Master's Degree in Electronic Engineering

Thesis

Front-end for long range capacitive sensor

Supervisors
Mihai Teodor Lazarescu
Luciano Lavagno

Candidate
Fotso Hondjie Christian Ulrich

Academic year 2018-2019

To my Family

Abstract

In the past decades, wireless localization technologies have undergone considerable progress. Since most people spend their time in indoor environment, indoor human localization play an important role in home automation, security system and assisted living scenario.They can be used to efficiently control and optimize the way our surrounding device consume energy in your daily life e.g. lighting system, heating system, security system.

Many technologies for human indoor tracking and localization have been developed over the years.The application for human localization are usually based on imaging techniques (camera, lidar), Capacitive sensing, microwave (RFID, WI-FI, ultrasound), inertial sensors dead reckoning (gyroscope and accelerometer). The main challenge here is to have a system that must combine robustness and reliability while meeting the requirements of accuracy, critical time, energy efficiency, privacy and deployment cost. Capacitve sensing seems to be a best candidate to fulfill those requirements. Capacitive sensing technology is widely used in smartphone, wearable device, automotive industry and smart objects. Despite their ubiquity, those application have in common that the sensing range is in order of centimeters or less.

Many research have been carried on capacitive sensor. Almost all of them were targeted for short-range application. Capacitive measurement made for long range suffer from low sensitivity, environmental noise and temperature change.

This thesis is embedded in a research project which purpose is to designed capacitive sensor for long range human tracking and localization. Previous experiments on long range capacitive sensor have been carried by the research group. They designed a node sensor based on the relaxation oscillator 555 timer configured as astable multivibrator oscillator. A change on the plate sensor capacitance lead to a change of frequency of the oscillator. Since the coupled capacitance of a person standing in from of the plate sensor depends on the the distance, this system have been used to evaluate the distance between the sensor plate and the

human body. The result obtained where quite good. Small sensor plates(4x4cm) could detect the presence only up to 80cm. 8x8cm plate can detect the presence up to 100cm. Finally a 16x16 cm plate showed reliable detection up to 140cm.

This promising approach raised some problems. The 555-based oscillator is very sensitive to voltage noise hence, to the environmental noise collected by the sensor plate. Also we have to wait 1s to perform one measurement. The dynamic of the sensor could change within this amount on time and affect the measurement.

To enhance the performance a new front-end have been designed by some members of the group research and has not been tested. My contribution was about the testing and the characterization of the new front end. During the testing part i realized that some components were not compliant with the design constraints, others were mounted in a wrong way on the pcb and some connections were not effective on the pcb board. I tried to solve the problems and the final results were quite good. A 16x16cm sensor plate was able to detect the presence of a person up to 150cm. The measuring time has been reduced to 53.248ms.

Acknowledgements

I would like to express my deepest gratitude to my supervisor Prof. Mihai Teodor Lazarescu and co-supervisor Prof. Luciano Lavagno,for giving me this thesis opportunity, for their useful guidance, and considerable encouragements throughout the thesis work.

I would also like to thanks Osama Bin Tariq and Usman Jamal for sharing their know how and assistance whenever I faced difficulties.

Finally,I would like to express my indebtedness to all of those who have given me constant support and love during the completion of the thesis, specially to Lyne.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview on indoor localization

The wide-scale proliferation of smart phones and other wireless devices in the last couple of years has resulted in a wide range of services including indoor localization. Indoor localization is the process of obtaining a device or user location in an indoor setting or environment. Indoor device localization has been extensively investigated over the last few decades, mainly in industrial settings and for wireless sensor networks and robotics[1].

However, it is only less than a decade ago since the wide-scale proliferation of smart phones and wearable devices with wireless communication capabilities have made the localization and tracking of such devices synonym to the localization and tracking of the corresponding users and enabled a wide range of related applications and services. User and device localization have wide-scale applications in health sector, industry, disaster management, building management, surveillance and a number of various other sectors [1].

As operating environments, homes place certain constraints on positioning systems. For instance, a 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. To be acceptable for home use, a positioning system needs to be passive, device-free, and unobtrusive.

In passive positioning, users do not need to perform any specific activities for the system to position them; the system operates automatically so that users do not necessarily notice its presence. Device-free means that people tracked by the system do not need to carry devices or tags. This is important, as people are generally unwilling to wear extra devices, especially at home. In addition, individuals can forget to put a device on, especially after changing their clothes. Another challenge is that every individual in a positioning area should have a device, which makes having visitors problematic.

Furthermore, mobile positioning devices use batteries and require regular monitoring and changing[2]. An non exhaustive exiting technologies on device-free for human human identification is presented here.

- **Pressure sensors** are the most traditional positioning system technology, the initial way to construct a positioning system was to install pressure-sensitive sensors on or under a floor surface. If installed under the floor surface, the sensors are invisible to the user, but the installation is difficult. This type of installation is also not always possible, as it requires flexible flooring and sufficient installation space beneath the floor surface. One benefit of this system is that it enables user identification, as the pressure generated corresponds to a persons unique weight and gait [2].

- **Image recognition** is to equip the place in which the location is desired with a set of sensors (cameras, LIDAR) which remain at a fixed and known position. These sensors will identify the object to be monitored and deduce the position. Imaging systems often achieve a very good accuracy in a closed environment, moreover they do not require any device attached to the user so It is device free. However this technique have several drawback.It requires to much power consumption to process the data. It require high cost of installation and maintenance. People may fell uncomfortable because their privacy is exposed.

- **Ultrasound** technology mainly relies on time of flight measurements of ultrasound signals (greater than 20KHz) and the sound velocity to calculate the distance between a transmitter and a receiver node. It has been shown to provide fine-grained indoor localization accuracy with centimeter level accuracy and track multiple mobile nodes at the same time with high energy efficiency and zero leakage between rooms. Usually, the ultrasound signal transmission is accompanied by an RF pulse to provide the necessary synchronization. However, unlike RF signals, the sound velocity varies significantly when humidity and temperature changes; this is why temperature sensors are usually deployed along with the ultrasound systems to account for these changes. Finally, although complex signal processing algorithms can filter out high levels of environmental noise that can degrade the localization accuracy, a permanent source of noise may still degrade the system performance severely [1].

- **RFID** technology is primarily intended for transferring and storing data using electromagnetic transmission from a transmitter to any Radio Frequency (RF) compatible circuit . An RFID system consists of a reader that can communicate with RFID tags. The RFID tags emit data that the RFID reader can read using a predefined RF and protocol, known to both the reader and tags a priori. There are two basic types of RFID systems :

  - **Active RFID**: Active RFIDs operate in the Ultra High Frequency (UHF) and microwave frequency range. They are connected to a local power source, periodically transmit their ID and can operate at hundreds of meters from the RFID reader. Active RFIDs can be used for localization and object tracking as they have a reasonable range, low cost and can be easily embedded in the tracking objects. However, the active RFID technology cannot achieve submeter accuracy and it is not readily available on most portable user devices.
  - **Passive RFID**: Passive RFIDs are limited in communication range (1-2m) and can operate without battery. They are smaller, lighter and cost less than the active ones; they can work in the low, high, UHF and microwave frequency range. Although they can be used as an alternative to bar-codes, especially when the tag is not within the line of sight of the reader, their limited range make them unsuitable for indoor localization. They can be used for proximity based services using brute force approaches7, but this will still require modifications to the existing procedure used by passive RFIDs such as transmitting an ID that can be used to identify the RFID and help [1].

- **Visible Light** is an emerging technology for high-speed data transfer that uses visible light between 400 and 800THz, modulated and emitted primarily by Light Emitting Diodes (LEDs). Visible light based localization techniques use light sensors to measure the position and direction of the LED emitters. In other words, the LEDs (acting like the iBeacons) transmit the signal, which when picked up by the receiver/sensor can be used for localization. For visible light, AoA(angle of Arrival) is considered the most accurate localization technique. The advantage of visible light based localization is its wide scale proliferation (perhaps even more than WiFi). However, a fundamental limitation is that line of sight between the LED and the sensor(s) is required for accurate localization[1].

The main challenge here is to have a system that must combine robustness and reliability while meeting the requirements of accuracy, critical time, energy efficiency, privacy and deployment cost. Capacitive sensing technique seems to be the best candidate to fulfill these requirements.

1.2 Capacitive sensing

Capacitive sensing technique exploit the fact that human body can interact with electric field and generate coupled capacitance with surrounding objects as we can see on hte figure1.1 from [3]. Many application are based on this technique e.g. touch screen for smartphone, wearable device, automotive industry.

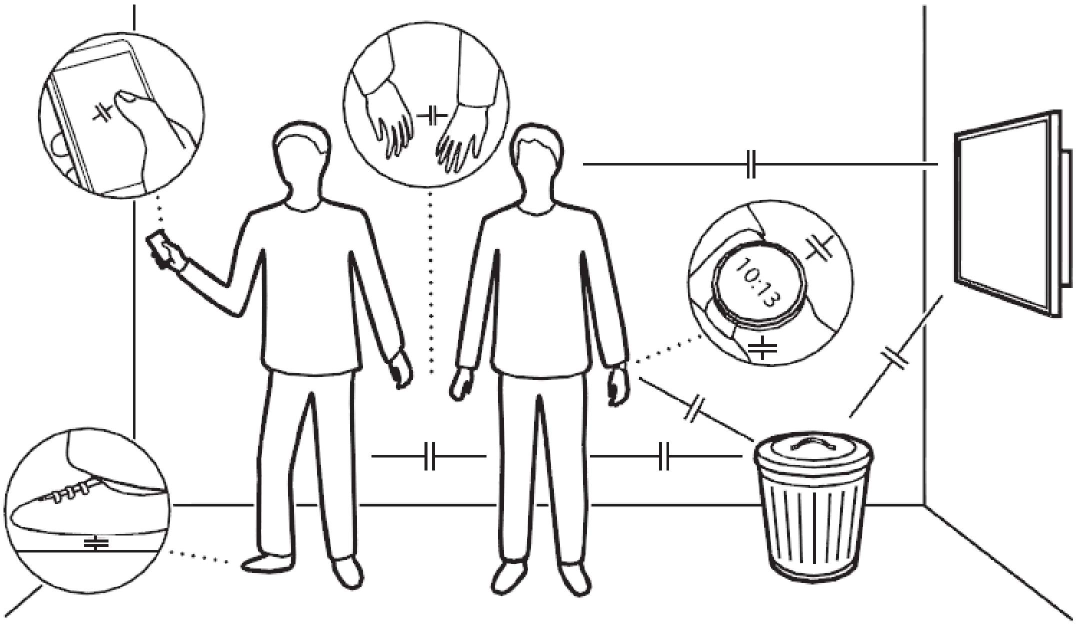

Figure 1.1. Coupling capacitances between human body and environment

The figugre 1.2 from [3] depict a lumped circuit model of human body interaction and environment.

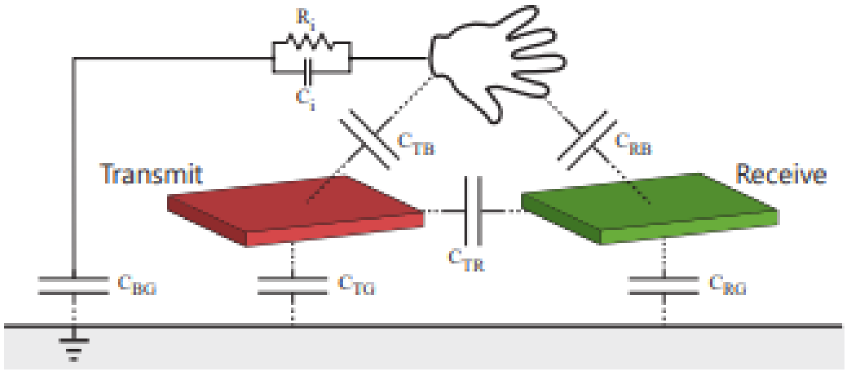

Figure 1.2. Lumped circuit model

1.2.1 Capacitive sensing technique

Sensing techniques can be described as either active or passive. They can also be grouped into four different operating modes: loading, shunt, transmit, and receive [3].

- **Active mode** In active capacitive sensing, a known signal is generated on the transmit electrode(s), capacitively coupled onto the body part, and then coupled into the receive electrode(s). The presence and movement of the body part can be sensed by measuring the strength of the signal coupled onto the receive electrode(s)[3].

- **Passive mode** While active sensing systems must actively generate an electric field, passive sensing systems rely on existingexternal or ambientelectric fields which are passively sensed[3].

- **Loading mode** is the simplest and most common type of capacitive sensing. Here, the same electrode is used for both transmit and receive. The body capacitively loads the electrode and causes a displacement current to flow through the body to ground. As the body gets closer to the electrode, the capacitive coupling (CT B) increases and so does the displacement current, which allows the system to sense the proximity of the body. Since the transmit and receive electrodes are the same, and are both part of the sensing system, there is no passive variant of loading mode [3].

- **Shunt mode** In shunt mode, the transmit and receive electrodes are distinct, and thus there is some capacitive coupling between them . When the human body is in proximity to the electrodes, it will capacitively couple to both the transmit and receive electrodes with about the same order of magnitude as the coupling between the electrodes . This will cause a displacement current to flow through the body to ground , and will thus reduce the displacement current flowing from the transmit to the receive electrode (. By measuring the decrease in displacement current at the receive electrode, the bodys proximity can be determined [3].

- **Transmit mode** Transmit mode is similar to shunt mode, except that the body is very close to the transmitter. This proximity means that the coupling between the body and the transmitter is much greater than the coupling between the body and the receiver or between the transmitter and the receiver [3].

- **Receive mode** : Receive mode is the inverse of transmit modethe body is very closely coupled to the receive rather than the transmit electrode.

A summary of the capacitive sensing methods is shown on 1.3.

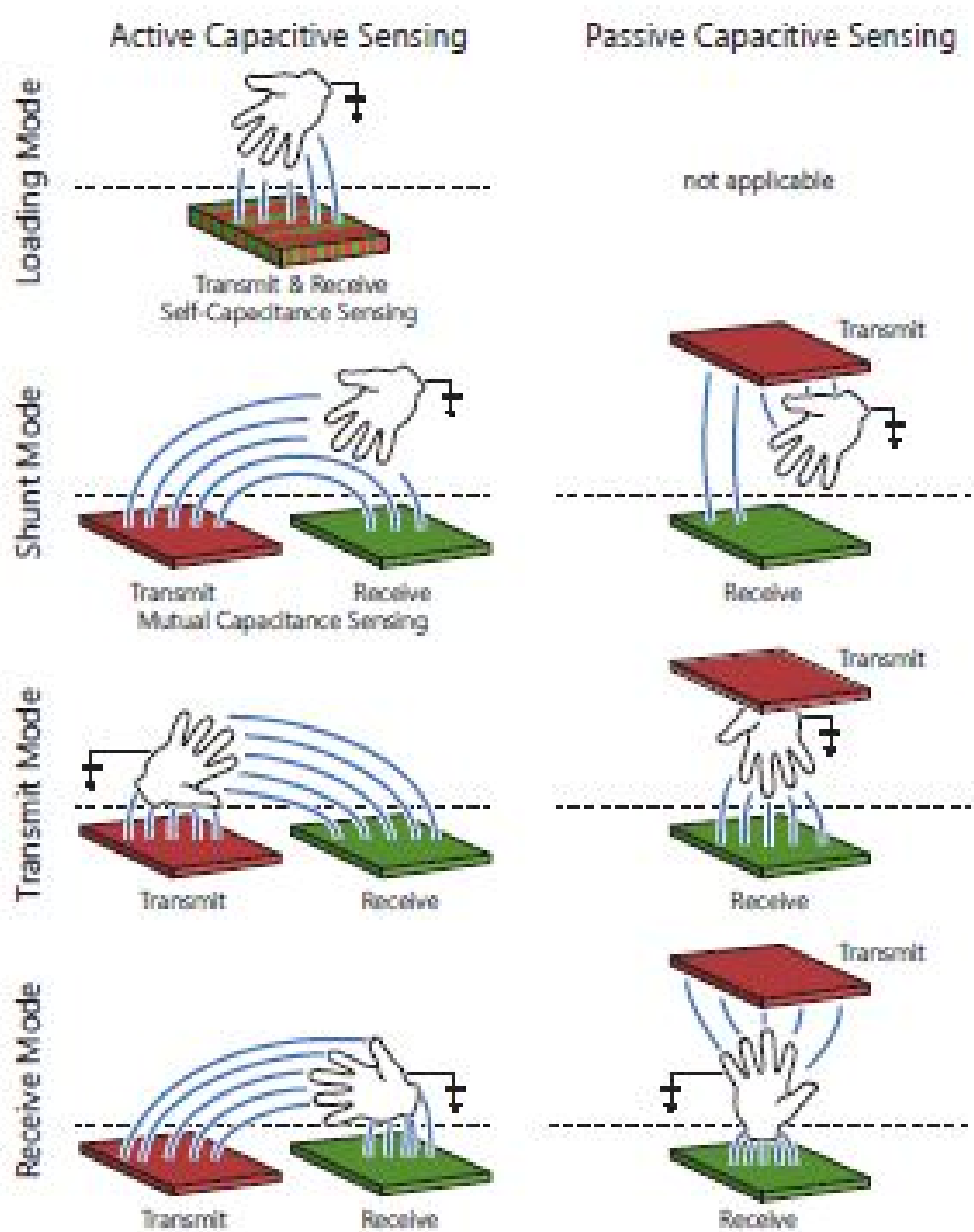

Figure 1.3. Capacitive sensing tecniques

1.2.2 Advantages of Capacitive Sensing

- **No line-of-sight needed**. While non-conductive objects like glass, plastics or wood change the measured capacitance through their dielectric value, this influence is low. When the distance between sensor electrode and non-conductive object stays the same, no capacitance changes are effected. In addition, these objects usually do not decrease sensor resolution, but even increase it. This is due to their dielectric constant, which is higher than that of air, causing the capacitance to increase slightly. This property enables sensors to be installed in walls or inside casings. However, the thicker the material, the greater the minimal distance between sensor electrode and tracked object - and thus the lower the available maximum resolution. [4]

- **Efficient data acquisition**. Post processing and analysis can be done in a microcontroller. By lowering the update rate, resolution can be easily increased.[4]

- **Cheap hardware**. Capacitive sensors can be built with few inexpensive off-the-shelf components.

1.2.3 Limitations of Capacitive Sensing

- **Ambiguous Data**. Capacitive sensors offer only one-dimensional data: the capacitance provided by the electrode. Thus the system has to decide, what a capacitance change means. Most times it is caused by a conductive object moving farther from or closer to the sensor electrode. However, changes in the dielectric value of the isolator between object and electrode, which is usually air, also affect the capacitance. [4]

- **Quickly Decaying Resolution**. A major hurdle for incorporating capacitive sensors in pervasive applications is their limited range. If the size of the object to be tracked and the dielectric value of the isolator can be controlled, the capacitance measured by a sensor depends mainly on the distance between electrode and object. The generic equation for the capacitance of a capacitor,$C \approx \frac{\epsilon.A}{d}$, implies that it is inversely proportional to the distance between the plates. However, this is not the case for greater distances between the plates. The farther apart the two plates are, the smaller their overlapping area gets relatively to their surrounding Thus a more realistic model is $C \approx \frac{\epsilon.A}{d^x}$; where x is between 1 3, depending on the environment. This rapid decrease of capacitance with increasing distance poses a problem when trying to track objects at distances greater than $\approx 10cm$. The lower resolution has to be accounted for, e.g. by placing another sensor opposite of the first so that the object is always close to one sensor. Furthermore, additional sensor data and sophisticated filtering can be useful[4].

- Exposure to External Influences As a result of the ambiguous data mentioned above, capacitive sensor measurements are also exposed to other, unwanted influences. If conductive objects which shall not be tracked get into sensor range, they increase capacitance, tainting the measurements. objects. Other unwanted influences include electric fields emitted by electric devices or humidity and temperature changes. Electric fields also negatively affect measurements. Changes in the dielectric (like increasing humidity) may cause erroneous measurements. Additionally, capacitors are sensitive to temperature changes. As many different effects influence measurements, capacitive sensors usually have to be calibrated, relating measured capacitance to proximity. Depending on the needed sensitivity, these calibrations have to be done regularly, especially when the environment of the sensor changes[4].

Chapter 2

Research project

2.1 Previous work

This research project goals is to used capacitive sensor of indoor human tracking and localization. The actual set up for experiment and testing is conducted in a in 3X3m room where four sensors are installed on as shown as on fig 2.1. Then a processing unit station compute the measurement and estimate the position using machine-learning algorithms.

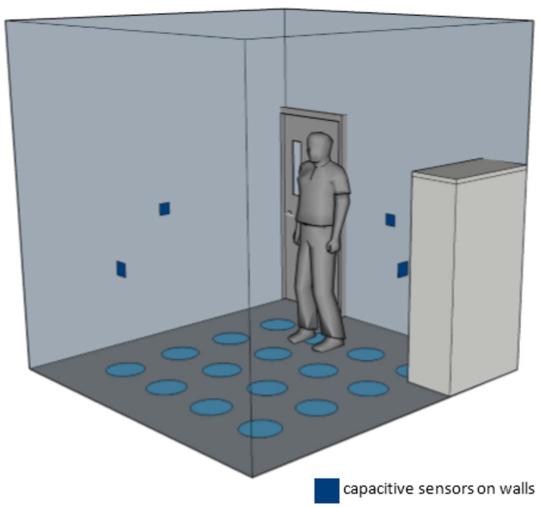

Figure 2.1. Sensor deployment set up

The sensor node used in the previous experiments, performs and indirect measurements of the capacitance by measuring the frequency of a relaxation oscillator.

It uses a transducer operating in load mode whose capacitance varies with the distance of a human body and other surrounding objects. The oscillator is based on a 555 timer on fig 2.2 integrated circuit set in astable mode. The output from the 555 timer is a continuous pulse waveform whose frequency depends on the values of the two resistors R1 and R2 and the capacitor C which represent the capacitance of the plate sensor(transducer).

The capacitance change is proportional to the distance between the sensor plate and the human body. Therefore, the distance between the body and the sensor plate can be estimated from the capacitance variation.

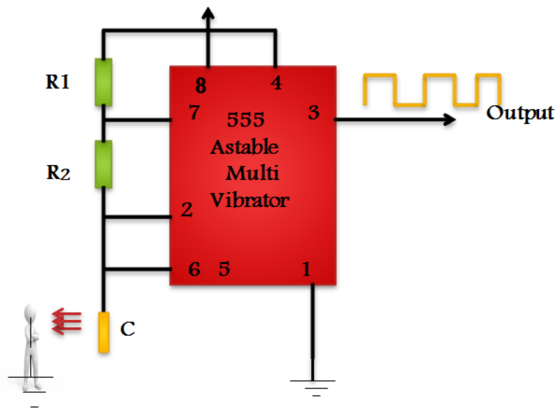

Figure 2.2. 555 IC set up

The frequency is measured by counting the periods of the oscillator in an interval of 1 s. Then, the data are acquired and prepared to be sent using an Arduino Uno board.Finally data are sent to to the processing using an Xbee radio. the x-Bee radio. The node is shown on fig 2.3

Their obtain remarkable result.For example a 16x16 plate showed reliable detection up to 140cm. This sensor node showed some limitation. To perform one measurement we have to wait one second.For localization systems this among of time is to high and also the dynamic of the sensor could change during that time and affect the measurement.Moreover, the 555-based RC oscillator that their used is very sensitive to voltage noise, hence to most environmental noise captured by the capacitive sensor plate[5].

In order to enhance the performance of the measurements a new front end has been designed and has not been tested.This thesis is focused on the testing and

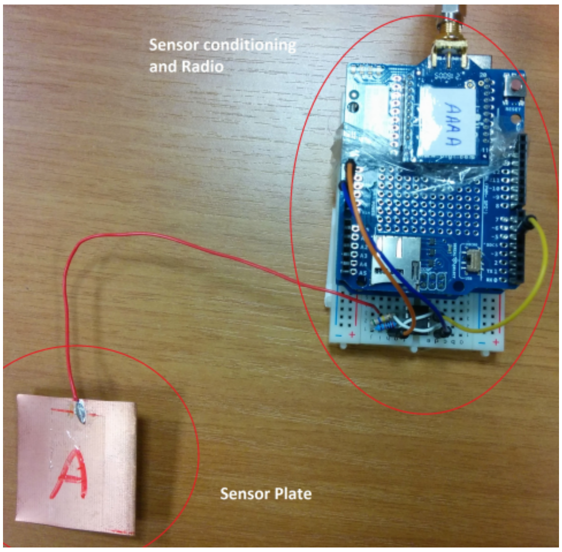

Figure 2.3. Sensor node based on 555 IC

the characterization of the new node.

2.2 New front-end description

The new front has been designed to measure the phase difference introduced by an excitation signal and the response signal form the exited system. The initial front end is shown on 2.4. On the green path, the excitation signal enters a low pass filter made of R1 and Cs, where Cs is the equivalent capacitance of the transducer of the sensor. The behavior of the filter depends on the value of Cs, whose value is proportional to the distance between the transducer plate and the human body. The output filter and the excitation signal then pass through two voltage follower. High pass filter after the repeater eliminate part of the low noise frequency from the buffered signal.

The phase shift sensitivity of an RC filter is maximum at the cut off frequency. Hence, to enhance the sensitivity of the sensor the R1 resistor can be adjustated in order to tune the filter R1Cs cut-off frequency to the excitation signal frequency 10Khz.

A narrow-band pass filter is then used to remove the noise captured by the plate of the capacitive transducer. It should be implemented using 4th order Butterworth bandpass filter centered at the excitation frequency (10 MHz) with a quality factor Q=5. It should be implemented using 4th order Butterworth bandpass filter centered at the excitation frequency (10 Mhz) with a quality factor Q=5.

The signal of the two front end paths are then each passed through a hysteresis comparator to prepare them for the phase discriminator circuit that is implemented using a XOR logic gate. The second a 2nd order lowpass filter at the output of the discriminator passes only the continue (DC) term. When no one is around the sensor, the cut-off frequency of the R1Cs filter is approximately the excitation signal frequency of 10 KHz and the resulting phase shift will be $\frac{\pi}{4}$. So when one is around the sensor there will be an static output due this initial phase shift difference. The AD8429 strumentation amplifier (Analog Devices, Inc) should remove this static difference. The strumentation amplifier output should be sampled by the atmega328p [6] microcontroller. Finally the xbee [7] radio should to send the sampled data to the processing using.

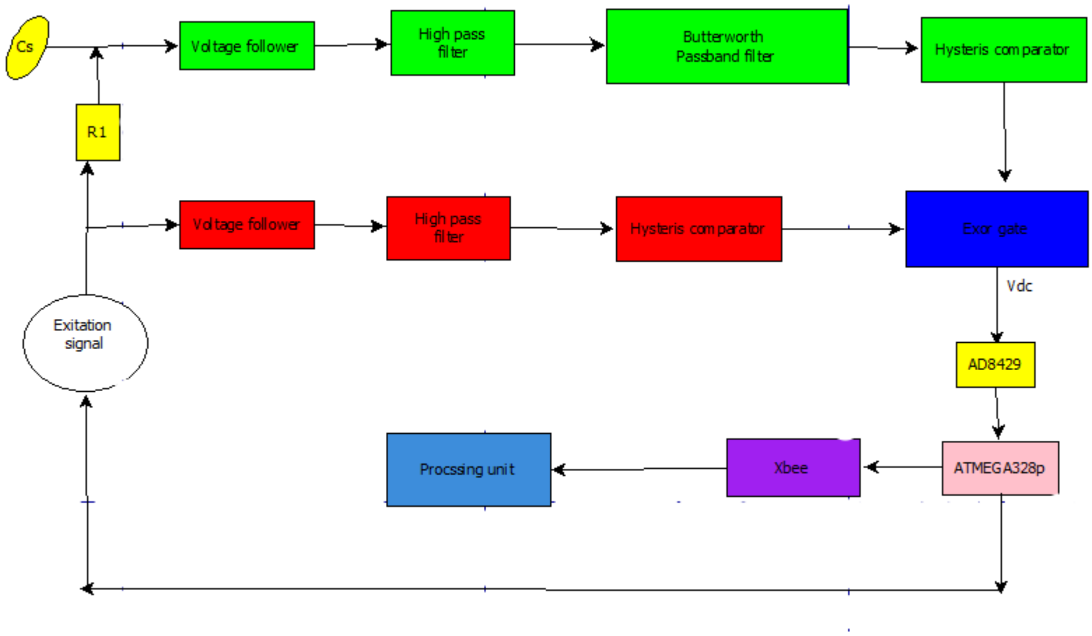

Figure 2.4. New end Front interface

2.3 Board inspection and testing

2.3.1 Board inspection

Since the board has been never test, i firstly check if the IC component mounted on the board were compliant with the design constraints. So i look at the datasheet of the components to ensure that there were the right one and also to check if there were configured in the proper way. It came out that the instrumentation amplifier AD8429 [8] was not compliant with the design constraints. From the datasheet of the AD8429 instrumentation amplifier, we can see that it is designed to work with in a dual power supply. In our case the the instrumentation amplifier should work with a single power single supply configuration and should be rail to rail. The AD623 [9] instrumentation amplifier turn out to be compliant with the design constraints. It can work with a single power supply and it is rail to rail. So the AD8429 instrumentation amplifier present on the board has been substituted by the AD623 instrumentation amplifier.

Then looking at the schematic of the new front end shown on 2.5 the new front end, it came out that some pin was floating where there should be connected. As example the connection of all the three potentiometer present on the pcb board have been placed in the wrong way as show on 2.7.

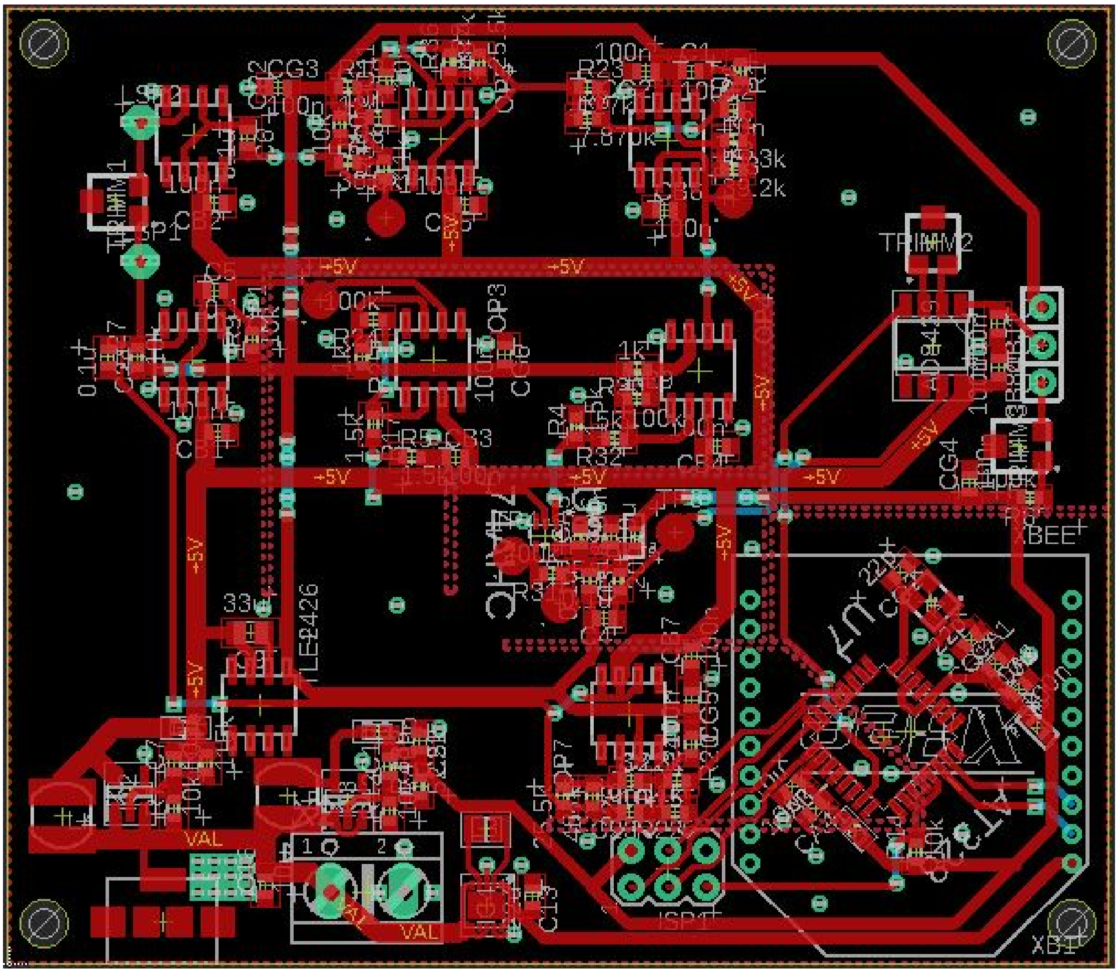

Figure 2.5. New Front end pcb

Due to delivery time, instead of ordering a new pcb the corrections has been done directly on the initial PCb by soldering new components. The front before testing can be is shown on figure 2.8.

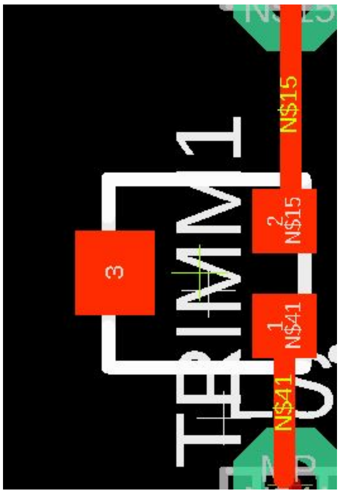

Figure 2.6. Wrong trimmer connection

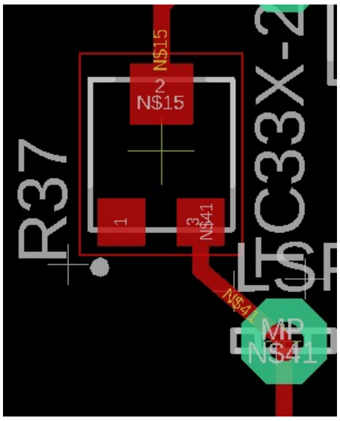

Figure 2.7. Right trimmer connection

2.3.2 Testing Setup

The result of your experiment depends on the sensitivity of the filter. As mention before the resistor R1 should be tuned in order to have the best sensitivity. We obtained no sensitivity for value of R1 lower than 100Kohm. For value higher then 100 Kohm the sensitivity started increasing.The best sensitivity has been obtained wih R1 = 1Mohm. The sine wave excitation signal is generated by the micro-controller. In [10] it is explained how it can be obtained a sine wave signal by using a High-speed PWM Varying Duty Cycle signal and a low pass filter. Analog waveforms can be generated by averaging the PWM signals over one period using simple low-pass filters.

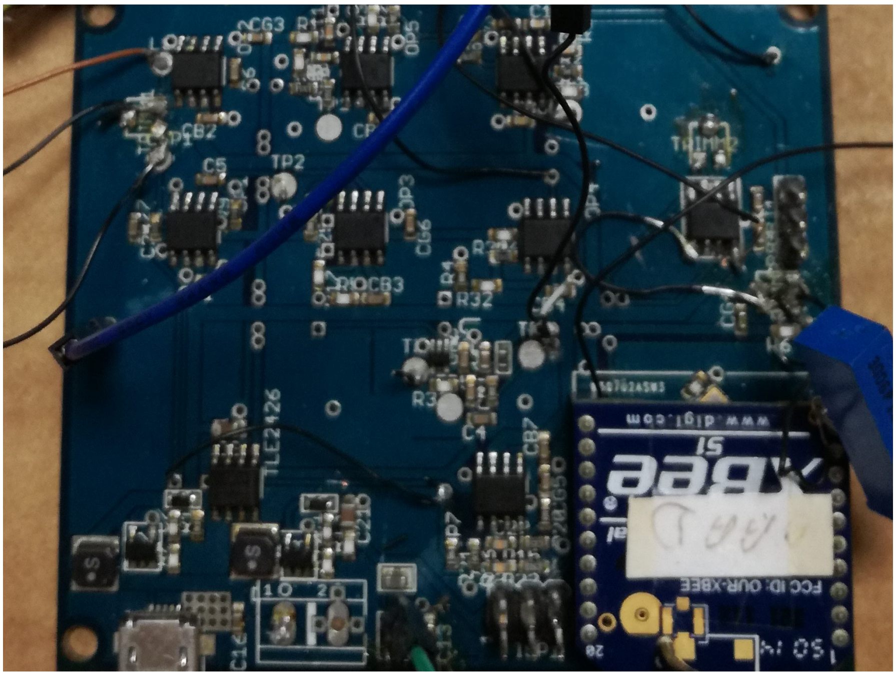

Figure 2.8. Front-end before testing

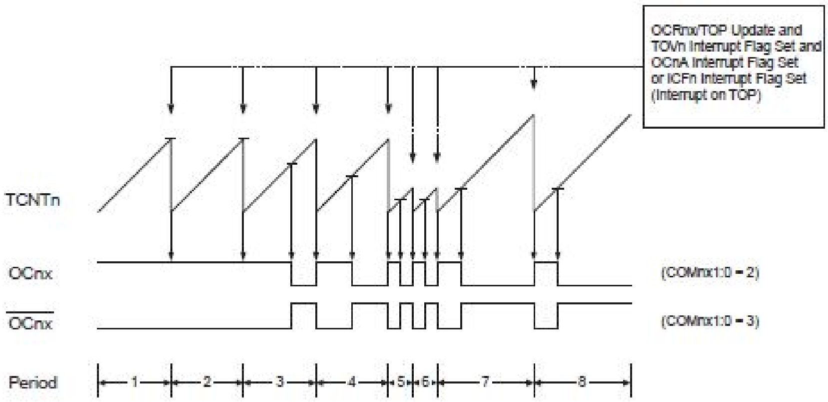

Figure 2.9. Fast PWM Mode, Timing Diagram of ATMEGA328P

2.3.3 Analog waveform generation from PWM signal

In this application note[10], implementation of sine wave generation from high-speed PWM output is explained. If high-speed PWM is used to generate analog signals, the step-size between the analog levels depends on the resolution of the PWM signal. Duty cycle of the PWM signal determines the amplitude of the analog waveform. A duty cycle of 50% gives an analog signal with half the supply voltage, while 75% duty cycle gives an analog signal with 75% supply voltage []. The analog low-pass filter could be a simple passive RC-filter for instance. The filter removes the high PWM base frequency and lets through the analog signal. The filter crossover frequency must be chosen high enough to not alter the analog signal of interest. At the same time, it must be as low as possible to minimize the ripple from the PWM base frequency. The higher the base frequency is, the easier it is to attenuate the base frequency and thereby minimize the signal ripple. The selection of resolution versus base frequency is thus an application dependent trade-off[10].

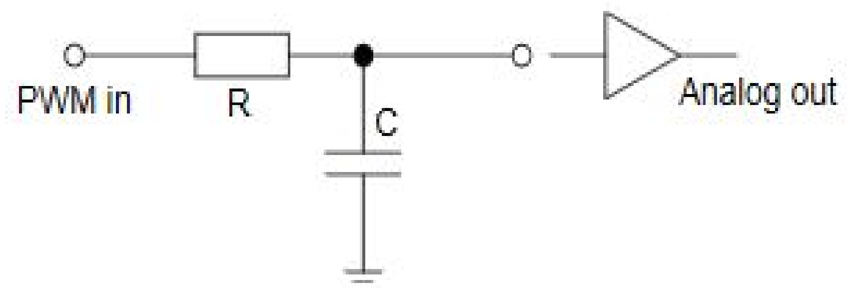

Figure 2.10. Analog waveform generation from PWM signal

2.3.4 Timer configuration for PWM generation

In this experiment, two timers (Timer0 and Timer1) have been used to generate the modulated PWM. the flowchart [10] to generate the modulated PWM signal is shown on figure 2.11

When the Timer0 value reaches the OCR0A value, the Timer Overflow interrupt service routine (ISR) is executed. The call to the ISR happen at the same frequency. When the ISR is executed, the value from sine table stored in the flash memory is loaded to OCR1A register in order to modulate the pwm.In this experiment the ucontroller have been programmed in order to obtain a sine wave signal with a frequency equal to 10 Khz and an amplitude Vp=1.5v. An ah doc low pass filter was already present on the pcb board.

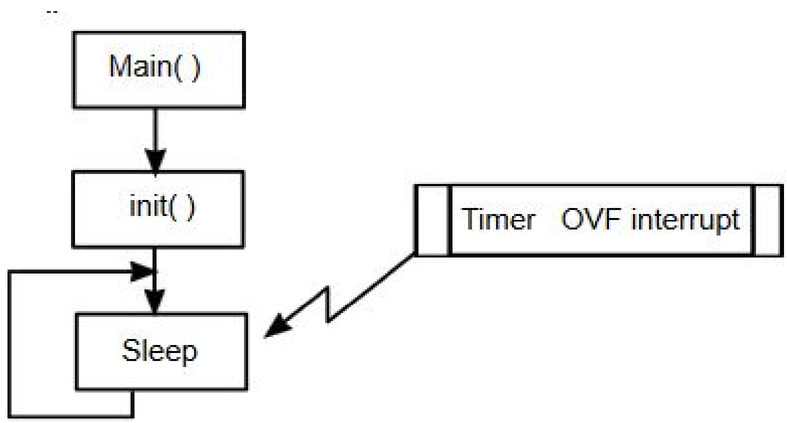

Figure 2.11. Flowchart Analog waveform generation from PWM signal

2.3.5 Testing result

The figure below highlight the output of the different stage of the conditioning circuitry. The gain of the instrumentation amplifier have been set to G=5.35 with Rg=23K.

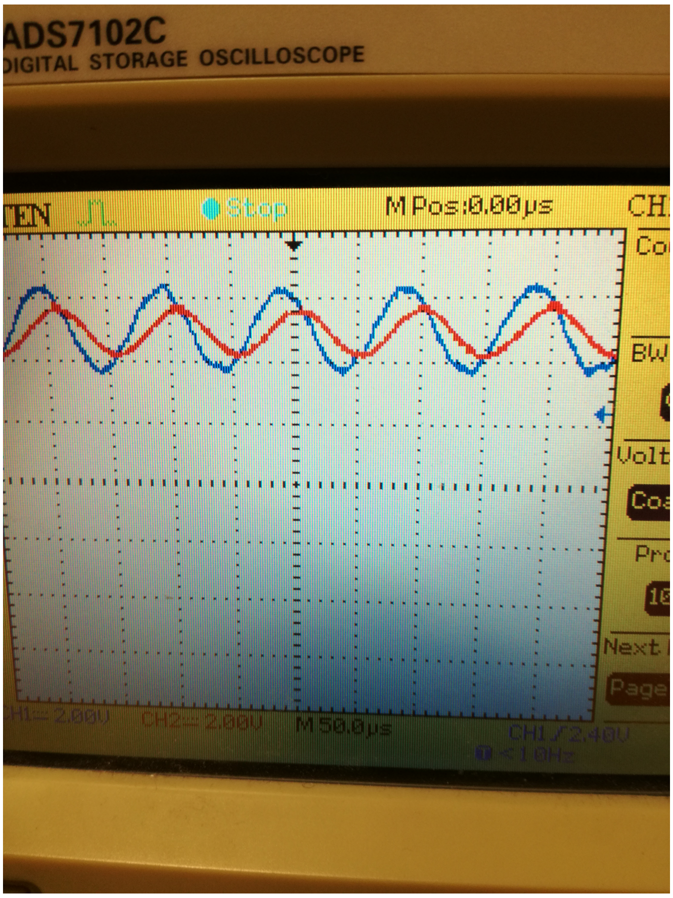

Figure 2.12. Voltage follower output

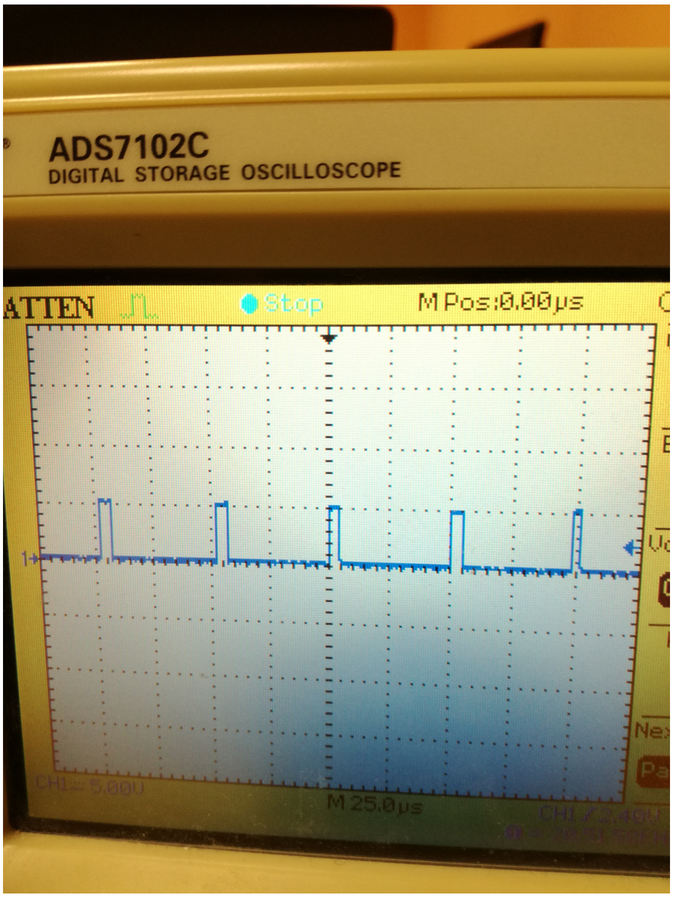

Figure 2.13. Exor output

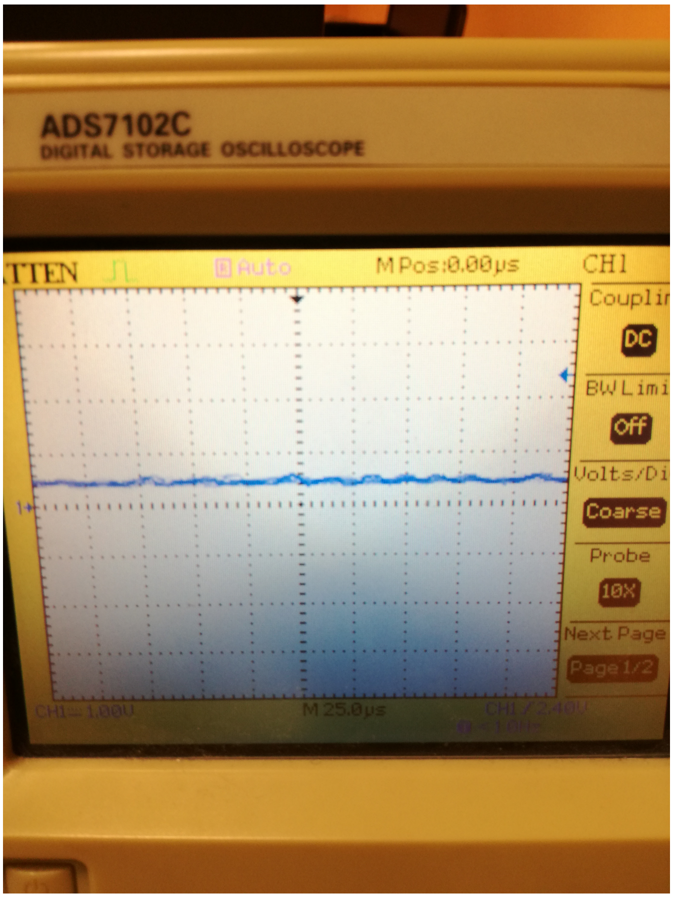

Figure 2.14. Filtered signal from exor output

I observed a misbehavior of the circuit. It wasn't work as expected. The output of the exor was getting lower as we approached the sensor. So I turned back on the the schematic to check the connection after the exor gate. We found out that the reference of the second order filter at the output of the Exor gate was not the good none. It should have been refered to ground. Due to a design error it has been connected to the VIRTUAL GROUND during the design phase 2.15. Also i found others faulty nets connection on the schematics. This lead to a misbehavior of the output signal.

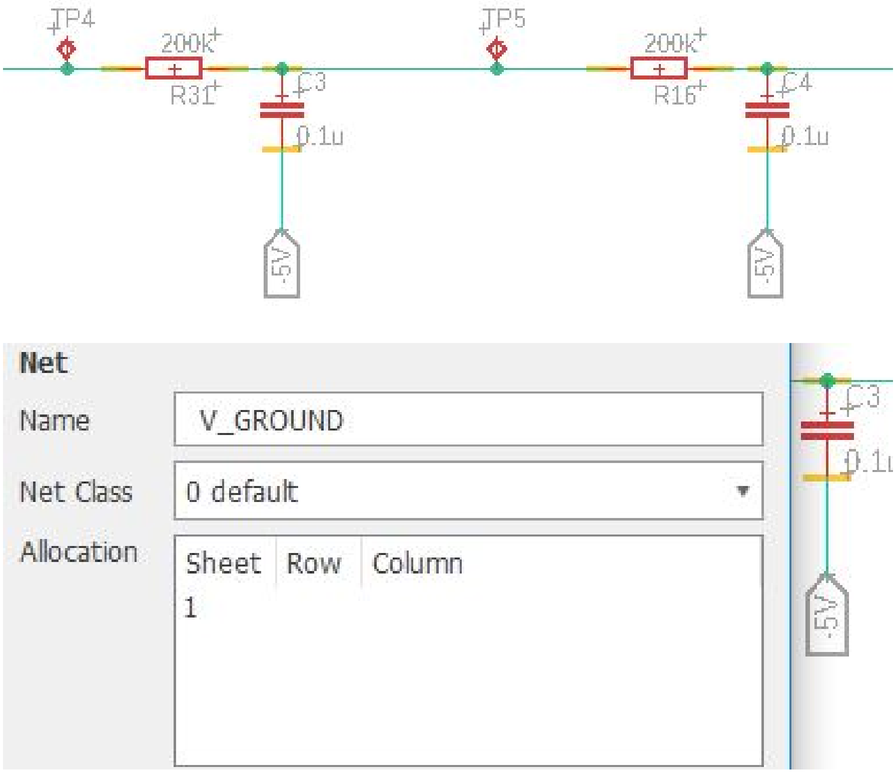

Figure 2.15. Error on the second order filter at the exor output

Due to time constraint i carried on the experiment with the new behavior of the sensor. Hence the output of the sensor decreasing while approaching the sensor.

As we can see in the 2.13 the output signal from the xor gate was noisy. as consequence the instrumentation amplifier output signal has been affected by noise as shown on picture 2.16

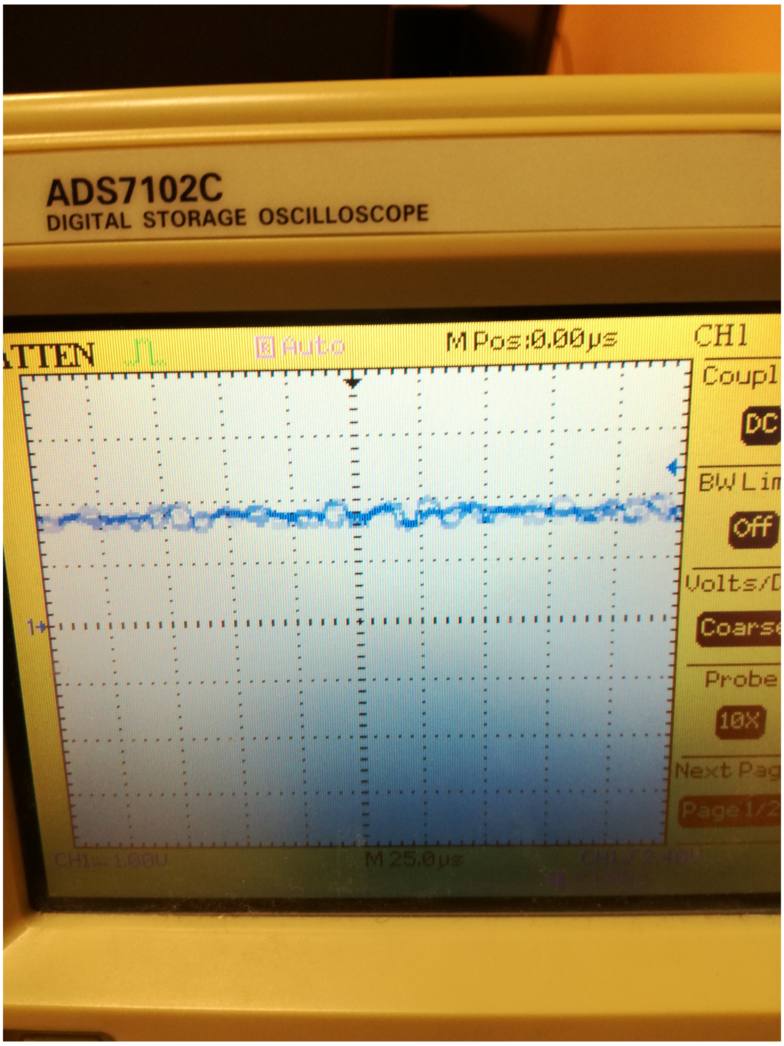

Figure 2.16. Instrumentation amplifier output

So an extra RC low pass filter ($R = 1M\Omega; C = 12$) with a cut-off frequency 13.36Hz have been added at the instrumentation amplifier output in order to eliminate most of the noise. The output after adding the new filter $Vout$ is shown on the 2.17.

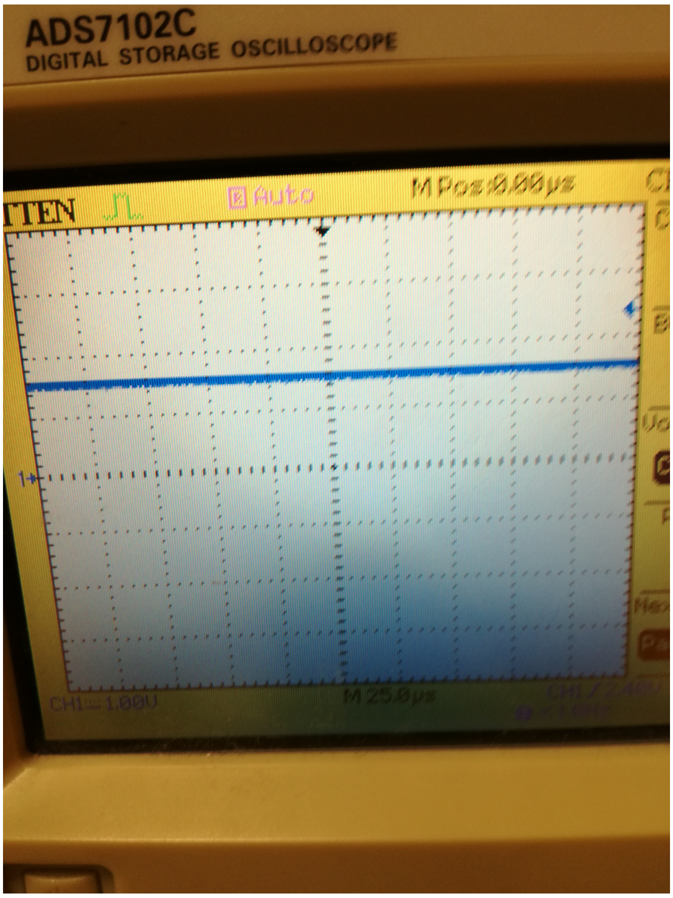

Figure 2.17. Filtered instrumentation amplifier output

The final front end is shown here 2.18.

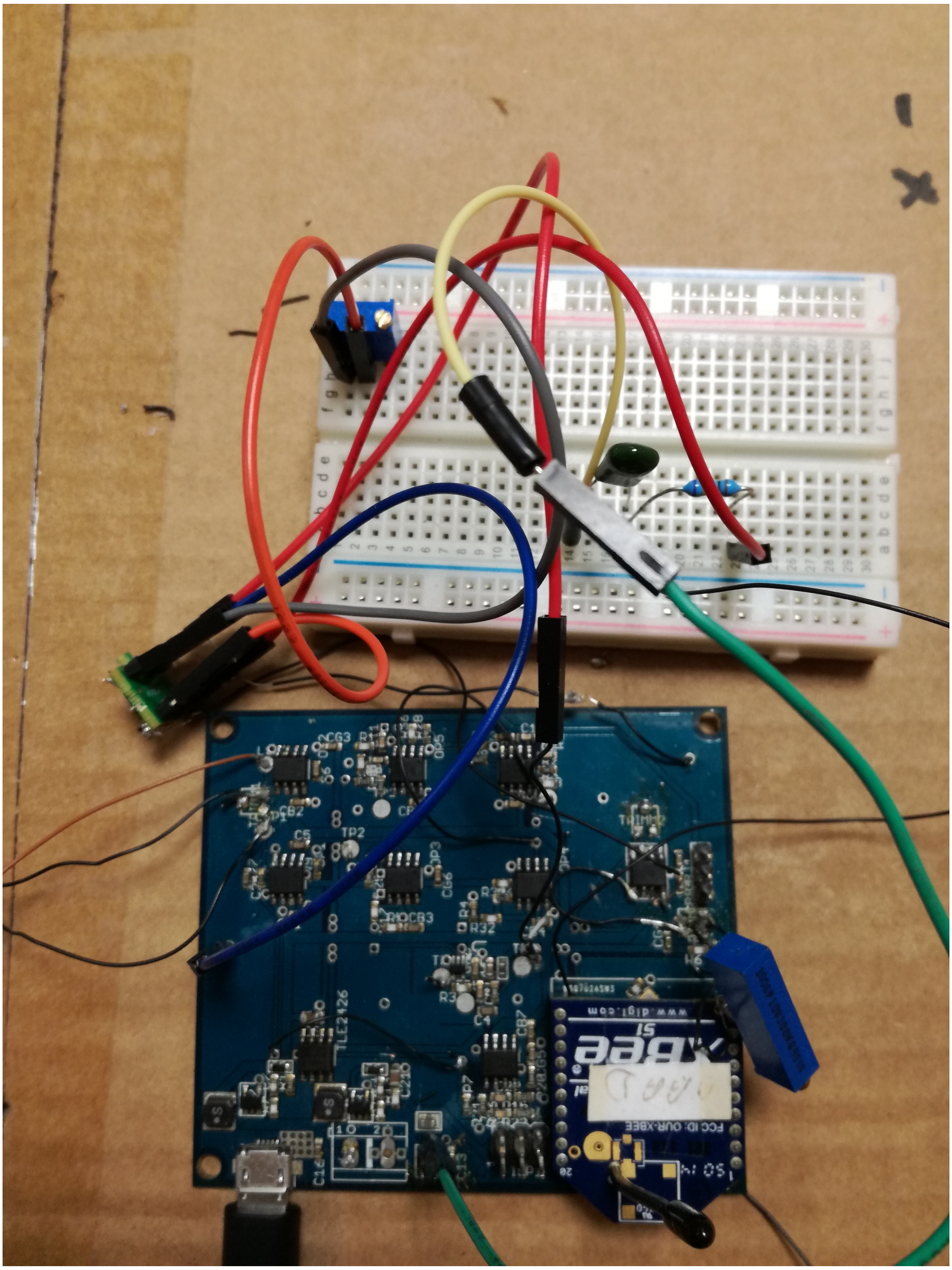

Figure 2.18. Front-end with filter

2.4 Data acquisition and results analysis

2.5 Data acquisition

The $Vout$ signal is then sampled by the atmega328p microcontroler . From the data-sheet [6] of the micro-controller, we note that the ADC clock frequency should be between 50KHz and 200KHz to get a 10 bits resolution on the conversion result which is the maximum resolution. When the ADC clock is 200kHz, the sampling frequency is al almost 15kSPS since one convertion requires approximatively 13 ADC clock cycles. This c onfines the upper frequency in the sampled signal to apporoximatively 7.5kHz. According to the datasheet, the ADC clock can be driven on frequencies up to 1Mhz, though this will lower the ENOB [6]. Since your clock frequency is 16 MHz in these experiment i set the prescaler of the adc at 16. Hence the ADC clock frequency was 1Mhz. The ADC has been set to operate as single conversion mode. At each reset the first sample result is discarded since it take 25 ADC clock cycle As specify in the datasheet 2.19.

Condition	Sample & Hold (Cycles from Start of Conversion)	Conversion Time (Cycles)
First conversion	13.5	25
Normal conversions, single ended	1.5	13
Auto Triggered conversions	2	13.5

Figure 2.19. Atmega328 ADC convertion time

Since the maximum resolution of the micro-controller is 10 bit, in order to enhance the resolution, i used oversampling technique as describe in [11].
The Nyquist theorem states that a signal must be sampled at least twice a fast as the bandwidth of the signal to accurately reconstruct the waveform; otherwise, we will observe aliasing phenomena. The minimum required sampling frequency, in accordance to the Nyquist Theorem, is the Nyquist Frequency $fnyquist$ [11]. The condition to have oversamplig is :

$$fnyquist > 2 \times \text{fsignal}$$

From [11] for each additional Bit of resolution n, the signal must be oversampled four times. hence, the oversampling frequency is :

$$foversampling = 4^n \text{fnyquist}$$

Criteria for noise, when using the decimation (over sampling) technique:

- The signal-component of interest should not vary significantly during a conversion.
- There should be some noise present in the signal.
- The amplitude of the noise should be at least 1 LSB.

Normally there will be some noise present during a conversion. The noise can be thermal noise, noise from the CPU core, switching of I/O-ports, variations in the power supply and others. This noise will in most cases be enough to make this method work [11].
The table below show the convertion time using using decimation tecnique with 1Mhz ADC clock frequency.

Bit	n	Samples	Time
11	1	4	52us
12	2	16	208us
13	3	64	832us
14	4	256	3.328ms
15	5	1024	13.312ms
16	6	4096	53.248ms

Figure 2.20. Convertion time using Over sampling tecnique

For this experiment i exploited oversampling technique to obtain a 16 bit resolution. The sampled data is then sent to the data processing unit using the built-in UART peripheral present on the atmega328p and the xbee radio.

2.5.1 Results

The results obtained are shown below.

i was able to detect the presence of a person at a distance up to 150cm. The measuring time has been reduce to 53.248ms. This measuring time is much lower than the one of the previous sensor node used in the project that was 1s.

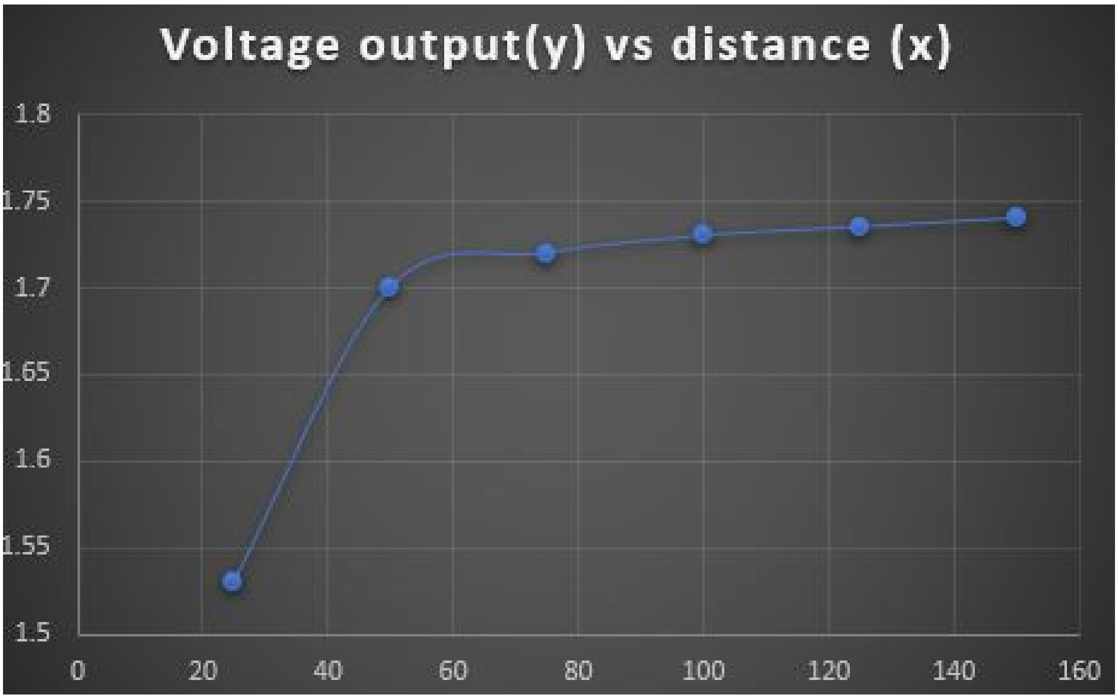

Figure 2.21. Voltage output VS distance

2.6 Conclusion and future work

Although many error have been found on the schematic, I tried to solve most of them by removing and soldering new component on the pcb board. Since i used many wires on the pcb, this technique may cause some problems. Wire may act like antenna and they may affect the results.

To obtain better results, the errors found on the schematic should be corrected to manufacture a new pcb to test.

Appendix A

C Code for PWM generation, ADC and data transmission

```c

/*
 * GccApplication4.c
 *
 *
 *
 */



#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;
  uint32_t total_input_periods;
} payload;

uint8_t *TX_payload = &payload;
uint32_t total_clocks_during_measurement = 0;
uint32_t total_input_periods = 0;
//float input_signal_frequency = 0.0;




const int sinewave_length=20;
volatile uint8_t end_isr  ;
volatile uint16_t sample;
volatile uint16_t sample;
 char from_slave;
const uint8_t sinewave_data[] PROGMEM =
    {20,23,26,29,32,34,36,38,39,40,40,40,39,38,36,34,32,29,26,23,20,17,14,11,8,6,4

uint32_t total;
//uint8_t LowBit;
//uint8_t HighBit;
//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};


ISR(TIMER0_COMPA_vect) {
  if (sample >= sinewave_length) {
    sample = -1;
  }
  else {
    OCR1A = pgm_read_byte(&sinewave_data[sample]);
  }
++sample;
//end_isr=1;
}



```

```

void setPrescaler(uint8_t prescaler)
{
  uint8_t mask = 0XF8;
  ADCSRA &=mask;
  ADCSRA|=prescaler;

}
void adc_init()
{
  setPrescaler(4);
  ADMUX |= (1<<REFS0);
  ADMUX &= ~(1<<REFS1);
  ADCSRA |= (1<<ADEN);                    //Power up the ADC
}


void adc_start()
{
  ADCSRA |= (1<<ADSC);    // star convertion
}


int main(void)
{
  DDRB |= (1 << DDB1)|(1 << DDB2);
  // PB1 and PB2 is now an output

  ICR1 = 40;
  // set TOP to 16bit

  OCR1A = pgm_read_byte(&sinewave_data[0]);;
  // set PWM for 25% duty cycle @ 16bit

  TCCR1A |= (1 << COM1A1)|(1 << COM1B1);
  // set none-inverting mode

  TCCR1A |= (1 << WGM11);
  TCCR1B |= (1 << WGM12)|(1 << WGM13);
  // set Fast PWM mode using ICR1 as TOP

  TCCR1B |= (1 << CS10);
  // START the timer with no prescaler

  //setting TIMER0 for updating the values for PWM period

  // Set up Timer 0 to send a sample every interrupt.
  cli();    // disable interrupts
  // Set CTC mode (Section 15.9.2 Clear Timer on Compare Match)
```

```c
  // WGM = 0b0100, TOP = OCR1A, Update OCR1A Immediate (Table 15-4)
  // Have to set OCR1A *after*, otherwise it gets reset to 0!
  TCCR0A = 0;
  TCCR0B = 0;

  TCCR0B |= (1 << WGM02);
  TCCR0A |= ((1 << WGM01) | (1 << WGM00));

  // No prescaler, CS = 0b001 (Table 15-5)
  TCCR0B = (TCCR0B & ~(_BV(CS02) | _BV(CS01))) | _BV(CS00);
  // Set the compare register (OCR1A).
  // OCR1A is a 16-bit register, so we have to do this with
  // interrupts disabled to be safe.
  OCR0A = 10;//F_CPU / SAMPLE_RATE;    // 16e6 / 8000 = 2000
  // Enable interrupt when TCNT1 == OCR1A (p.136)
  TIMSK0 |= _BV(OCIE0A);

  sample = 0;
  sei();  // enable interrupts
  adc_init();
  XbeeUSART_init();
  adc_start(); //start the first conversion and ignore it
  while(ADCSRA & (1<<ADSC));
  while (1)
  {
    //_delay_ms(10);
    //sum all measurement
    total = 0;
    for(int i=0;i<4096;i++)
    {

      adc_start();
      while(ADCSRA & (1<<ADSC));
      total+=ADC;
    }

    payload.total_clocks_during_measurement = total;//total_clocks_during_measurement;
    payload.total_input_periods = 1234;//total_input_periods; THIS IS DUMMY

    XBee_TX_Request(SERVER_ADDRESS, TX_payload, sizeof(struct payload_s));

    // we have a working Fast PWM
  }
}
```

Appendix B

Matlab code for data reception

```
%%%% 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;

file_num =5;
minutes_of_run = 15;
samples_per_rot = 50;
no_of_steps = 5;

turn = 1; %the current

%[y,Fs] = audioread('gong.wav');            %soundplay
%serialPort = '/dev/ttyACM0';              % define COM port #
%serialPort = '/dev/cu.usbserial-A8003LKe';     % define COM port #
serialPort = 'COM7';
s = serial(serialPort);
set(s,'BaudRate',9600);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
j=int32(0);
time= zeros(j,1);

%Request Packet

beacon_addr = uint8(hex2dec('5'));
type_of_pkt = uint8(hex2dec('47'));
code_of_data_a = uint8(hex2dec('11'));%little endian
code_of_data_b = uint8(hex2dec('00'));%little endian
reserved_1 = uint8(hex2dec('04'));
reserved_2a = uint8(hex2dec('00'));
reserved_2b = uint8(hex2dec('00'));
```

```
data_to_read = uint8(hex2dec('16'));
reserved_3a = uint8(hex2dec('00'));
reserved_3b = uint8(hex2dec('00'));
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

no_of_samples_per_loc = 12 ;
location_change  = 1 ;
location_change_limit = no_of_samples_per_loc ;


A_RX_packet_count = zeros(5000,1);


A_total_clocks_during_measurement = zeros(500000,1);


A_total_input_periods = zeros(5000,1);

time_A = zeros(5000,1);


freq_A = zeros(5000,1);


% x_axis = zeros(no_of_samples_per_loc*8*3+no_of_samples_per_loc,1);
% y_axis = zeros(no_of_samples_per_loc*8*3+no_of_samples_per_loc,1);

fopen(s);
%pause(5);
start_time = clock;
flushinput(s);
flushoutput(s);

data_index = 1 ; %array index to store US/cap sensor values
i = 1;
TX_pkt_count = 1;
senT_counter = 1;
RX_pkt_counter = 0;


pause(1);
beep;

end_at = 60*minutes_of_run;

t0 = clock;

while(1)

```

```
    while etime(clock, t0) < end_at

      if (s.BytesAvailable > 16)
          [address, total_clocks_during_measurement,
    total_input_periods] = rec_API_2b(s);

              if (total_clocks_during_measurement > 0)
              A_total_clocks_during_measurement(i) = (double(
    total_clocks_during_measurement)/64.0)*(5.0/65536.0);
              A_total_input_periods = total_input_periods;
              disp(A_total_clocks_during_measurement(i));
%               disp(A_total_input_periods)
              i=i+1;
              end
          end
      end
  end

fclose(instrfind);
```

Bibliography

[1] F. Zafari, A. Gkelias, and K. Leung, "A survey of indoor localization systems and technologies. arxiv 2017," *arXiv preprint arXiv:1709.01015*.

[2] T. Kivimäki, T. Vuorela, P. Peltola, and J. Vanhala, "A review on device-free passive indoor positioning methods," *International Journal of Smart Home*, vol. 8, no. 1, pp. 71–94, 2014.

[3] 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*, pp. 3293–3315, ACM, 2017.

[4] R. Wimmer, M. Kranz, S. Boring, and A. Schmidt, "A capacitive sensing toolkit for pervasive activity detection and recognition," in *Fifth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom'07)*, pp. 171–180, IEEE, 2007.

[5] J. Iqbal, M. T. Lazarescu, O. B. 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)*, pp. 189–194, IEEE, 2017.

[6] A. Coorporation, "Atmel atmega328p datasheet," *Accessed: Apr*, vol. 26, p. 2018, 2011.

[7] X. P. D. Sheet, "nd¡ www. sparkfun. com/datasheets/wireless/zigbee," *XBee-Datasheet. pdf*.

[8] "Analog device - ad8429 datasheet." Available at https://www.analog.com/media/en/technical-documentation/data-sheets/AD8429.pdf.

[9] "Analog device - ad623 datasheet." Available at https://www.analog.com/media/en/technical-documentation/data-sheets/ad623.pdf.

[10] H.-s. P. P. Clock, "Avr131: Using the avrs high-speed pwm,"

[11] A. Enhancing, "resolution by oversampling," *Application Note AVR121, Atmel, URL: http://www. atmel. com/Images/doc8003. pdf,[Zugriff am 28.10. 2013]*.