 10 kHz, so that the fundamental frequency of the output of the XOR gate, V_{out}, is double, f_0 = 20 kHz. The non-zero average signal V_{out} can be represented as a Fourier series.

$$V_{out} = \frac{5T_1}{T} + \sum_{k=1}^{\infty}(A_k cos(2k\pi f_0 t) + B_k sin(2k\pi f_0 t)), \tag{1.15}$$

with

$$A_k = \frac{5}{k\pi}[sin(\frac{2\pi kT_1}{T})], \tag{1.16}$$

$$B_k = \frac{5}{k\pi}[1 - cos(\frac{2\pi kT_1}{T})]. \tag{1.17}$$

T_1 depends directly on the variable phase shift introduced by the R_1C_s filter based on the value of the overall capacitance of the sensor plate which depends on the position of the human body.

After the 2nd-order lowpass filter, we get the DC term:

$$V_{out}(dc) = \frac{5T_1}{T}. \tag{1.18}$$

After adjusting R_G to set the amplifier gain to G,

$$V_{dc} = G(V_{out}(dc) - V_R). \tag{1.19}$$

Thus, the output voltage V_{dc} changes with C_s, so that if the person approaches the sensor, the value when the person is out of the range of the sensor will be increased.

Chapter 2

The Project

2.1 Description

The circuit has been designed to measure the phase shift difference caused by an excitation signal and the response signal form the exited system. As shown on Figure 2.1, the microcontroller generates PWM signal with 10 kHz. One signal enters a low pass filter with R and C_s, where C_s is the capacitance of the sensor whose value depends on the distance between the plate and the human body. To achieve the maximum sensitivity of the RC filter, the R resistor should be adjustable in order to tune the RC filter at the cut-off frequency which is 10 kHz. The output signal from the RC filter and the original signal then are passed through two voltage follower. High pass filter after them eliminate the low frequency noise from the buffered signal.
To remove most of the noise, a narrow band-pass filter is used. The filter is only placed on the upper path of the sensor frontend and is implemented using a 4th order butterworth band-pass filter with a quality factor Q = 5.
The signals of the two frontend paths are then passed through a hystersis comparator to prepare them for the phase discriminator circuit that is implemented using a XOR gate. After the XOR gate, with a 2nd-order low-pass filter, the output is the DC term of the phase discriminator output. With no one in the range of the sensor, there is a phase difference between the two signals which is $\frac{\pi}{4}$. Thus we can obtain a static output value. To obtain V_{dc}, using a precision instrumentation amplifier AD623, and an adjustable resistor is implemented to set the instrumentation amplifier gain. The ATMAGA328p microcontroller samples the output from the amplifier. And then is used the XBEE to send the sampled data to the PC.

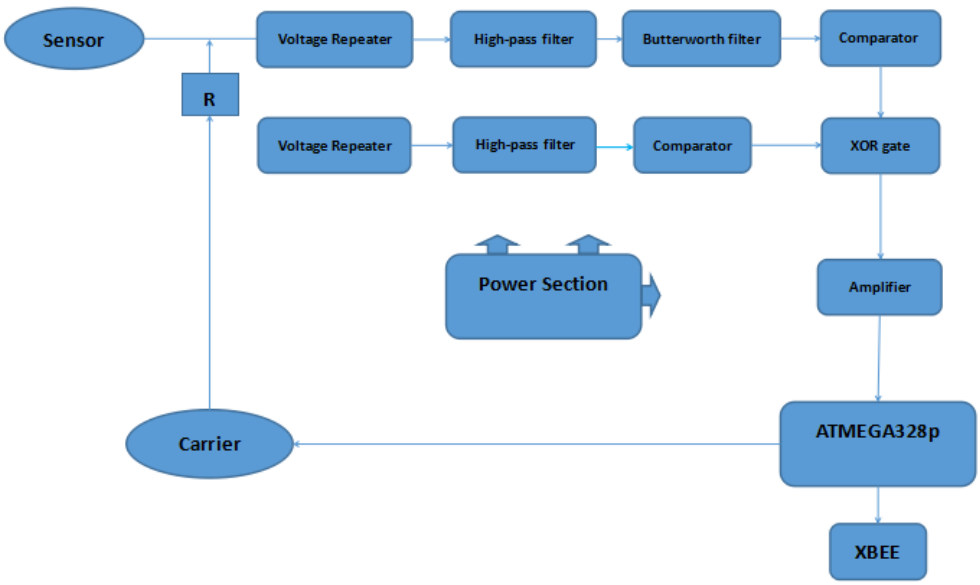

Figure 2.1: Block diagram of frontend interface.

2.2 Board testing

The board had already been checked by a previous researcher, so the function of the board basically worked. I started my work from the schematics and the datasheets of the IC components of the board. Ahead of all, I needed to understand every component and every section part. Since the previous researcher changed the board without modifying the schematic, firstly I modified the schematic to match the tested board, the layout is shown in Figure 2.2. Then I tested the board to verify it.

Firstly I tested the functionality of the board. My method was to run the board and verify the whole function of the circuit with digital multimeter, digital oscilloscope. I found the circuit is worked in correct behavior. Then I used the MATLAB to get the value through the wireless reception. The value changes with the human body movement. Before testing, I had to do a fine tuning to the circuit. I started from tuning the frequency of R_1C_s (R_1 is the adjustable resistor of the low-pass filter) at the cut-off frequency of 10 kHz, according to that the amplitude of the output of R_1C_s should be 0.707 of the original sinusoid when no one is in the range of the sensor plate. Another tuning is adjusting the trimmer implemented with the instrumentation amplifier to set the gain, so that the input of the ADC on

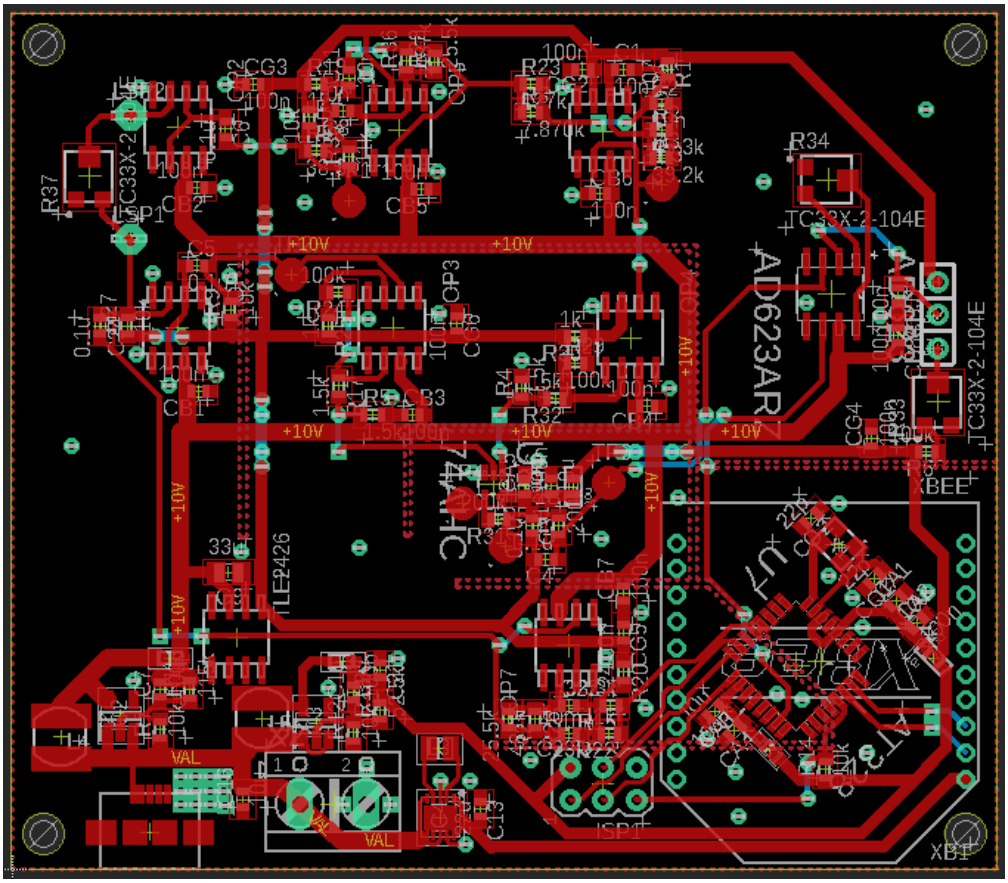

Figure 2.2: Modified board PCB.

microcontroller is within range of ADC under all conditions. I have a MATLAB test code named "check sensitivity" to test the sensitivity of the board from the distance of 0 cm to the distance of 200 cm between human body and sensor plate. And another test code named "check stability" to test the stability of a long period running.

2.3 Problems

After the board testing, I found some problems through the plot of values. To check the sensitivity of the sensor, a person was asked to stay in front of the sensor with some predefined distances and the readings of V_{dc} were recorded by MATLAB. The man moved to the certain distances according to test scenario. The result is shown Figure 2.3. From the result, it shows the noise which is unwanted of more than 12 mV(12 mV/2.976 V=0.403%).

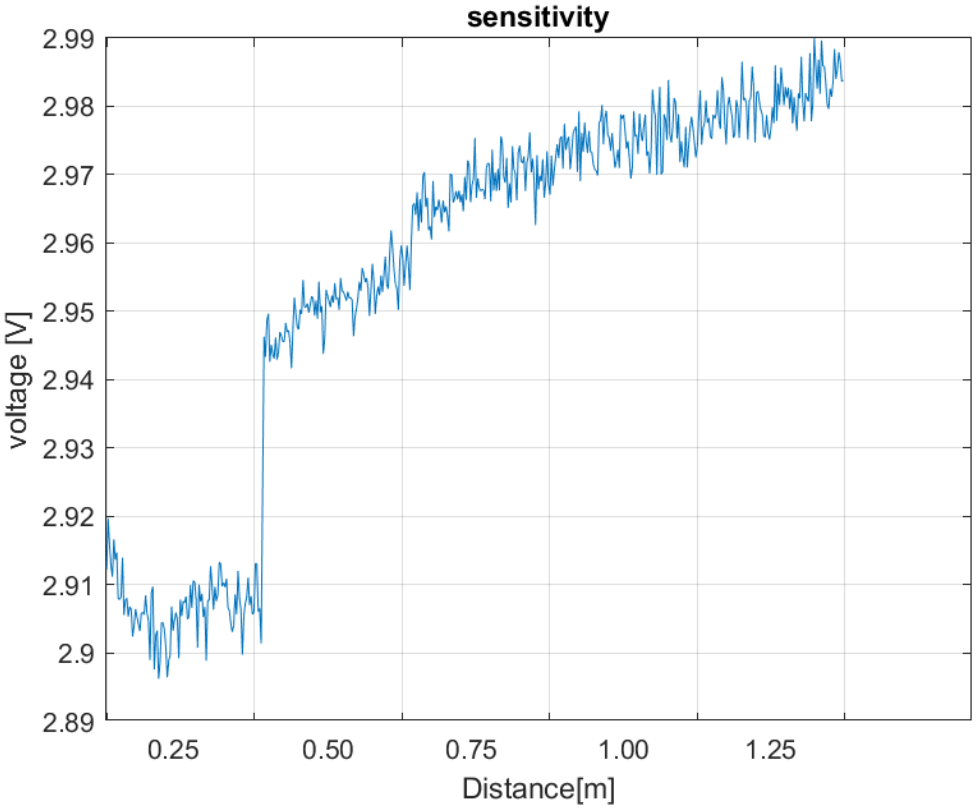

Figure 2.3: Result of sensitivity test of distance from 25 cm to 125 cm.

To check the stability, I kept the board running for several hours with no person in a room. Figure 2.4 shows a result of stability check. There is a huge drift of nearly 70 mV over 21 hours(70 mV/3.64 V=1.9%) which is very high.

2.4 Solutions to improve the front-end performance

Some drift could be caused by the environment changes such as temperature, humidity of the room, but these effects should be small. Then I found the 4th order butterworth band-pass filter (shown in Figure 2.5) also caused the drift. To solve this problem, I bypassed the BP filter and ran again the stability test. I found that

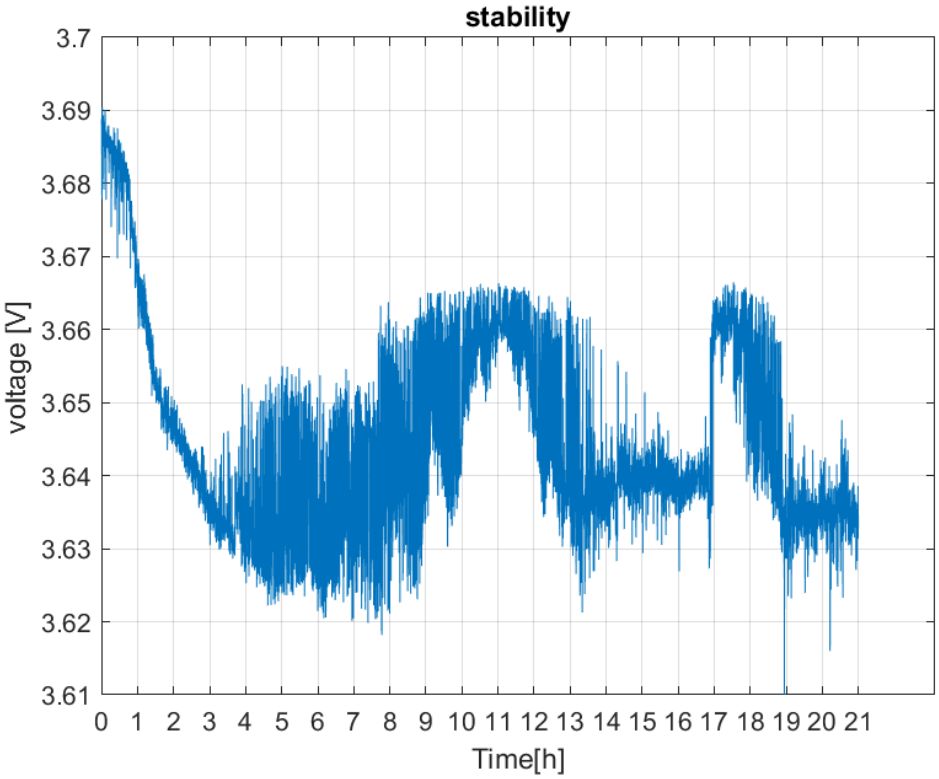

Figure 2.4: Result of stability for 21 hours.

the drift was reduced and is acceptable. The result is shown Figure 2.6.

The problem of noise could not be eliminated completely especially when the 4th order butterworth BP filter is bypassed. Firstly, I did some steps to reduce the noise in circuit. I reduced the length of the wire which connects circuit to the sensor plate. I used the trimmer TC33X-2-104E attached to board instead of using big size trimmer soldering on the board with wires. Also I resoldered some components on the board instead of using another board connected with wires.

Then I checked the power supplies. I used power supply with 5 V and 400 mA to power up the circuit, there was no any current limiting and voltage drop. There are 3 different voltage supplies on the board which are 10 V, 5 V (Virtual ground), and 3.3 V. Two components NCP1403 generate 10 V and 5 V. And the virtual ground 5 V is generated by a rail spiltter TLE2426. It can generate the voltage exactly in middle of 10 V and ground. ADP2503 generates the 3.3 V which is only for the power supply of XBEE part.

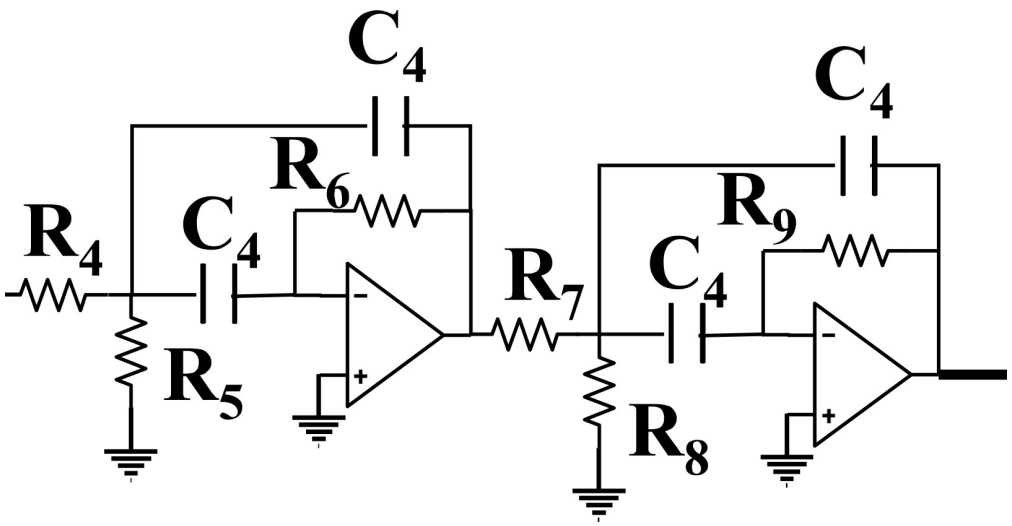

Figure 2.5: 4th order butterworth BP filter.

I noticed that some samples are missed when the data transmitted to the PC through the XBEE. With checking the power supply of XBEE, the voltage sometimes droped below 2V when loaded with XBEE part. So that XBEE would switch off if the voltage was not in its correct voltage supply. I checked both the datasheet of ADP2503 and the layout of board. I saw that the datasheet recommends to implement both input and output capacitor which should be very close to the component (as shown in Figure 2.7). While in layout, the input capacitor of ADP2503 was shared with 10 V regulator and 5 V regulator (as shown highlighted in Figure 2.8).

So that I soldered a 10uF capacitor very close to the regulato, then the problem was solved and the XBEE worked in correct transmission.

2.5 Component modification

The goal of the project is to design a small size and less power consumption circuit. So that I began to try finding some better components instead of the components now used.
Initially for the operation amplifier, we had a requirement that the amplifier should

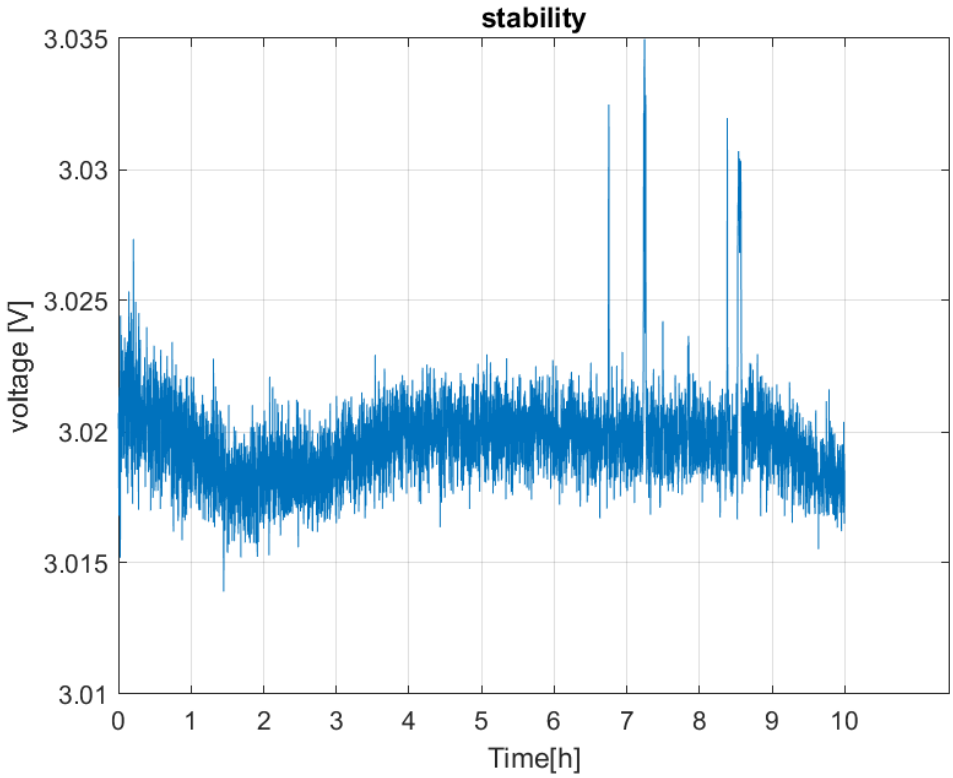

Figure 2.6: Result of stability bypassing the butterwrth BP filter.

be single or dual rail to rail, input impedance should be larger than 10 MΩ, power supply should over than 5 V, and slew rate should be more than 0.5 V/us. From the datasheet of OP184 which was now used, it basically achieves the requirement. But when considering the power consumption, its supply current is 1.45 mA which is quite large. So that I chose a rail to rail, low power op-amp OPA2990(As shown in Figure 2.9). It highly achieves the requirement, moreover, its supply current is only 120 uA which is a very small power consumption. Also it works well below 60 °C. Since OPA2990 consists of dual op-amp, we could save space on the PCB board.

For the comparator part, we need to achieve that power supply is larger than 5 V, slew rate is over than 1 V/us or 10 V/us. The comparator now used is implemented of op-amp which is not good because it could not achieve the full output. Here I chose the low power, rail to rail comparator MAX9142 (as shown in Figure 2.10). It is high speed whose propagation delay is 40 ns, while supply current is only 150 uA per comparator. It works well below 60 °C. Most importantly, its output stage

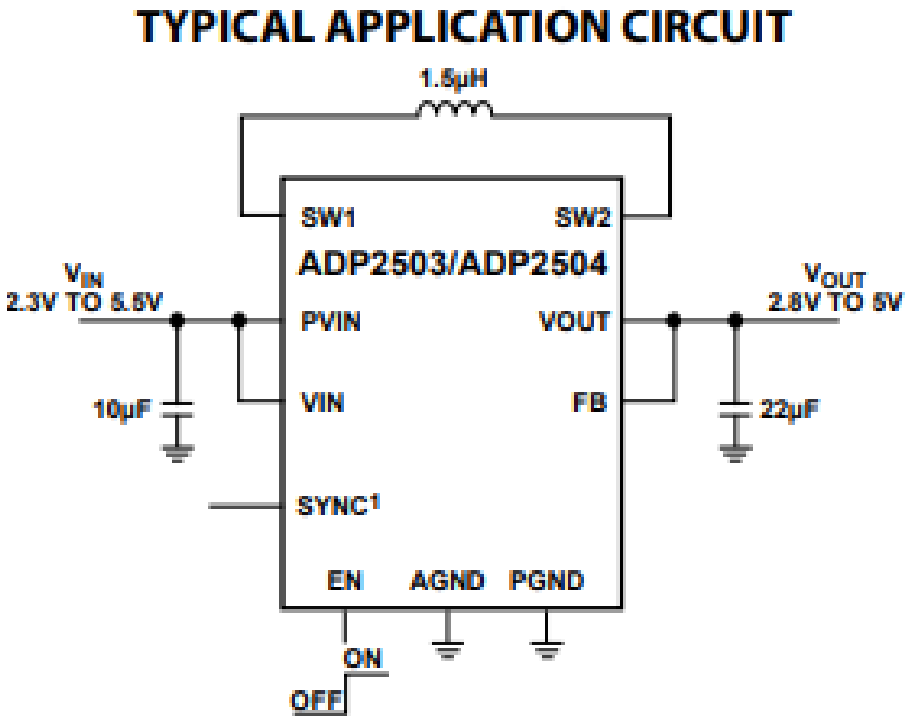

Figure 2.7: Typical application circuit of AD2503.

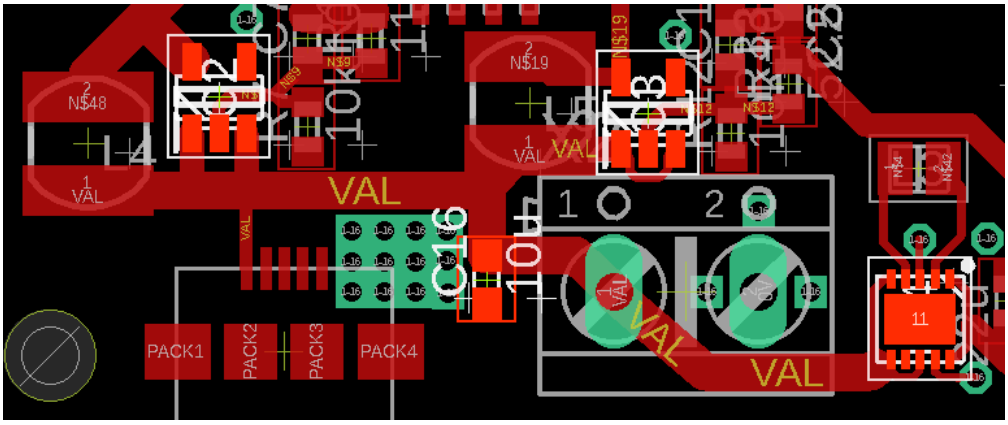

Figure 2.8: Input capacitor shared within three regulators.

circuitry (as shown in Figure 2.11) is what we need. During an output transition, I_{SOURCE} or I_{SINK} is pushed or pulled to the output pin. The output source or sink current is high during the transition, creating a rapid slew rate. Once the output voltage reaches V_{OH} or V_{OL}, the source or sink current decreases to a small

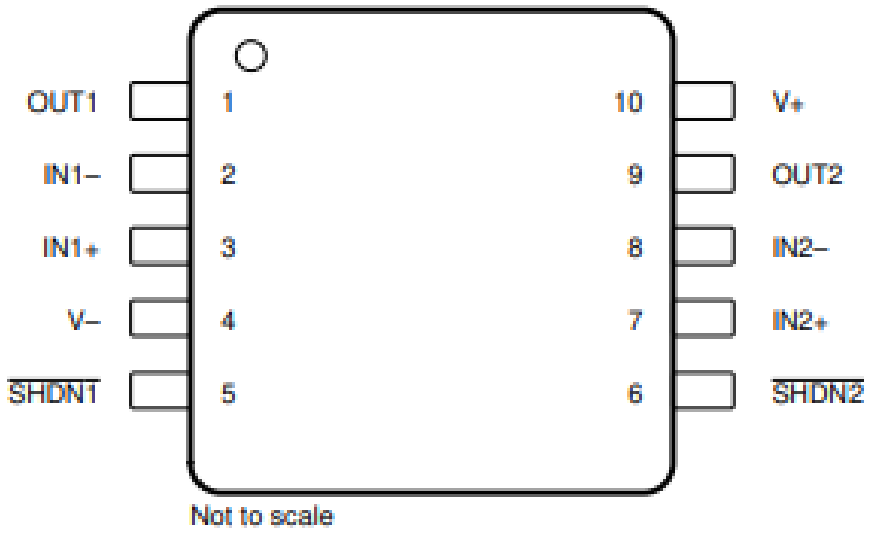

Figure 2.9: Top view of OPA2990.

value, capable of maintaining the V_{OH} or V_{OL} static condition. This significant decrease in current conserves power after an output transition has occurred. One consequence of a current-driven output stage is a linear dependence between the slew rate and the load capacitance. A heavy capacitive load will slow down a voltage output transition. This can be useful in noise-sensitive applications where fast edges may cause interference.

Because of the change of some components, then I considered to redesign the power section part. According to the datasheets of the components I used, I need a 3.3 V voltage regulator for XBEE part, an analog voltage regulator of 5 V, a digital voltage regulator of 5 V, and a voltage reference whose output is half of the analog voltage regulator output.
I found a high-precision, dual-output series voltage reference MAX6072 (as shown in Figure 2.12). The MAX6072 is a dual-output precision series voltage reference. The product features two outputs, $+V_{REF}$ and $+\frac{V_{REF}}{2}$. The device exhibits a very low 1/f noise of 2 ppm (peak-to-peak). Each output can source and sink 10 mA and has an independent sense line. This product has a temperature drift of 6 ppm/°C (max) over the ambient temperature range of -40 °C to +125 °C and an initial accuracy of 0.04 %. And a pair of output voltages is 5 V/2.5 V which is available to my case. Also its supply current is low which is 150 uA.

For the digital voltage regulator and 3.3 V voltage regulator, to reduce the PCB size, I found a dual output, step-down charge pump with automatic mode

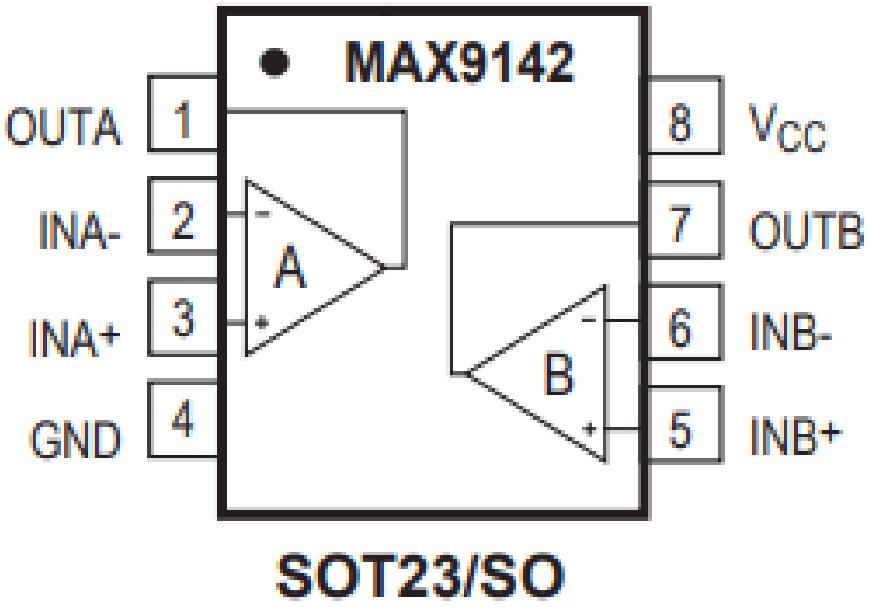

Figure 2.10: Top view of MAX9142.

switching, LTC3256 (as shown in Figure 2.13). The LTC3256 is a wide input range switched capacitor step-down DC/DC converter that produces two regulated outputs: a 5 V output via direct connection to the charge pump output, and a 3.3 V output via a low dropout (LDO) linear post-regulator. The device provides up to 350 mA of total output current. At 12 V V_{IN} and maximum load on both outputs, power dissipation is reduced by over 2 W compared to a dual LDO regulator solution. The LTC3256 has numerous safety features including overcurrent fault protection, overtemperature protection and tolerance of 38 V input transients.

Unfortunately, a problem occurs if we use the MAX6072. Because of the supply voltage of MAX6072 is 5.2 V to 5.5 V which is over the output voltage of power bank. Thus, I need design an extra external boost regulator. Here I chose XC6372 (as shown in Figure 2.14). It is a PWM controlled step-up DC/DC converters. The built-in 1.4 Ω switching transistor type enables a step-up circuit to be configured using only three components, a coil, a diode, and a capacitor. Output voltage can be selectable in the range from 2.0 V to 7.0 V in increments of 0.01 V (accuracy: ±2.5 %).

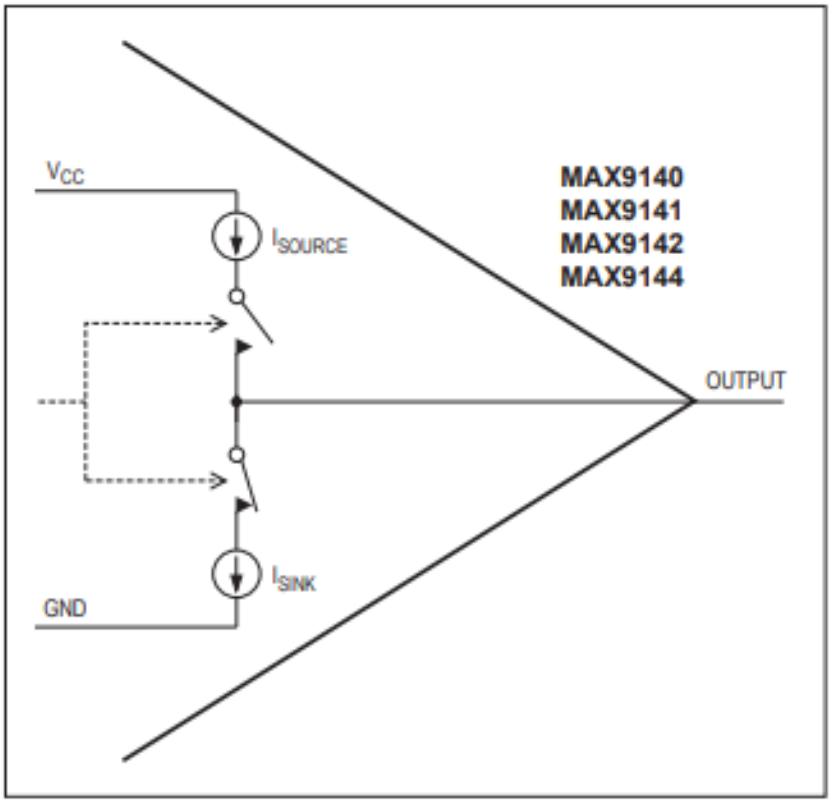

Figure 2.11: Output stage circuitry of MAX9142.

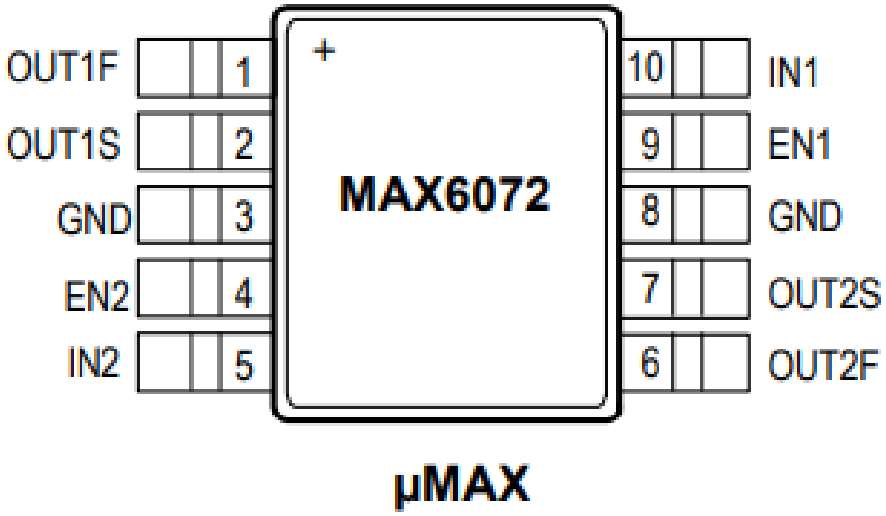

Figure 2.12: Top view of MAX6072.

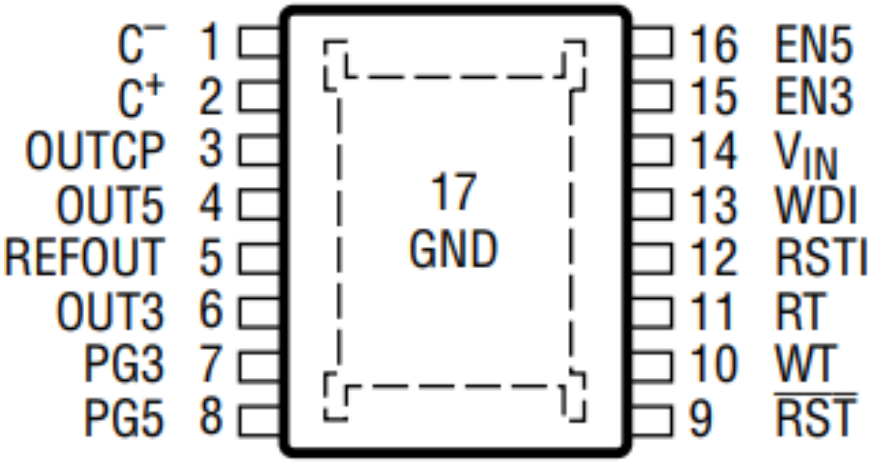

Figure 2.13: Top view of LTC3256.

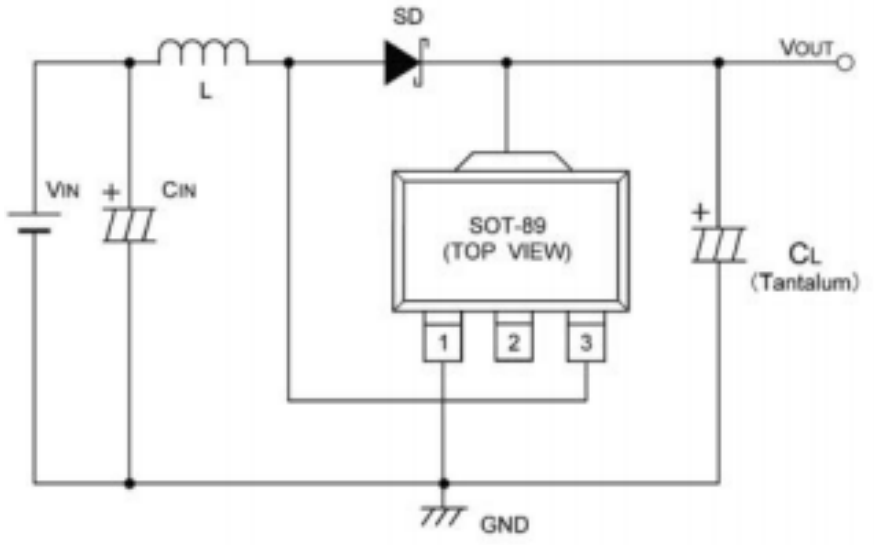

Figure 2.14: XC6372.

Chapter 3

The Results

3.1 Experimental setup

As shown in Figure 3.1, the sensor plate which is 16 cm×16 cm was mounted on a wall consists of boxes at a certain height. The board was put on the top of the box away from the power bank which would reduce the noise from the power bank. XBEE 802.15.4 RF modules by Digi International were used at both transmitting end and receiving end. And MATLAB was used to received the data on PC, then plot the data values.

Figure 3.1: Experimental setup.

3.2 Sensitivity

As shown in Figure 3.2 and Figure 3.3. The highest noise is around 4 mV (4 mV/2.604 V=0.154 %) with average noise 2 mV. The result is much better than the previous circuit. Comparing the results of these two tests, the noise is still visible which may be caused by person slightly movement during the test, electrical wiring in floor, etc. Moreover, if we just consider the value of certain distance, it could be considered that the circuit can detect a man up to 1.5 m.

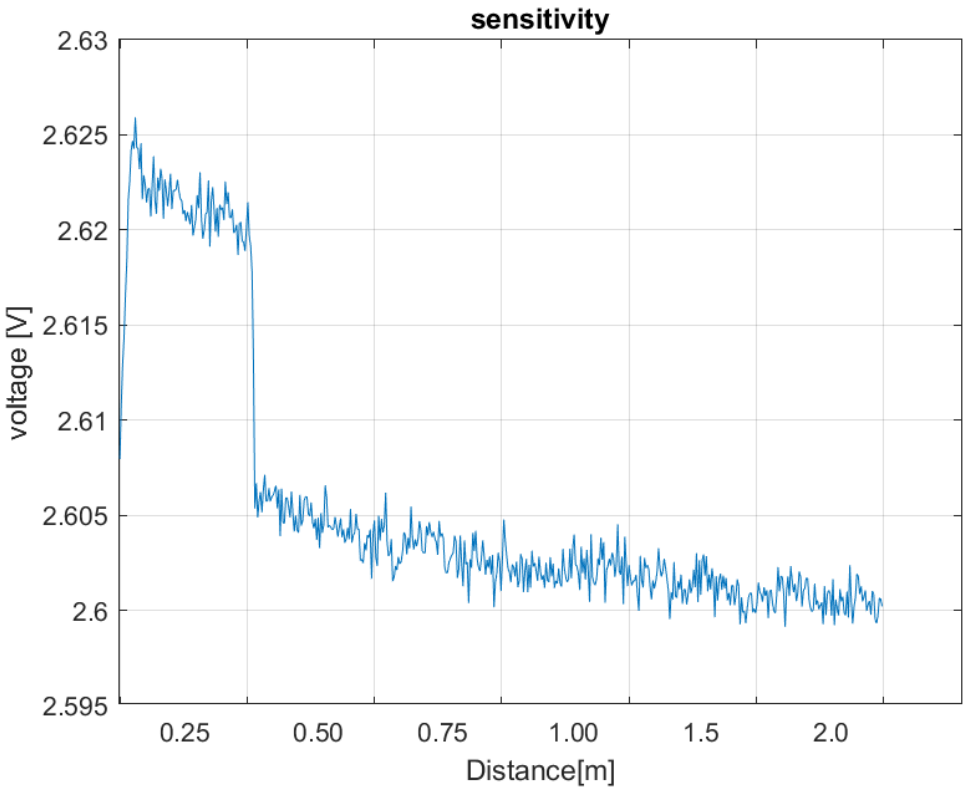

Figure 3.2: Sensitivity test 1.

Table 3-1 shows a comparison of the previous front-end and the improved front-end in terms of sensitivity. SNR can be calculated as maximum change in voltage from 0.25 m to 1.5 m divided by mean amplitude of noise.

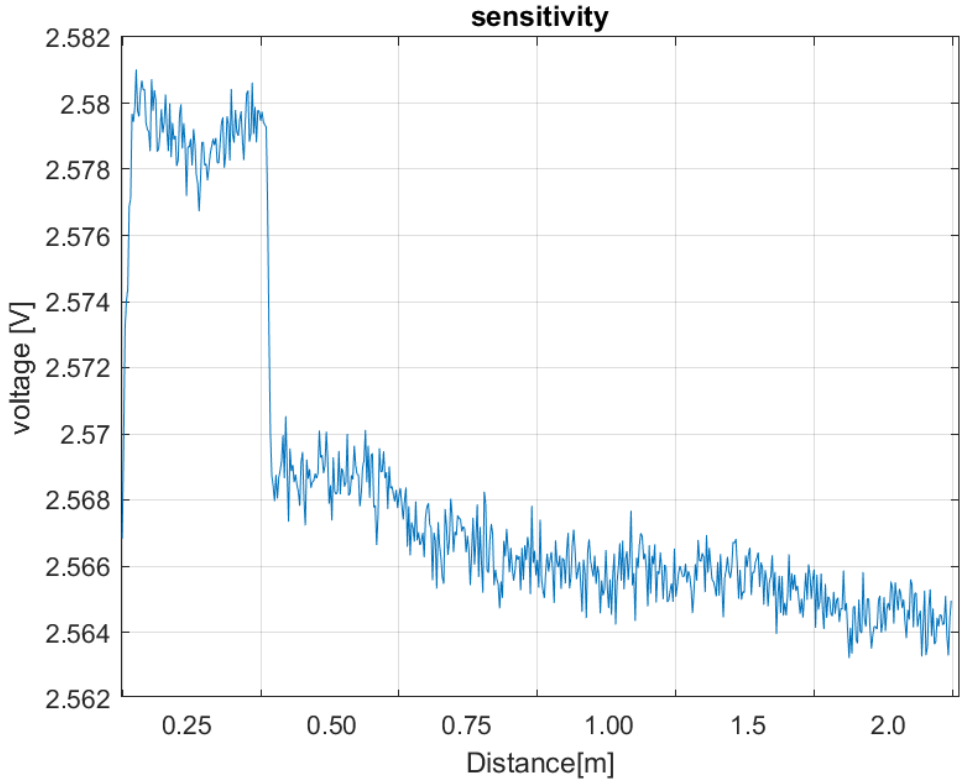

Figure 3.3: Sensitivity test 2.

Attribute	Previous Design	Improved Design
Peak noise	12 mV	4 mV
Average noise %	0.403 %	0.154 %
Max. Change in V	80 mV	20 mV
SNR	16.5	14.0
Range of detection	1.25 m	1.5 m

Table 3.1: Comparison between the previous circuit and the improved circuit in terms of sensitivity

3.3 Stability

As shown in Figure 3.4 and Figure 3.5. To avoid other disturbance with the purpose only testing stability, the sensor plate was replaced by a fixed value of 18 uF capacitor. From the results, there was a small drift around 15 mV for 11 hours (1.36 mV/h and 15 mV/2.414 V=0.621 %), which may be caused by environment

affects such as humidity, temperature, etc. However, the stability of the circuit is better than before.

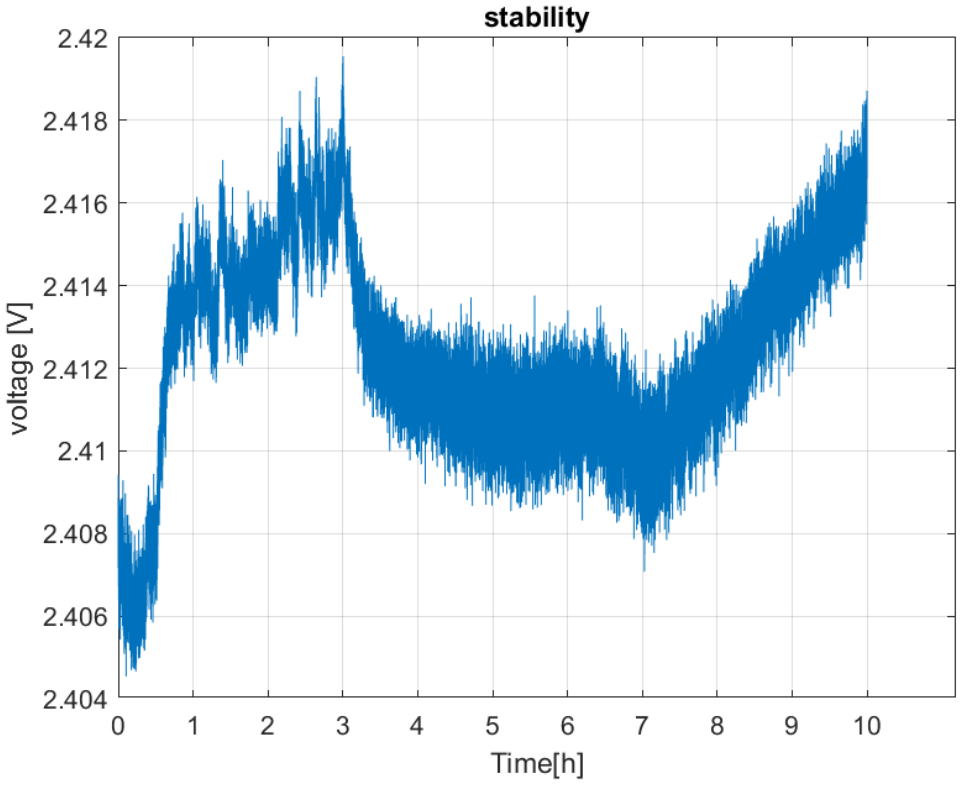

Figure 3.4: Stability test 1.

Table 3-2 shows a comparison of the previous front-end and the improved front-end in terms of stability.

Attribute	Previous Design	Improved Design
Max. Change in V	21 mV	17 mV
Drift in percentage	0.695 %	0.621 %

Table 3.2: Comparison between the previous circuit and the improved circuit in terms of stability

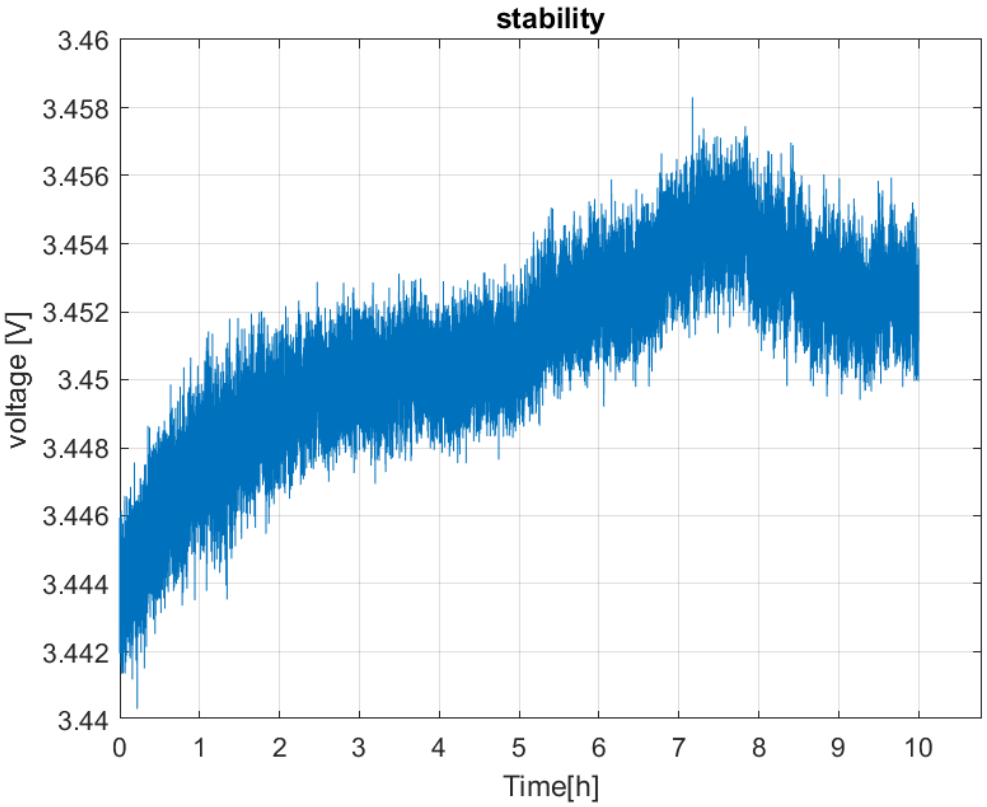

Figure 3.5: Stability test 2.

3.4 PCB design

Figure C.4 shows the main board of the circuit and Figure C.5 shows the external voltage adaptor. The size of the board is reduced to 45 mm×45 mm which is less small than the previous board (65.5 mm×75 mm).

Chapter 4

Conclusion and future work

With resoldering some part of the previous circuit, correcting several faults within the circuit and changing some components of the circuit, the results of the final circuit testing is achieved the goal of this project.
For future work, according to the new designed circuit layout, we could manufacture the PCB board and soldering the components on it to test. It should be better than it is now.

Appendix A

Microcontroller code

```
#define F_CPU  16000000UL

#include <stdlib.h>
#include <util/delay.h>

#include <stdint.h>
#include <avr/interrupt.h>
#include <avr/io.h>
#include <avr/pgmspace.h>
#include "USART.h"
#include "XBEE.h"
#include "XBEE_reception.h"

// Constants
//



// Serial port setup
#define BAUDRATE 9600
#define BAUD_PRESCALLER (((F_CPU / (BAUDRATE * 16UL))) - 1)

// Server protocol
#define SERVER_REQUEST_OFFSET 8
#define SERVER_REQUEST_START_MEASUREMENT 49
#define SERVER_ADDRESS 0xABCD

// Local variables
//
struct payload_s {
    uint32_t total_clocks_during_measurement;
```

```
32      uint32_t total_input_periods;
33  } payload;
34
35  uint8_t *TX_payload = &payload;
36  uint32_t total_clocks_during_measurement = 0;
37  uint32_t total_input_periods = 0;
38  //float input_signal_frequency = 0.0;
39
40
41
42
43  const int sinewave_length=20;
44  volatile uint8_t end_isr  ;
45  volatile uint16_t sample;
46  volatile uint16_t sample;
47   char from_slave;
48  const uint8_t sinewave_data[] PROGMEM =
        {20,23,26,29,32,34,36,38,39,40,40,40,39,38,36,34,32,29,26,23,20,
49  17,14,11,8,6,4,2,1,0,0,0,1,2,4,6,8,11,14,17};
50  uint32_t total;
51  //uint8_t LowBit;
52  //uint8_t HighBit;
53  //const uint8_t sinewave_data[] PROGMEM = {20, 26, 32, 36, 39, 40,
        39, 36, 32, 26, 20, 14, 8, 4, 1, 0, 1, 4, 8, 14};
54
55
56  ISR(TIMER0_COMPA_vect) {
57      if (sample >= sinewave_length) {
58          sample = -1;
59      }
60      else {
61          OCR1A = pgm_read_byte(&sinewave_data[sample]);
62      }
63  ++sample;
64  //end_isr=1;
65  }
66
67
68
69
70  void setPrescaler(uint8_t prescaler)
71  {
72      uint8_t mask = 0XF8;
73      ADCSRA &=mask;
74      ADCSRA|=prescaler;
75
76  }
77  void adc_init()
78  {
```

```
79      setPrescaler(4);
80      ADMUX |= (1<<REFS0);
81      ADMUX &= ~(1<<REFS1);
82      ADCSRA |= (1<<ADEN);                    //Power up the ADC
83  }
84
85
86  void adc_start()
87  {
88      ADCSRA |= (1<<ADSC);    // star convertion
89  }
90
91
92  int main(void)
93  {
94      DDRB |= (1 << DDB1)|(1 << DDB2);
95      // PB1 and PB2 is now an output
96
97      ICR1 = 40;
98      // set TOP to 16bit
99
100     OCR1A = pgm_read_byte(&sinewave_data[0]);;
101     // set PWM for 25% duty cycle @ 16bit
102
103     TCCR1A |= (1 << COM1A1)|(1 << COM1B1);
104     // set none-inverting mode
105
106     TCCR1A |= (1 << WGM11);
107     TCCR1B |= (1 << WGM12)|(1 << WGM13);
108     // set Fast PWM mode using ICR1 as TOP
109
110     TCCR1B |= (1 << CS10);
111     // START the timer with no prescaler
112
113     //setting TIMER0 for updating the values for PWM period
114
115     // Set up Timer 0 to send a sample every interrupt.
116     cli();   // disable interrupts
117     // Set CTC mode (Section 15.9.2 Clear Timer on Compare Match)
118     // WGM = 0b0100, TOP = OCR1A, Update 0CR1A Immediate (Table 15-4)
119     // Have to set OCR1A *after*, otherwise it gets reset to 0!
120     TCCR0A = 0;
121     TCCR0B = 0;
122
123     TCCR0B |= (1 << WGM02);
124     TCCR0A |= ((1 << WGM01) | (1 << WGM00));
125
126     // No prescaler, CS = 0b001 (Table 15-5)
127     TCCR0B = (TCCR0B & ~(_BV(CS02) | _BV(CS01))) | _BV(CS00);
```

```
128     // Set the compare register (OCR1A).
129     // OCR1A is a 16-bit register, so we have to do this with
130     // interrupts disabled to be safe.
131     OCR0A = 10;//F_CPU / SAMPLE_RATE;     // 16e6 / 8000 = 2000
132     // Enable interrupt when TCNT1 == OCR1A (p.136)
133     TIMSK0 |= _BV(OCIE0A);
134 
135     sample = 0;
136     sei();  // enable interrupts
137     adc_init();
138     XbeeUSART_init();
139     adc_start(); //start the first conversion and ignore it
140     while(ADCSRA & (1<<ADSC));
141     while (1)
142     {
143         //_delay_ms(10);
144         //sum all measurement
145         total = 0;
146         for(int i=0;i<4096;i++)
147         {
148 
149             adc_start();
150             while(ADCSRA & (1<<ADSC));
151             total+=ADC;
152         }
153 
154         payload.total_clocks_during_measurement = total;//
    total_clocks_during_measurement;
155         payload.total_input_periods = 1234;//total_input_periods;
    THIS IS DUMMY
156 
157         XBee_TX_Request(SERVER_ADDRESS, TX_payload, sizeof(struct
    payload_s));
158 
159         // we have a working Fast PWM
160     }
161 }
```

Appendix B

Server codes

Check sensitivity

```
%%%% External functions used in this script %%%%
% send_API_2.m --> sends request to gather frequency readings
% rec_API_2.m --> receives freuency readings from cap-sensor nodes
% send_US_req.m --> sends request for Ultrasound sensor's readings
clear all;
close all;
clc;
%fclose(instrfind);

minutes_of_run = 1;     %each sub-run
hours = 6;               %total run

serialPort = 'COM4';
s = serial(serialPort);
set(s,'BaudRate',9600);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

data_received = zeros(200000,1);

fopen(s);

start_time = clock;
flushinput(s);
flushoutput(s);

data_index = 1 ; %array index to store US/cap sensor values
i = 1;
pause(1);
```

```
30 end_at = 60*minutes_of_run;
31 j=0;
32
33 while j < hours
34 t0 = clock;
35
36 while etime(clock, t0) < end_at
37
38     if (s.BytesAvailable > 16)
39
40
41         [address, total_clocks_during_measurement,
    total_input_periods] = rec_API_2b(s);
42
43             if (total_clocks_during_measurement > 0)
44
45              data_received(i) =(double(total_clocks_during_measurement
    )/64.0)*(5.0/65536.0);
46
47              disp(data_received(i));
48
49              i=i+1;
50
51             end
52       end
53 end
54
55 t0=clock;
56
57 j=j+1;
58
59 data=nonzeros(data_received);
60 save(sprintf('frontend2_%s.mat', datestr(now,'mm-dd HH-MM')),'data');
61
62 end
63
64 data=nonzeros(data_received);
65 save(sprintf('frontend2_%s.mat', datestr(now,'mm-dd HH-MM')),'data');
66
67 figure(1);
68 plot(data);
69 ylabel('voltage [V]');
70 title('sensitivity');
71 grid on;
72
73 xticks(1:(length(data)/6):length(data)+1);
74 xlabel('Distance [m]');
75 xticklabels({'0.25','0.50','0.75','1.00','1.50','2.00'});
```

```
saveas(gcf,sprintf('frontend2_FINAL_25cmto200cm_NOPERSON_%s.png',
    datestr(now,'mm-dd HH-MM')));
```

Check stability

```
%%%%% External functions used in this script %%%%%
% send_API_2.m --> sends request to gather frequency readings
% rec_API_2.m --> receives freuency readings from cap-sensor nodes
% send_US_req.m --> sends request for Ultrasound sensor's readings
clear all;
close all;
clc;
%fclose(instrfind);

minutes_of_run = 60;     %each sub-run
%sub_hours = 1;
hours = 12;               %total run

serialPort = 'COM4';
s = serial(serialPort);
set(s,'BaudRate',9600);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

data_received = zeros(200000,1);

fopen(s);

start_time = clock;
flushinput(s);
flushoutput(s);

data_index = 1 ; %array index to store US/cap sensor values
i = 1;
pause(1);
end_at = 60*minutes_of_run;
j=0;

while j < hours
t0 = clock;

while etime(clock, t0) < end_at

    if (s.BytesAvailable > 16)


```

```
42          [address, total_clocks_during_measurement,
    total_input_periods] = rec_API_2b(s);
43
44              if (total_clocks_during_measurement > 0)
45
46               data_received(i) =(double(total_clocks_during_measurement
    )/64.0)*(5.0/65536.0);
47
48               disp(data_received(i));
49
50               i=i+1;
51
52              end
53        end
54 end
55
56 t0=clock;
57
58 j=j+1;
59
60 data=nonzeros(data_received);
61 save(sprintf('frontend2_%s.mat', datestr(now,'mm-dd HH-MM')),'data');
62
63 end
64
65 data=nonzeros(data_received);
66 save(sprintf('frontend2_%s.mat', datestr(now,'mm-dd HH-MM')),'data');
67
68 figure(1);
69 plot(data);
70 ylabel('voltage [V]');
71 title('stability');
72 grid on;
73 xticks(1:(length(data)/11):length(data)+1);
74 xlabel('Time[h]');
75 xticklabels({'0','1','2','3','4','5','6','7','8','9','10'});
76 saveas(gcf,sprintf('frontend2_FINAL_stability_start_17_30_%s.png',
    datestr(now,'mm-dd HH-MM')));
```

Appendix C

PCB design

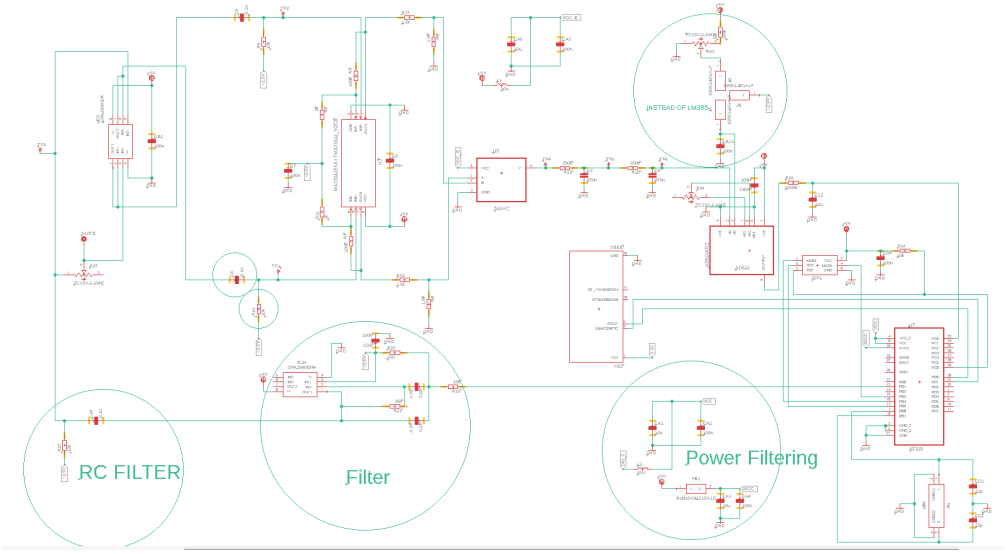

Figure C.1: Schematic of main circuit

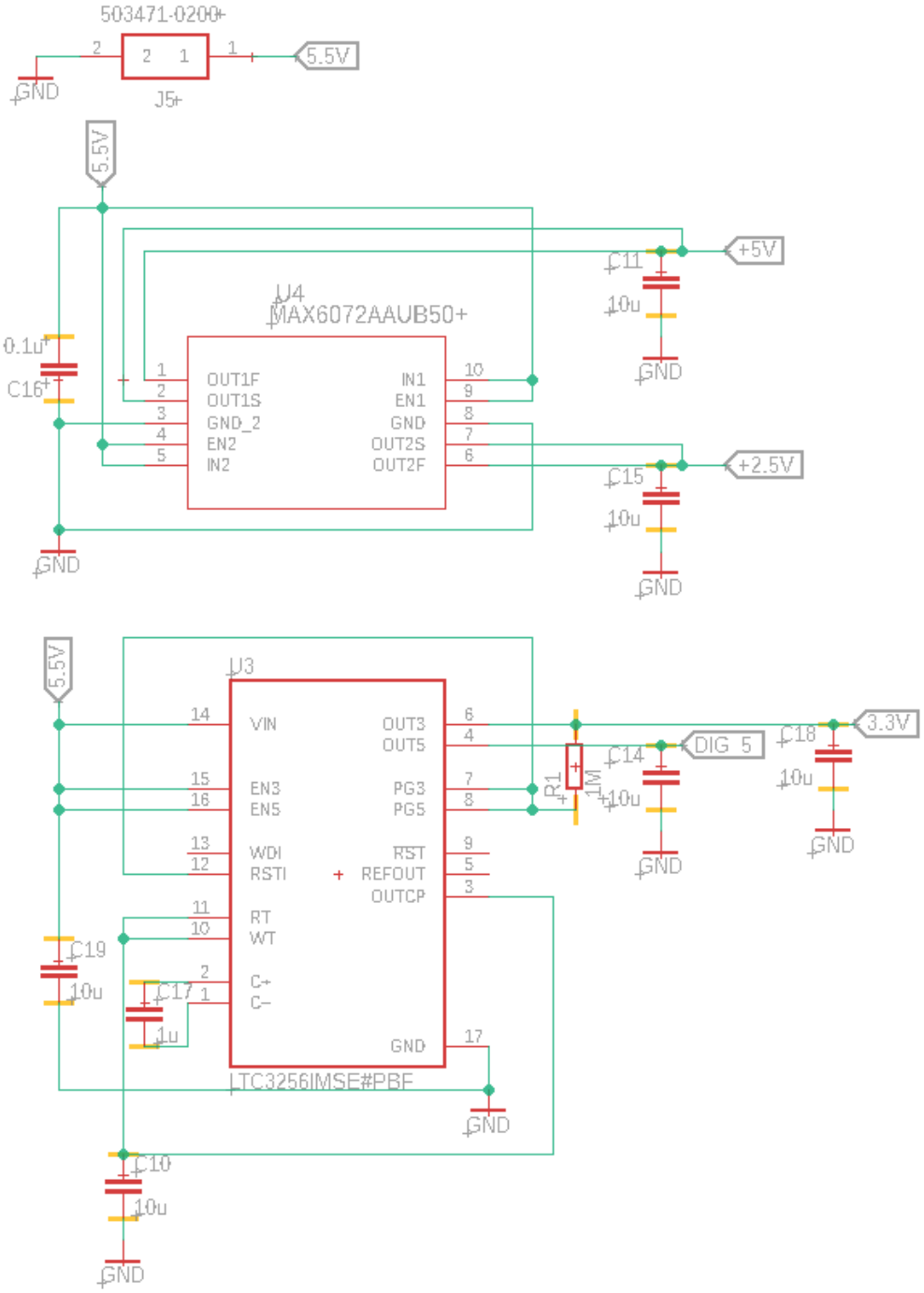

Figure C.2: Schematic of power supply

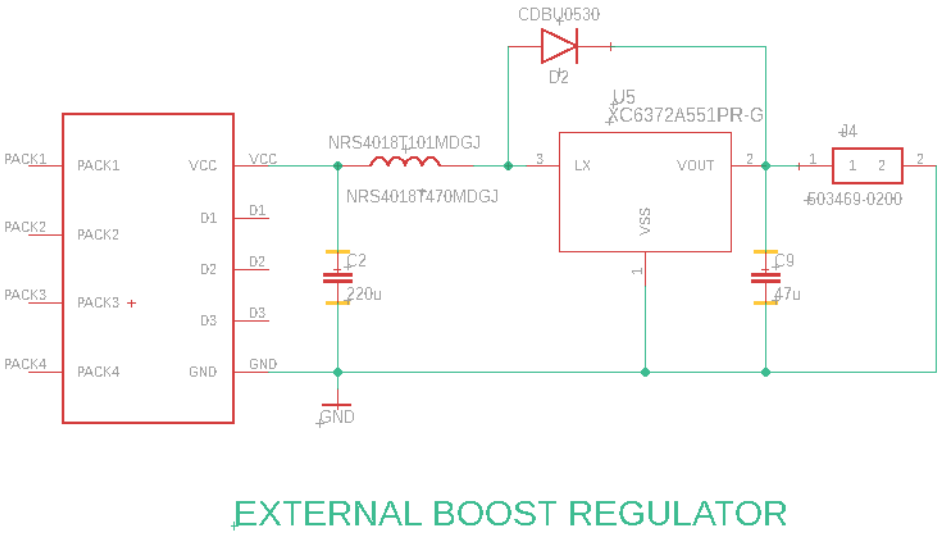

Figure C.3: Schematic of external boost circuit

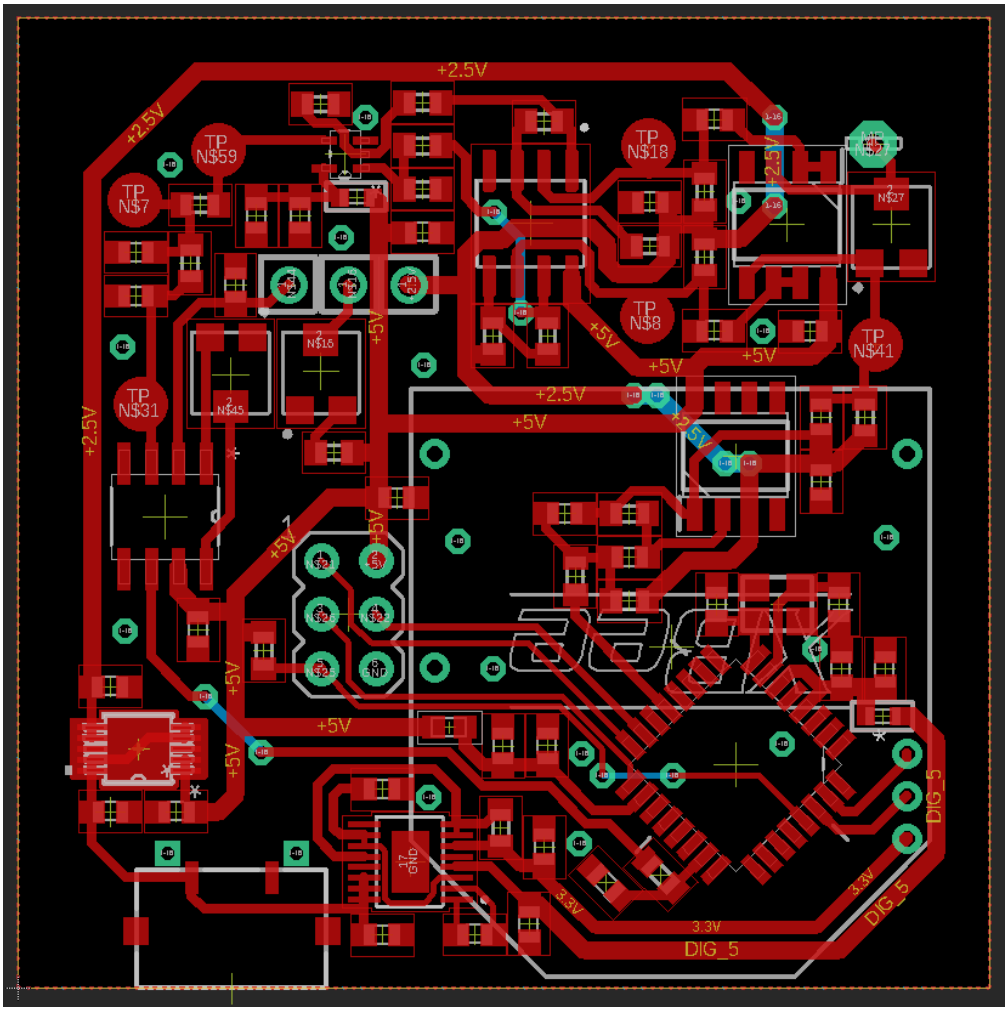

Figure C.4: Layout of main board

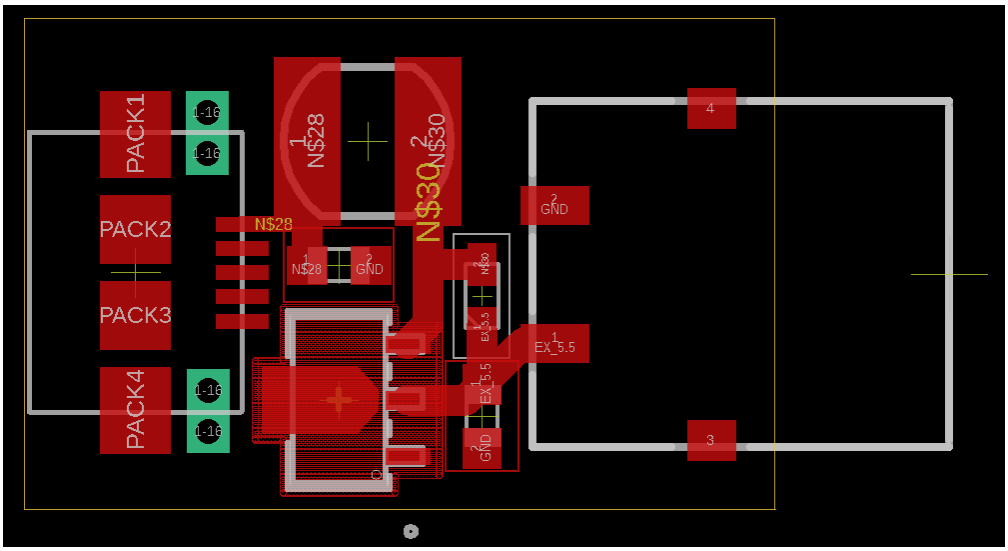

Figure C.5: Layout of external boost circuit

Bibliography

[1] T. Kivimaki, T. Vuorela, T. Vuorela, and J. Vanhala. «A Review on Device-free passive indoor positioning methods». In: *International Journal of Smart Home* vol. 8, no. 1 (2014), pp. 71–94 (cit. on p. 1).

[2] F. Zafari, A. Gkelias, and K. Leung. «A survey of indoor localization systems and technologies». In: *IEEE Communications Surveys and Tutorials* , vol. 21, Issue: 3 (2019), pp. 2568–2599 (cit. on p. 2).

[3] M. Youssef, M.Mah, and A. Agrawala. «Challenges: Device-free passive localization for wireless environments». In: *Proceedings of the 13th Annual International Conference on Mobile Computing and Networking* (2007), pp. 222–229 (cit. on p. 2).

[4] T. Grosse-Puppendahl, C. Holz, G. Cohn, R. Wimmer, O. Bechtold, S. Hodges, M. S. Reynolds, and J. R. Smith. «Finding common ground: A survey of capacitive sensing in human-computer interaction». In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (2017), pp. 3293–3315 (cit. on p. 3).

[5] A. Ramezani Akhmarehl, M. T. Lazarescu, O. Bin Tariq, and L. Lavagno. «A tagless indoor localization system based on capacitive sensing technology». In: *Sensors* , vol. 16, no. 9 (2016) (cit. on p. 3).

[6] J. Iqbal, M. T. Lazarescu, O. Bin Tariq, and L. Lavagno. «Long range, high sensitivity, low noise capacitive sensor for tagless indoor human localization». In: *2017 7th IEEE International Workshop on Advances in Sensors and Interfaces (IWASI)* (2017) (cit. on p. 4).

[7] J. Smith, T. White, C. Dodge, J. Paradiso, N. Gershenfeld, and D. Allport. «Electric field sensing for graphical interfaces». In: *IEEE Computer Graphics and Applications* , vol. 18 (May 1998), pp. 54–60 (cit. on p. 5).

[8] P. Poolad, supervisors L. Lavagno, and M. T. Lazarescu. «Optimization of capacitive sensor front-end for indoor human localization and identification». In: (2017) (cit. on p. 5).

[9] R. Gambotto, supervisor L. Lavagno, and M. T. Lazarescu. «Analog front-ends for long range capacitive sensing for IoT indoor human localization». In: (2017) (cit. on p. 5).

[10] F. H. Christian, supervisor L. Lavagno, and M. T. Lazarescu. «Front-ends for range capacitive sensor». In: (2018) (cit. on p. 5).

[11] J. Iqbal, M. T. Lazarescu, A. Arif, and L. Lavagno. «High sensitivity, low noise front-end for long range capacitive sensors for tagless indoor human localization». In: *IEEE 3rd International Forum on Research and Technologies for Society and Industry* (2017) (cit. on pp. 6, 7).