

Small, Lightweight Speed and Distance Sensor for Skiers/Snowboarders

Air Force Research Laboratory



Design Team 5

Kunal Verma - Team Manager
Justin Erskine - Web Master
Tim Ross - Documentation Coordinator
Temika Cage - Presentation Coordinator
Ben Guild - Laboratory Coordinator

Dr. Shanker Balasubramaniam - Faculty Advisor
Dr. Stephen W. Schneider - Sponsor Contact

Executive Summary

Performance analysis aids/devices are an essential component of training methodologies in modern athletics. In recent years, professional grade equipment is becoming more readily available to amateur and recreational athletes. In snow sports like skiing, this top down migration of technologies for training has been slow to develop. Now is an excellent time to leap into the market for portable devices targeted at ski enthusiasts. For skiers, the primary information necessary to compare the performance of individuals is speed; a device that would allow users to measure their speed, in order to record progress, would be a welcome entry to this consumer sector.

This document presents our design for an inexpensive speed and distance sensor that can be used by snow sports enthusiasts of all skill levels. By using low cost materials, maintaining minimal power consumption of on-board electrical devices and harnessing the benefits of high frequency Doppler radar our device has been determined to be a useful and robust product for all skiers and snowboarders. The project is sponsored by the Air Force Research Laboratory and is the third iteration of a speed/distance recorded for skiers.

Acknowledgements

Design Team 5 would like to acknowledge the generous contributions of our sponsor, the Air Force Research Laboratory, as well as our sponsor contact Dr. Schneider for allowing us to work on this project. The Team would also like to thank Dr. Balasubramaniam for helping us throughout the semester with our project, we are grateful for his generosity, guidance, his moral support and fair critique of our project throughout its development. We would also like to give recognition to three ECE staff members: Gregg Mulder, Brian Wright, and Roxanne Peacock. Without their help we would never have been able to finish our project. Lastly we would like to thank the ECE 480 professors Dr. Goodman and Dr. Shanblatt, as well as all of the presenters for the course this semester that provided us with incredible advice for both our project and our professional careers as engineers. Thank you all!

Table of Contents

Section 1 - Introduction and Background	4
Section 2 - Design Solution	6
Section 3 - Design Process	12
Section 4 - Performance Data	17
Section 5 - Summary	18
Appendix A - Team Members Roles and Contributions	21
Appendix B - Literature and References	25
Appendix C - Extras	26

Section 1: Introduction and Background

Introduction

For this project our goal was to create a personal electronic device capable of measuring speed and distance, designed specifically for athletes and enthusiasts in winter sports such as skiing and snowboarding. The device can also be used by rollerbladers, speed skaters, skateboarders and other sports in which velocity is relatively stable over time. We were given several design criteria at the beginning of the design process; the most important of these are the ability to record maximum speed, total distance travelled and average speed. These metrics should allow athletes to gauge their performance in order to help them in their training. Speed is a particularly important metric for skiers and sportsman; in freestyle competitions, achieving the optimal speed for lift off is necessary to safely hit a jump. For downhill and slalom competition the fastest competitor wins, so knowing one's speed is extremely vital to success. Ergonomics, ease of use, robustness and power performance are also aspects of the design we were asked to consider during the development of our project.

Speed Detection Methods

Previous ECE design teams have tackled this project in the past, with varying degrees of success. The first attempt was during the spring of 2008. The design team created a device that implemented GPS as its basis of measurement, and paired it with a Bluetooth device. The biggest problem encountered was the large error when turns were involved in the route. Other problems with the device included lack of waterproofing and a bulky size. The second attempt, during autumn of 2009, used a system which relied on GPS and an Inertial Navigation System. Inertial Navigation System (INS) uses multi-axis accelerometers and gyroscopes to measure the speed. It is good for short term measurements, but accuracy decreases over time. The combined system used GPS as a long-term reliable source of input, and INS as a local short term error checking. This combined system offered accurate results. Unfortunately, this design attempt failed due to the complexity and time limitations.

Our primary objective in this project was to use a different means of speed measurement, namely Doppler sensing, in order to provide a simple and accurate measurement for speed while also

simplifying the overall design hierarchy. All parts used in our product were chosen for ease of acquisition and low cost, as opposed to previous designs that required expensive GPS hardware and software packages.

Wireless Doppler detection is our means of speed measurement, with integration of an accelerometer to detect movement of the user. Doppler detection will provide a far more reliable source of velocity measurement than a GPS position tracking/velocity derivation method. Doppler frequency detection allows us to calculate velocity using a completely linear model, with one degree of variation that can be calibrated without worry of stability issues in velocity measurement, an issue often present in INS/accelerometer systems.

There are two primary methods of radar detection: continuous wave radar and pulse radar; pulsed radar uses pulse broadcast and detection; using this method is not as useful for velocity detection and harder to process due to inherently hampered frequency detection. Continuous wave radar allows us to process the Doppler signal using simple analog circuitry. The only downside is continuous power consumption, but due to our proposed battery capacity we did not expect this detection method to pose to great of an issue.

Current Products

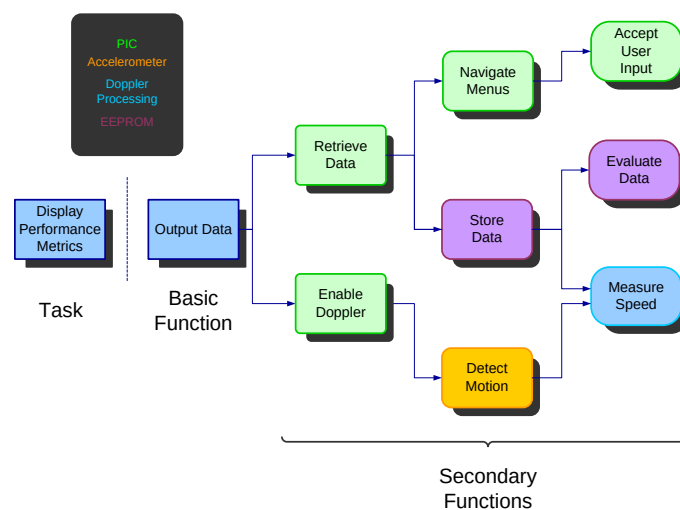
Our device competes directly with an existing device, the Silva Tech Ski Speedometer Watch. The retail device comes in two components, a Doppler Module and a watch receiver for viewing instantaneous speed and run data. Looking at reviews of the product on Amazon, the device either doesn't work or is far too complicated for use by the average consumer. The cost is also slightly prohibitive, although Amazon now sells the seemingly defective product for \$49.90; the MSRP is listed as \$199.99. Our goal with this project is to provide a cheap alternative to the Silva Tech device, simple to put together at home for a DIY project, and easy to use.

Section 2: Design Solution

The design criteria provided by our sponsor is presented below:

- **Measurements** (minimum one minute intervals)
 - Average speed
 - Max Speed
 - Distance Travelled
- **Safety**
 - Lightweight (< 2 lbs.)
 - Disabled display during recording
 - Weather resistance
 - Low temperature operation (-10F)
- **Power and Efficiency**
 - Auto-off after 10 minutes of inactivity
 - At least 2 hours of battery life
- **Operation**
 - Data storage requirement for at least 10 minutes of run data
 - Data report on LCD or data export to external device

We determined that we would require 3 basic subsystems for our device: velocity acquisition using Doppler radar, user interface and control via a microprocessor.



Fast Diagram
(Developed to guide design process)

Radar Device

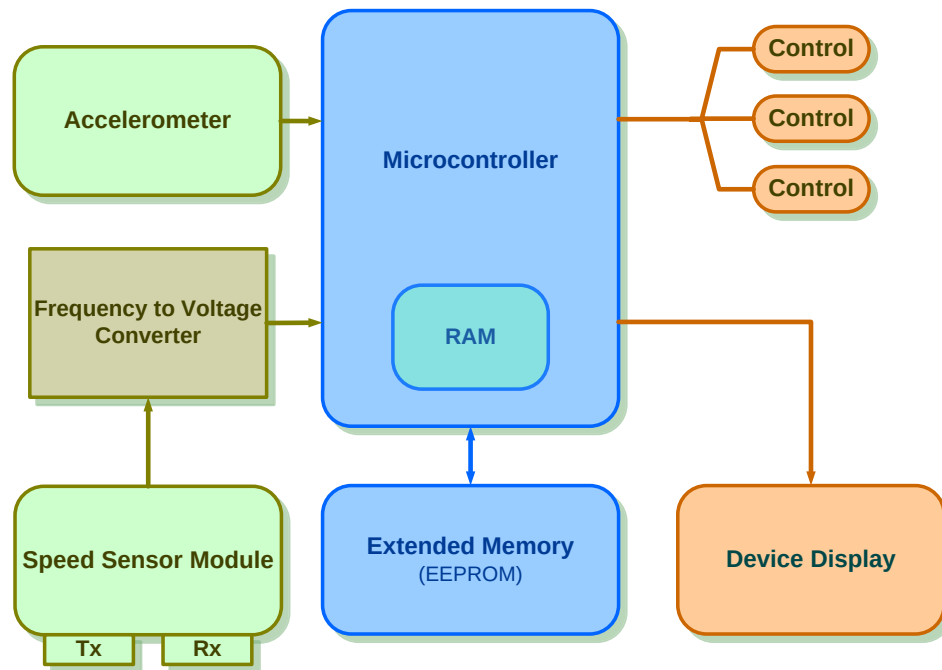
The radar device first transmits an EM wave at a fixed frequency at whatever object we wish to observe, in this case the moving ground below a skier. The wave reflects off the receiving object and returns a wave with a shifted frequency. This frequency deviation allows us to calculate the difference in velocity between the sending object (skier) and the observed object (ground). Because our frame of reference is the skier, we can reduce the complexity of the Doppler derivation, making the sending object stationary and the observed object motive, or vice versa. In either case we can develop any easy method for measuring the magnitude of velocity, or speed. This value is sent to the microcontroller for recording and analysis. We also tested an accelerometer as a backup device but it is unimplemented in the final design.

Control System

The microcontroller will provide the control of the overall device including menu navigation logic, read/write operations to data storage, ADC conversion of the Doppler signal into speed data, motion detection with accelerometer as well as computation of the desired statistics for a given run. It will also provide I/O capabilities for interaction with the user.

User Interface

The user interface consists of an LCD screen that displays the menu options. At the users command, the device will be able to retrieve previously stored data, or begin the tracking process. The user will be able to navigate the options through the use of pushbuttons for easy manipulation. The UI design must be simple and intuitive for use by any kind of skier/snowboarder.



For our final design we were able to accommodate every design criteria, save two. Because of the extra engineering time it would take to create an airtight enclosure along with the increased cost due (purchasing MILSPEC components) we decided to forgo weather resistance and low temperature operation. With proper time and funding we might have been able to include environment protection, however we decided it would be more prudent to finish a product capable of performing our primary functions: speed and distance measurement.

Meeting these objectives required special consideration for design constraints imposed by each criterion. For instance, disabling the display during recording would require motion detection. We accomplished this through the use of logical operations on the data coming to the PIC from the ADC. We set up thresholds for when the user is in motion and the LCD will be disabled during the active run.

From the outset of the project we had to decide whether we wanted to design our own Doppler module or purchase a prefabricated device. We chose the latter; a prefabricated product would allow us to spend more time designing the system rather than focus on engineering a very complicated component. We were not sure if the modules we were researching would provide us with the type of signal we required, so we decided to purchase test units. We had no evidence to see if this could work, one project in a Swedish journal had to drop the velocity measurement

component of their ski analysis system because they could not get the Doppler device to work. After testing we were satisfied that we could use this type of speed detection for our system.

The weight of our product was another matter. While most components have negligible weight the device enclosure, link cable and battery all had to have a combined weight of less than 2lbs. We purchased these specific parts with our weight constraint.

The battery and power efficiency goal was incredibly easy to manage, enough to allow us to consider many different choices in the power design. We needed a Vcc that could provide the 5V required of our Doppler module, LCD screen and microcontroller. Trying to attain a level 5V from a battery is difficult due to a decreasing terminal voltage as the cell discharges. A simple voltage regulator was chosen, with nominal output voltage of 4.98V. We also chose a 9V rechargeable battery with enough capacity to allow our device to last longer than 2 hours on a single charge. We measured supply current throughout the project to ensure that the battery capacity would be able to meet our demands.

Our initial budget for this project was very limited in scope; it did not take into account many additional items that we added later on in the project. Below is a table containing our initial estimation of cost to build our device:

Cost Breakdown	
Item	Cost
Microcontroller	\$4.00
LCD Display	\$20.00
Radar Module	\$15.00
EEPROM	\$5.00
Battery	\$10.00
Project Box	\$10.00
Total Cost	\$64.00

There are a number of risks associated with our system that the Design Team took into consideration. Most of the risks focus on the accuracy of our system and safety for the user, which are the two highest priorities.

In terms of accuracy, we had to make sure the Doppler module was precise in its detection of velocity. Without correct results, our system would be useless. We also needed to factor in that quick turns and twists are used for tricks and to control speed in snowboarding and skiing. If the device is mounted incorrectly on our user and our sampling rate is mistuned, sharp turns could skew our detected velocity, and the device will report incorrect speeds.

The device is intended to be worn on the user, or mounted on skis or a snowboard. In sports such as snowboarding and skiing, participants can fall from large heights or at high speeds. The location and casing of our device should have taken these issues into consideration in order to minimize damage to the user. However, due to design limitations and a larger than desired size for the Doppler module we had to do the best we could with the design of Doppler module housing. In the future a combined design of a proprietary Doppler module in conjunction with slimmer housing could allow us to maintain our goal to keep the user safe.

One of our biggest concerns with the final device was the fact that it probably could not be used reliably in a skiing environment. Again, due to time and resource limitations we were unable to create housing to ensure proper weatherproofing.

Section 3: Design Process

Each subsystem of our proposed device was worked on separately, with certain responsibilities given to each team member. Justin Erskine and Kunal Verma performed signal analysis and integration between the microcontroller and both the accelerometer and Doppler Module. The LCD programming, user interface, data storage and other programming responsibilities were given to Tim Ross, Ben Guild, and Temika Cage. Because our conceptual design had been more or less decided from the beginning of the design process (Doppler, Control, UI) we decided to focus our feasibility matrix more towards component choices in our proposed design. The following considerations came into mind during component research:

- **Component Complexity**
Ease of programming or system integration.
- **Component Cost**
Price in comparison to lowest cost option.
- **Power Consumption**
Subjective power consumption rank, also based off of lower power consumption of component category.
- **Size**
Subjective measure for volume budget in the final design.
- **Temperature Limits**
Top marks for meeting or exceeding -10°F , mid marks for $\leq 0^{\circ}\text{F}$, low marks otherwise.

Specific performance metrics of the radar subsystem included:

- **Detection Distance**
Max detection range using pulses, with good marks around 15ft.
- **Detection Mode**
Continuous wave mode, pulse mode, or both

Our feasibility matrices are shown below:

Feasibility Matrix	Component	Doppler Module			
Factor	Weight	Custom Module	AgilSense HB100	Infineon RON-7701	Sky Microwave 4177
Low Complexity	5	1	4	4	4
Low Cost	2	2	4	2	2
Power Consumption	4	2	4	4	4
Size	4	1	4	4	4
Temperature Limits	3	2	2	2	2
Detection Distance	4	4	1	4	4
Detection Mode	1	4	1	4	4
Total		47	71	82	82

Feasibility Matrix	Component	Microcontroller	
Factor	Weight	PIC18F4520	TI MSP430
Low Complexity	5	4	1
Low Cost	2	4	2
Power Consumption	4	2	2
Size	4	1	1
Temperature Limits	3	1	4
Detection Distance	4	-	-
Detection Mode	1	-	-
Total		43	33

Feasibility Matrix	Component	Display	
Factor	Weight	LCD	OLED
Low Complexity	5	4	2
Low Cost	2	2	2
Power Consumption	4	4	4
Size	4	2	4
Temperature Limits	3	1	4
Detection Distance	4	-	-
Detection Mode	1	-	-
Total		51	58

The results of design analysis determined that our three primary components would be an Infineon RON-7701 or a Sky Microwave 4177 Doppler module, a PIC18F4520 microcontroller, and an OLED display. When we attempted to purchase the two Doppler modules we were unable to find retail sellers or even private merchants on eBay, AliBaba, or other such electronic sites. We were forced to purchase the HB100 Doppler module from AgilSense, which happened to be our testing modules. We had many PIC48F4520 microcontrollers available in the lab for use, so we were comfortable using the microcontroller knowing that so many would be available should we encounter a malfunction. As for the OLED we decided to move to a simpler LCD model once we had decided to forgo our weather/temperature requirements. The complexity of programming the OLED also became a factor in choosing to use the LCD, the OLED required a separate controller just for the display itself which in turn required the use of 8-10 ports on the PIC, space we could not afford to lose.

In order to program the microprocessor to control the device we used Microchip's MPLAB Integrated Development Environment in conjunction with the MPLAB ICD2 Programmer. This system allows the microprocessor to be programmed in C, making the process simpler and allowing for more flexibility in software design.

Doppler Signal Processing

After testing the Doppler Module we found that the output voltage from the intermediate frequency pin was roughly 20mVpp, so our first order of business was to amplify the signal to a level where a comparator could easily detect zero crossings of the sinusoidal signal. We designed a single sided op-amp with a gain of roughly 50 to modify the Doppler signal to a 1Vpp level. The frequency to voltage converter was fed with this signal and we designed the converter to map 0 – 400 Hz to output voltages of 0 – 4 VDC. The ADC on the microcontroller was then fed by our DC signal from the converter and sampled to reproduce an 8bit value, with a comparison point of 4V to increase accuracy range.

External EEPROM

Hardware

For the EEPROM we chose to use a **256K I2C™ CMOS Serial EEPROM**. The reasons for choosing this EEPROM device were the benefits it provided to our design. This EEPROM offered 32,000 8bit-Addresses available for writing. This is more than ample amount of space to store current and previous run data. It also offered Low Power Consumption, which was critical to extending battery life. It is I²C Compatible, which is a simple procedure to implement and comprehend. Lastly, it operates in designated range of temperatures, -40°C to 85°C.

For wiring the EEPROM, we wired pins A0, A1, and A2 to Ground. These pins are selection pins in I²C for the case when we have multiple devices on the bus. Vss is wired to Ground and Vcc is wired to 5 volts, in order to operate the chip at default specifications. Lastly the Clock and Data pins have small pull-up resistors.

For wiring the PIC, we must wire the SCL and SDA pins to their respective EEPROM pins for communication.

Software

Configuration of PIC and EEPROM.

```
OpenI2C( MASTER, SLEW_ON );
```

The EEPROM is defaulted to work as a Slave Device in High Speed Mode due to the 5V difference in Vcc and Vss. Next, the PIC was configured to work as a Master Device in high speed mode.

Sending and Receiving Data

Functions provided by Microchip were used to write and read data over the I²C bus. Many different options were available for writing and reading, but we chose functions which wrote and read only 1 byte at a time.

Results

In the end, we could not get the external EEPROM to work correctly. Due to either a wiring mistake or software mistake, the EEPROM would not communicate correctly with the microcontroller. We abandoned attempts to get the external EEPROM to work in order to focus on developing the main portion of our design.

USART (LCD Interface)

In order to make use of the values from Doppler module, we needed to convert them to digital signals to make our calculations. The ADC converts the Voltage received from the frequency to voltage converter using the formula: $\frac{(Volt)*1024}{5V}$. We store these values into our internal EEPROM every 2 seconds to give us a sample velocity.

LCD Communication

For our device we chose the CrystalFontz CFA632YFBKS LCD display. This display utilizes USART communication, or alternatively SPI, in order to pass information on its single DATA_IN pin. This makes communication with the PIC much simpler than other multi-wire approaches and requires less hardware, wiring, and communication overhead. The display also has configurable, through the use of jumpers and switches, baud rates and built in graphics operations that make the display flexible and capable of all the things that we require of it.

In order to do this, we must configure the microprocessor for USART communication. We also needed to configure the LCD's interface options for communication with an embedded microcontroller's built in USART.

Configuration of USART

```
OpenUSART(USART_TX_INT_OFF & USART_RX_INT_OFF &  
USART_ASYNCH_MODE & USART_EIGHT_BIT & USART_CONT_RX  
& USART_BRGH_LOW, 63);
```

USART_TX_INT_OFF and USART_RX_INT_OFF turn off transmit and receive interrupts respectively. This configuration was chosen because our device only transmits data when available and doesn't receive data from the LCD, eliminating the need for interrupts.

USART_ASYNCH_MODE configures our processor to transmit data on our USART asynchronously as opposed to on a clock signal.

USART_EIGHT_BIT sets our USART to transmit 8 bits of data per transmission. This is in contrast to the 9 bit transmission mode.

USART_CONT_RX allows for continuous receiving by the USART. As the LCD does not transmit, this configuration may not be necessary.

USART_BRGH_LOW configures the USART to use low baud rates in asynchronous transmission mode.

63 is the designated baud rate. As the baud rate is automatically detected in asynchronous mode, this configuration setting is likely not necessary

These settings were chosen because they provide the functionality we need in the USART communication between the PIC and the LCD.

Section 4: Performance Data

Performance testing went as expected. Velocity measurements were perfectly turned on the microcontroller after testing using calibration formulas. Our metric for testing was always max velocity, and in each test run max velocity matched on the device and in real life. We were unable to check in time to see if turning velocity was effected. The following are our run results.

Run 1		
Time Mark	Velocity	
0	5 mph	
30	END	
Results	Average Velocity	Total Time
Expected	5 MPH	30 Secs
Measured	4 MPH	34 Secs

Run 2		
Time Mark	Velocity	
0	10 mph	
30	END	
Results	Average Velocity	Total Time
Expected	10 MPH	30 Secs
Measured	7 MPH	46 Secs

Run 3		
Time Mark	Velocity	
0	15 mph	
45	END	
Results	Average Velocity	Total Time
Expected	15 MPH	45 Secs
Measured	14 MPH	43 Secs

Section 5: Summary

Microcontroller

There are a vast variety of microcontrollers to choose from in the marketplace. The ECE shop has provided all ECE 480 students with a PIC18F4520 and the design team has determined that this microcontroller provides sufficient support for our proposed design. The ease of use for this microcontroller makes it a great candidate for our initial prototyping. Later on in the design process we may decide to upgrade to a better microcontroller for improved data handling and processing.

Display

The LCD display will be the main interface between the user and the device. Information taken during the ski run will be shown on the LCD screen when the user requests it. Most LCD screens are listed in the given price range. They are relatively cheap and will provide an effective display for our initial prototype. Budget permitting, the design team would like to switch to an OLED display to meet the temperature requirements of the device specification.

Radar Module

The motion sensor is the most important part related to the project. All the measurement data will be taken from it and processed to develop the information that will eventually be read by the user. Due to the complexity of building a radar module the design team decided to use an on-the-shelf part. Though there are several products to choose from, we originally decided to use an AgilSense HB100 for testing purposes. In the future we may upgrade to a model that has better signal range.

Persistent Memory

EEPROM will store all the long term data and information for future use. The cost of this item is low for the many different types available. For the initial prototype we will use the internal RAM of the microcontroller to store data and then interface the EEPROM to dump data.

Power Source

Rechargeable batteries are very affordable and reliable. The battery we will use can provide more than enough power to fit the design specifications. The higher cost buy-in of rechargeable batteries will be recouped through the multiple reuses of the rechargeable battery. It is imperative that we develop a power regulator for our first prototype, giving us time for other design work later in the development cycle.

Enclosure

The project box must be very practical since this device is supposed to be mobile. The components of the device need to be contained for the purpose of mobility. For the prototype we will use a pre-manufactured standard case but in the future we plan to create a custom housing for the internal circuitry in order to protect the device from tough weather conditions.

Final Cost Breakdown			
Item	Price	Our Costs	Production Cost
PIC	\$6.32	\$0.00	\$3.95
Battery	\$10.88	\$10.88	\$7.61
Switch	\$1.17	\$1.17	\$1.17
Large Project Box	\$3.69	\$3.69	\$3.69
Small Project Box	\$2.29	\$2.29	\$2.29
Printed Circuit Boards	\$3.98	\$3.98	\$3.98
LCD	\$33.00	\$0.00	\$25.00
Doppler Module	\$15.00	\$15.00	\$15.00
LM324N	\$0.46	\$0.00	\$0.29
LM2907	\$2.02	\$2.02	\$1.46
LM7805	\$0.60	\$0.00	\$0.38
MX045	\$3.35	\$0.00	\$1.91
Capacitors (x5)	\$0.10	\$0.00	\$0.10
Inductors (x2)	\$1.80	\$0.00	\$1.80
Resistors (x14)	\$0.98	\$0.00	\$0.98
Phono Jack Set (x2)	\$0.60	\$0.60	\$0.60
Savings			\$16.03
Total	\$86.24	\$39.63	\$70.21

Schedule

The Gantt chart for our project is attached in Appendix C.

Final Thoughts

During the design process, it was found that the Doppler module was not powerful enough to ping the ground if mounted on the waist or hip of the user. Therefore, the device was divided into two components instead of one.

The project had many successes. The internal EEPROM was programmed and can effectively run and store data. The frequency-to-voltage converter (LM2907) converted the frequency from the Doppler module to voltage. The device is inexpensive with a final cost of \$86.24. On internal power, the device can run 3.5 hours which is much more than the recommended 2 hours. Also, the design was finished on time.

Although the project can be considered a success, it had many failures. The external EEPROM could not be implemented forcing the team to use the EEPROM inside the PIC Microcontroller. The external EEPROM has more memory and was a better choice. The project was also bulky. There was not enough time to design a project box specifically for the project. As a result, standard project boxes had to be purchased from a retailer. Furthermore, the device was not designed to perform in cold and/or wet conditions.

For the future, it is hoped that external EEPROM and a more powerful Doppler module can be applied. With these improvements, the device would be only need one component. It would also hold more data.

Appendix A

Team Member Roles and Contributions



Kunal Verma

Kunal Verma was lead for the analog design portion of the project. He was the main designer behind the Doppler circuit signal processing scheme. When the Doppler module first arrived Kunal, with the help of Justin, performed testing on the device to determine how the signal needed to be modified in order for it to be read by the ADC of the microcontroller. Kunal determined the circuit topology necessary to perform this modification and designed both the signal amplifier and frequency to voltage converter.

During this time Kunal helped the team build the debug breadboard circuit for ease of use and clean layout. He helped in the programming with suggestions, comments, and modification of the code as needed. While the accelerometer was still being considered as a motion detector Kunal programmed the logic code to read from accelerometer using the CCP functions of the microcontroller.

Once all the final parts had been determined Kunal made the final board layout and performed much of the soldering and circuit testing. He also prepared and cut the project boxes, linking cable, and interconnection between components. Kunal was also a primary tester for the final product, including benchmark testing and calibration.



Justin Erskine

Justin's technical contribution to the project, besides in helping with the overall vision of the project, had mostly to do with the analog components of the build. In the beginning it was decided that we would search for a device on the market that has Doppler capabilities while trying to design our own device to do it. To this end Justin experimented with different patch antenna designs for implementation in a custom Doppler circuit. He loaded multiple patch antenna geometries into electromagnetic protocol software, NEC2, and ran multiple simulations to try to find an antenna that would work for our needs. His final task with antenna design was to try to compensate for dielectrics that would be present in the final design when the team found a suitable Doppler module that could be implemented into our design.

After the Doppler module arrived Justin assisted in the initial testing of the device. The signal coming from the module needed to be amplified so that we could easily process its frequency into a steady voltage. A frequency-to-voltage IC was found and Justin used the datasheet to find a correct method for implementing the IC. Initially, neither the amplifier circuit nor the

frequency-to-voltage converter worked, so with the help of Kunal Verma, Justin performed multiple tests to analyze why the circuit was not working as planned. Eventually the issue was taken care of.

To get the intended 5V voltage level for the circuit, Justin initially tried to implement an adjustable voltage regulator into the design. While the circuit wasn't terribly difficult to design the idea was scrapped for a simpler 5V regulator with a large input voltage range.

For the rechargeable battery Justin repurposed the use of the adjustable regulator as a source of constant current to recharge the 9V Nickel Metal Hydride cell. Unfortunately the team ran out of time and was unable to implement the recharge circuit into the final product; it was eventually used as a secondary component of the project.

Justin was also responsible for much of the solder work when we moved our final design from the breadboard to the perforated circuit boards we used for the final prototype.



Tim Ross

Tim Ross had a large technical role throughout the development of the Speed and Distance Sensor. He was primarily responsible for software implementation throughout the design, making him the Lead Programmer for our project. He also was partially responsible for interfacing the microprocessor with the EEPROM. As part of the team in charge of EEPROM interfacing, Tim researched and implemented I²C communication. After much work spent with the EEPROM design and many errors, the team was forced to abandon EEPROM interfacing and instead use an internal EEPROM on the microcontroller.

As Lead Programmer for the Design Team, Tim first spent time designing the Menu System. A simple intuitive system was required for our device, especially one that utilized limited input. This setup was implemented early in the design because it only required communication with the LCD, and no user input. After setting up the menu system, he implemented optional controls to the system, such as contrast and backlight control. He also developed small write functions to help with outputting data on the LCD.

After the decision to abandon EEPROM interfacing, Tim worked hard on using the microprocessor's internal EEPROM to store data. Due to the severely limited amount of memory of the EEPROM, Tim worked hard to manage the data efficiently, using dynamic allocation techniques. He developed and perfected the method used for reading data from the ADC and placing into memory. He developed the storage of run data and the method to access this data in memory. He implemented the system to delete run data and move them to maximize available space. He tirelessly debugged errors in code and output on the LCD throughout development of the program. He also analyzed code to minimize execution time and increase efficiency overall.



Temika Cage

During the beginning of the semester, when the work was divided, Temika's original responsibility was programming the microcontroller. She felt she was more skilled in programming than the electromagnetic and analog portion of the design. Also, time permitting; she would work on designing the project box in a computer aided design program.

Due to unfamiliarity with the language and coding technique used to work with the microcontroller, Temika refocused her efforts on creating the printed circuit boards. She researched several programs and decided to try Express PCB because it was a free design suite. While creating test circuits in Express PCB, Temika decided that the PCB software was slightly too complicated for the needs of the project. After contacting the ECE Shop staff, Temika was suggested to use Eagle design software because most students in past projects had preferred it over other programs. While learning the software program, the overall circuit topology for the final design was under constant modification. The team was not entirely sure how the circuit layout should be handled. Once a final design was created, time restraints restricted the design, simulation and production of a custom PCB.

Also, with the project being finished later than expected there was no time to create custom project boxes. Instead, standard boxes were purchased from the ECE Shop and RadioShack. Temika helped design both project boxes to contain all the necessary circuitry and created cutting diagrams in Microsoft Visio for proper machining of the project boxes.



Benjamin Guild

Ben Guilds technical contribution to the project mainly comprised of circuit layout, initial microprocessor programming, and assisting in program debugging and layout. He also contributed to component selection for our device.

In the initial stages of our product development, Ben developed code to read the ADC and display it on the LCD screen in order to give a testing baseline for our radar module. The code read the output of the radar module and frequency to voltage converter and converted it to values that could be displayed on the LCD screen and updated on the press of a button. Ben also contributed to the LCD screen selection and development.

While looking over application notes from previous semesters design teams, Ben found a reference to interfacing a PIC microcontroller to LCD screen document that only required the use of a single PIC pin and contained a very useful C library for using the LCD. He recommended this to the team and implemented the LCD screen and code, with modifications, in the design. He also added to the library a function to convert numerical values to characters so that they

could be displayed on the LCD. The LCD also had to have jumpers soldered for correct operation and the baud rate set through switches on the back of the device.

Tim Ross was the lead programmer on the project. Ben contributed to the later iterations of our code through assisting in debugging and memory management. The original thought was to use an external EEPROM device, but the programmers ran into trouble implementing the device and implemented memory on the microprocessor EEPROM as an alternative. Tim and Ben developed the memory layout scheme which segmented the memory in software to separate the run data as it is collected from the final data collected for each run and the configuration data. This allows the device to keep records of past runs without the possibility of overwriting previously saved data.

Appendix B

Literature and References

"Information Links." *ECE Design Team 10*. Web. 07 Feb. 2010.

<http://www.egr.msu.edu/classes/ece480/goodman/spring08/group10/documents/documents.htm>

"ECE480: Design Team 6." *ECE Design Team 6*. Web. 07 Feb. 2010.

<http://www.egr.msu.edu/classes/ece480/goodman/fall09/group06/>

"Amazon.com: Tech 4 O SI- Ski 1 Speedometer: Sports & Outdoors." *Amazon.com: Online Shopping for Electronics, Apparel, Computers, Books, DVDs & more*. Web. 08 Feb. 2010.

http://www.amazon.com/Tech-2831000-SI-Ski-Speedometer/dp/B000EQ82IM/ref=pd_sxp_f_r/191-3981142-2188238

"HB100 Motion Sensor Module" *AgilSense*. Web. 28 Apr. 2010.

<http://www.agilsense.com/pdf/x-band%20sensor/MSAN001-V101.pdf>

"LM324N Low Power Quad Operational Amplifiers" *National Semi*. Web. 28 Apr. 2010.

<http://www.national.com/ds/LM/LM124.pdf>

"LM2907 Frequency to Voltage Converter" *National Semi.*. Web. 28 Apr. 2010.

<http://www.national.com/ds/LM/LM2907.pdf>

"PIC12F4520 40 Pin Microcontroller" *Microchip*. Web. 28 Apr. 2010.

<http://ww1.microchip.com/downloads/en/DeviceDoc/39631E.pdf>

Appendix C

The following pages contain our Gantt chart, PIC code, and other extraneous documents.