RMI Summer Project Report – 2018

A Simple ultrasound GPS system for Indoor Mobile Robot System

-Deepak S.V., 1st Year EEE, NIT-Trichy, India

Preface

I once again thank all the RMI Seniors for have given me the opportunity to take part in the RMI Summer Project division & giving me necessary support in doing a project which I was curious about for the past 2 years. In the span of 1 and half months of doing my project, I faced many challenges & overcoming each of them was a great experience.

In my project I was supposed to make a device to achieve indoor localisation just by using ultrasonic sensors. In this report I would like to briefly recall the abstract of my project basic & advanced tasks first and then produce a timeline of how I proceeded through my project and completed it.

Project Intro

Basic Task:

In the basic task I intend to make a simple ultrasound GPS system by getting distances of target from ultrasound satellites and calculating its position, by intersection of louses (circles) of target w.r.t. the satellites.

Advanced Task 1:

This task involves application of Kalman filter in code to minimize errors caused in ultrasound sensors to get a more precise location.

Advanced Task 2:

This task involves making a bot that traverses paths using the continuous coordinates received from the ultrasonic sensors.

Timeline

Due to shortage of time in creating the report & to give more clarity on how I overcame each problem, I decided to make my Project report in the form of a weekly timeline. If feel it would better for me to discuss about some tough challenges I faced in the SAC rather than mentioning them in detail in this report.

May 3rd Week:

I reached my home by the end of the 2nd week of May & I started my project on the 3rd week of May.

First of all a main part of my project was to calibrate time between 2 Arduino boards (1 connected to ultrasonic transmitters (or) Satellites & 1 connected to the ultrasonic receivers in the bot), for the sake of measuring the time gap between the transmission & reception of the ultrasonic pulse.

I initially planned to used IR transmitters & TSOP receivers to calibrate the time between the 2 Arduinos. I took this decision as IR transmitters at 38KHz & TSOP reception will have a good signal strength & I could also use more than 1 TSOP receivers if I wished to have more than 1 receiver robots in the tracking area & calibrate time in more than 1 Arduinos w.r.t. the transmitter Arduino.

I then started developing codes to calibrate time between 2 Arduinos & then check how accurately it has been calibrated using a wired connection setup between the 2 Arduinos. I developed some 3 to 4 algorithms with multiple approaches (I can elaborate those algorithms used in SAC if someone is interested) & the best algorithm gave me a time calibration accuracy of (+/-) 30us . But that was not satisfactory enough as sound would travel about 1cm in 30us which will affect the accuracy of my localisation.

Conclusion of the week: No matter how much ever I break my head, IR transmitter & receivers can never be used to calibrate time between 2 Arduinos in a scale close to +/- 1us.

May 4th Week:

Then after some researching on the web I thought of trying 434MHz radio frequency transmitter & receiver chips to calibrate time between the Arduino boards. And similar to IR transmitters , rf transmitters also had the flexibility to have more than 1 receivers for the same transmitter.

Then I developed a “new complex algorithm” with reference to an algorithm used for IR transmitter in the previous week. This algorithm has some amount of noise during

calibration, but in most of the time it gave a very good accuracy to a level of +/- 1us, which was my target.

So I happily moved to my next task of measuring distance between an ultrasonic transmitter and receiver after calibration of time between the Arduinos. And just 1 day later I completed that and 4 hours later I implemented the algorithm to find distance between 2 ultrasonic transmitters and 1 receiver at the same time with an frequency of **50 distance values per second.** Then I posted its working video in my Mentors group.

Result of the week: Time calibration between Arduinos with rf waves successful & measured 25 sets of distance values between 2 ultrasonic transmitter & 1 receiver per second.

June 1st Week:

The goal of this week was to fix a localisation area & position the ultrasonic transmitters above them at a position where the pulses of both transmitters will cover the entire localisation area.

After some planning & considering my room size I decided to fix a target localisation area of **120cm x 120cm .** Through certain experiments I initially found that the ultrasonic transmitters were capable of spreading its pulse beam by 30 degrees & it could be captured by an ultrasonic receiver which is 2m far from it.

By using the above logic I started using a 3D plotting software in my PC to decide the height & distance of placement from origin of tracking area of the ultrasonic transmitters. Here I created a 3D cone with divergent angle of 30 degrees, placed them at some distance from origin and at some heights above the xy plane so that, the intersection of these cones with the xy planes gives ellipses & the intersection area of the ellipses formed by both the transmitters contains the whole of the target localisation area within it.

Result of the week: Fixed the positions of the transmitters above the ground such that they can beam their pulses over the target localisation area with satisfactory beam strength.

June 2nd Week:

After setting the target localisation area, I realised that the previous RF time calibration algorithm developed on May 4th week, had some noises & demerits like “time calibration can be done only once in 500ms (within which 25 distance values from sensors need to be read)” . This caused errors due to the fact that the 16MHz Quartz crystal in

Arduino has +/- 0.5% inaccuracy which gets added up over each cycle of Timer 1 & caused more deviation in the distance measured.

So my full focus of this week was to develop a new algorithm that was capable of calibrating time between the Arduino boards much more frequently & with less noise & must not be affected much by the +/- 0.5% inaccuracy of the Quartz oscillator.

Result of the week: After spending a full week, a far superior timer calibration algorithm which calibrated time for every cycle of Timer 1 at prescaler of 8 was developed & I started researching about the working of Kalman Filter.

June 3rd Week:

My father sent me for a 1 week internship to a German MNC called **"Auma Drives"** in Bengaluru who basically manufacture electronic valves which can be controlled from a central control room. It was a great experience interacting with the RND department over there…

June 4th Week:

I then tried to measure the distance, but the ultrasonic sensors still gave me a deviation of +/- 5cm in the values received in the localisation area the ultrasonic transmitter & receiver cannot be always facing them along a line always. So I decided to implement Kalman Filter to these measured distances first & then proceed with the localization.

So after Spending 3 days of research on the working of Kalman Filter I was flawlessly able to implement it in my code & it gave amazing results.

Result of the week: Kalman filter was successfully implemented in my code & it **improved the accuracy of the sensors from +/- 5cm to +/- 3mm** . It was a like a turning point to my project.

July 1st Week:

This was the most critical week for my project. Barely slept for 3 hours a day to complete my project.

First I developed & made the intersection of circles equation & outputted (x,y) coordinates of the localizing area via UART to the Serial Monitor.

Then I learnt a software called **"Processing"** & coded in it & I was able to display the (x,y) coordinate as a point on the graph. **(With this my basic & Advanced Task 1 Got Over together).**

By the 2nd half of July 1st week the accuracy of distance achieved till Advanced Task 1 was sufficient enough to proceed with Advanced Task 2.

Then I started to build a differential drive robot with 2 ultrasonic receivers on the same bot & redesigned the circuit and code of the receiver bot. I placed the ultrasonic receivers at the front & back of the bot to calculate the position & direction at which the bot is facing.

After this I was left up with 1 day within which I made the bot to trace hardcoded path & draw any given pattern (consisting of straight lines) on the floor, **thereby completing my Advanced Task 2 & 100% of my Project`s abstract on the last day!**

Result of the week: Successfully completed of Basic Task, Advanced Task 1 & Advanced Task 2 of my Project & came back to the campus on 8/7/18.