Project Specification for the Navigation Assistance for the Vision Impaired System

Jacy Barr Samuel Jia Khai Lee Sean Sylwester

Abstract— **This document presents the specifications, designs, and test results for the Wearable Sensor for the Blind ECE Capstone project. The project seeks to design and implement a wearable system to help the blind navigate more safely by using ultrasonic and laser time-of-flight distance sensors to detect nearby obstacles, and haptic feedback modules to relay that information to the user. This document is intended to be sufficient for other engineers to entirely recreate this project using only this as reference. To that effect, the project's specifications, system-level and black-box block diagrams with accompanying block explanations, and testing procedures and results are all included.**

I. Introduction

The Navigation Assistance for the Vision Impaired ECE Capstone group's purpose is to develop wireless depth sensing modules which relay the distance of obstacles through haptic feedback motors. This document seeks to describe the research process and technical aspects of our design with enough detail such that the project could be recreated by other engineers. A comprehensive block diagram is included to give an overview of the system architecture, with each block being expanded upon to explain every component of the overall design. A review of the system requirements, design methodology, and project timeline are included to show the process by which the design was carried out.

II. Project Summary

We are developing a wearable system of sensors, NAVI (Navigation Assistance for the Vision Impaired), to act as a supplemental tool alongside the typical White Cane to help vision impaired individuals navigate. We are attempting to accomplish this by translating visual information of the user's surroundings that they may be missing into tactile feedback that they can respond to more easily.

Our project design process began by establishing the expectations of our stakeholder. Once the key features and requirements of our design were agreed upon, we broke the project down into blocks representative of each functional component of our system that we could develop independently. When integrated, these individual blocks would culminate in a system consisting of both a method of detecting nearby obstacles and a way to relay this information back to the user, all while remaining comfortable for the user to wear for extended periods of time. With this design approved our project stakeholder, we were able to start researching the appropriate hardware with which we could accomplish this task.

We had originally settled on a system comprised of several individual units, each of which contained a single long-range ultrasonic range detector and vibration motor. We quickly concluded that this design was limited in both scope and functionality. To address these concerns, we worked towards a new design that would provide 360° object detection in a more user-friendly form factor. The new design would consist of a headband of sensors pointing in all directions, and a waistband mounted with haptic feedback modules to relay the sensor information back to the user.

Although individual blocks were designed and implemented with ease, poor scheduling and delayed construction of our non-graded blocks resulted in a slew of problems when integrating the full system. A delayed PCB delivery and many wire connectivity issues led to an accidental short-circuiting that destroyed several components for which we had no replacement. We didn't have enough slack-time left to meet our initial deadline, so we needed to ask for a two-week extension. Thankfully, this extension was approved, and we were able to use this additional time to make the necessary changes to the PCB layout, enclosure, connectors, and wiring. We were then able to successfully pass our system checkoff, only falling slightly short of two out of our ten system requirements. Our team is now working on fixing those remaining issues and making quality-of-life improvements to ensure that our system is demo-ready for the Engineering Expo.

Reflecting on our project, there are several key lessons that we've learned along the way. Most importantly, we now recognize the necessity of planning for failure. Had everything worked on our first attempt, we would have had plenty of time to meet our initial deadline. But even minor mistakes or unexpected developments can compound on each other, leading to major problems and delays. We also now have a much better understanding of project development processes. It's crucial to begin with a solid design with as much room for error and as little risk as possible. Though even when following that mindset, we realize that it's often still useful to iterate on an initial design, using the knowledge gained from the first attempt to improve the design and implementation of the following versions. We will use what we have learned from this process to avoid some of these challenges in future endeavors.

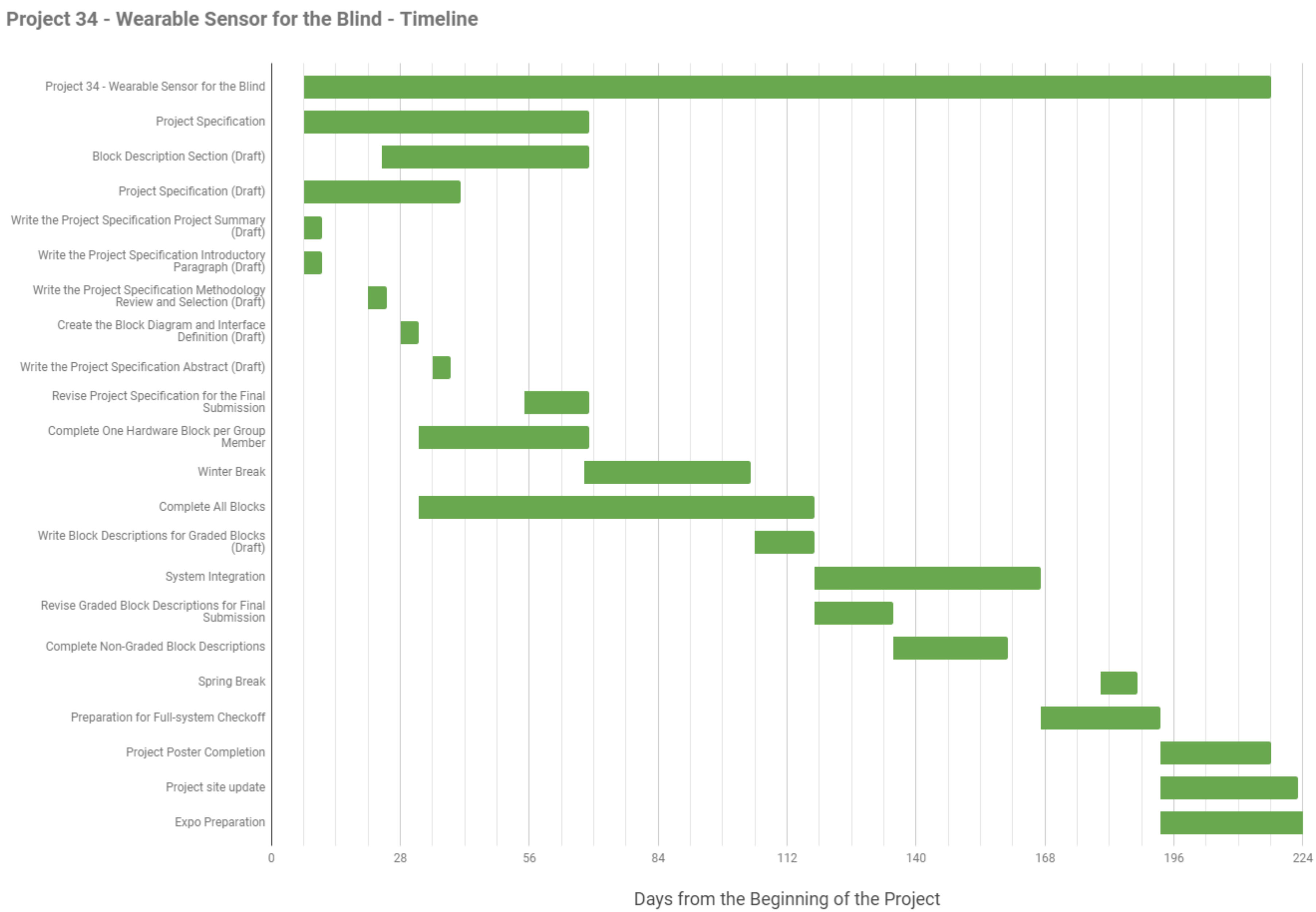


FIGURE 1. PROJECT TIMELINE

III. ENGINEERING REQUIREMENTS AND TESTING PROCEDURES

This section of the document will present the requirements and testing procedures for the Navigation Assistance for the Vision Impaired ECE Capstone project. Each engineering requirement (ER) is presented with the associated customer requirement (CR) provided by the project's stakeholders, and a step-by-step testing procedure to verify that the system meets the requirement.

1. **Battery Life**
    - 1.1. CR: Battery must last a long time.
    - 1.2. ER: The system will operate for at least 12 hours on battery power while objects are within 20cm of the system in all directions.
    - 1.3. Test Steps:
        - 1.3.1. Charge sensor module and feedback module batteries according to the Charging Time engineering requirement.
        - 1.3.2. Place the sensor band in an enclosed space so that all sensors detect objects within 20cm for maximum motor vibration.
        - 1.3.3. Leave the system in this configuration for 12 hours of continuous operation.
    - 1.4. Test Pass Condition: After 12 hours, the system is still functional.
2. **Charging Time**
    - 2.1. CR: Battery must be rechargeable.
    - 2.2. ER: Battery will charge from the level that the system doesn't turn on to the level that the system passes the Battery Life Engineering Requirement (ER1) in less than 15 hours.
    - 2.3. Test Steps:
        - 2.3.1. Switch on the system and leave it to discharge to the point that neither the sensor band nor the feedback band can be powered on.
        - 2.3.2. Plug in both the sensor band and feedback band and let them charge for just under 15 hours.
        - 2.3.3. Perform the Battery Life test (ER1)
    - 2.4. Test Pass Condition: The battery charges from the level that the system doesn't turn on to the level that the system passes the Battery Life Engineering Requirement (ER1) in less than 15 hours.
3. **Feedback Differentiation**
    - 3.1. CR: Users must be able to accurately sense object distance
    - 3.2. ER: After a 30-minute training session, 9/10 users will discern the distance to an object with a precision of 1m at 100% accuracy for objects 1-5m directly in front of the user, and with an accuracy of 20cm for objects within 1m of the user in all directions.
    - 3.3. Test Steps:
        - 3.3.1. Give at least 10 users a 30-minute training session on differentiating system feedback.
        - 3.3.2. Blindfolded/blind user will remain stationary while a 1x1m wide object is placed in one of the defined intervals.
    - 3.4. Test Pass Condition: 9/10 users can correctly identify the distance interval in which the object was placed to within 1m directly in front of the user, and to within 20cm in any direction.
4. **Low Power Mode**
    - 4.1. CR: Battery must last a long time.
    - 4.2. ER: System has an optional low power mode where: objects will be indicated with a precision of 1m at 100% accuracy for objects 1-5m directly in front of the user and an accuracy of 20cm within 1m of the user in all directions; the system will refresh in less than 300ms; and the system will operate for at least 24 hours while objects are within 10cm of the system in all directions.
    - 4.3. Test Steps:
        - 4.3.1. Configure the system for maximum power draw by placing objects within 20cm of every sensor.
        - 4.3.2. Switch the system to low power mode
        - 4.3.3. Leave system to run for 24 hours
    - 4.4. Test Pass Condition: After 24 hours, the system is still functional.
5. **Navigation**
    - 5.1. CR: Users must be able to navigate with the product without hitting anything.
    - 5.2. ER: 9/10 blind or blindfolded users will walk down a hall of at least 20m long and at least 1m wide and turn into a doorway without touching the walls, after up to 30 minutes of training and practice
    - 5.3. Test Steps:
        - 5.3.1. Give at least 10 test subjects a 30-minute training and practice session with the system.
        - 5.3.2. Have 10 test subjects walk down a hall of at least 20m long and at least 1m wide and turn into a doorway.
    - 5.4. Test Pass Condition: 9/10 users successfully walk down the hall and enter the doorway without hitting any walls.
6. **Object Direction**
    - 6.1. CR: User must be able to accurately sense obstacle direction.
    - 6.2. ER: System feedback module will indicate the direction to an object to within 30° of the direction measured by the system sensor module.
    - 6.3. Test Steps:
        - 6.3.1. Place a 1x1m wide object in a fixed location in range of the sensors
        - 6.3.2. Leaving the feedback band stationary, rotate the sensor band a full 360° in place.
    - 6.4. Test Pass Condition: The direction indicated by the feedback modules is always within 30° of the actual direction of the object for the full rotation of the sensor band.
7. **Response Time**
    - 7.1. CR: The feedback must update quickly
    - 7.2. ER: System feedback responds to changes in object position in less than 200ms

7.3. Test Steps:
7.3.1. Place tape over all but one sensor in a group corresponding to a single motor.
7.3.2. Place a switch such that in one position it covers the remaining sensor, and in the other position it does not.
7.3.3. Attach a battery and an oscilloscope to the switch so that in one position, the oscilloscope reads the battery voltage, and in the other position it reads 0V.
7.3.4. Connect another oscilloscope probe to the signal pin of the motor being tested.
7.3.5. Configure the oscilloscope to measure the time between the switch flipping to cover the sensor, and the motor reflecting that change.
7.4. Test Pass Condition: The feedback updates within 200ms.

8. Water Resistance
8.1. CR: Product must be durable.
8.2. ER: Every component in the system is water resistant according to the IPX4 standard.
8.3. Test Steps:
8.3.1. For 5 minutes, splash 10 liters per minute on the system at a pressure of 80 kPa.
8.3.2. Repeat step 1 for every component in the system
8.4. Test Pass Condition: No water enters any enclosure during any of the tests

9. Wearable Comfort
9.1. CR: Product must be comfortable.
9.2. ER: The system weighs less than 900g, and after 15 minutes of use, 9/10 users report the system is comfortable.
9.3. Test Steps:
9.3.1. Have at least 10 users wear the system for 15 minutes.
9.4. Test Pass Condition: 9/10 users think that it is comfortable, and the system weighs less than 900g.

10. Wireless Communication
10.1. CR: System must include a sensor band that reliably communicates wirelessly with a separate feedback band.
10.2. ER: Wireless connection between the sensor and feedback bands remain uninterrupted for 12 hours.
10.3. Test Steps:
10.3.1. After performing the 12-hour Battery Life Engineering Requirement test, check to see if the sensor band and feedback band are still connected wirelessly.
10.4. Test Pass Condition: After the 12-hour Battery Life Engineering Requirement test, the sensor band and feedback band are still connected wirelessly.

11. Drop Durability
11.1. CR: System must be durable.
11.2. ER: System functionality is not affected in any way by a drop from 2m.
11.3. Test Steps:
11.3.1. Drop the system from a height of 2m.
11.4. Test Pass Condition: System functionality is not affected in any way by the drop.

12. Temperature
12.1. CR: System must be durable.
12.2. ER: System distance accuracy is not affected by changes in temperature between -20°C and 60°C
12.3. Test Steps:
12.3.1. In a walk-in freezer, place a 10x10cm object 0.5m directly in front of the system sensors.
12.3.2. In a room-temperature room, place a 10x10cm object 0.5m directly in front of the system sensors.
12.3.3. In a sauna, place a 10x10cm object 0.5m directly in front of the system sensors
12.4. Test Pass Condition: All 3 measurements have the same system feedback output.

13. Price
13.1. CR: System must not cost too much.
13.2. ER: System component cost sums to less than $200.
13.3. Test Steps:
13.3.1. Sum the cost of all components in the system.
13.4. Test Pass Condition: Sum is less than $200.

14. Body Measurements
14.1. CR: System must be wearable by most people
14.2. ER: System harnesses will fit on people with body measurements within 2 standard deviations of the mean.
14.3. Test Steps:
14.3.1. Measure the minimum and maximum chest and head circumferences supported by the system.
14.4. Test Pass Condition: The size range of the two bands would accommodate people within 2 standard deviations of mean head and chest circumferences.

15. Enclosure Size
15.1. CR: System must not be too bulky.
15.2. ER: Enclosure is no larger than 6x5x4cm.
15.3. Test Steps:
15.3.1. Measure the length, width, and height of the enclosure.
15.4. Test Pass Condition: Enclosure is no larger than 6x5x4cm.

16. Minimum Object Width
16.1. CR: System must detect thin objects.
16.2. ER: The system indicates objects as thin as 1m at 5m directly in front of the user, and as thin as 30cm at 1m from the user in any direction.
16.3. Test Steps:
16.3.1. Place a 1x1m object 5m directly in front of the user and note if the object is detected.
16.3.2. Place a 30x30cm object 1m at 90° off center in front of the user.
16.4. Test Pass Condition: Both objects are detected.

17. Enclosure Durability
17.1. CR: System must be durable
17.2. ER: Enclosure withstands 20lbs of weight placed directly on top of it for 30s without breaking in any way.
17.3. Test Steps:
17.3.1. Lay the enclosure on a flat surface.
17.3.2. Place a 20lbs weight on top of the enclosure.

17.4. Test Pass Condition: After 30s, the enclosure has not broken in any way.

18. Feedback Intensity

18.1. CR: User must be able to control the feedback intensity

18.2. ER: Feedback intensity is user-configurable on a range from 30-100% maximum intensity.

18.3. Test Steps:

18.3.1. Measure feedback intensity at the minimum and maximum settings.

18.4. Test Pass Condition: Feedback at minimum settings is less than 30% of the maximum setting.

19. Wireless Consistency

19.1. CR: The wireless link must be reliable

19.2. ER: The wireless link between the sensor and feedback bands maintain 100% uptime when separated by 1m.

19.3. Test Steps:

19.3.1. Separate the sensor and feedback bands by 1m, and switch both on.

19.3.2. Leave the system running for 12 hours

19.4. Test Pass Condition: The wireless link is still connected after 12 hours.

20. Feedback Module Locations

20.1. CR: Feedback modules must stay in the same relative positions around the user as the harness is adjusted to fit different body sizes.

20.2. ER: Location of feedback modules on the feedback band do not vary by more than 20° between the minimum and maximum allowed dimensions.

20.3. Test Steps:

20.3.1. Measure the angles of each feedback module on each band relative to the front at the minimum and maximum band size settings.

20.4. Test Pass Condition: No feedback module varies by more than 20° between the minimum and maximum band size settings.

IV. Methodology Review And Selection

This section of the document lists comparison tables for the sensors, batteries, Bluetooth modules, microcontrollers, haptic feedback modules, and magnetometers that could potentially be used in the system, along with a selection for the best component for this design, and reasoning for the selection. Also included is a comparison table of similar projects, which highlights key design considerations that were used to make this design more successful.

Developing a wearable sensor for the blind to satisfy the project's stakeholders, this design must remain comfortable and accurate without sacrificing longevity. To achieve this, power consumption, weight, and accuracy must be prioritized while remaining within a tight budget. A design using ultrasonic and IR sensors along with haptic feedback motors driven by efficient controllers provides the best performance, energy efficiency, and comfort. Utilizing HC-SR04 and VL53L0X sensor modules for their range, price, and efficiency; Alkaline AAA batteries for their high capacity and small form factor; Arduino Pro Minis with the HC-05 Bluetooth modules for their efficiency and small size; and coin-cell haptic feedback motor for their price, size, and efficiency, our design will ensure measurement accuracy, low cost, power efficiency, and user comfort.

Table I. Sensor Comparisons

Sensor Name	Price	Sensor Type	Microcontroller Included?	Range (m)	Angle	Voltage (V)	Current (mA)	Notable Features
JSN-SR04T	$15.39	Ultrasonic 40kHz	Yes	0.02 - 4.5	75°	5	30	Waterproof sensor
HC-SR04	$0.97	Ultrasonic 40kHz	Yes	0.02 - 4	15°	5	15	
SEN0001	$13.90	Ultrasonic	Yes	0.05 - 5	N/A	3.3-5	20	Temperature Correction, Built-in Servo Controller
Intel® RealSense ™ Camera (R200)	$109.00	IR Light Projection	Yes	0.4 - 2.8	46°	3.3	500	
IR Distance Sensor	$24.95	IR	No	1 - 5.5	59°	5	30	
VL53L0X	$3.95	IR	No	0.02-1.2m	25°	2.8	20	Extremely short measurement time (20ms)

Table I compares six different sensor types that could be used for this design. The three primary sensor types capable of measuring several meters, as required for this design, are: ultrasonic sonar, RGB-D cameras, and infrared laser rangefinders. Since this project has a tight budget, and the product must be comfortably wearable, the ideal sensor for this design would be inexpensive, small, low power, and accurate. While being very accurate, RGB-D cameras fail the first 3 of those criteria, generally costing hundreds of dollars, and being bulky, and power-hungry. This leaves ultrasonic and medium-range IR sensors as the best choice for this project, meeting all 4 criteria.

The inexpensive ultrasonic distance sensor market is dominated by a single module: the SR04. The HC-SR04 module is the simplest module available using the SR04 sensor, and can be found for less than a dollar. Other modules add useful features such as waterproof sensors, temperature correction, and integrated servo controllers, but these all come at a significant price, with the cheapest options still costing ~15x more than the barebones model. Due to the budget and space constraints of our design, the base HC-SR04 module best fits our medium-range sensor needs.

Similarly, the VL53L0X dominates the medium-range IR sensor market. Most other IR sensors have significantly longer range, which increases cost, power, and measurement time. Since this project only requires a range of 1m from the laser sensors, increasing the range farther is not beneficial, so the VL53L0X is the best solution for the short-range sensor due to its low cost, low power consumption, and useful range.

TABLE II. POWER SUPPLY COMPARISONS

Type	Composition	Manufacturer	Size	ANSI	Dimensions (mm)	Nominal Voltage (V)	Capacity (mAh)	Weight (g)	Price ($)
Primary	Alkaline	Atomic Ultra Alkaline	AA	15A	14.5 x 50.5	1.5	2000	23	0.247
		Energizer E91	AA	15A	14.5 x 50.5	1.5	3000	23	0.7
		Duracell Coppertop	AA	15A	14.5 x 50.5	1.5	2850	22.7	0.62
		Rayovak	AAA	24A	9.5-10.5 x 43.3-44.5	1.5	1123	11	0.49
		Panasonic	AAA	24A	9.5-10.5 x 43.5-44.5	1.5	794	10.83	0.35
		Fujitsu	AAA	24A	10.5 x 44.5	1.5	1140	11	0.295
	Zinc-Carbon	Eveready	AA	15D	14.5 x 50.5	1.5	1100	15	0.297
		Panasonic	AA	15D	13.9 x 50.25	1.5	1100	17.5	0.25
		Panasonic	AAA	15D	10.5 x 44.5	1.5	N/A	9.7	0.25
		Eveready	AAA	15D	10.5 x 44.5	1.5	540	9.7	0.413
Secondary	NiCd	Tenergy	SubC	N/A	23 x 43	1.2	2200	51	2.04
		Power Portable	F	N/A	33 x 91	1.2	7000	187	11.99
		Yellow Sleeve	AA	N/A	14 x 49	1.2	1000	22	2.89
		White Sleeve	A	N/A	17 x 49	1.2	1600	30	2.69
		White Sleeve	4/5 A	N/A	17 x 43	1.2	1200	31	2.59
	NiMH	Green Sleeve	F	N/A	33 x 91	1.2	14000	246	16.99
		Green Sleeve	4/3 AF	N/A	18 x 67	1.2	4500	62.3	5.79
		Green Sleeve	4/3 AF	N/A	18 x 67	1.2	4200	62.3	5.29
		Green Sleeve	4/3 AF	N/A	18 x 66	1.2	3800	53	4.39
		Sanyo	A	N/A	17 x 49	1.2	2100	36	3.99
		Green Sleeve	A	N/A	17 x 49	1.2	2700	36	3.89

TABLE II compares 21 different batteries that could be used for this design. Reviewing the possible power sources used in portable devices, there is a clear divide among types. Batteries are defined as either primary(disposable) or secondary(rechargeable) with each supporting respective nominal voltages. Comparing NiCd and NiMH secondary compositions, NiCd batteries are lower in weight and cost, but sacrifice capacity. NiMH on the other hand offers higher capacities, weight, and price points. Secondary types are available in a variety of forms, but do so at a higher cost than disposable. Primary cells are more energy dense for their weight and size and are available for a fraction of the price in comparison. The Alkaline composition out performs Zinc-Carbon in each for each of the aforementioned features.

An important aspect of our design is the longevity of the device's battery while remaining comfortable for the user during extended use. Working from a limited budget, we can conclude that for a power source to accomplish this, a high capacity battery with minimal size, weight, and cost should be utilized. While secondary units are available in a variety of sizes, the low nominal voltage and added bulk needed to implement a charging circuit does not align with our projects design. Primary AAA sizes however (particularly alkaline compositions), offer respectable capacities at low weights and volumes. The low cost and higher nominal voltage of an alkaline AAA primaries marks them as good candidates for our design.

TABLE III. BLUETOOTH MODULES

Bluetooth Name	Manufacturer	Dimensions	Description	Price ($)
HC-05	DSD Tech	1.1 x 0.6 x 0.1"	Bluetooth 2.0 Master/Slave Mode Min Baud Rate: 9600 Max Baud Rate: 460800 Current 30~40mA	7.99
HC-06	DSD Tech	1.1 x 0.6 x 0.1"	Bluetooth 2.0 Slave Mode Min Baud Rate: 1200 Max Baud Rate: 1382400 Current 30~40mA	8.49
EZ-Link	BlueFruit	1.6 x 0.8 x 0.2"	Bluetooth 2.0 Slave Mode Min Baud Rate: 1200 Max Baud Rate: 1382400	22.50
BlueSMiRF	SparkFun	1.7 x 0.6 x 0.15"	Bluetooth 2.0 Slave Mode Min Baud Rate: 1200 Max Baud Rate: 1382400	19.96

TABLE III compares four different Bluetooth module that could be used for this design. The dimensions of the components are similar and small in form. Based on the review of several variations of Bluetooth modules, the major difference between each component is the ability to have Master and Slave mode which exists in HC-05. To build a connect by sending and receiving data between microcontrollers, the master mode is required. Therefore, the DSD Tech HC-05 Bluetooth module is chosen for this project.

TABLE IV. MICROCONTROLLER COMPARISONS

Microcontroller Name	Manufacturer	Dimensions	Core	Description	Price ($)
DEV TEENSY LC 13305	Sparkfun Electronics	1.4 x 0.7"	ARM® Cortex®-M0+	3.3V or 5V to other devices at up to 5-20mA	14.38
ARDUINO NANO	Arduino	0.73" x 1.70"	ATmega328	DC Current per I/O Pin 40mA Operating Voltage (logic level) 5 V Input Voltage (limits) 6-20 V Input Voltage (recommended) 7-12 V	21.49
DEV TEENSY 3.2 13736	SparkFun Electronics	1.4 x 0.7"	ARM® Cortex®-M4	3.3V to other devices at up to 100mA	22.50
ARDUINO MICRO	Arduino	1.9" x 0.69"	ATmega32U4	Operating Voltage 5V Input Voltage (recommended) 7-12V Input Voltage (limits) 6-20V DC Current per I/O Pin 40 mA DC Current for 3.3V Pin 50 mA	24.95
TRINKET MINI MCU BOARD 5V	Adafruit Industries LLC	1.05 x 0.6"	ATtiny85	3.3V or 5.0V power regulator with 150mA output capability	6.95
BEETLE	DFRobot	0.79 x 0.87"	ATmega32u4	Operating Voltage: 5V DC	7.90
ATmega328 Pro Trinket	Adafruit Industries LLC	1.5" x 0.7"	ATmega328P	Operating Voltage: – 1.8 - 5.5V On-board 5.0 V power regulator with 150 mA output capability	9.95
Adafruit Metro Mini 328	Adafruit Industries LLC	0.7" x 1.7"	ATmega328	Input voltage (Vin): 6-16V (9V recommended) USB-to-Serial built in 5V onboard regulator with 150mA out, 3.3V 50mA	12.50
Bluno Beetle	DFRobot	1.13" x 1.30"	ATmega328	Operating voltage: 5V DC Wireless Programming via BLE	14.90
Arduino Pro Mini 3.3V / 5V	SparkFun Electronics	1.3x0.70"	ATmega328	3.3V at 8MHz Max 150mA output 5V at 16MHz Max 150mA	9.95

TABLE IV compares ten different microcontrollers that could be used for this design. One of the most crucial elements for choosing the right microcontroller for this project is the size. To create a wearable device that is comfortable for a long period of time, the best choice is to have a physically small microcontroller. The table above compares the differences between multiple possible microcontrollers in terms of their sizes, core, operating voltages and prices. Each microcontroller has similar operating voltages which are either 3.3V or 5V. In order to interact with other hardware components such as the sensors and feedback modules effectively, the microcontroller will need to operate at low power to conserve the battery life to achieve a long battery life. With a simple modification (e.g. by disabling the power LED), a microcontroller (in this case, a 5V Arduino Pro Mini), we could lower the power consumption to 23 µA in Power Down Sleep (PDS) and 16.9 mA on Active Mode (ACT) from the stock version of 3.14 mA and 19.9mA respectively. With a limited budget, a low cost and a small size microcontroller is preferable.

TABLE V. HAPTIC FEEDBACK MODULE COMPARISONS

Type	Model	Dimensions (mm)	Voltage (V)	Current (mA)	Weight (g)	RPM	Price ($)
button	301-101	11.8 x 10 x 3.55	2.5-3.8	75-85	1.2		
coin	ROB-08449	55 x 10.1 x 3.65	2.5-3.6	60		13000	4.95
disc	VMD0290916	2.3 x 12	2.2-2.6	60		13000	
cylinder	307-100	72.2 x 8.8 x 8.8	0.75-3.6	250-680	4.6	13500	

TABLE V compares four different haptic feedback module types that could be used for this design. Based on the review of several variations of feedback modules, more costly models are those with higher rpm ratings. The dimensions of the unit do not appear to have much significance, as most of the motors are small in form. The servo and coreless models are significantly larger and would add significant weight to the completed design. Several coreless motors able to withstand more current while others do not. The brand also plays a role on the prices charged for each item. Although not all were, many feedback module's prices appeared to be scaled with the amount current they withstand.

TABLE VI. SIMILAR PROJECT COMPARISONS

Project Name	Sensor Type	Sensor Location	Direction Scanned	Feedback Type	Range	Unique Features
Smart Cane	Ultrasonic	Cane	Direct front, above belt	Vibration of Cane	3m	In production in India for $90
Silicon Eyes	Ultrasonic	Glove	Wherever the glove is pointed	Vibration of Glove	3m	Braille keyboard input
A Path Force Feedback Belt	IR Camera	Hand-held	360°	Vibration on Belt	Depends on camera	
A design of blind-guide crutch based on multi-sensors	Ultrasonic	Cane	Direct front and sides, above belt	Vibration of Cane, Audio Cues	3m	
Navigation Assistance Using RGB-D Sensor with Range Expansion	RGB-D Sensor	Hung around neck	Wide range in front, ground	Verbal Cues, Audio Cues		
Ultrasonic Assistive Headset for visually-impaired people	Ultrasonic	Headset	360°	Audio Cues	5m	Solar Powered
A Mobility Device for the Blind with Improved Vertical Resolution Using Dynamic Vision Sensors	Dynamic Vision Sensor	Headset	Direct Front, above belt	3D Audio		Very low power consumption
SUGAR System	Ultra-Wide Band Sensor	Wall-Mounted, smartphone interface	Entire room	Audio Cues	50m	Extremely long range

TABLE VI compares eight projects similar to this design. There are dozens of projects similar to our own, with vastly different implementations, all attempting to solve the deceptively simple problem of helping the blind navigate. The most common sensor type is ultrasonic, especially for the lower-budget projects most similar to ours. More sophisticated sensor systems typically don't leave the research lab, or are too expensive to be viable for mass production. Feedback types vary significantly, ranging from headset-given audio cues to haptic feedback vibration motors installed into wearable gloves. However, audio feedback has the drawback that it obstructs hearing ability, which the blind rely heavily on for navigation. The projects that are inexpensive, simple, and effective enough for our team to model our designs off all use ultrasonic sensors with one or more vibration motors, mounted on convenient locations such as the existing white cane, or a small glove.

TABLE VII. MAGNETOMETER COMPARISONS

Name	Price ($)	Manufacturer	Heading Accuracy (°)	Interface	Sensitivity (μT)	Voltage (V)	IC
SEN12670	14.95	Sparkfun	n/a	I2C	0.1	1.95-3.6	MAG3110
SEN10530	n/a	Sparkfun	1-2	I2C	n/a	2.16-3.6	HMC5883L
1528-1226-ND	34.95	Adafruit	2.5	I2C	0.3	2.4-3.6	BNO055
MMC5883MA (IC only)	1.98	Mesmic Inc.	1	I2C	0.7	2.16-3.6	MMC5882MA
LSM303	14.95	Adafruit	n/a	I2C	0.5	3-5	LSM303DLHC

TABLE VII compares several models of magnetometers that could be utilized to orient the devices in our design. After reviewing the available models, there are very few that come with an accompanying breakout board and many that are only available in large quantities as surface mount parts. For simplicities sake and cost, a unit shipped with a breakout board would be ideal. Our project utilizes both 3.3 and 5v voltages provided through regulators, so of the reviewed magnetometers, there wouldn't be an issue using any one of them. All models have relatively high accuracy and similar sensitivities which would work well for what we are trying to accomplish. Therefore, among the compared options, the SEN12670 magnetometer would be the ideal choice for our design regarding price, availability, and simplicity.

V. BLOCK DIAGRAM AND INTERFACES

This section shows the block diagrams for the system, as well as a complete listing of the interfaces between the blocks and the properties of each interface. The black-box diagram in Fig. 2 shows the inputs and outputs to the system as a whole, which the top-level block diagram in Fig. 3 expands upon to show the individual blocks that this design implements, and finally TABLE VIII defines the properties of each interface shown in the top-level block diagram.

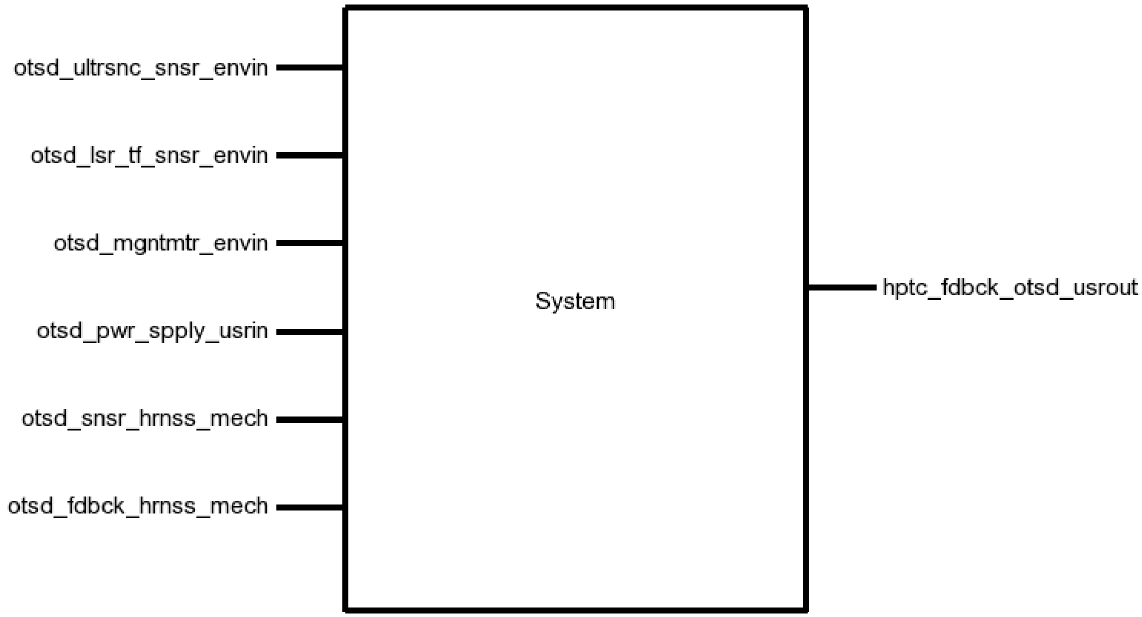


FIG. 2. SYSTEM BLACK-BOX DIAGRAM

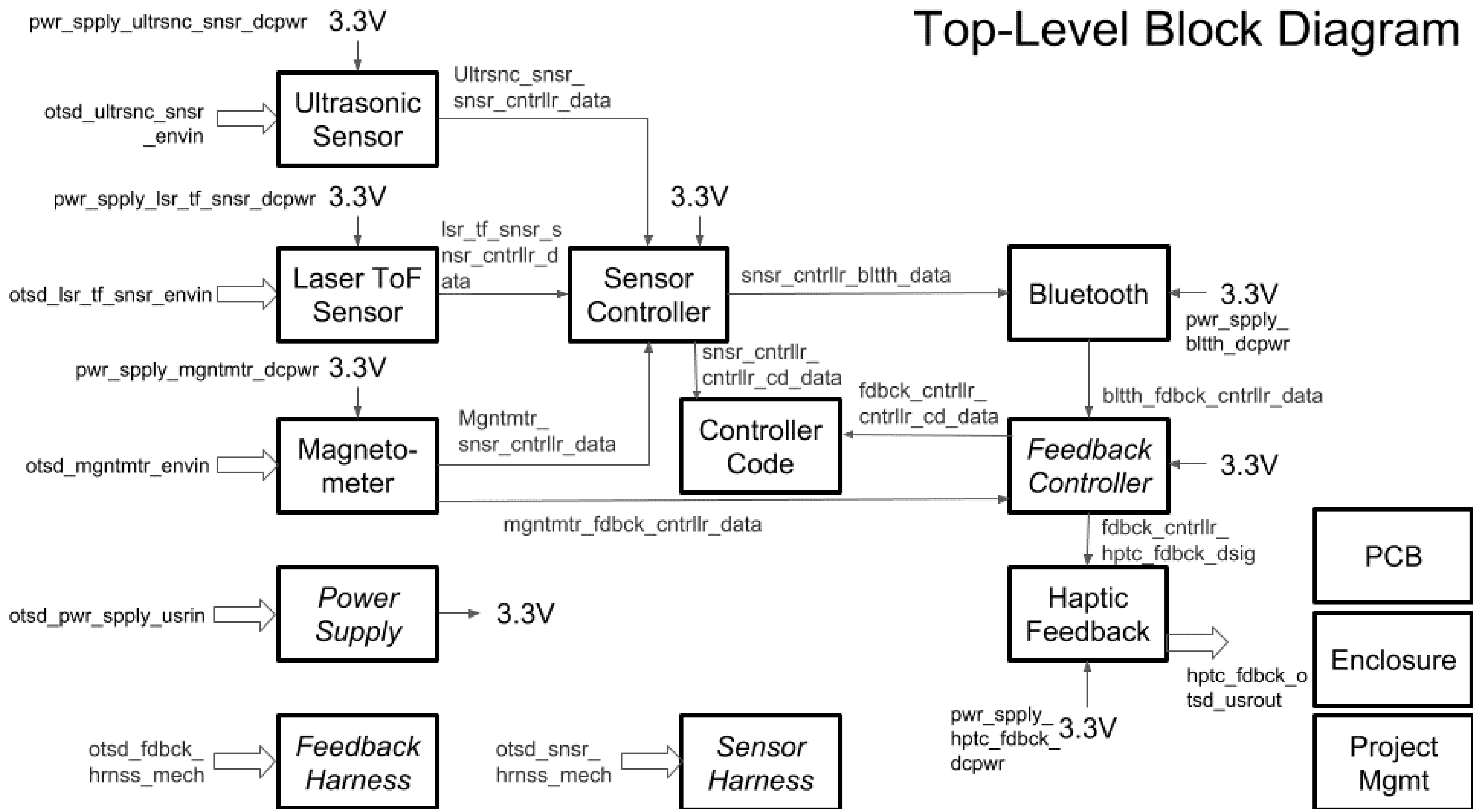


FIG. 3. SYSTEM TOP-LEVEL BLOCK DIAGRAM

TABLE VIII. INTERFACES

Name	Properties
otsd_ultrsnc_snsr_envin	Other: Accuracy: 10cm Other: Angle: 75° Other: Frequency: 40kHz Other: Range: 20-500cm
otsd_lsr_tf_snsr_envin	Other: Measurement Time: 60ms Other: Accuracy: 12% Other: Range: 1m(white) 0.4m(black) Other: Angle: 5°
otsd_mgntmtr_envin	Electromagnetic: Minimum Magnetic Flux: -1mT Electromagnetic: Magnetic Flux Resolution: 1µT Electromagnetic: Maximum Magnetic Flux: 1mT Other: Direction Measurement Accuracy: ±15°
otsd_usr_cntrls_usrin	Type: Sensor Band Switch (Off/On/LowPower) Type: Battery Compartment Type: Feedback Module Switch (Off/On) Type: Knob to control feedback intensity (Analog Off to Full Power)
ultrsnc_snsr_cntrllr_cd_data	Other: Measurement Time: 150µs-25ms (38ms if no obstacle) Other: Pulse Voltage: 5V Other: Echo Pin Level During Measurement: 1 Protocol: TTL
lsr_tf_snsr_cntrllr_cd_data	Datarate: 115.2kHz Messages: Sensor Data Other: Measurement Time: 60ms Protocol: I2C
mgntmtr_cntrllr_cd_data	Datarate: 112.5kHz Messages: Sensor Data Other: Measurement Frequency: 20Hz Protocol: I2C
pwr_spply_ultrsnc_snsr_dcpwr	Inominal: 5mA ± 10% Ipeak: 30mA Vmax: 5.5V Vmin: 3V
pwr_spply_lsr_tf_snsr_dcpwr	Inominal: 20µA ± 10% per sensor (between measurements) Ipeak: 30mA per sensor (during measurement) Vmax: 5V Vmin: 3V
pwr_spply_mgntmtr_dcpwr	Inominal: 5µA (standby) Ipeak: 0.5mA (measurement) Vmax: 3.6V Vmin: 1.95V
pwr_spply_mcrcntrllr_dcpwr	Inominal: 9mA ± 10% Ipeak: 20mA Vmax: 5.5V Vmin: 1.8V
pwr_spply_bltth_dcpwr	Inominal: 30mA ± 10% (during pairing) Inominal: 8mA ± 10% (after pairing) Vmax: 6V Vmin: 3.6V
pwr_spply_hptc_fdbck_dcpwr	Ipeak: 60mA per motor Vmax: 3.6V Vmin: 3V Vnominal: 3.3 ± 10% at 100% duty cycle
usr_cntrls_pwr_spply_usrin	Type: Power Switch (Off/On) Type: Battery Compartment Usability: 9/10 Users will change the batteries in less than 2 minutes
usr_cntrls_cntrllr_cd_usrin	Type: Low Power Mode Switch (Off/On) (Active HIGH)
cntrllr_cd_ultrsnc_snsr_data	Messages: Trigger Measurement: 0x55 Other: Pulse Width: 10µs Other: Pulse Voltage: 5V Protocol: TTL
cntrllr_cd_lsr_tf_snsr_data	Datarate: 115.2kHz Messages: Address Programming, Measurement Trigger, Measurement Read Other: Measurement Time: 60ms Protocol: I2C

VI. Block Descriptions

This section of the document details the design of each blocks shown in Fig. 3. Each description includes an overview of the function of the block within the system, a schematic or block diagram describing the design of the system, a wiring diagram for integrating it with the rest of the system, a listing of the properties of the interfaces into the system, step-by-step testing procedures for testing each property, and validation for the design choices made. They are listed alphabetically according to the names shown in Fig. 3.

1. Laser ToF Sensor

Block Owner: Sean Sylwester

1.1. Introduction

The purpose of this document is to describe the Laser Time of Flight (ToF) Sensor block of the Wearable Sensor for the Blind ECE Senior Capstone Project to other engineers with enough detail such that they would be able to reproduce and test this block using only this document as reference. The Laser ToF block is implemented using a band of many VL53L0X ToF Ranging Sensor from ST Microelectronics on custom-built PCBs. Included in this document are an overview of the block, a schematic and wiring diagram, the properties of the interfaces with the other blocks in the system and testing procedures for each, and reasoning for why this design is the best solution for this block.

1.2. Block Overview

The Laser ToF Sensor block will provide the distance to the nearest object in eight 45° zones around the user to the system, which will use that information to set the system haptic feedback module that will convey that distance information to the user. The measurements taken need to be fast and accurate, so that the user can quickly and confidently navigate around obstacles in their environment. Fig. 4 below shows the black box diagram of the system. *otsd_lsr_tf_snsr_envin* represents the environmental input to this block, namely the distance to the nearest objects. *lsr_tf_snsr_cntrllr_cd_data* represents the I^2C input from the microcontroller, which triggers measurements for the sensors and programs the ranging profile. *pwr_spply_lsr_tf_snsr_dcpwr* represents the connection to the power supply, which is regulated to 3.3V, and should draw about 30mA per sensor during measurements. And finally, *cntrllr_cd_lsr_tf_snsr_data* represents the I^2C data output from this block to the microcontroller. All the properties for these interfaces can be found in Table IX.

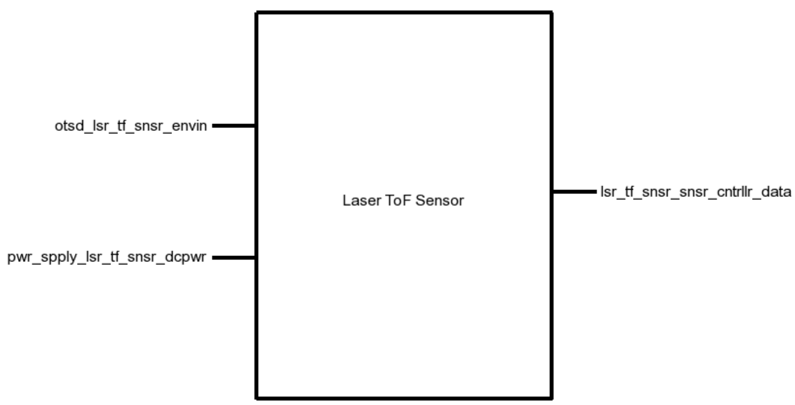


Fig. 4. Black Box Diagram of the Laser ToF Sensor Block

Table IX. Laser ToF Sensor Block Interfaces And Properties

Name	Properties
otsd_lsr_tf_snsr_envin	1. Other: Accuracy: 12% 2. Other: Range: 0.4m(Black) 1m(White) 3. Other: Angle: 5° 4. Other: Measurement Time: 60ms
lsr_tf_snsr_snsr_cntrllr_data	1. Datarate: 100kHz 2. Messages: Address Programming, Measurement Trigger, Measurement Read 3. Other: Measurement Time: 60ms 4. Protocol: I2C
pwr_spply_lsr_tf_snsr_dcpwr	1. Inominal: 35µA ± 10% per sensor (between measurements) 2. Ipeak: 30mA per sensor (during measurement) 3. Vmax: 3.5V 4. Vmin: 2.6V

1.3. Verification

This section details the testing procedures to verify every property listed in Table IX. All tests must be passed successfully before this block will be integrated into the rest of the system. Testing will be completed using a single VL53L0X sensor board.

Power and Communication Testing

This test will verify the power requirements for this block on the *pwr_spply_lsr_tf_snsr_dcpwr* interface, and the communication requirements on the *cntrllr_cd_lsr_tf_snsr_data* and *lsr_tf_snsr_cntrllr_cd_data* interfaces.

1. Connect a VL53L0X sensor to a DC power supply that can display current, set to 2.6V, via the *pwr_spply_lsr_tf_snsr_dcpwr* interface.
2. Connect a microcontroller using I2C at 100kHz to the *otsd_lsr_tf_snsr_envin* interface.
3. Load a test program on the microcontroller than takes a measurement for 200ms, then idles for 200ms.
4. Note the current displayed on the DC power supply during a measurement, and during idle.
5. Repeat steps 1-4 with the DC power supply set to 3.5V.

PASS: This test passes if the current draw never exceeds 30mA, and if the I2C interface communication links at 100kHz.

Link to Video

Measurement

This test will verify the measurement requirements for this block on the *otsd_lsr_tf_snsr_envin* interface.

1. Connect a VL53L0X sensor to a DC power supply set to 3.3V via the *pwr_spply_lsr_tf_snsr_dcpwr* interface.
2. Connect a microcontroller using I2C at 100kHz to the *otsd_lsr_tf_snsr_envin* interface.
3. Load a test program on the microcontroller than takes a measurement for 60ms, then prints out the distance measurement, then idles for 40ms.
4. Place a white object 1m directly in front of the sensor and note if the sensor detects the object.
5. Place a black object 0.4m directly in front of the sensor and note if the sensor detects the object.
6. Place a white object 30cm away, and 3cm off-center (5°) and note if the sensor detects the object.
7. Place a white object 30cm directly in front of the sensor and note if the distance measurement is within 10% of 30cm.

PASS: If the measurements complete in less than 60ms, objects in steps 4, 5, and 6 were detected, and the object in step 7 was detected within 10%.

Link to Video

1.4. Design

The schematic in Fig. 5 presents this block's wiring diagram, including the interfaces of this block to the rest of the system. Fig. 6 shows the schematic for the custom-build sensor interface PCB, and Fig. 7 shows the physical layout of this PCB.

This block is implemented using 30 VL53L0X Laser ToF Sensors mounted on a headband to achieve the 360° sensing system requirement. All the sensors are connected to the same I^2C (SDA and SCL) bus. Since all the sensors have the same default I^2C address on startup, they need to be enabled one-at-a-time to reprogram each address to be unique. This is done with a MAX6895 Delay IC, which propagates a boot signal through each sensor in turn with a 30ms delay. In the 30ms window after a sensor boots, that sensor will be the only one on the I^2C bus with the default address, so it can be reprogramed without any conflict.

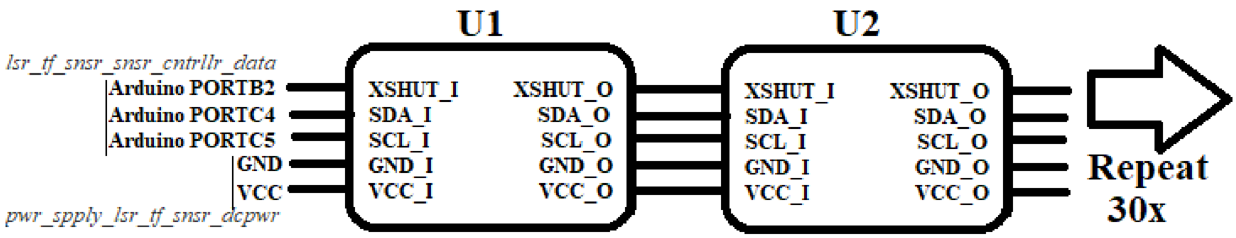


FIG. 5. WIRING DIAGRAM FOR THE LASER TOF SENSOR BLOCK

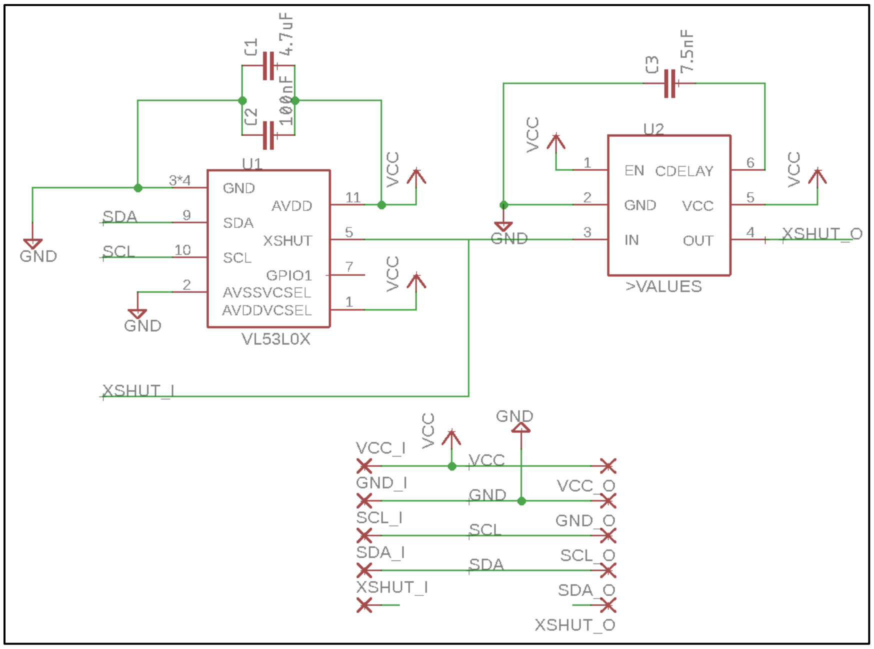


FIG. 6. VL53L0X INTERFACE PCB SCHEMATIC

The values for smoothing capacitors C1 and C2, 4.7µF and 100nF, respectively, were indicated in Figure 3 in the VL53L0X datasheet [8]. Capacitor C3 was chosen to set the delay of the MAXIM Delay IC to be 30ms, which was calculated using the equation $t_{DELAY} = [C_{CDELAY} \text{ x } 4.0 \text{ x } 106 + 40\mu s]$ from page 7 of the MAX6895 datasheet [9].

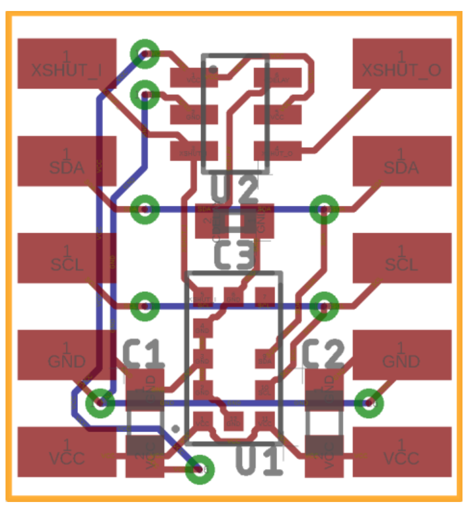


FIG. 7. VL53L0X INTERFACE PCB LAYOUT

1.5. Design Validation

For this block, the VL53L0X sensor was used because it fits the low cost, low power, medium range, high accuracy, and fast performance that this project required of the Laser ToF Sensor block. Table X below validates all the system properties using the VL53L0X Datasheet.

TABLE X. INTERFACE PROPERTY VALIDATION FOR THE LASER TOF SENSOR BLOCK

Property	**Validation**
otsd_lsr_tf_snsr_envin	
Other: Accuracy: 12%	Table 12 of the VL53L0X Datasheet states that the worst-case accuracy is 12% [8].
Other: Range: 0.4m(Black) 1m(White)	Table 11 of the VL53L0X Datasheet states that the minimum range indoors is 1.2m for a white object, and 0.7m for a grey object [8].
Other: Angle: 5°	VL53L0X Datasheet states that the FOV of the sensor is 25° [8].
Other: Measurement Time: 60ms	Table 13 of the VL53L0X Datasheet states that minimum timing budget is 20ms [8].
lsr_tf_snsr_snsr_cntrllr_data	
Datarate: 100kHz	Table 3 of the VL53L0X Datasheet states that the maximum I2C operating frequency is 400kHz [8]
Messages: Address Programming, Measurement Trigger, Measurement Read	Section 2 of the VL53L0X Datasheet states that Address Programming, Measurement Trigger, and Measurement Read commands can be sent over I2C [8].
Other: Measurement Time: 60ms	Table 13 of the VL53L0X Datasheet states that minimum timing budget is 20ms [8].
Protocol: I2C	Section 3 of the VL53L0X Datasheet states that I2C is the control interface required [8].
pwr_spply_lsr_tf_snsr_dcpwr	
Inominal: 35µA ± 10% per sensor (between measurements)	Table 9 of the VL53L0X Datasheet states that the inter-measurement current consumption is 16µA [8].
Ipeak: 30mA per sensor (during measurement)	Table 9 of the VL53L0X Datasheet states that the measurement current consumption is 19mA [8].
Vmin: 2.6V	Table 9 of the VL53L0X Datasheet states that the minimum operating voltage is 2.6V [8].
Vmax: 3.5V	Table 9 of the VL53L0X Datasheet states that the minimum operating voltage is 3.5V [8].

1.6. Bill of Materials

TABLE XI. BILL OF MATERIALS FOR THE LASER TOF SENSOR BLOCK

Reference Designator	Value	Manufacturer	Manufacturer Part Number	Suppliers	Quantity	Unit Price
C1	4.7uF	Taiyo Yuden	LMK107BJ475KA-T	Mouser	30	$0.18
C2	100nF	Vishay	VJ0603Y104KXXCW1BC	Mouser	30	$0.12
C3	7.5nF	Murata	GRM155R71E752JA01D	Mouser	30	$0.10
U1	VL53L0X	STMicroelectronics	VL53L0CXV0DH/1	Mouser	30	$5.30
U2	MAX6895 Delay IC	MAXIM	MAX6895AAZT+T	Mouser	30	$2.00

2. PCB

Block Owner: Sean Sylwester

2.1. Introduction

The purpose of this document is to describe the PCB Block of the Wearable Sensor for the Blind ECE Senior Capstone Project to other engineers with enough detail such that they would be able to reproduce this block using only this document as reference. Included in this document are an overview of the block, a schematic and layout diagram, and a complete bill of materials for the PCB.

2.2. Block Overview

The PCB block links together all the individual components of the system. It provides the necessary connectors to wire together the system processing blocks to the sensor blocks and haptic feedback block. It will also hold the 3.3V regulator, which will power every component in the system. Fig. 8 below shows all the connectors needed on the PCB.

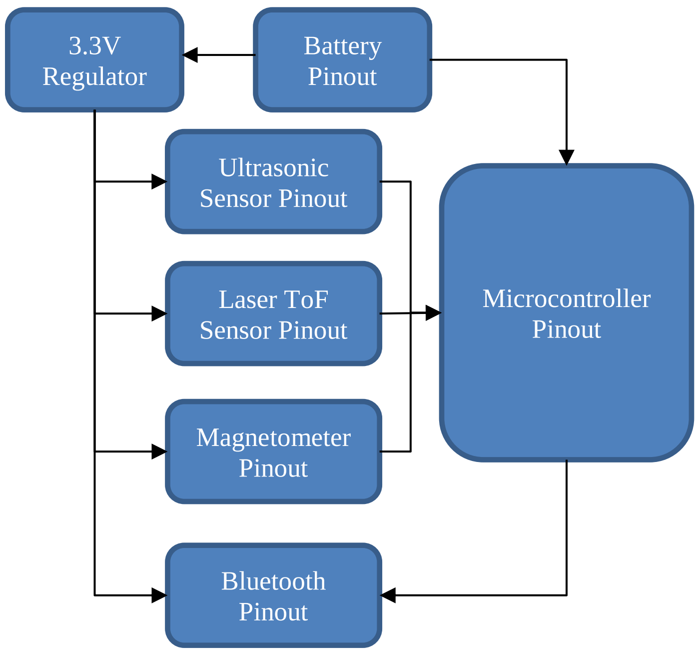


Fig. 8. Block Diagram of the PCB Block

2.3. Schematic

Fig. 9 below shows the schematic of the PCB corresponding to the block diagram shown in Fig. 8.

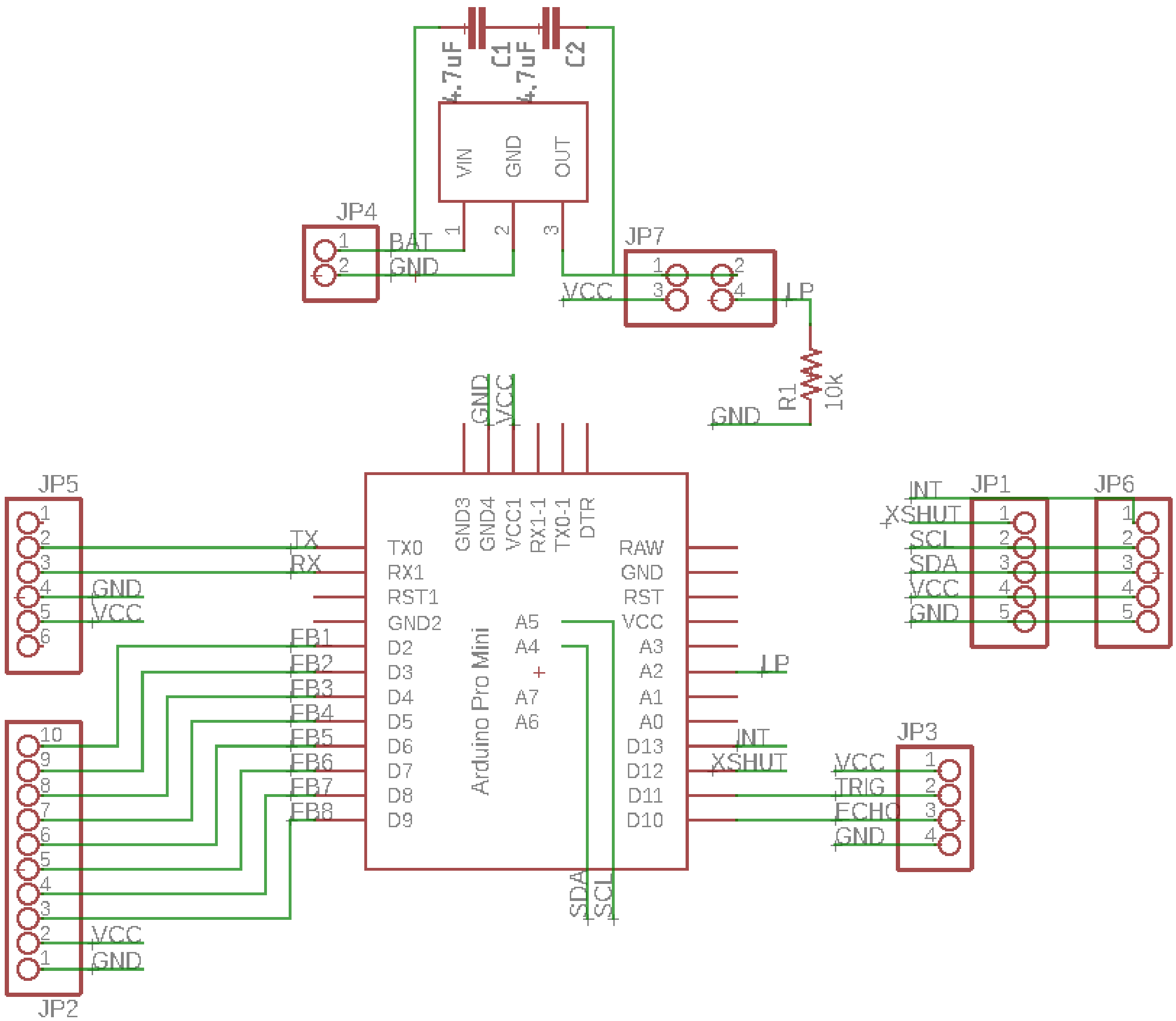


Fig. 9. PCB Schematic

schematic.pdf

2.4. PCB Artwork

Fig. 10 and Fig. 11 below show the physical layout of the PCB according to the schematic shown in Fig. 9.

[Link to Gerber Files]

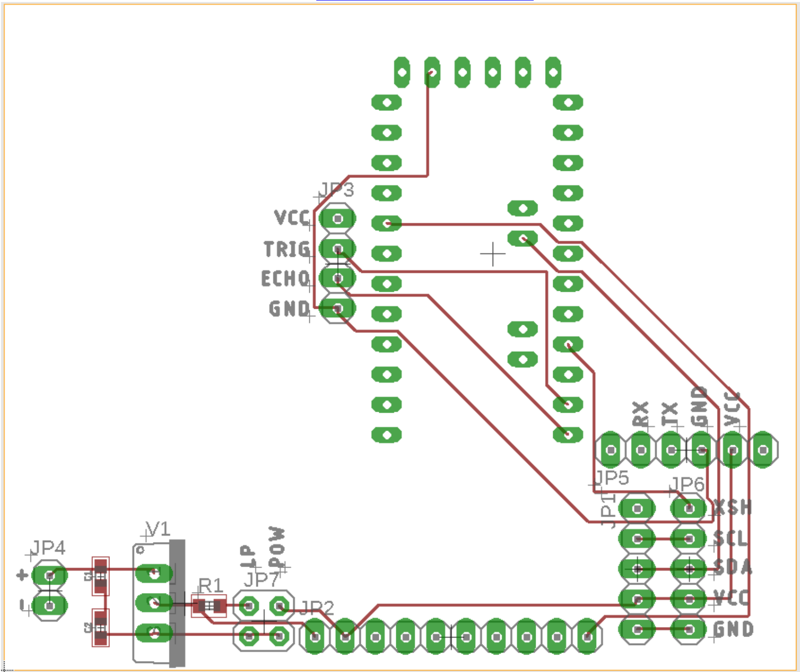


Fig. 10. PCB Layout Top

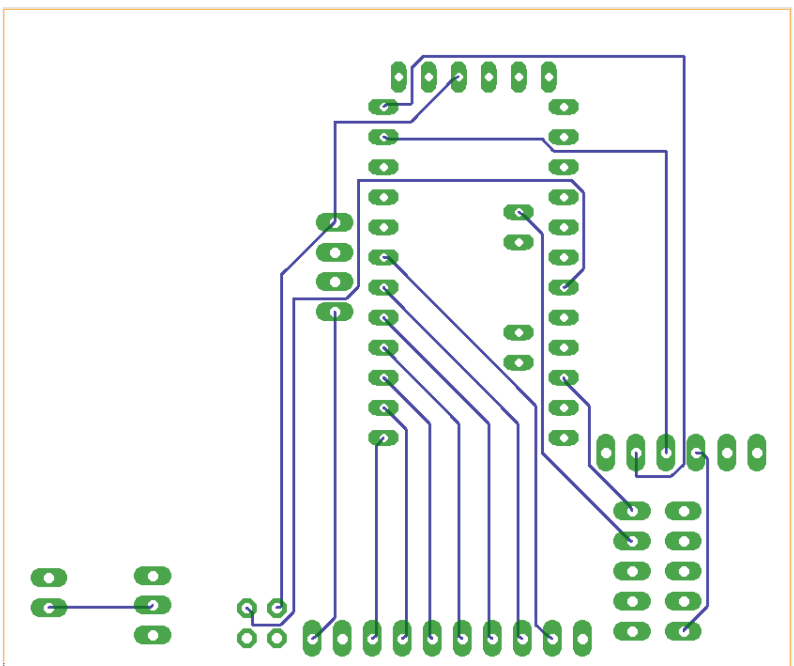


Fig. 11. PCB Layout Bottom

layout.pdf

2.5. Bill of Materials

TABLE XII. BILL OF MATERIALS FOR THE PCB BLOCK

Reference Designator	Description	Manufacturer	Manufacturer Part Number	Suppliers	Package	Quantity	Unit Price
C1 C2	Multilayer Ceramic Capacitors MLCC - SMD/SMT 4.7uF 10V X5R +/-10% 0603 Gen Purp	Taiyo Yuden	LMK107BJ475KA-T	Mouser	0603 in 1608 mm	2	$0.22
R1	RES SMD 10K OHM 1% 1/8W 0603	Vishay Beyschlag	MCT06030C1002FP500	Digi-Key	0603 in 1608 mm	1	$0.18
JP1, JP6	1x5 Female Header	Sullins Connector Solutions	PPTC051LFBN-RC	Digi-Key	1x5	2	$0.47
JP2	1x10 Female Header	Sullins Connector Solutions	PPTC101LFBN-RC	Digi-Key	1x10	1	$0.65
JP3	1x4 Female Header	Sullins Connector Solutions	PPTC041LFBN-RC	Digi-Key	1x4	1	$0.45
JP4	1x2 Female Header	Sullins Connector Solutions	PPTC021LFBN-RC	Digi-Key	1x2	1	$0.32
JP5	1x6 Female Header	Sullins Connector Solutions	PPTC061LFBN-RC	Digi-Key	1x6	1	$0.52
JP7	2x2 Female Header	Sullins Connector Solutions	PPTC022LFBN-RC	Digi-Key	2x2	1	$0.57
U1	Arduino Pro Mini	Arudino	Pro Mini 3.3V	Digi-Key	Pro Mini	1	$14.99
U2	LDO Voltage Regulators Low Power 3 Ampere	Texas Instruments	UCC283T-5	Mouser	TO220	1	$7.88

2.6. Approval Log

Table 13 below records the approval signatures for the PCB’s schematic, layout and bill of materials.

TABLE 13. PCB APPROVAL LOG

Schematic Approved	Signature: Sean Sylwester	Date: 3/12/18
PCB Approved	Signature: Sean Sylwester	Date: 3/13/18
Bill of Materials Complete	Signature: Sean Sylwester	Date: 3/13/18

3. Power Supply

Block Owner: Sean Sylwester

3.1. Introduction

The purpose of this document is to describe the Power Supply block of the Wearable Sensor for the Blind ECE Senior Capstone Project to other engineers with enough detail such that they would be able to reproduce and test this block using only this document as reference. The Power Supply block is implemented using 4 AA batteries in series connected to a 3.3V, 3A Regulator from Texas Instruments. Included in this document are an overview of the block, a schematic and wiring diagram, the properties of the interfaces with the other blocks in the system and testing procedures for each, and reasoning for why this design is the best solution for this block.

3.2. Block Overview

The Power Supply block will provide power to every component in the system. The output voltage must be stable since sensitive sensors are being used. Roughly an Amp will be drawn continuously by the system when in use, so a high current capability is required. Fig. 4 below shows the black box diagram of the block. All the properties for these interfaces can be found in Table IX.

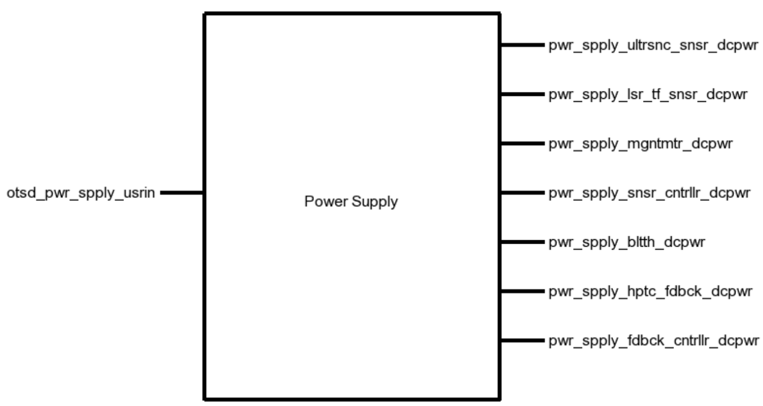


Fig. 12. Black Box Diagram of the Power Supply Block

TABLE XIV. POWER SUPPLY BLOCK INTERFACES AND PROPERTIES

Name	**Properties**
otsd_pwr_spply_usrin	1. Timing: Low-Power Mode Battery Life: 24h 2. Timing: Standard Mode Battery Life: 12h 3. Type: AA Batteries (4x) 4. Usability: Time to Change Batteries: <1min
pwr_spply_ultrsnc_snsr_dcpwr	1. Inominal: 5mA ± 10% 2. Ipeak: 30mA 3. Vmax: 5.5V 4. Vmin: 3V
pwr_spply_lsr_tf_snsr_dcpwr	1. Inominal: 35µA ± 10% per sensor (between measurements) 2. Ipeak: 30mA per sensor (during measurement) 3. Vmax: 3.5V 4. Vmin: 2.6V
pwr_spply_mgntmtr_dcpwr	1. Inominal: 5µA ± 10% (standby) 2. Ipeak: 0.5mA (measurement) 3. Vmax: 3.6V 4. Vmin: 1.95V
pwr_spply_snsr_cntrllr_dcpwr	1. Inominal: 35mA ± 10% (Bluetooth Paired, with laser, Bluetooth, ultrasonic) 2. Inominal: 60mA ± 10% (Bluetooth Not Paired, with laser, Bluetooth, ultrasonic) 3. Vmax: 12V 4. Vmin: 3.4V
pwr_spply_bltth_dcpwr	1. Inominal: 35mA ± 10% (During Pairing) 2. Inominal: 15mA ± 10% (after pairing) 3. Vmax: 6V 4. Vmin: 3.6V
pwr_spply_hptc_fdbck_dcpwr	1. Inominal: 50mA ± 20% at 100% duty cycle 2. Ipeak: 75mA (per motor) 3. Vmax: 3.3V 4. Vmin: 2.7V
pwr_spply_fdbck_cntrllr_dcpwr	1. Inominal: 9mA ± 10% 2. Ipeak: 20mA 3. Vmax: 5.5V 4. Vmin: 1.8V

3.3. VERIFICATION

This section details the testing procedures to verify every property listed in Table IX. All tests must be passed successfully before this block will be integrated into the rest of the system.

Battery Testing

This test will verify the battery properties in the *otsd_pwr_spply_usrin* interface, and all the power interfaces to this block. The currents being drawn from the system more than cover the nominal and peak current requirements in each power mode.

1. Connect 4 new AA batteries in series to the 3.3V voltage regulator.
2. Connect a DC load set to the nominal low-power mode current draw of 140mA to the output of the voltage regulator.
3. Wait 24h, then check the output voltage of the regulator and the battery output voltage.
4. Connect 4 new AA batteries in series to the 3.3V voltage regulator.
5. Connect a DC load set to the nominal standard mode current draw of 250mA to the output of the voltage regulator.

6. Wait 12h, then check the output voltage of the regulator and the battery output voltage.

PASS: The power supply continues to provide 3V-3.3V from the regulator, and 3.6V-6V from the raw battery output after 24h runtime in low-power mode, and after 12h in standard power mode.

Link to Video

User Interface Testing

This test will verify the user interface property in the *otsd_pwr_spply_usrin* interface

1. Blindfold a tester.
2. Give the blindfolded tester 4 new AA batteries, a screwdriver, and the enclosure with the batteries to be replaced.
3. Time how long it takes for the blindfolded tester replace the batteries.

PASS: 9/10 blindfolded testers are able to change the batteries in less than 1 minute.

Link to Video

3.4. Design

The schematic in Fig. 5 presents this block's wiring diagram, including the interfaces of this block to the rest of the system. Fig. 6 shows the schematic for the power supply portion of the PCB.

This block is implemented using 4 AA batteries connected to the input of a UCC283T-3 Low-Dropout Linear Voltage 3.3V/3A Regulator. Two 4.7µF smoothing capacitors are placed at the input and output of the regulator. Two switches are used: one for the main power on/off, and one to enable/disable the low-power mode. The low-power mode switch has a 10kΩ pull-down resistor connected to it to prevent a floating signal when low-power mode is disabled.

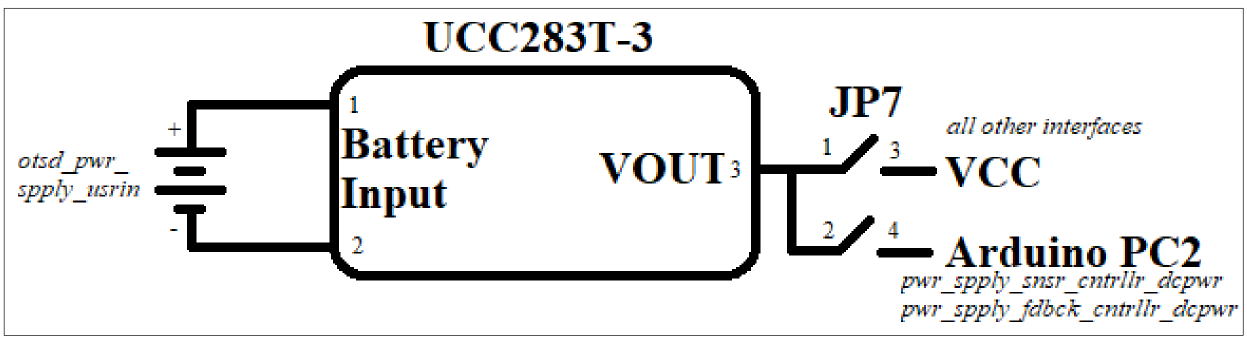


FIG. 13. WIRING DIAGRAM FOR THE POWER SUPPLY BLOCK

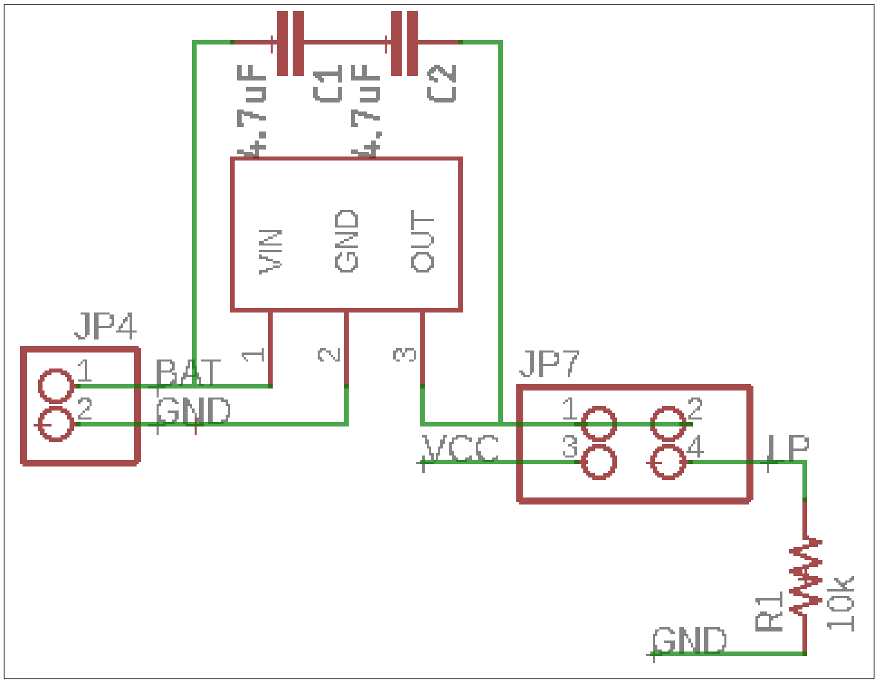


FIG. 14. POWER SUPPLY SCHEMATIC

3.5. DESIGN VALIDATION

For this block, the UCC283T-3 regulator was used because it more than covers the 3.3V/1A power requirement that this project required of the Power Supply block and can be powered continuously with 4 AA batteries as their voltage drops over usage time. Table X on the next page validates all the system properties using the UCC283T-3 Datasheet.

TABLE XV. INTERFACE PROPERTY VALIDATION FOR THE LASER TOF SENSOR BLOCK

Property	**Validation**
otsd_pwr_spply_usrin	
Timing: Low-Power Mode Battery Life: 24h	(AA Battery capacity: 3500mAh [2]) / (low-power mode current: 140mA) > 24h
Timing: Standard Mode Battery Life: 12h	(AA Battery capacity: 3500mAh [2]) / (standard mode current: 250mA) > 12h
Type: AA Batteries (4x)	4xAA Battery voltage: 4V-6V. This lies within the 3.45-9V input voltage range for the regulator, shown in page 3 of the UCC283T-3 datasheet [1].
Usability: Time to Change Batteries: <1min	The user needs to remove two screws, shake out the old batteries, put it in the new batteries, and replace the screws. In our preliminary testing, this process took less than 30s, so <1min is reasonable.
pwr_spply_ultrsnc_snsr_dcpwr	
Inominal: 5mA ± 10%	The regulator will output up to 3A given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Ipeak: 30mA	The regulator will output up to 3A given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Vmax: 5.5V	The regulator will output 3.3V given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Vmin: 3V	The regulator will output 3.3V given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
pwr_spply_lsr_tf_snsr_dcpwr	
Inominal: 35µA ± 10% per sensor (between measurements)	The regulator will output up to 3A given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Ipeak: 30mA per sensor (during measurement)	The regulator will output up to 3A given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Vmax: 3.5V	The regulator will output 3.3V given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Vmin: 2.6V	The regulator will output 3.3V given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
pwr_spply_mgntmtr_dcpwr	
Inominal: 5µA ± 10% (standby)	The regulator will output up to 3A given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Ipeak: 0.5mA (measurement)	The regulator will output up to 3A given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Vmax: 3.6V	The regulator will output 3.3V given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Vmin: 1.95V	The regulator will output 3.3V given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
pwr_spply_snsr_cntrllr_dcpwr	
Inominal: 35mA ± 10% (Bluetooth Paired, with laser, Bluetooth, ultrasonic)	The regulator will output up to 3A given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Inominal: 60mA ± 10% (Bluetooth Not Paired, with laser, Bluetooth, ultrasonic)	The regulator will output up to 3A given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Vmax: 12V	4xAA Batteries when fully charged will have a maximum voltage of 6V [2].
Vmin: 3.4V	4xAA Batteries when fully depleted will have a minimum voltage of 4V [2].
pwr_spply_bltth_dcpwr	
Inominal: 35mA ± 10% (During Pairing)	The regulator will output up to 3A given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Inominal: 15mA ± 10% (after pairing)	The regulator will output up to 3A given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Vmax: 6V	4xAA Batteries when fully charged will have a maximum voltage of 6V [2].
Vmin: 3.6V	4xAA Batteries when fully depleted will have a minimum voltage of 4V [2].
pwr_spply_hptc_fdbck_dcpwr	
Inominal: 50mA ± 20% at 100% duty cycle	The regulator will output up to 3A given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Ipeak: 75mA (per motor)	The regulator will output up to 3A given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Vmax: 3.3V	The regulator will output 3.3V given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Vmin: 2.7V	The regulator will output 3.3V given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
pwr_spply_fdbck_cntrllr_dcpwr	
Inominal: 9mA ± 10%	The regulator will output up to 3A given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Ipeak: 20mA	The regulator will output up to 3A given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Vmax: 5.5V	The regulator will output 3.3V given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].
Vmin: 1.8V	The regulator will output 3.3V given our 4V-6V input range according to page 3 of the UCC283T-3 datasheet [1].

3.6. Bill of Materials

TABLE XVI. BILL OF MATERIALS FOR THE POWER SUPPLY BLOCK

Reference Designator	**Description**	**Manufacturer**	**Manufacturer Part Number**	**Suppliers**	**Package**	**Quantity**	**Unit Price**
C1 C2	Multilayer Ceramic Capacitors MLCC - SMD/SMT 4.7uF 10V X5R +/-10% 0603 Gen Purp	Taiyo Yuden	LMK107BJ475KA-T	Mouser	0603 in 1608 mm	2	$0.22
R1	RES SMD 10K OHM 1% 1/8W 0603	Vishay Beyschlag	MCT06030C1002FP 500	Digi-Key	0603 in 1608 mm	1	$0.18
S1 S2	125VAC 6A Amps On/On/ 2 Position Terminal SPDT Latching Mini Toggle Switch	Gadgeter	B01JU6KBH6	Amazon	SPDT	2	$0.53
U2	LDO Voltage Regulators Low Power 3 Ampere	Texas Instruments	UCC283T-5	Mouser	TO220	1	$7.88

4. Ultrasonic Sensor

Block Owner: Sean Sylwester

4.1. Introduction

The purpose of this document is to describe the Ultrasonic Sensor block of the Wearable Sensor for the Blind ECE Senior Capstone Project to other engineers with enough detail such that they would be able to reproduce and test this block using only this document as reference. The Ultrasonic Sensor block is implemented using a JSN-SR04T waterproof ultrasonic module. Included in this document are an overview of the block, a schematic and wiring diagram, the properties of the interfaces with the other blocks in the system and testing procedures for each, and reasoning for why this design is the best solution for this block.

4.2. Block Overview

The Ultrasonic Sensor block will provide the distance to the nearest object in the direction that the user's head is pointing to the system, which will use that information to set the system haptic feedback module that will convey that distance information to the user. The measurements taken need to be fast and accurate, so that the user can quickly and confidently navigate around obstacles in their environment. The sensor will determine the Fig. 15 below shows the black box diagram of the system. *otsd_ultrsnc_snsr_envin* represents the environmental input to this block, namely the distance to the nearest object. *ultrsnc_snsr_snsr_cntrllr_data* represents the sensor data link to the microcontroller. And finally, *pwr_spply_ultrsnc_snsr_dcpwr* represents the connection to the power supply, which is regulated to 5V, and should draw about 5mA during measurements.

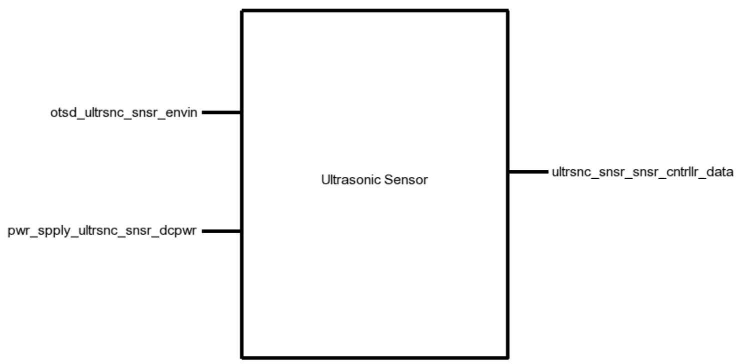


Fig. 15. Black Box Diagram of Ultrasonic Sensor Block

Table XVII. Display Block Interfaces And Properties

Interface	Properties
otsd_ultrsnc_snsr_envin	1. Other: Accuracy: 10cm 2. Other: Angle: 37.5° 3. Other: Range: 20-500cm
ultrsnc_snsr_snsr_cntrllr_data	1. Messages: Object Distance 2. Other: Measurement Time: <38ms 3. Other: Pulse Voltage: 3.3V 4. Protocol: TTL
pwr_spply_ultrsnc_snsr_dcpwr	1. Inominal: 5mA ± 10% 2. Ipeak: 30mA 3. Vmax: 5.5V 4. Vmin: 3V

4.3. Verification

This section details the testing procedures to verify every property listed in Table XVII. All tests must be passed successfully before this block will be integrated into the rest of the system.

Measurement Testing

This test will verify the measurement requirements for this block on the *otsd_ultrsnc_snsr_envin* interface.

7. Connect the JSN-SR04T ultrasonic sensor to a DC power supply set to 3.3V via the *pwr_spply_ultrsnc_snsr_dcpwr* interface.
8. Connect the JSN-SR04T Echo and Trigger pins to a microcontroller.
9. Load a test program on the microcontroller that continuously triggers a measurement with a 3.3V pulse, reads the sensor data with a TTL protocol, and prints the result to a computer.
10. Place an object 20cm directly in front of the sensor and note if the sensor detects the object.
11. Place an object 500cm directly in front of the sensor and note if the sensor detects the object.
12. Place an object 100cm away, and 75cm off-center (37.5°) and note if the sensor detects the object.
13. Place an object 100cm directly in front of the sensor and note if the sensor measurement is within 10cm of 100cm.

PASS: If the measurement completes in less than 38ms, objects in steps 4, 5, and 6 were detected, and the object in step 7 was detected within 10cm.

Link to Video

Power Testing

This test will verify the power requirements for this block on the *pwr_spply_ultrsnc_snsr_dcpwr* interface.

1. Connect the JSN-SR04T ultrasonic sensor to a DC power supply set to 5.5V via the *pwr_spply_ultrsnc_snsr_dcpwr* interface.
2. Connect the JSN-SR04T Echo and Trigger pins to a microcontroller.
3. Trigger a measurement by pulling the Trigger pin to 3.3V for 10µs.
4. Note the current displayed on the DC power supply during a measurement, and during idle.
5. Repeat steps 1-4 with the DC power supply set to 3V.

PASS: If the current never exceeds 30mA, a pulse on the Echo pin is returned with the 3V and 5.5V supplies.

Link to Video

4.4. Design

The schematic in Fig. 16 presents this wiring diagram for the JSN-SR04T sensor used in this block, including the interfaces of this block to the rest of the system. The timing diagram in Fig. 17 shows the input and output data from the block. Table XVIII provides validation for each of the properties listed in Table XVII using the JSN-SR04T datasheet.

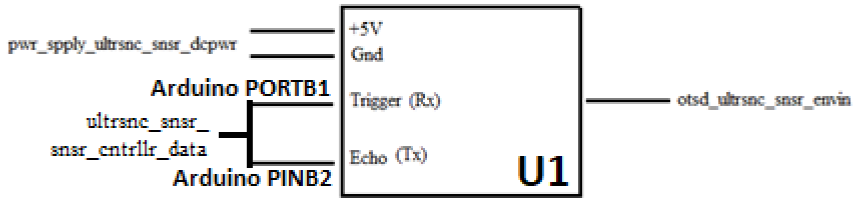


Fig. 16. Wiring Diagram for the Ultrasonic Module Block

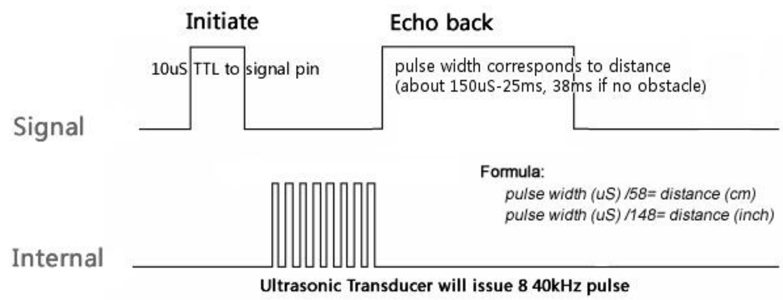


FIG. 17. MEASUREMENT TIMING DIAGRAM FOR THE ULTRASONIC MODULE BLOCK

4.5. DESIGN VALIDATION

For this block, the JSN-SR04T ultrasonic sensor was used. This sensor was chosen because it fits the water resistance, measurement distance, accuracy, and angle of detection metrics required of the Ultrasonic Sensor Block. Table XVIII below validates each property listed in Table XVII. Note that the sensor used in the JSN-SR04 is identical to the one used in the HC-SR04. Neither of these have a datasheet, but the HC-SR04 has a reference document with enough information to validate these properties.

TABLE XVIII. INTERFACE PROPERTY VALIDATION FOR THE ULTRASONIC SENSOR BLOCK

Property	**Validation**
pwr_spply_ultrsnc_snsr_dcpwr	
Min Voltage: 3 V	The HC-SR04 datasheet says that the minimum supply voltage is 3 V [13]
Max Voltage: 5.5 V	The HC-SR04 datasheet says that the maximum supply voltage is 5.5 V [13]
Nominal Current: 5mA	The HC-SR04 datasheet says that the nominal working current nominal is <8mA [13]
Peak Current: 30mA	The HC-SR04 datasheet says that the quiescent current is less than 2 mA [13]
otsd_ultrsnc_snsr_envin	
Range: 20 cm - 500cm	The HC-SR04 datasheet says that the distance range is 20-500cm [13]
Range accuracy: 10cm	The HC-SR04 datasheet says that the distance measurement is accurate to ±0.3cm [13]
Angle: 15°	The HC-SR04 datasheet says that angle of the ultrasonic is at least 15° [13]
ultrsnc_snsr_snsr_cntrllr_data	
Pulse Protocol: TTL	The HC-SR04 datasheet says that a Time-to-Life (TTL) signal is used to encode the distance measurement [13]
Other: Measurement Time: <38ms	The HC-SR04 datasheet says that the width of the TTL signal is 150µs to 25ms, and 38ms if no object is detected [13]
Pulse Voltage: 5 V	The HC-SR04 datasheet says that the Echo and Trigger pins function with a 3.3V pulse signal [13]
Messages: Sensor Data	The HC-SR04 datasheet says that a Time-to-Life (TTL) signal is used to encode the distance measurement [13]

4.6. BILL OF MATERIALS

TABLE XIX. BILL OF MATERIALS FOR THE ULTRASONIC SENSOR BLOCK

Reference Designator	**Description**	**Manufacturer**	**Manufacturer Part Number**	**Suppliers**	**Quantity**	**Price**
U1	Ultrasonic Sensor	JSN	JSN-SR04T	Digi-Key	1	$15.39

6. SENSOR CONTROLLER BLOCK

Block Owner: Samuel Lee

6.1. *Introduction*

The purpose of this section is to describe the Sensor Controller block of the ECE44x Wearable Sensor for the Blind project to 4th year ECE students. The Sensor Controller block is implemented using Arduino Pro Mini 3.3V module. This document provides an overview of the overall block function including interface properties and a schematic, verification for the design in the form of a step-by-step testing process, and support for the validity of the design in the form of outside research and numerical justification addressing individual properties.

6.2. *Design Details*

The wiring diagram (Fig 18) presents the block design, including the interfaces of the block. These interfaces are further elaborated through validation information given in TABLE XX, which demonstrates externally-supported validly for each of the block's properties.

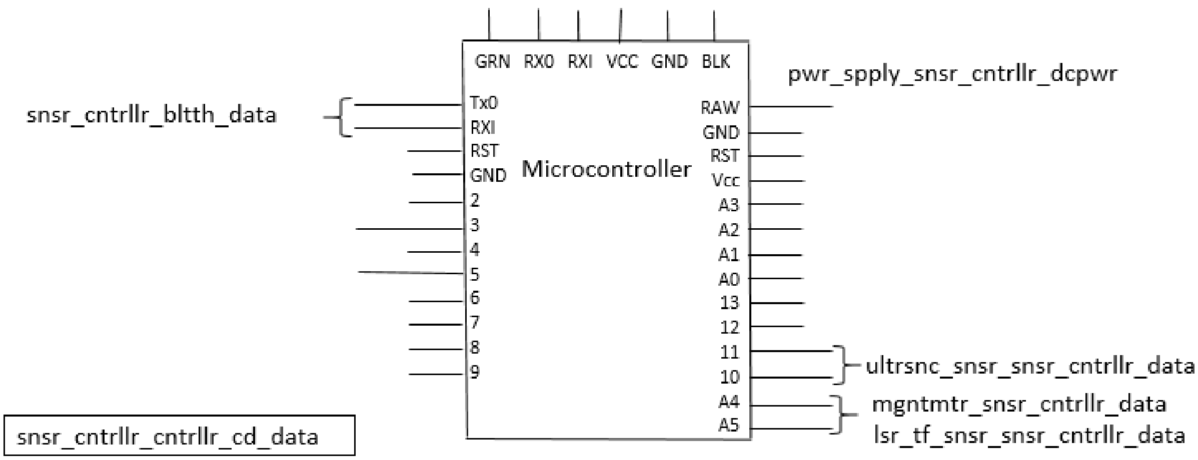


FIG 18. WIRING DIAGRAM OF SENSOR CONTROLLER BLOCK

6.3. *Block Overview*

The block is powered from **pwr_spply_snsr_cntrllr_dcpwr** interface which takes in voltages between 3.4V and 12V through the RAW pin. The voltage applied through this pin is regulated to 3.3V before it gets to the processor. Fig 18 shows the black box diagram. The Sensor Controller receives data from the system Ultrasonic Sensor, Laser ToF Sensor, and Magnetometer Sensor. Data are sent and received via a serial port protocol over the **snsr_cntrllr_bltth_data**, which will be received by the system Feedback band. The function of the Sensor Controller block is to collect and processing data, reorganize packets and running the I2C interfaces of the system.

A full listing of interface properties can be found in Table XX.

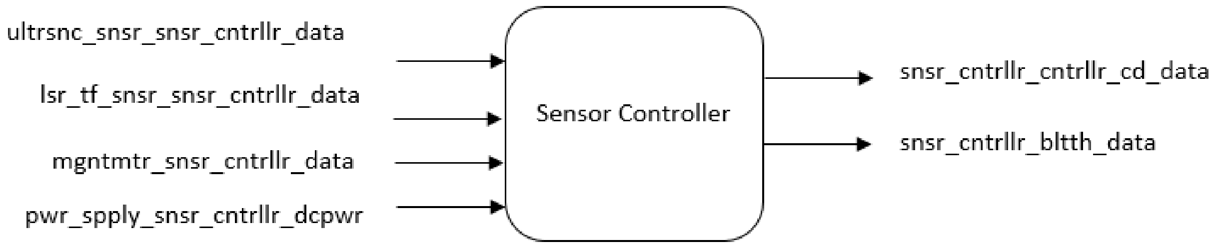


FIG 19. BLACK BOX DIAGRAM OF SENSOR CONTROLLER BLOCK

`

TABLE XX. SENSOR CONTROLLER BLOCK INTERFACES AND PROPERTIES

Interface	Properties
ultrsnc_snsr_snsr_cntrllr_data	1.Messages: Object Distance 2.Protocol: TTL 3.Other: Measurement Time < 38ms 4.Other: Pulse Voltage 3.3V
lsr_tf_snsr_snsr_cntrllr_data	1.Messages: Address Programming, Measurement Trigger, Measurement Read 2.Datarate: 100kHz 3.Other: Measurement Time <60ms 4.Protocol I2C
mgntmtr_snsr_cntrllr_data	1.Messages: Sensor Data 2.Datarate: 100kHz 3.Other: Measurement Frequency 20Hz 4.Protocol I2C
pwr_spply_snsr_cntrllr_dcpwr	1.Inominal: 35mA ± 10% (Bluetooth Paired, with laser, Bluetooth, ultrasonic) 2.Inominal: 60mA ± 10% (Bluetooth Not Paired, with laser, Bluetooth, ultrasonic) 3.Vmax: 12V 4.Vmin: 3.4V
snsr_cntrllr_cntrllr_cd_data	1.Protocol: Avrdude 2.Other: Arduino 3.Other: Code Size less than 25KB
snsr_cntrllr_bltth_data	1.Protocol: Serial Port Protocol 2.Protocol: 1Stop Bit, No Parity Bit 3.Other: Name: ece34_M 4.Other: Password: group34 5.Other: Range >1m

6.4. *Block Verification*

Based on the interfaces for this block, a verification (testing) process needs to be indicated. This will allow the final constructed design to be tested verifying that all the interface properties have been met and that the block is ready for integration into the system.

Link to All Sensor Controller Test Videos

1. ***Setup Test (snsr_cntrllr_cntrllr_cd_data)***

    a. Connect the microcontroller to a 3.3V power supply via VCC pin or 3.4V-12V via RAW pin, and the **snsr_cntrllr_cntrllr_cd_data** interface to a PC using an USB-to-TTL serial converter.

    b. Load the test program to the microcontroller using the “Arduino.exe” via the usbasp programmer.

c. To enable “Arduino.exe” to output compilation steps through the log, go to Preferences -> Settings -> At “Show verbose output during”, check “compilation” and “upload”.

d. Check that program has been compiled and uploaded to the microcontroller through “Arduino.exe” log window.

e. The test program simulates the real code by having arrays of data as input and output them onto the serial monitor.

PASS:

1. Verify that the program is uploaded using avrdude through the log of “Arduino.exe”.
2. Verify that the program written in Arduino.
3. Verify that the code size is less than 25KB for the **snsr_cntrllr_bltth_data.**
4. Verify that data are printed into the serial monitor.

2. Power Test (pwr_spply_snsr_cntrllr_dcpwr)

a. Connect the microcontroller to a laser sensor, an ultrasonic sensor, and a Bluetooth module.

b. Connect the microcontroller to a 3.4V power supply via RAW pin.

c. Load test program to the microcontroller using “Arduino.exe” via usbasp programmer.

d. Measure the current used through the power supply before the Bluetooth is paired.

e. Measure the current used through the power supply after the Bluetooth is paired.

f. Repeat step b-d for a 12V power supply.

PASS:

1. Verify that the microcontroller successfully powers up (Blinking LEDs on the board) with 3.4V and 12V supplied to the RAW pin.
2. Verify that the current used before Bluetooth pairing is around 60mA.
3. Verify that the current used after Bluetooth pairing is around 35mA.

3. Bluetooth Test (snsr_cntrllr_bltth_data)

a. Setup Bluetooth based on *Bluetooth to Bluetooth Testing* at *Bluetooth Block section IV.C*.

b. Load test program to the microcontroller using “Arduino.exe” via usbasp programmer.

c. Test program will send an array of numbers through Bluetooth to be received by the feedback controller’s Bluetooth module.

PASS:

1. Verify that the feedback controller receives the numbers and printed onto the Serial Monitor.

4. Ultrasonic Sensor Test (ultrsnc_snsr_snsr_cntrllr_data)

a. Setup the Ultrasonic Sensor to the microcontroller by connecting the trigger pin to pin 10, echo pin to pin 11, VCC and GND of the sensor to the VCC and GND of the microcontroller,

b. Load test program to the microcontroller using “Arduino.exe” via usbasp programmer.

c. Test program will trigger and read the measurement from the ultrasonic sensor and send it to the Serial Monitor.

PASS:

1. Verify that the sensor controller receives the ultrasonic sensor data and print it onto the Serial Monitor.

5. I2C Test (lsr_tf_snsr_snsr_cntrllr_data, mgntmtr_snsr_cntrllr_data)

a. We'll be using the Laser ToF sensor to proof the functionality of the I2C of the microcontroller.

b. Setup the laser sensor to the microcontroller by connecting the SDA pin of the laser sensor to A4 pin of the microcontroller, SCL pin to A5 pin, VCC and GND of the sensor to the VCC and GND of the microcontroller.

c. Load test program to the microcontroller using "Arduino.exe" via usbasp programmer.

d. Test program will trigger and read the measurement from the laser sensor and send it to the Serial Monitor.

PASS:

1. Verify that the sensor controller receives the laser sensor data and print it onto the Serial Monitor.

If the block passes all the listed tests, all interface properties have been verified and the block is ready for inclusion into the system.

6.5. *Design Validation*

For this block, an off the shelf solution was chosen. The ATmega328P on the Arduino Pro Mini microcontroller fits the needs of the block interfaces without needing to be modified. Table XXI includes the interface property validation for this block. All the interface properties have been addressed and the design meets or exceeds the properties.

TABLE XXI. INTERFACE PROPERTIES VALIDATION FOR THE SENSOR CONTROLLER BLOCK

Interface	Properties
ultrsnc_snsr_snsr_cntrllr_data	
Messages: Object Distance	The JSN-SR04T Ultrasonic Waterproof Range Finder can provide 20cm-600cm distance sensing function. (Overview, JSN-SR04T-2.0 Datasheet [1])
Protocol: TTL	JSN-SR04T datasheet states that Time-to-Life(TTL) signal is used to encode distance measurement. (Specification, JSN-SR04T-2.0 Datasheet [1])
Other: Measurement Time < 38ms	JSN-SR04T datasheet states that the width of the TTL signal is 150µs to 25ms, and 38ms if no object is detected. (Specification, JSN-SR04T-2.0 Datasheet [1])
Other: Pulse Voltage 3.3V	JSN-SR04T datasheet states that the pulse voltage ranges from 3-5.5V. (Specification, JSN-SR04T-2.0 Datasheet [1])
lsr_tf_snsr_snsr_cntrllr_data	
Messages: Address Programming, Measurement Trigger, Measurement Read	VLX53L0X datasheet states that Address Programming, Measurement Trigger, and Measurement Read commands can be sent over I2C. (Section 2, VLX53L0X Datasheet [2])
Datarate: 100kHz	VLX53L0X has a maximum operating frequency of 400kHz. (Table 3, VLX53L0X Datasheet [2])
Other: Measurement Time <60ms	VLX53L0X has a smallest range timing budget of 20ms. (Table 13, VLX53L0X Datasheet [2])
Protocol I2C	VLX53L0X uses I2C as its control interface. (Section 3, VLX53L0X Datasheet [2])
mgntmtr_snsr_cntrllr_data	
Messages: 3-axis sensor data	MAG3110 datasheet states it provides digital 3-axis data. Register descriptions states 16bits for each X, Y, Z measurement. (Section 5, MAG3110 Datasheet [3])
Datarate: 100kHz	MAG3110 has a maximum operating frequency of 400kHz for Fast mode, 100kHz for normal mode. (Table 7, MAG3110 Datasheet [3])
Other: Measurement Frequency 20Hz	MAG3110 triggered measurements uses 10Hz. (Section 4.2.7, MAG3110 Datasheet [3])
Protocol: I2C	MAG3110 uses I2C as its control interface. (Section 2.4, MAG3110 Datasheet [3])
pwr_spply_snsr_cntrllr_dcpwr	
Inominal: 35mA ± 10% (Bluetooth paired, with laser, Bluetooth, Ultrasonic)	Arduino Pro Mini 3.3V/8MHz (DEV-11114) has a maximum current of 150mA. (Power, Arduino Pro Mini Datasheet [4])
Inominal: 60mA ± 10% (Bluetooth not paired, with laser, Bluetooth, Ultrasonic)	Arduino Pro Mini 3.3V/8MHz (DEV-11114) has a maximum current of 150mA. (Power, Arduino Pro Mini Datasheet [4])
Vmax: 12V	Arduino Pro Mini 3.3V has a on board voltage regulator. Voltage supplied through RAW pin will be regulated to 3.3V. (Page5-6, Arduino Pro Mini Tutorial [5])
Vmin: 3.4V	Arduino Pro Mini 3.3V has a on board voltage regulator. Voltage supplied through RAW pin will be regulated to 3.3V. (Page5-6, Arduino Pro Mini Tutorial [5])
snsr_cntrllr_cntrllr_cd_data	
Protocol: Avrdude	Arduino IDE uses Avrdude to upload compiled code to the board. (Tools, Arduino IDE 1.5 Hardware Specification [6])
Other: Language: Arduino	Controller code is programmed using Arduino to interact with Arduino Pro Mini. (Arduino Pro Mini 3.3V Tutorial [5])
Other: Code Size less than 25KB	ATmega328P has 32K Bytes of In-System Self-Programmable Flash program memory which is sufficient for system code. (Page 1, ATmega328 Datasheet [7])
snsr_cntrllr_bltth_data	
Protocol: Serial Port Protocol	Controller code is programmed by using Arduino IDE to interact with the Bluetooth module. (Arduino Pro Mini 3.3V Tutorial [5])
Protocol: 1Stop Bit, No Parity Bit	The extra stop bit is useful to allow the Bluetooth module to know that the connection is still valid.
Other: Name: ece34_M	Name is set for project ownership.
Other: Password: group34	Password is set for security reasons.
Other: Range >1m	The Bluetooth module has a distance range of approximately 10 meters, it is sufficient for a head-to-waist distance of approximately 1 meter. (General Specification, HC-05 Bluetooth Module Breakout Board [8])

6.6. *Bills of Materials*

Table 3 lists the bills of materials used for the Sensor Controller block.

TABLE XXII. BILLS OF MATERIALS FOR THE SENSOR CONTROLLER BLOCK

Item	Where to buy	Manufacturer	Manufacturer Part Number	Price
Arduino Pro Mini 3.3V	Digikey	SparkFun Electronics	DEV-11114	$9.95
SH_BT_Board HC-05 Bluetooth Module	Amazon	DSD Tech	-	$9.99
VL53L0X Laser ToF Sensor	Digikey	Adafruit Industries LLC	3317	$14.95
Triple Axis Magnetometer – MAG3110	Digikey	SparkFun Electronics	SEN-12670	$14.95
Ultrasonic Sensor	Digikey	DFRobot	SEN0208	$16.00

7. Feedback Controller Block

Block Owner: Samuel Lee

7.1. *Introduction*

The purpose of this section is to describe the Feedback Controller block of the ECE44x Wearable Sensor for the Blind project to 4th year ECE student. The Feedback Controller block is implemented using Arduino Pro Mini 3.3V module. This document provides an overview of the overall block function including interface properties and a schematic, verification for the design in the form of a step-by-step testing process, and support for the validity of the design in the form of outside research and numerical justification addressing individual properties.

7.2. *Design Details*

The wiring diagram (Fig 18) presents the block design, including the interfaces of the block. These interfaces are further elaborated through validation information given in Table XX, which demonstrates externally-supported validly for each of the block's properties.

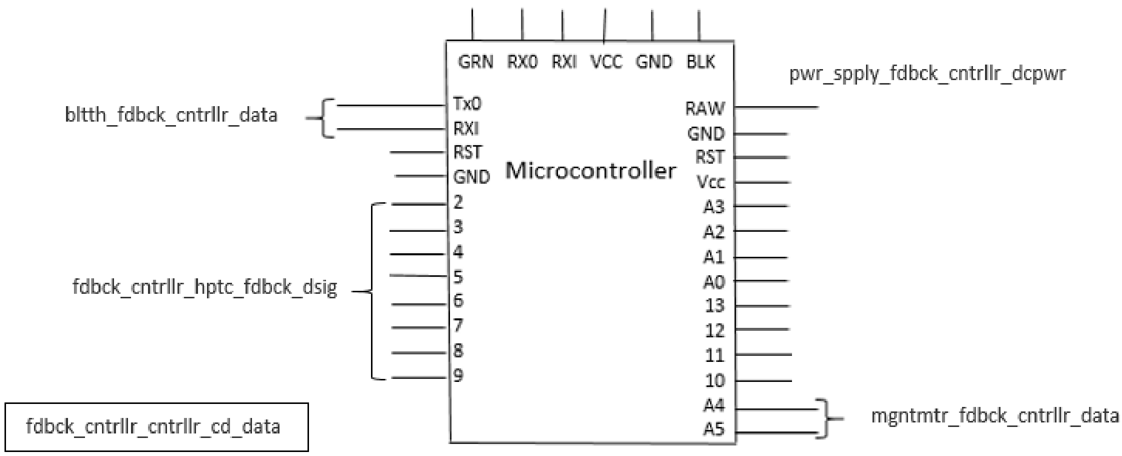


Fig 20. Wiring Diagram of Feedback Controller Block

7.3. *Block Overview*

The block is powered from **pwr_spply_fdbck_cntrllr_dcpwr** interface which takes in voltages between 3.4V and 12V through the RAW pin. The voltage applied through this pin is regulated to 3.3V before it gets to the processor. Fig 18 shows the black box diagram. The Feedback Controller receives data from the system Magnetometer Sensor. Data are sent and received via a serial port protocol over the **fdbck_cntrllr_bltth_data**, which will be received by the system Controller band. The function of the Feedback Controller block is to collect data from the Sensor Controller to relay data to the haptic feedback.

A full listing of interface properties can be found in Table XX.

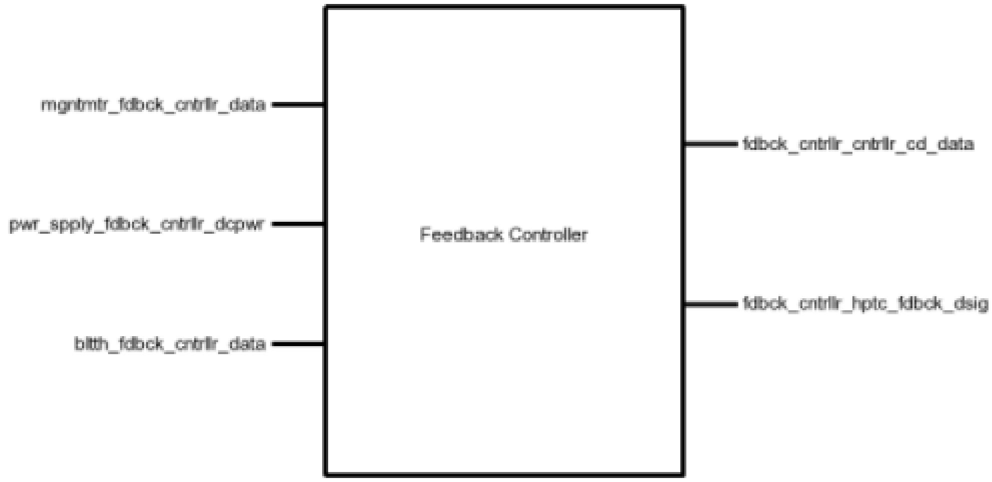


FIG 21. BLACK BOX DIAGRAM OF FEEDBACK CONTROLLER BLOCK

`

TABLE XXIII. FEEDBACK CONTROLLER BLOCK INTERFACES AND PROPERTIES

Interface	**Properties**
mgntmtr_fdbck_cntrllr_data	1.Messages: Sensor Data 2.Datarate: 100kHz 3.Other: Measurement Frequency 20Hz 4.Protocol I2C
pwr_spply_fdbck_cntrllr_dcpwr	1.Inominal: 9mA ± 10% 2.Vmax: 12V 3.Vmin: 3.4V
fdbck_cntrllr_cntrllr_cd_data	1.Protocol: Avrdude 2.Other: Language: Arduino 3.Other: Code Size less than 10KB
fdbck_cntrllr_bltth_data	1.Datarate: 38400 Baud Rate 2.Protocol: 1Stop Bit, No Parity Bit 3.Protocol: Serial Port Protocol 4.Other: Name: ece34_S 5.Other: Password: group34 6.Other: Range >1m
fdbck_cntrllr_hptc_fdbck_dsig	1.Fall Time: <100ms 2.Logic-Level: 3.3V 3.Rise Time: <100ms

7.4. *Block Verification*

Based on the interfaces for this block, a verification (testing) process needs to be indicated. This will allow the final constructed design to be tested verifying that all the interface properties have been met and that the block is ready for integration into the system.

Link to All Feedback Controller Test Videos

1. ***Setup Test (fdbck_cntrllr_cntrllr_cd_data)***

   a. Connect the microcontroller to a 3.3V power supply via VCC pin or 3.4V-12V via RAW pin, and the **fdbck_cntrllr_cntrllr_cd_data** interface to a PC using an USB-to-TTL serial converter.

b. Load the test program to the microcontroller using the “Arduino.exe” via the usbasp programmer.

c. To enable “Arduino.exe” to output compilation steps through the log, go to Preferences -> Settings -> At “Show verbose output during”, check “compilation” and “upload”.

d. Check that program has been compiled and uploaded to the microcontroller through “Arduino.exe” log window.

e. The test program simulates the real code by having arrays of data as input and output them onto the serial monitor.

PASS:

5. Verify that the program is uploaded using avrdude through the log of “Arduino.exe”.
6. Verify that the program written in Arduino.
7. Verify that the code size is less than 10KB for the **fdbck_cntrllr_bltth_data.**
8. Verify that data are printed into the serial monitor.

2. Power Test (pwr_spply_fdbck_cntrllr_dcpwr)

a. Connect the microcontroller to a haptic feedback motor and a Bluetooth module.

b. Connect the microcontroller to a 3.4V power supply via RAW pin.

c. Load test program to the microcontroller using “Arduino.exe” via usbasp programmer.

d. Measure the current used through the power supply before the Bluetooth is paired.

e. Measure the current used through the power supply after the Bluetooth is paired.

f. Repeat step b-d for a 12V power supply.

PASS:

4. Verify that the microcontroller successfully powers up (Blinking LEDs on the board) with 3.4V and 12V supplied to the RAW pin.
5. Verify that the current used before Bluetooth pairing is around 9mA.

3. Bluetooth Test (fdbck_cntrllr_bltth_data)

a. Setup Bluetooth based on *Bluetooth to Bluetooth Testing* at *Bluetooth Block section IV.C*.

b. Load test program to the microcontroller using “Arduino.exe” via usbasp programmer.

c. Test program will send an array of numbers through Bluetooth to be received by the feedback controller’s Bluetooth module.

PASS:

2. Verify that the feedback controller receives the numbers and printed onto the Serial Monitor.

4. I2C Test (mgntmtr_fdbck_cntrllr_data)

a. We’ll be using the Laser ToF sensor to proof the functionality of the I2C of the microcontroller.

b. Setup the laser sensor to the microcontroller by connecting the SDA pin of the laser sensor to A4 pin of the microcontroller, SCL pin to A5 pin, VCC and GND of the sensor to the VCC and GND of the microcontroller.

c. Load test program to the microcontroller using “Arduino.exe” via usbasp programmer.

d. Test program will trigger and read the measurement from the laser sensor and send it to the Serial Monitor.

PASS:

2. Verify that the feedback controller receives the laser sensor data and print it onto the Serial Monitor.

If the block passes all the listed tests, all interface properties have been verified and the block is ready for inclusion into the system.

7.5. *Design Validation*

For this block, an off the shelf solution was chosen. The ATmega328P on the Arduino Pro Mini microcontroller fits the needs of the block interfaces without needing to be modified. Table XXI includes the interface property validation for this block. All the interface properties have been addressed and the design meets or exceeds the properties.

TABLE XXIV. INTERFACE PROPERTIES VALIDATION FOR THE FEEDBACK CONTROLLER BLOCK

Interface	Properties
fdbck_cntrllr_hptc_fdbck_dsig	
Fall Time: <100ms	To provide clear instances of unique vibrations, the time needed to fall from 3.3V to 0V should be less than 100ms
Logic-Level: 3.3V	To produce the full-strength vibration, the powered motor must reach the minimum rated voltage of 3.0V via page 3 of the datasheet [1]
Rise Time: <100ms	To provide clear instances of unique vibrations, the time needed to fall from 3.3V to 0V should be less than 100ms
mgntmtr_fdbck_cntrllr_data	
Messages: 3-axis sensor data	MAG3110 datasheet states it provides digital 3-axis data. Register descriptions states 16bits for each X, Y, Z measurement. (Section 5, MAG3110 Datasheet [3])
Datarate: 100kHz	MAG3110 has a maximum operating frequency of 400kHz for Fast mode, 100kHz for normal mode. (Table 7, MAG3110 Datasheet [3])
Other: Measurement Frequency 20Hz	MAG3110 triggered measurements uses 10Hz. (Section 4.2.7, MAG3110 Datasheet [3])
Protocol: I2C	MAG3110 uses I2C as its control interface. (Section 2.4, MAG3110 Datasheet [3])
pwr_spply_fdbck_cntrllr_dcpwr	
Inominal: 9mA ± 10%	Arduino Pro Mini 3.3V/8MHz (DEV-11114) has a maximum current of 150mA. (Power, Arduino Pro Mini Datasheet [4])
Vmax: 12V	Arduino Pro Mini 3.3V has a on board voltage regulator. Voltage supplied through RAW pin will be regulated to 3.3V. (Page5-6, Arduino Pro Mini Tutorial [5])
Vmin: 3.4V	Arduino Pro Mini 3.3V has a on board voltage regulator. Voltage supplied through RAW pin will be regulated to 3.3V. (Page5-6, Arduino Pro Mini Tutorial [5])
fdbck_cntrllr_cntrllr_cd_data	
Protocol: Avrdude	Arduino IDE uses Avrdude to upload compiled code to the board. (Tools, Arduino IDE 1.5 Hardware Specification [6])
Other: Language: Arduino	Controller code is programmed using Arduino to interact with Arduino Pro Mini. (Arduino Pro Mini 3.3V Tutorial [5])
Other: Code Size less than 10KB	ATmega328P has 32K Bytes of In-System Self-Programmable Flash program memory which is sufficient for system code. (Page 1, ATmega328 Datasheet [7])
fdbck_cntrllr_bltth_data	
Protocol: Serial Port Protocol	Controller code is programmed by using Arduino IDE to interact with the Bluetooth module. (Arduino Pro Mini 3.3V Tutorial [5])
Protocol: 1Stop Bit, No Parity Bit	The extra stop bit is useful to allow the Bluetooth module to know that the connection is still valid.
Other: Name: ece34_S	Name is set for project ownership.
Other: Password: group34	Password is set for security reasons.
Other: Range >1m	The Bluetooth module has a distance range of approximately 10 meters, it is sufficient for a head-to-waist distance of approximately 1 meter. (General Specification, HC-05 Bluetooth Module Breakout Board [8])

7.6. *Bills of Materials*

Table 3 lists the bills of materials used for the Feedback Controller block.

TABLE XXV. BILLS OF MATERIALS FOR THE FEEDBACK CONTROLLER BLOCK

Item	Where to buy	Manufacturer	Manufacturer Part Number	Price
Arduino Pro Mini 3.3V	Digikey	SparkFun Electronics	DEV-11114	$9.95
SH_BT_Board HC-05 Bluetooth Module	Amazon	DSD Tech	-	$9.99
Triple Axis Magnetometer – MAG3110	Digikey	SparkFun Electronics	SEN-12670	$14.95

8. Controller Code Block

Block Owner: Samuel Lee

8.1. *Introduction*

The purpose of this section is to describe the design and behavior of the Controller Code in the system. The Controller Code block is written in Arduino language which is merely a set of C/C++ functions for the Arduino Pro Mini microcontroller. Most of the codes are independent from various system sensors such as the Ultrasonic Sensor, Laser ToF Sensor and Magnetometer, system feedback such as the Haptic Feedback module. This document provides an overview of the overall block function including interface properties, verification, design and validation. This block is completed by Samuel Lee.

8.2. *Block Overview*

The Wearable Sensor for the Blind project receives Ultrasonic sensor data, Laser ToF data, and Magnetometer data as inputs and provides feedback to the users with vibrations through the Haptic Feedback. As the system sensor band and the system feedback band are separated, the Bluetooth module will be the bridge between the two system bands. To accomplish this, the system code must be able to read data from the system sensors and provide them to the system feedback modules. The code will be downloaded once to system via the USB-to-TTL serial converter. Fig 22. Black Box Diagram Of Controller Code BlockFig 22 shows the black box diagram of the controller code block.

A full listing of interface properties can be found in Table XXVI.

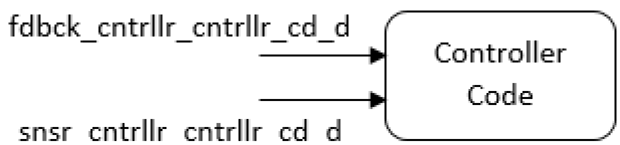


Fig 22. Black Box Diagram Of Controller Code Block

Table XXVI. Display Block Interfaces and Properties

Interface	Properties
fdbck_cntrllr_cntrllr_cd_data	1.Protocol: Avrdude 2.Other: Language: Arduino 3.Other: Code Size 10kB
snsr_cntrllr_cntrllr_cd_data	1.Protocol: Avrdude 2.Other: Language: Arduino 3.Other: Code Size less than 25kB

8.3. *Block Verification*

Based on the interfaces for this block, a verification (testing) process needs to be indicated. This will allow the final constructed design to be tested verifying that all the interface properties have been met and that the block is ready for integration into the system.

Link to Controller Code Test Videos

1. Connect the microcontroller to a 3.3V power supply via VCC pin or 3.4V-12V via RAW pin, and the programming interface either **snsr_cntrllr_cntrllr_cd_data** or **fdbck_cntrllr_cntrllr_cd_data** to a PC using an USB-to-TTL serial converter.
2. Load the test program to the microcontroller using the “Arduino.exe” via the usbasp programmer.
3. To enable “Arduino.exe” to output compilation steps through the log, go to Preferences -> Settings -> At “Show verbose output during”, check “compilation” and “upload”.
4. Check that program has been compiled and uploaded to the microcontroller through “Arduino.exe” log window.
5. The test program simulates the real code by having arrays of data as input and output them onto the serial monitor.

PASS:

9. Verify that the program is uploaded using avrdude through the log of “Arduino.exe”.
10. Verify that the program written in Arduino.
11. Verify that the code size is less than 25KB for the **snsr_cntrllr_bltth_data** and 10KB the **fdbck_cntrllr_cntrllr_cd_data.**
12. Verify that data are printed into the serial monitor.

If the block passes all the listed tests, all interface properties have been verified and the block is ready for inclusion into the system.

8.4. *Design Details*

The pseudocode presents the block design, including the interfaces of the block. These interfaces are further elaborated through validation information given in Table 2, which demonstrates externally-supported validly for each of the block’s properties.

8.4.1. Code Description (Pseudocode)

Sensor Controller Band:

```
Init:
  initialize serial for bluetooth
  initialize laser sensor
  initialize magnetometer
loop:
  check if low-power mode has been enabled
      enable low power mode

  read sensor band magnetometer
  read feedback band magnetometer from bluetooth
  calculate band offset

  trigger laser sensor measurements
  trigger and read ultrasonic measurement
  read laser sensor measurements

  construct data package with offset sensor data
  send data package to feedback controller through bluetooth
```

Feedback Controller band:

```
Init:
  initialize serial for bluetooth
  initialize magnetometer
loop:
  read magnetometer
  send magnetometer data to sensor controller through bluetooth

  read sensor data packages
  output data to haptic feedback modules
```

8.5. *Design Validation*

For validating the Controller Code block, both functional and interface validation will be explored. As the function of the code has an impact on the final system as a whole, the addition of the function validation is needed.

Functional Validation

a. The system needs to measure distance.
The code design shown includes specific step for measuring distance. This step is repeated each time through the main process loop.

b. The system needs to provide user interface.
The code contains a step to generate vibrations through haptic feedback module when needed.

Interface Validation

TABLE XXVII. INTERFACE PROPERTIES VALIDATION FOR THE SENSOR CONTROLLER BLOCK

Interface	Properties
fdbck_cntrllr_cntrllr_cd_data	
Protocol: Avrdude	The Arduino Pro Mini's Atmega328P is programmable via the Avrdude through Arduino IDE. (Tool, Custom Board Menu, Arduino IDE 1.5 Hardware Specification [1])
Other: Language: Arduino	Controller code is programmed by using Arduino IDE to interact with the Bluetooth module. (Arduino [2])
Other: Code Size less than 10kB	ATmega328P has 32K Bytes of In-System Self-Programmable Flash program memory which is sufficient for system code. (Page 1, ATmega328 Datasheet [3])
snsr_cntrllr_cntrllr_cd_data	
Protocol: Avrdude	The Arduino Pro Mini's ATMEGA328P is programmable via the Avrdude through Arduino IDE. (Tool, Custom Board Menu, Arduino IDE 1.5 Hardware Specification [1])
Other: Language: Arduino	Controller code is programmed using Arduino to interact with Arduino Pro Mini. (Arduino [2])
Other: Code Size less than 25kB	ATmega328P has 32K Bytes of In-System Self-Programmable Flash program memory which is sufficient for system code. (Page 1, ATmega328 Datasheet [3])

8.6. *Bills of Materials*

Table XXVIII lists the bills of materials used for the Controller Code block.

TABLE XXVIII. BILLS OF MATERIALS FOR THE CONTROLLER CODE BLOCK

Item	Where to buy	Manufacturer	Manufacturer Part Number	Price
Arduino Pro Mini 3.3V	Digikey	SparkFun Electronics	DEV-11114	$9.95
Arduino IDE	Arduino Software	Arduino	-	FREE
Adafruit_VL53L0X	Adafruit Github	Adafruit	-	FREE

9. BLUETOOTH BLOCK

Block Owner: Samuel Lee

9.1. *Introduction*

The purpose of this section is to describe the Bluetooth block of the ECE44x Wearable Sensor for the Blind project to 4th year ECE students with the intention of these students being able to build and verify the block without further research. The Bluetooth block is implemented using a SH_BT_Board V1.3 HC-05 Bluetooth module. This document provides an overview of the overall block function including interface properties and a schematic, verification for the design in the form of a step-by-step testing process, and support for the validity of the design in the form of outside research and numerical justification addressing individual properties.

9.2. *Design Details*

The wiring diagram (Fig 23) presents the block design, including the interfaces of the block. These interfaces are further elaborated through validation information given in Table XXIX, which demonstrates externally-supported validly for each of the block's properties.

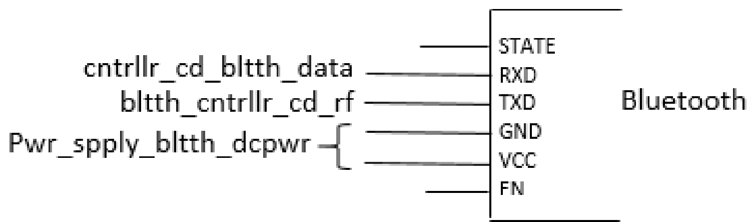


FIG 23. WIRING DIAGRAM OF BLUETOOTH BLOCK

9.3. *Block Overview*

The Bluetooth transmit serial data from the system controller to the system feedback module. The block is powered from **pwr_spply_bltth_dcpwr** interface usually around 5V with a nominal current draw of 50 mA. Fig 24 shows the black box diagram. Data are sent via a serial port protocol over the **snsr_cntrllr_bltth_data** interface. Finally, the output is represented by the **bltth_fdbck_cntrllr_data** interface. The function of the Bluetooth block is to transmit data serially between system sensor module and system feedback module. This block is completed by Samuel Lee.

A full listing of interface properties can be found in Table XXIX.

FIG 24. BLACK BOX DIAGRAM OF BLUETOOTH BLOCK

`

TABLE XXIX. BLUETOOTH BLOCK INTERFACES AND PROPERTIES

Interface	Properties
pwr_spply_bltth_dcpwr	1.Min Voltage: 3.6V 2.Max Voltage: 6V 3. Nominal Current: 30mA ±10% (During pairing) 4. Nominal Current: 8mA ±10% (After pairing)
snsr_cntrllr_bltth_data	1.Language: Arduino 2.Transmission Rate: 38400 Baud 3.Program Size: 2050 bytes (Master code) / 2256 bytes (Slave code) 4.Extra bit: 1 Stop bit, 0 Parity bit
bltth_fdbck_cntrllr_data	1.Protocol: Serial Port Protocol 2.Transmission rate: 38400 Baud 3.Range: At least 1 meter 4.Other: name – ece34_M (Master mode) / ece34_S (Slave mode) 5.Other: password – group34 6.Extra bit: 1 Stop bit, 0 Parity bit

9.4. *Block Verification*

Based on the interfaces for this block, a verification (testing) process needs to be indicated. This will allow the final constructed design to be tested verifying that all the interface properties have been met and that the block is ready for integration into the system.

Link to Bluetooth Test Videos

*Power Testing (**pwr_spply_bltth_dcpwr**)*

This test will verify that the Bluetooth module operates after providing **pwr_spply_bltth_dcpwr** interface.

1. Connect the Bluetooth block to power via the **pwr_spply_bltth_dcpwr** interface at system microcontroller's VCC pin OR use a power supply at 5V to the VCC on the Bluetooth.

2. Connect a multimeter to the **pwr_spply_bltth_dcpwr** to check the voltage.

3. Connect a light bulb at pin 7 of the system's microcontroller, TX pin on the system microcontroller to RX pin on Bluetooth; RX pin on the microcontroller to TX pin on Bluetooth; VCC and GND pins on microcontroller to VCC and GND pins on the Bluetooth respectively.

4. Detach VCC pin on the Bluetooth, then load a test program to the system microcontroller using "arduino.exe" via the usbasp programmer. It does the following: when received data '1', write to pin 7 to light up the light bulb for 2 seconds. Reattach VCC pin.

5. Connect a mobile device to the Bluetooth module and use a "bluetooth controller app", *ArduDriod*, to send data to the system.

PASS: This test passes if 9 out of 10 individuals are able to successfully power up the Bluetooth by illuminating the LED on the Bluetooth and light bulb by using a mobile device.

*Bluetooth Configuration Testing (**snsr_cntrllr_bltth_data**)*

This test will verify that the configuration of the Bluetooth module by using AT mode in Table XXIX. The communication interface **snsr_cntrllr_bltth_data** will be verified to the interface properties as expected.

1. Connect the Bluetooth block to power via the **snsr_cntrllr_bltth_data** interface at system microcontroller's VCC pin OR use a power supply at 5V to the VCC on the Bluetooth.

2. Connect the TX pin on the system microcontroller to the TX pin on the Bluetooth; RX pin on the microcontroller to the RX pin on the Bluetooth; VCC pin on the microcontroller to VCC and EN pins on the Bluetooth; GND pin on the microcontroller to the GND pin on the Bluetooth.

3. Detach VCC pin on the Bluetooth, then load an empty program to the mimcrocontroller using "arduino.exe" via the USB Asp programmer. Reattach the VCC pin.

4. Open the Serial Monitor on the Arduino IDE and use AT Commands: AT, AT+UART?, AT+NAME?, AT+PSWD?, AT+ROLE?.

PASS: This test passes if all AT command returns "OK" and values of the parameters.

*Bluetooth to Bluetooth Testing (**bltth_fdbck_cntrllr_data**)*

This test will verify that Master and Slave Bluetooth module will communicate to each other by sending data using **bltth_fdbck_cntrllr_data**.

1. Based on *Bluetoth Configuration Testing*, configure the first Bluetooth module to be in Slave mode and the second Bluetooth module to be in Master mode using **bltth_fdbck_cntrllr_data** interface.

2. Configure Bluetooth module to Slave mode with the following setting and command:
   a. NAME = ece34_S ; Command: AT+NAME=ece34_S
   b. PASSWORD = group34 ; Command: AT+PSWD=group34
   c. UART = 38400 Baud Rate, 1 Stop bit, 0 Parity bit ; Command: AT+UART=38400,0,0
   d. ROLE = 0 (Slave) ; Command: AT+ROLE=0
   e. Returns ADDRESS (14:3:620fe) ; Command: AT+ADDR?

3. Configure Bluetooth module to Master mode with the following setting and command:
   a. NAME = ece34_M ; Command: AT+NAME=ece34_M
   b. PASSWORD = group34 ; Command: AT+PSWD=group34
   c. UART = 38400 Baud Rate, 1 Stop bit, 0 Parity bit ; Command: AT+UART=38400,0,0
   d. ROLE = 1 (Master) ; Command: AT+ROLE=1
   e. CMOD = 1 ; Command: AT+CMODE=1
   f. BIND = 14:3:620fe ; Command: AT+BIND=14,3,620fe

4. Detach VCC pin on the Slave Bluetooth module, then load a test Slave program to the system microcontroller using "arduino.exe" via the USB Asp programmer. It does the following: reads the serial input via **bltth_fdbck_cntrllr_data**, waits for data '1' : Turns light bulb on for 100ms. Reattach VCC pin.

5. Detach VCC pin on the Master Bluetooth module, then load a test Master program to the system microcontroller using "arduino.exe" via the USB Asp programmer. It does the following: for every 2 second, '1' is sent serially via **bltth_fdbck_cntrllr_data** and turns off local light bulb, send '0' and turns on local light bulb. Reattach VCC pin.

PASS: This test passes if the light bulb in the Slave Bluetooth module lights up when the light bulb in the Master Bluetooth module turns off.

If the block passes all the listed tests, all interface properties have been verified and the block is ready for inclusion into the system.

9.5. *Design Validation*

For this block, an off the shelf solution was chosen. The HC-05 Bluetooth module fits the needs of the block interfaces without needing to be modified except minor configurations. Table XXX includes the interface property validation for this block. All the interface properties have been addressed and the design meets or exceeds the properties.

TABLE XXX. INTERFACE PROPERTY VALIDATION FOR THE BLUETOOTH BLOCK

Interface	Properties
pwr_spply_bltth_dcpwr	
Min Voltage: 3.6V	SH_BT_Board V1.3 HC-05 Bluetooth has an input voltage range of 3.6 – 6V via **pwr_spply_bltth_dcpwr** interface. BC417 Chip runs on 3.3V but a voltage regulator is used. (Section 1 SH_BT_Board Datasheet, Page 2[1])
Max Voltage: 6V	SH_BT_Board V1.3 HC-05 Bluetooth has an input voltage range of 3.6 – 6V via **pwr_spply_bltth_dcpwr** interface. BC417 Chip runs on 3.3V but a voltage regulator is used. (Section 1 SH_BT_Board Datasheet, Page 2 [1])
Nominal Current: 30mA ±10% (During pairing)	Operating current during pairing is 30mA. (Section 1 SH_BT_Board Datasheet, Page 3 [1])
Nominal Current: 8mA ±10% (After pairing)	Operating current after pairing is 8mA during communication. (Section 1 SH_BT_Board Datasheet, Page 3 [1])
snsr_cntrllr_bltth_data	
Language: Arduino	Controller code is programmed by using Arduino IDE to interact with the Bluetooth module. (Arduino [3])
Transmission Rate: 38400 Baud	Serial baud rate is used by the system microcontroller to transfer data.
Program Size: 2050 bytes (Master code) / 2256 bytes (Slave code)	ATmega328P has 32K Bytes of In-System Self-Programmable Flash program memory which is sufficient for system code. (ATmega328 Datasheet Page 1 [4])
Extra bit: 1 Stop bit, 0 Parity bit	The extra stop bit is useful to allow the Bluetooth module to know that the connection is still valid.
bltth_fdbck_cntrllr_data	
Protocol: Serial Port Protocol	The HC-05 Bluetooth module uses Bluetooth SSP (Serial Port Protocol) for transparent wireless serial connection setup. (Overview of Serial Port Bluetooth Module HC-05 [6])
Transmission rate: 38400 Baud	Supported baud rate: 4800, 9600,19200, 38400, 57600, 1152000, 230400, 460800, 921600, 1382400. Baud rate is the capability of the serial port to transfer a maximum of 38400 bits per second. (Section 3.1 AT Command Mode, SH_BT_Board Datasheet, Page 12 [1])
Range: At least 1 meter	The Bluetooth module has a distance range of approximately 10 meters, it is sufficient for a head-to-waist distance of approximately 1 meter. (General Specification, HC-05 Bluetooth Module Breakout board Page 1 [5])
Other: name – ece34_M (Master mode) / ece34_S (Slave mode)	The HC-05 Bluetooth module is Master and Slave mode compatible. Master role have paired memory to remember an address of a Slave Bluetooth module to pair with it automatically. (HC Serial Bluetooth Instruction Manual, Page 3 [2])
Other: password – group34	Password is set for security reasons.
Extra bit: 1 Stop bit, 0 Parity bit	The extra stop bit is useful to allow the Bluetooth module to know that the connection is still valid.

9.6. *Bills of Materials*

Table XXXI lists the bills of materials used for the Bluetooth block.

TABLE XXXI. BILLS OF MATERIALS FOR THE BLUETOOTH BLOCK

Item	Where to buy	Manufacturer	Manufacturer Part Number	Price
SH_BT_Board HC-05 Bluetooth Module	Amazon	DSD Tech	-	$9.99

1. ENCLOSURE – BLOCK OWNER: JACY BARR

***Abstract*—This document covers the design of the Enclosure block for the ECE442 Wearable Sensor for the Blind Project. The enclosure is representative of a rigid case which contains all power, communication, and processing system blocks in the design as well as a structure to orient the ultrasonic sensor. These are designed to be connected to a fabric/elastic harness either worn around a subject's head or waist while remaining durable, user-accessible, and resistant to water. 3D printed component housing in combination with metal machine screws and rubber gaskets provide isolation from environmental stresses without sacrificing serviceability.**

1.1. INTRODUCTION

The purpose of this document is to describe the Enclosure block of the Wearable Sensor for the Blind project to 4th year ECE students and engineers with the intention of these individuals being able to build and verify the blocks functionality without further research. This document describes the casing that will secure the internal electronic components that are enclosed within and validates the chosen design based on those dimensions.

1.2. BLOCK OVERVIEW

The Enclosure block consists of two main bodies that secure the electronics. This is implemented as designed 3D printed enclosures with a removable lid that provides access to the user replaceable AA batteries and a form-fitting sensor mount. The case enclosure contains the processing, power, Bluetooth, magnetometer, voltage regulators, and custom PCB that connects the individual components. The sensor mount holds only the ultrasonic sensor for proper orientation. The enclosure case is designed be a rigid hard case that holds the electronics and resists water penetration to allow for outdoor use. The block was designed by first identifying the dimensions of the components to be used and then designing around them using CAD software. The external form mount was decided upon based on the natural curve of the human head while the case was optimized for minimal space occupancy, as the sum of components requires a great deal of space as a wearable accessory.

1.3. INTERFACES

The Enclosure block does not have any specific interfaces defined, but is characterized by the ability to contain and support the electronic components within the units. As the system is designed to be used outdoors and indoors, it is necessary that it remains functional for extended periods of time while exposed to environmental factors such as rainfall. Additionally, as the units are to be worn by the user, the casings must allow for some type of strap or band to attach externally. Fig 25. shows the black box diagram.

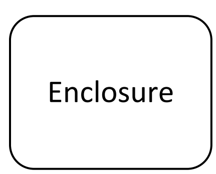


FIG 25 BLACK BOX DIAGRAM OF ENCLOSURE BLOCK

1.4. VERIFICATION

The enclosure block does not have any defined interfaces, so the only testing to verify this blocks functionality is related to the ability to resist water, rigidly contain all other blocks of the system, and allow for some type of strap or band to be connected. Performing the following will verify such.

Test 1: Case-Component & Sensor-Mount Containment

1. Procure each completed block for including into the enclosure. If a block is not yet available, produce a decoy matching the dimensions of each component and attach them to the PCB. These are then to be inserted into the enclosure.
2. Insert components or decoys to the enclosure case and close it with the necessary tools(screwdriver) and insert ultrasonic sensor into mount.
3. Briefly inspect each system, give the case and mount a gentle shake.

PASS: This test passes if all the blocks/representations of blocks noted fit with in the enclosure and after shaking and none of the decoys inside are dislodged from their original position.

Test 2: Case Water Resistance

1. Obtain a small dry sheet of paper and place it in the case enclosure
2. Close the case with the lid and necessary screws
3. Lay the unit in a dry bucket and splash a cup of water over it
4. Remove unit from bucket and dry off the outside

PASS: This test passes if the paper inside once removed from the case enclosure is still dry

Test 3: Strap/Band Connection Presence

1. Pass a strip of paper 25mm or less in width through enclosure case side loop or mount side loop
2. Connect opposing ends of strip of paper and pinch with fingers

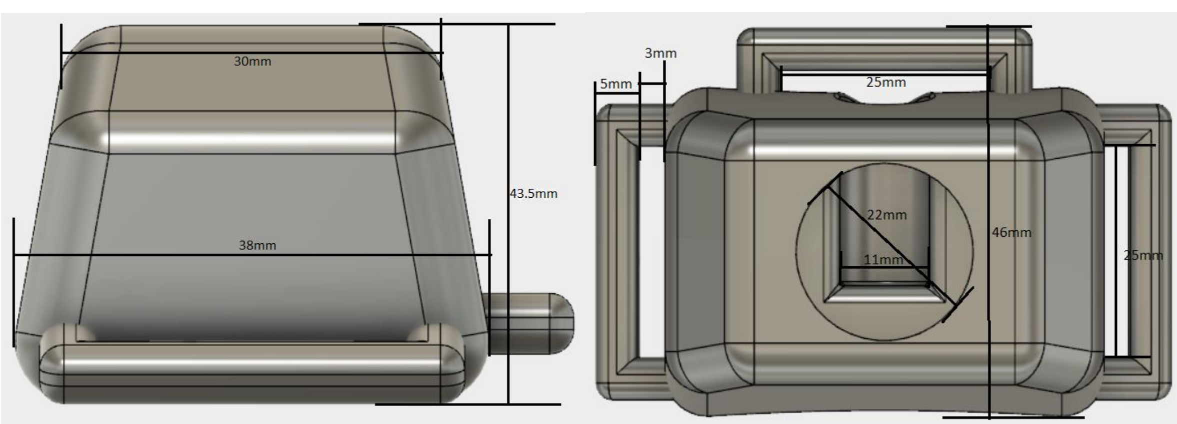
PASS: This test passes if strip of paper does not disconnect from enclosure after ends are connected

Link to Video

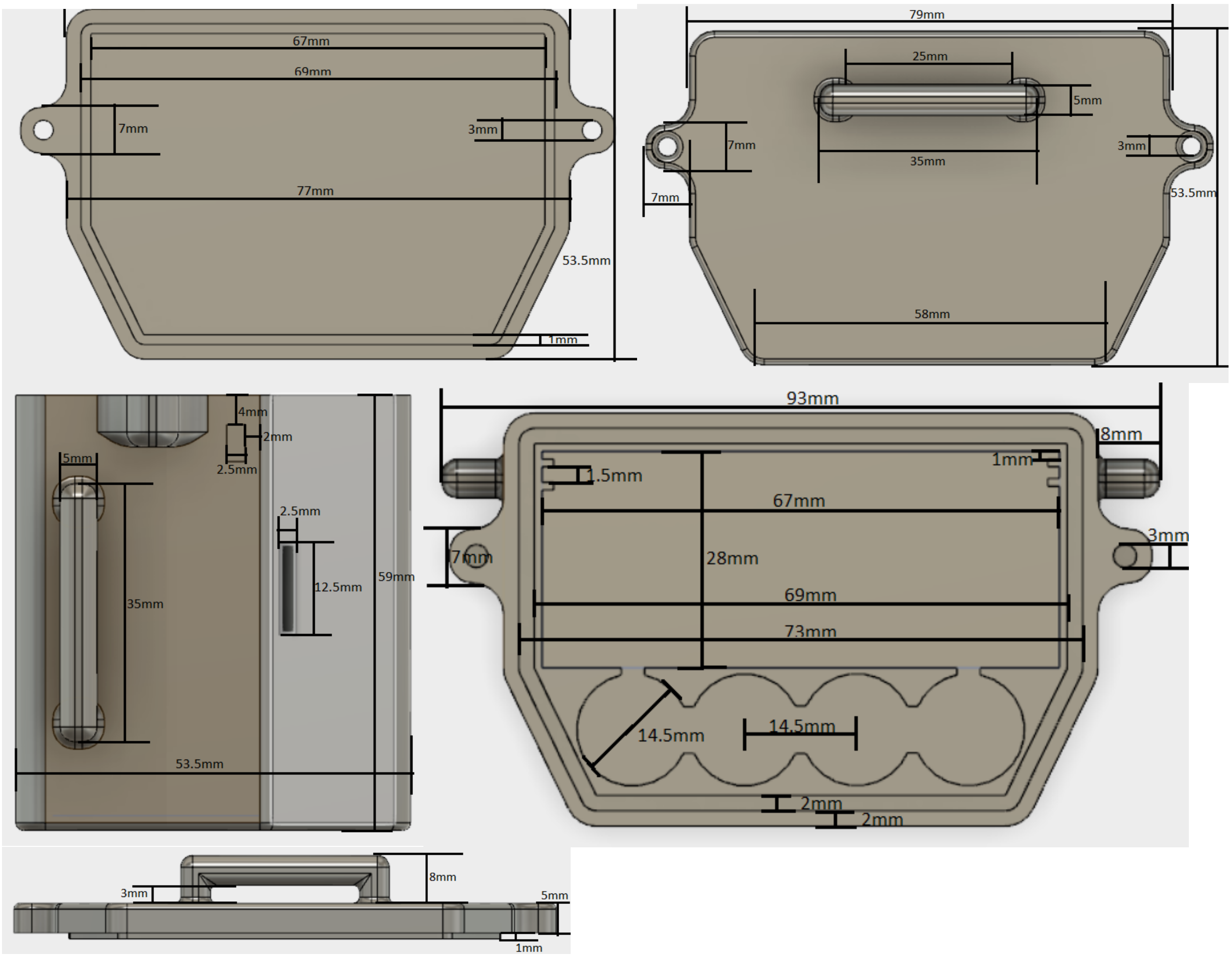
1.5. DESIGN

1.5.1.Mechanical Drawings

1.5.1.1. Ultrasonic Sensor Mount



1.5.1.2. Sensor Band Enclosure



1.5.1.3. Feedback Band Enclosure

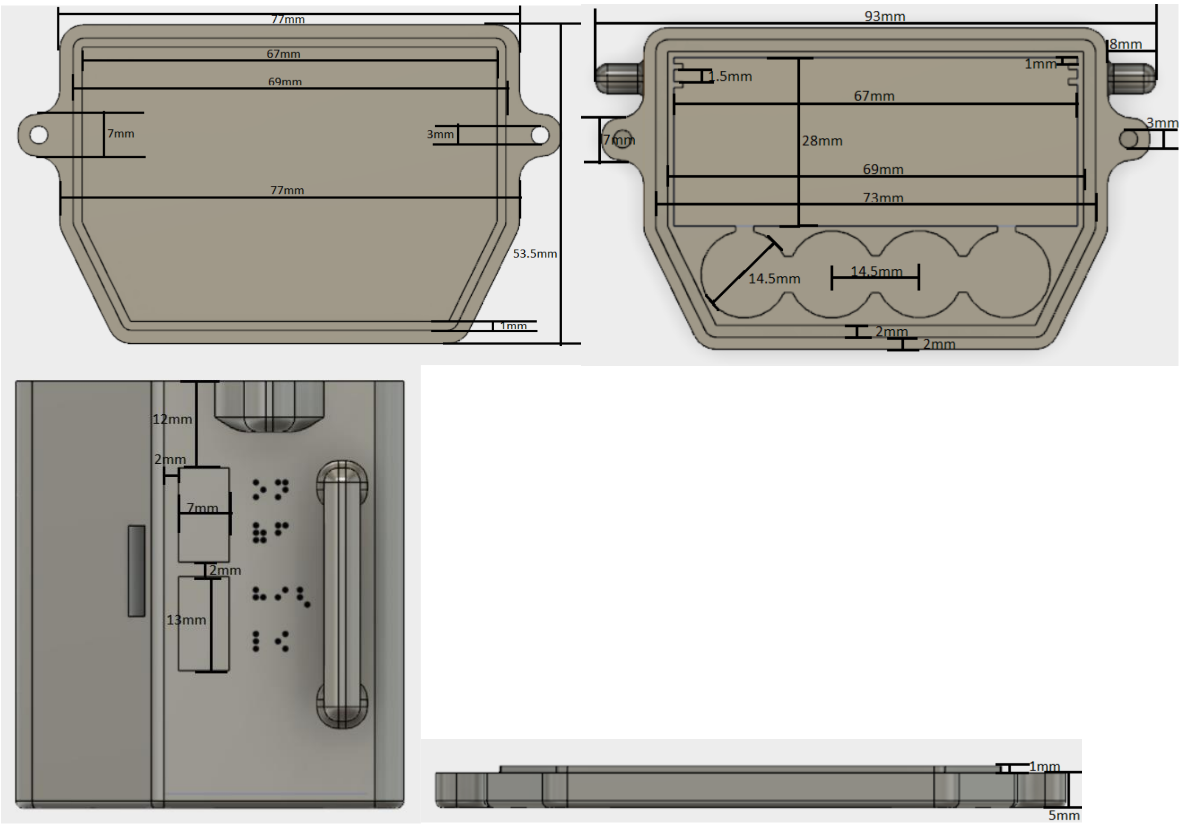


FIG 26. ENCLOSURE MODELS

1.5.2.Design Validation

In the absence of any specifically identified interfaces, validation will cover each of the other blocks to be included and show that they fit with in the enclosure. As shown in Fig 26, all the other components fit within the case as described. Drawings provided in the design section allow for dimensional verification that the components have adequate room to be mounted securely while allowing for each part to seamlessly connect together with the necessary hardware.

1.6. BILL OF MATERIALS

Reference Designator	Component Value	Manufacture	Part Number	Supplier	Quantity	Unit Price
ABS Filament	N/A	MakerBot	N/A	OSU Printing	~80g	$0.10/g
Rubber O-ring	2mm x 90mm	The O-ring Store	N2.00X042	The O-ring Store	2	$0.35
Metal Bolts	3mm x 6mm	Bolt Depot	6811	Bolt Depot	4	$0.06
Toggle Switches	125V/6A SPDT	Gadgeter	TS-2T-S	Amazon	4	$0.53

2. SENSOR HARNESS – BLOCK OWNER: JACY BARR

2.1. *Introduction*

The purpose of this section is to describe the Sensor Harness block of the ECE44x Wearable Sensor for the Blind project to 4th year ECE students. This section describes the sensor harness which acts to connect the sensor enclosure, ultrasonic sensor and laser ToF sensor modules to gather information of the user's surrounding.

2.2. *Design Details*

The diagram (Fig 18) presents the block design, including the interfaces of the block. These interfaces are further elaborated through validation information given in TABLE XX, which demonstrates externally-supported validly for each of the block's properties.

Fig 27. Diagram of Feedback Harness Block

Block Overview

The Sensor Harness block will hold the sensor enclosure, ultrasonic sensor and laser ToF sensor modules.

A full listing of interface properties can be found in Table XX.

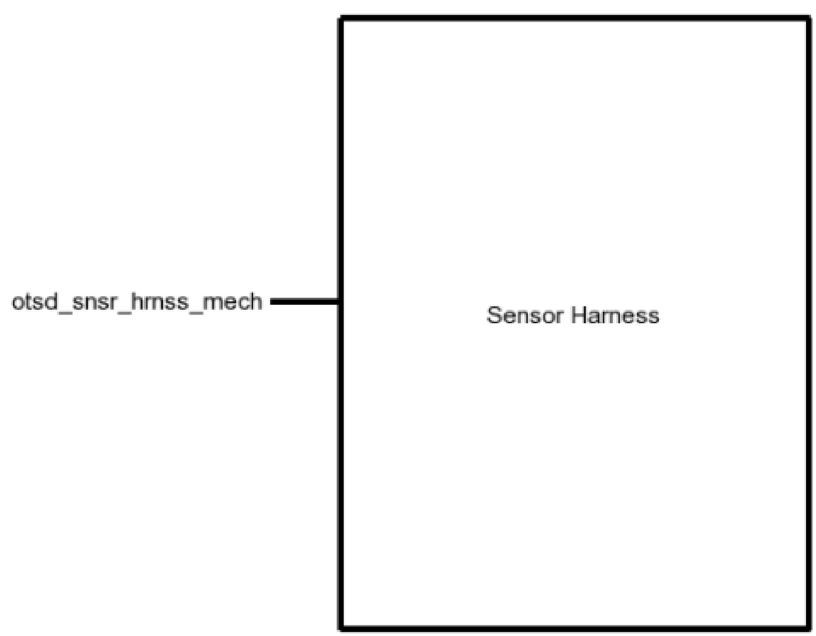


FIG 28. BLACK BOX DIAGRAM OF SENSOR HARNESS BLOCK

`

TABLE XXXII. SENSOR HARNESS BLOCK INTERFACES AND PROPERTIES

Interface	**Properties**
ostd_snsr_hrnss_mech	1.Other: Minimum Circumference: 50cm 2.Other: Maximum Circumference: 60cm 3.Other: Maximum Weight: 100g

2.4. *Block Verification*

The Feedback Harness block only has a defined interface, so the only testing to verify this block's functionality is related to the ability to expand contract to accommodate to the various physics of individuals participating in using the system.

Link to All Sensor Harness Test Videos

1. ***Expandability Test (otsd_snsr_hrnss_mech)***
    a. Connect the headband to the user.

PASS:

13. Verify that the buckle locks.
14. Verify that the harness weights less than 100g.
15. Verify that the harness expands with 60 cm circumference.
16. Verify that the harness contract back to 50 cm circumference.

If the block passes all the listed tests, all interface properties have been verified and the block is ready for inclusion into the system.

2.5. *Design Validation*

In the absence of any specifically identified interfaces, validation will cover each of the other blocks to be included and show that they fit with in the feedback harness. Drawings provided in the design section allow for dimensional verification. The document will be updated to match the new information.

2.6. *Bills of Materials*

Table 3 lists the bills of materials used for the Sensor Controller block.

TABLE XXXIII. BILLS OF MATERIALS FOR THE SENSOR HARNESS BLOCK

Item	**Where to buy**	**Manufacturer**	**Manufacturer Part Number**	**Price**
Non-Roll Woven Elastic	Amazon	Dritz	9507B	$4.95
Quick release Buckle	Amazon	Coghlan	0180	$4.64

3. Feedback Harness - Block Owner: Jacy Barr

3.1. *Introduction*

The purpose of this section is to describe the Feedback Harness block of the ECE44x Wearable Sensor for the Blind project to 4th year ECE students. This section describes the feedback harness which acts to connect the haptic feedback modules to relay feedback vibrations back to the user.

3.2. *Design Details*

The diagram (Fig 18) presents the block design, including the interfaces of the block. These interfaces are further elaborated through validation information given in Table XX, which demonstrates externally-supported validly for each of the block's properties.

Fig 29. Diagram of Feedback Harness Block

Block Overview

The Feedback Harness block will hold each haptic feedback modules and connects with the feedback enclosure.

A full listing of interface properties can be found in Table XX.

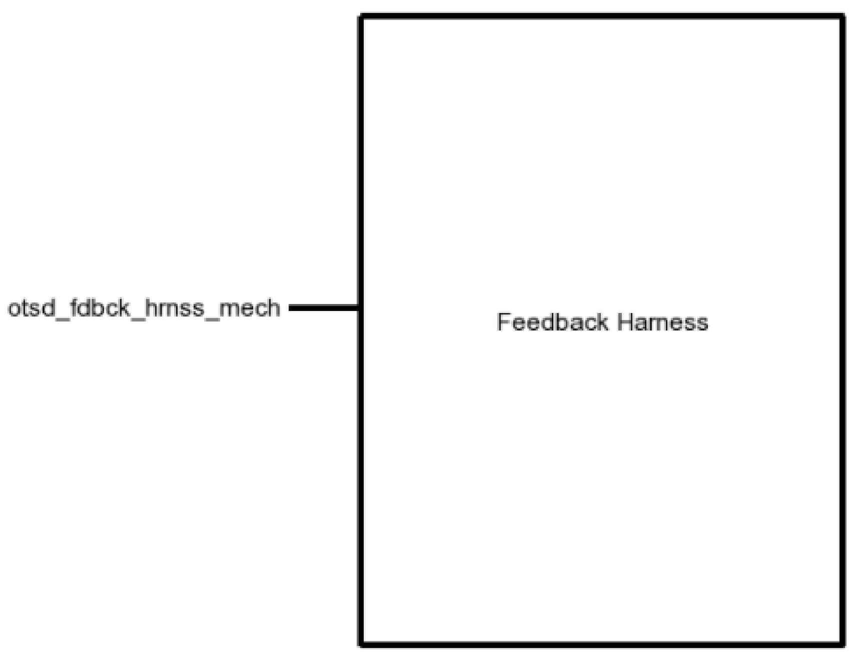


Fig 30. Black Box Diagram of Feedback Harness Block

`

Table XXXIV. Feedback Harness Block Interfaces And Properties

Interface	Properties
ostd_fdbck_hrnss_mech	1.Other: Minimum Circumference: 84cm 2.Other: Maximum Circumference: 110cm 3.Other: Weight: 100g

3.4. *Block Verification*

The Feedback Harness block only has a defined interface, so the only testing to verify this block's functionality is related to the ability to expand contract to accommodate to the various physics of individuals participating in using the system.

Link to All Feedback Harness Test Videos

6. Expandability Test (otsd_fdbck_hrnss_mech)

a. Connect the waistband to the user.

PASS:

17. Verify that the buckle locks.
18. Verify that the harness weights less than 100g.
19. Verify that the harness expands with 110 cm circumference**.**
20. Verify that the harness contract back to 84 cm circumference**.**

If the block passes all the listed tests, all interface properties have been verified and the block is ready for inclusion into the system.

3.5. *Design Validation*

In the absence of any specifically identified interfaces, validation will cover each of the other blocks to be included and show that they fit with in the feedback harness. Drawings provided in the design section allow for dimensional verification. The document will be updated to match the new information.

3.6. *Bills of Materials*

Table 3 lists the bills of materials used for the Feedback Controller block.

TABLE XXXV. BILLS OF MATERIALS FOR THE FEEDBACK HARNESS BLOCK

Item	Where to buy	Manufacturer	Manufacturer Part Number	Price
Non-Roll Woven Elastic	Amazon	Dritz	9507B	$4.95
Quick release Buckle	Amazon	Coghlan	0180	$4.64

1. HAPTIC FEEDBACK - BLOCK OWNER: JACY BARR

1.1. INTRODUCTION

The purpose of this document is to describe the Haptic Feedback block of the ECE441 Wearable Sensor for the Blind project so fellow 4th year ECE students will be able to reproduce and verify this component's functionality without further research. The haptic Feedback block is implemented using eight Coin Vibration Motors and eight N-type surface mount MOSFET transistors. This document provides an overview of the overall block function including interface properties, schematic, and verification for the design in the form of a step-by-step testing process, and support for the validity of the design in the form of outside research and numerical justification addressing individual properties.

1.2. BLOCK OVERVIEW

The MOSFETs included in this block receive a digital signal from the microcontroller which allows current to flow from the power supply to the Coin Vibration Motors. Varying the intensity of vibrations via use of pulse width modulation, each motor is then able to produce a range of distinct patterns which are associated with measured proximity data. The block must be capable of producing a range of distinct intensities of vibrations while maintaining a short rise and fall time to relay the information processed by the system back to the user. Fig 31 shows the black box diagram of the described functionality. The block is powered from the **pwr_spply_hptc_fdbck_dcpwr** interface from a 3.3V source with a nominal current draw of 50mA +/- 20% at a full duty cycle. Signals indicating when to power the motor are sent via digital pins over the **fdbck_cntrllr_hptc_fdcbk_dsig** interface and are defined by the rise and fall time of signals which affect how quickly the vibrations can initiate and cease. Finally, the output is represented by the **hptc_fdbck_otsd_usrout** interface. Due to the nature of the output interface, its properties rely on the user's interpretation of the vibration, so to ensure the user experiences match the signals passed, the properties are defined around the ability of motor to produce distinct vibrations and user's ability to distinguish them. The full listing of interface properties is listed below in Table 36.

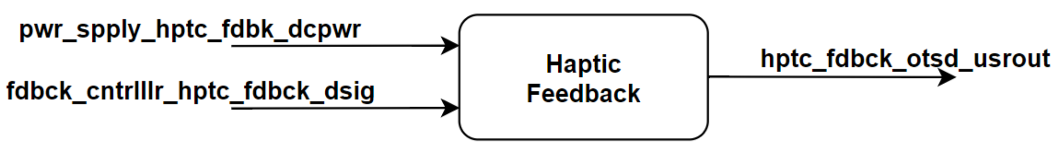


FIG 31. BLACK BOX DIAGRAM OF HAPTIC FEEDBACK BLOCK

TABLE 36 HAPTIC FEEDBACK INTERFACES AND PROPERTIES

Interface	Properties
hptc_fdbck_otsd_usrout	1. Type: Haptic Feedback 2. Usability: 9/10 users are able to detect the vibration intensity changing with 100% accuracy 3. 5 Distinct Intensities 4. Dimension: LxWxH <50mm^3
pwr_spply_hptc_fdbck_dcpwr	1. Vmin: 2.7V 2. IPeak: 75mA (per motor) 3. VMax: 3.3V 4. INominal: 50mA +/- 20% at 100% Duty Cycle
fdbck_cntrllr_hptc_fdbck_dsig	1. Fall Time: <100ms 2. Rise Time: <100ms 3. Logic-Level: 3.3V

1.3. VERIFICATION

Based on the interfaces for this block, a verification (testing) process is provided below to verify interface property functionality. This information will allow for the final implementation of this block to be tested, verifying that each interface property has been satisfied and that the block is ready for integration into the complete system.

1.3.1. Power Testing

This test will verify that the haptic feedback motor will not exceed the current draw threshold and function properly with the provided voltage.

1. Connect feedback motor and additional testing components to breadboard according to wiring diagram in Fig 32
2. Connect the Haptic Feedback Block to power via the **pwr_spply_hptc_fdbck_dcpwr** interface at 3.3V and 2.7V using a DC power supply. Insert a current meter (DMM) inline OR use a power supply that displays the used current.
3. Connect a microcontroller to the **fdbck_cntrllr_hptc_fdbck_dsig** interface.
4. Mount breadboard(via tape or weight) to surface to prevent unit from vibrating too violently.
5. Load test program to the microcontroller that produces a series of ten distinct frequencies followed by a max duty cycle.

PASS: This test passes if peak current does not exceed 80mA, unit continues to produce varying frequencies at both 3.3V/2.7V, and nominal voltage during max duty cycle stays within 10% of 3V on **pwr_spply_hptc_fdbck_dcpwr** interface.

Link to Video

1.3.2. Signal Output(User) Testing

For this test, we will verify that the output, **hptc_fdbck_otsd_usrout** is able to produce the intensities provided by the **fdbck_cntrllr_hptc_fdbck_dsig** interface with enough clarity for the user to identify when they change while remaining small in form.

1. Connect feedback motor and additional testing components to breadboard according to the wiring diagram in Fig 32
2. Connect oscilloscope probes to the negative and positive terminals of the haptic feedback motor and configure the oscilloscope to measure the pulse width of the square wave.
3. Load a test program to the microcontroller which steps through each intensity for 30s each
4. Measure the total time the motor is powered high for each period to determine the relative intensity from each interval
5. Have testing subjects place their finger on the vibration motor while the intensity stepping code runs.

PASS: This test passes if 9/10 users are able to distinguish when there is a change in intensity of vibration with 100% confidence and the oscilloscope measurements are within 10% of the data signals provided by **fdbck_cntrllr_hptc_fdbck_dsig**.

Link to Video

1.3.3. Signal Input Testing

This test will verify that the digital signal interface **fdbck_cntrllr_hptc_fdcbk_dsig** is able to provide the haptic feedback motor with a data signal that has both rise and fall times less than 100ms at a logic level of 3.3V.

1. Connect feedback motor and additional testing components to breadboard according to the wiring diagram in Fig 32
2. Connect oscilloscope probes to ground and the digital signal **fdbck_cntrllr_hptc_fdbck_dsig** pin connected to the rail of the breadboard and configure the oscilloscope to measure time difference and amplitude.
3. Load a test program to the microcontroller that produces the a 1Hz frequency signal.
4. Measure the rise time, fall time, and amplitude of the signal.

PASS: This test passes if the oscilloscope frequency measurements provided by the **cntrllr _hptc_fdbck_data** interface produces a rise and fall time less than 100ms each at 3.3V.

Link to Video

If the block passes all of the listed tests, all interface properties have been verified and the block is ready for inclusion into the system.

1.4. Design

The Wiring Diagram (Fig 32) presents the block design, including the interfaces of the block. The full system will include eight identical configurations of this spaced equidistant from one another. These interfaces are further elaborated through the validation information provided in Table 37 Interface Property Validation For The Display Block, which demonstrates the feedback motors ability to satisfy each defined interface property.

1.4.1. Wiring Diagram

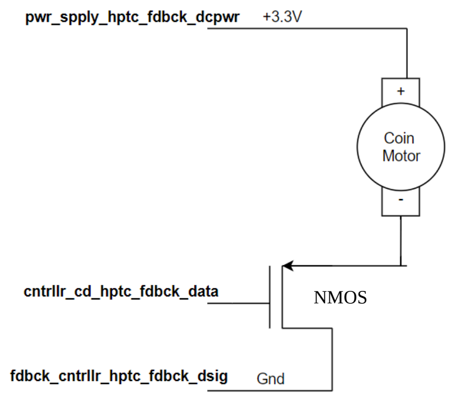


FIG 32. WIRING DIAGRAM FOR THE HAPTIC FEEDBACK BLOCK

1.4.2. Design Validation

For this block, a coin vibration motor was chosen to relay processed sensory information back to the user. The 3V C0720B015F model of vibration motor accompanied by a 2N7000 N-Type MOSFET to act as a switch for the data signal pins suits the needs of the system into interface properly with the block.

Table 37 Interface Property Validation For The Display Block includes the interface property validation for this block. All interface properties have been addressed and the design meets or exceeds the properties.

TABLE 37 INTERFACE PROPERTY VALIDATION FOR THE DISPLAY BLOCK

Property	**Validation**
hptc_fdbck_otsd_usrout	
Usability: 9/10 users are able to detect the vibration intensity changing with 100% accuracy	To provide clear instances of vibrations, the duration of motor being powered high varied through the use of pulse width modulation. Adjusting the time in which the motor is powered high by more than 15% for each interval gives enough differentiation for each variation
Dimension: LxWxH <50mm^3	Datasheet specifications for 3V Coin motor C0720B015F dimensions as stated on page 9 [1] prove volume is less than specified property
Type: Haptic feedback	Project is developed for individuals with impaired vision, so a haptic motor provides the clearest method of relaying sensory information without over stimulating the users auditory sense.
5 Distinct Intensities	The user's ability to distinguish the distance from nearby obstacles is a defining feature of our design, so ensuring the intervals of feedback remain clear is crucial.
pwr_spply_hptc_fdbck_dcpwr	
V_{min}: 2.7V	Datasheet specifications for 3V Coin motor C0720B015F minimum operation voltages stated as 2.7V on page 4 [1]
I_{Peak}: 75mA (per motor)	Datasheet specifications for 3V Coin motor C0720B015F maximum current rating stated as 75mA on page 5 [1]
V_{Max}: 3.3V	Datasheet specifications for 3V Coin motor C0720B015F maximum operation voltages stated as 3.3V on page 4 [1]
$I_{Nominal}$: 50mA +/- 20% at 100% duty cycle	Initial powering of the motor creates a spike in current draw which quickly settles to around 50mA at 100% duty cycle.
fdbck_cntrllr_cd_hptc_fdbck_data	
Fall Time: <100ms	To provide clear instances of unique vibrations, the time needed to fall from 3.3V to 0V should be less than 100ms
Rise Time: <100ms	To provide clear instances of unique vibrations, the time needed to fall from 3.3V to 0V should be less than 100ms
Logic-Level: 3.3V	To produce the full-strength vibration, the powered motor must reach the minimum rated voltage of 3.0V via page 3 of the datasheet [1]
Rise Time: <100ms	To provide clear instances of unique vibrations, the time needed to rise from 0V to 3.3V should be less than 100ms

1.5. Bill of Materials

Reference designator	Component Value	Manufacture	Part number	Supplier	Quantity	Unit price
N-Type MOSFET	N/A	ON Semiconductor	2N7000FS-ND	Digikey	8	$0.47
Coin Motor	3V	Jinlong Machinery & Electronics Co.	C0820B002F	Xump	8	$0.99

1. MAGNETOMETER BLOCK

Block Owner: Jacy Barr (ungraded)

1.1. Introduction

The purpose of this section is to describe the Magnetometer block of the ECE44x Wearable Sensor for the Blind project to 4th year ECE students. The Magnetometer block is implemented using a pair of Triple Axis Magnetometer from SparkFun Electronics. Included in this section are an overview of the block, a schematic and wiring diagram, the properties of the interfaces with the other blocks in the system and testing procedures for each, and reasoning for why this design is the best solution for this block.

1.2. Design Details

The wiring diagram (Fig 1) presents the block design, including the interfaces of the block. These interfaces are further elaborated through validation information given in TABLE I, which demonstrates externally-supported validity for each of the block's properties.

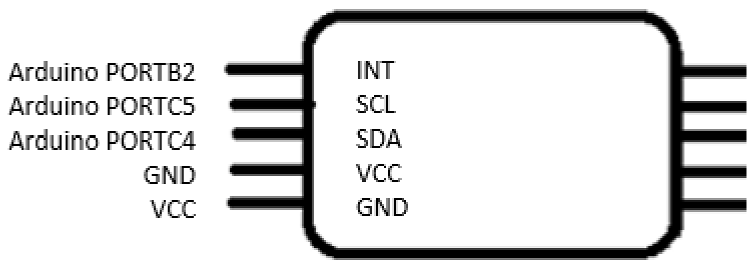


Fig 1. Wiring Diagram of Feedback Harness Block

1.3. Block Overview

The SEN12670 breakout board for this block detects three axes (xyz) of surrounding magnetic fields to determine orientation with respect to the magnetic north and south poles. This polled information is relayed back to a microcontroller via I2C communication to align head mounted sensors with waist mounted vibrations motors. The block must be capable of sensing the earth's magnetic field orientation and outputting data over I2C to communicate this data back to the microcontroller. Fig. 2 shows the black box diagram. The block is powered from the **pwr_spply_mgnmtr_dcpwr** interface from a 3.3V source with a nominal current draw of 5uA during standby mode. Communication to the magnetometer will be via I2C over the **cntrllr_cd_mgntmr_data** interface while processed data from the magnetometer will be sent over the **cntrllr_cd_hptc_fdbck_**data interface. Finally, the environmental magnetic fields is measured in terms of flux density with respect to each axes which is represented by the **otsd_mgntmtr_envin** interface. The full listing of interface properties are listed below in Table 1.

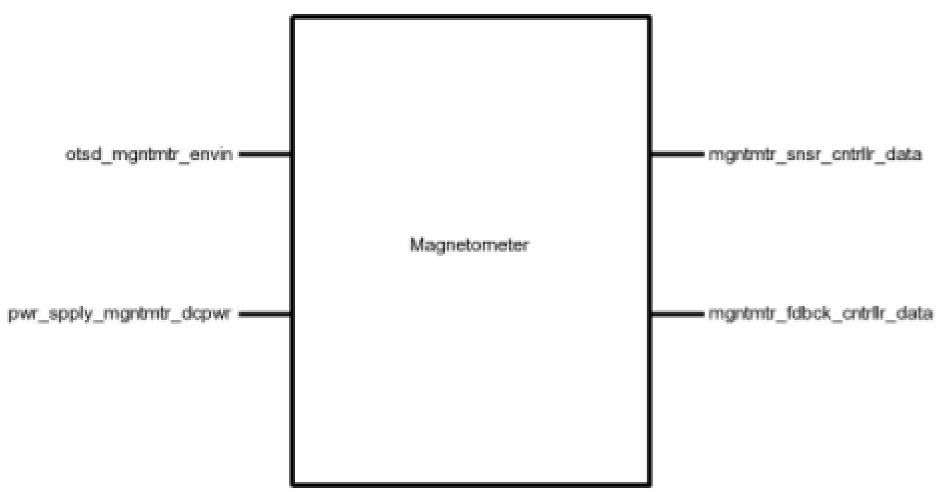


FIG 2. BLACK BOX DIAGRAM OF MAGNETOMETER BLOCK

`

TABLE I. MAGNETOMETER BLOCK INTERFACES AND PROPERTIES

Interface	Properties
ostd_mgntmtr_envin	1.Other: Direction Measurement Accuracy: ±15°
mgntmtr_snsr_cntrllr_data	1.Datarate: 100kHz 2.Messages: Sensor Data 3.Other: Measurement Frequency: 20 Hz 4. Protocol: I2C
mgntmtr_fdbck_cntrllr_data	1.Datarate: 100kHz 2.Messages: Sensor Data 3.Other: Measurement Frequency: 20 Hz 4. Protocol: I2C
pwr_spply_mgntmtr_dcpwr	1.Inominal: 5µA ± 10% (standby) 2.Ipeak: 0.5mA (measurement) 3.Vmax: 3.6V 4.Vmin: 1.95V

1.4. Block Verification

Based on the interfaces for this block, a verification (testing) process needs to be indicated. This will allow the final constructed design to be tested verifying that all the interface properties have been met and that the block is ready for integration into the system.

This test will verify the power requirements for this block on the *pwr_spply_mgntmtr* interface

A. Power Testing

1. Connect a magnetometer to a DC power supply that can display current, set to 3.6V, via the *pwr_spply_mgntmtr interface*
2. Connect a microcontroller using I2C at 100kHz to the *mgntmre_snsr_cntrllr_data* interface
3. Load a test program on the microcontroller than takes a measurement for 200ms, then idles for 200ms.
4. Note the current displayed on the DC power supply during a measurement, and during idle.
5. Repeat steps 1-4 with the DC power supply set to 1.95V.

PASS: This test passes if the current draw never exceeds 0.5mA during the measurement phase, nominal current stays within 10% of 5uA, and the magnetometer continues to function for both the min and max voltages

Link to Video

This test will verify the ability to detect and accurately read heading with 15 degree of accuracy

B. Environmental Testing

1. Connect magnetometer according to wiring diagram in fig 1 and obtain a compass that is calibrated
2. Run code to allow for magnetometer to read heading and compare serial monitor output to compass reading

PASS: This test passes if the heading angle read from the serial monitor is within 15 degrees of that shown on the calibrated compass

Link to Video

This test will verify the ability of the system to communicate via I2C

C. Communication Testing

1. Connect microcontroller to components according to wiring diagram in fig 1 using I2C at 100kHz to the *mgntmtr_snsr_cntrllr_data* interface

2. Load a test program on the microcontroller than takes a measurement for 200ms, then idles for 200ms.
3. Connect an oscilloscope to the I2C lines and configure to measure frequency

PASS: This test passes if the I2C interface communication links at 100kHz and the measured frequency is roughly 20kHz

Link to Video

If the block passes all of the listed tests, all interface properties have been verified and the block is ready for inclusion into the system.

1.5. Design Validation

For this block, an off the shelf solution was chosen. SparkFun's Triple Axis Magnetometer Breakout – MAG3110 fits the needs of the block interfaces without needing to be modified except minor configurations. Table II includes the interface property validation for this block. All the interface properties have been addressed and the design meets or exceeds the properties.

TABLE II. INTERFACE PROPERTY VALIDATION FOR THE BLUETOOTH BLOCK

Property	Validation
ostd_mgntmtr_envin	
Other: Direction Measurement Accuracy: ±15°	
mgntmtr_snsr_cntrllr_data	
Datarate: 100kHz	Table 7 of the Magnetometer Datasheet states that the maximum I2C operating frequency is 400kHz [8]
Messages: Sensor Data	magnetometer Datasheet states the type of sensor data transmission type [3]
Other: Measurement Frequency: 20 Hz	magnetometer Datasheet states that 20Hz is the measurement frequency [8].
Protocol: I2C	Section 3 of the magnetometer Datasheet states that I2C is the control interface required [8].
mgntmtr_fdbck_cntrllr_data	
Datarate: 100kHz	Table 7 of the Magnetometer Datasheet states that the maximum I2C operating frequency is 400kHz [8]
Messages: Sensor Data	magnetometer Datasheet states the type of sensor data transmission type [3]
Other: Measurement Frequency: 20 Hz	magnetometer Datasheet states that 20Hz is the measurement frequency [8].
Protocol: I2C	Section 3 of the magnetometer Datasheet states that I2C is the control interface required [8].
pwr_spply_mgntmtr_dcpwr	
Inominal: 5µA ± 10% (standby)	Table 7 of the Magnetometer Datasheet states that the nominal current draw is 5uA [8]
Ipeak: 0.5mA (measurement)	Table 7 of the Magnetometer Datasheet states that the maximum current draw is 05mA [8]
Vmax: 3.6V	Table 7 of the Magnetometer Datasheet states that the maximum operating voltage is 3.6V [8]
Vmin: 1.95V	Table 7 of the Magnetometer Datasheet states that the minimum operating voltage is 1.95V [8]

1.6. Bills of Materials

Table 3 lists the bills of materials used for the Sensor Controller block.

TABLE III. BILLS OF MATERIALS FOR THE MAGNETOMETER BLOCK

Item	Where to buy	Manufacturer	Manufacturer Part Number	Price	Quantity
Triple Axis Magnetometer – MAG3110	Digikey	SparkFun Electronics	SEN-12670	$14.95	2

VII. References

[1] SH_BT_Board Datahseet, <http://www.ram-e-shop.com/ds/general/Bluetooth_TRx_Module_New.pdf>

[2] HC Serial Bluetooth Instruction Manual, <https://cdn.makezine.com/uploads/2014/03/hc_hc-05-user-instructions-bluetooth.pdf>

[3] Getting Started with the Arduino Pro Mini, <https://www.arduino.cc/en/Guide/ArduinoProMini#toc2>

[4] Atmega328 Datasheet, <http://www.atmel.com/images/Atmel-8271-8-bit-AVR-Microcontroller-ATmega48A-48PA-88A-88PA-168A-168PA-328-328P_datasheet_Complete.pdf>

[5] HC-05 Bluetooth Module Breakout board Page1 <http://e-gizmo.com/oc/kits%20documents/HC-05%20Bluetooth%20Module%20breakoutboard/HC05%20Bluetooth%20module%20breakout%20board.pdf>

[6] Serial Port Bluetooth Module HC-05 <https://www.itead.cc/wiki/Serial_Port_Bluetooth_Module_(Master/Slave)_:_HC-05>

[7] http://www.vibration-motor.com/products/download/C0720B001F.pdf

[8] ST Microelectronics, “World smallest Time-of-Flight ranging and gesture detection sensor,” VL53L0X production datasheet, May 2016

[9] Adafruit and T. Dicola, “Adafruit VL53L0X Time of Flight Micro-LIDAR Distance Sensor Breakout,” Adafruit VL53L0X Time of Flight Micro-LIDAR Distance Sensor Breakout, 23-Nov-2016. [Online]. Available: https://learn.adafruit.com/adafruit-vl53l0x-micro-lidar-distance-sensor-breakout/overview. [Accessed: 03-Dec-2017].

[10] Atmega328 Datasheet, <http://www.atmel.com/images/Atmel-8271-8-bit-AVR-Microcontroller-ATmega48A-48PA-88A-88PA-168A-168PA-328-328P_datasheet_Complete.pdf>

[11] Arduino Pro Mini, <https://store.arduino.cc/usa/arduino-pro-mini>

[12] Getting Started with the Arduino Pro Mini, <https://www.arduino.cc/en/Guide/ArduinoProMini#toc2>

[13] Adruino, “JSN-SR04T 20-60 0cm ultrasonic waterproof Range Finder ” JSN-SR04T Datasheet, https://drive.google.com/a/oregonstate.edu/file/d/1mWfcWH5f5-8lk2r7KeLZ4dzk0CqiVCk0/view?usp=sharing

[14] Electrosome, “Interfacing HC-SR04 Ultrasonic sensor with PIC Microcontroller,” https://electrosome.com/hc-sr04-ultrasonic-sensor-pic/, accessed: 12/2/2017

VIII. REVISION HISTORY

TABLE XXXVIII. REVISION HISTORY

Date	Section	Editor	Description of Change
10/6/17	Project Summary	Sean Sylwester	Draft created for peer review
10/6/17	Introduction	Sean Sylwester	Draft created for peer review
10/13/17	Engineering Requirements	Sean Sylwester	Draft created for peer review
10/19/17	Methodology Review and Selection	Sean Sylwester	Draft created for peer review
10/20/17	Project Timeline	Sean Sylwester	Draft created for peer review
10/23/17	Block Diagram and Interfaces	Sean Sylwester	Draft created for peer review
11/2/17	Abstract	Sean Sylwester	Draft created for peer review
11/4/17	Revision History	Sean Sylwester	Created
11/4/17	Introduction	Sean Sylwester	Added project title, and reworded some sentences.
11/4/17	Project Summary	Sean Sylwester	Removed unnecessary background information, reworded some sentences, and added more technical design details.
11/4/17	Engineering Requirements	Sean Sylwester	Removed some unnecessary requirements, reworded some, and added testing procedures for each of the 10 graded requirements.
11/4/17	Methodology Review and Selection	Sean Sylwester	Added project title, reworded some sentences, and added more analysis.
11/4/17	Block Diagram and Interfaces	Sean Sylwester	Completely reworked the block diagram layout and presentation, added a microcontroller code block, and combined the two regulator blocks into one.
12/3/17	Methodology Review and Selection	Sean Sylwester	Magnetometer and Bluetooth comparison tables added.
12/3/17	Block Diagram and Interfaces	Sean Sylwester	Block diagram and interfaces updated
12/3/17	All sections	Sean Sylwester	Writing updated based on peer and instructor feedback
12/3/17	Process Memo	Sean Sylwester	Section added
1/12/18	Block Diagram	Samuel Lee	Updated block diagram to fix misunderstanding of "controller code" and "microcontroller"
1/21/18	Laser ToF Sensor Block Description	Sean Sylwester	Updated some wording, updated the interfaces and properties due to a change in our block diagram
1/21/18	Ultrasonic Sensor Block Description	Sean Sylwester	Section Created
1/21/18	PCB Block Description	Sean Sylwester	Section Created
1/21/18	Bluetooth Block Description	Samuel Lee	Updated Bluetooth block diagram with new interfaces names
1/21/18	All sections	Samuel Lee	Updated Bluetooth, Controller Code, and Sensor Controller block descriptions drafts
1/21/18	Revision History	Samuel Lee	Revision History Updated
1/21/18	Process Memo	Samuel Lee	Section Added
2/6/28	Block Diagram	Samuel Lee	Updated block diagram for Bluetooth, Controller Code, and Sensor Controller block description
2/7/18	Laser ToF Sensor Block Description	Sean Sylwester	BoM, and design and validation sections updated.
2/7/18	Ultrasonic Sensor Block Description	Sean Sylwester	BoM, and validation section updated
2/7/18	PCB Block Description	Sean Sylwester	BoM updated and figure background colors changed from black to white
2/7/18	Haptic Feedback Block	Jacy Barr	Reformatted everything into IEEE, included information about multiples of the same circuit, added BOM, updated interfaces, fixed wiring diagram
2/7/18	Project Management Block	Jacy Barr	Reformatted everything into IEEE, improved project narrative for specificity, altered communication protocols to table format
2/7/18	Enclosure Block	Jacy Barr	Reformatted everything into IEEE, included CAD models instead of drawings, redefined 'interfaces', improved testing procedures, updated/included BOM, added dimensions to components
3/14/18	Power Supply Block	Sean Sylwester	Block added
3/14/18	Sensor Harness, Feedback Harness, Magnetometer Block	Jacy Barr	Block added
3/14/18	Feedback Controller Block	Samuel Lee	Block added
3/14/18	PCB Block	Sean Sylwester	Updated block with new schematic, layout, and BOM.
3/14/18	Haptic Feedback Block	Jacy Barr	Updated with PCB Layout
3/23/18	PCB, Enclosure, Power Supply, and Haptic Feedback Block	All	Changed PCB and enclosure layouts to better accommodate connectors, and to clip onto user clothing instead of hanging off the bands. Changed from AA batteries to rechargeable Li-Ion batteries, with a charging circuit included (improves usability). Changed to a smaller haptic feedback motor and added a metal pad to make the vibrations more localized and intense.
4/6/18	Introduction, Project Summary, Timeline, Requirements	All	Updated the introduction, project summary, and timeline to reflect the previously described system changes. Updated engineering requirements to make them attainable and testable.

IX. Process Memo

Based on the comments received from our peers and instructors, we implemented changes to our project summary to remove unnecessary background information and add more technical information that would benefit our stakeholder and fellow engineering students looking to reproduce our work. In our initial submission of the project specification document, we had forgotten to include our project timeline, customer requirements, and the non-graded requirements. Updating our document with these inclusions provided clarity to the reasons behind our initial engineering requirements. To further develop our requirements, we added testing procedures for our requirements and the customer requirements that prompted them.

Following our submission of our first project specification document for peer review, the final engineering requirements meeting with Don informed us that our overall design lacked complexity for our group size. To improve our project, we brainstormed additional features that when implemented would be suitable for our four-man team. Such drastic changes prompted several new components to be researched, compared, analyzed, and described in the final project specification document. As a group, we held several meetings to work on combining low weighted engineering requirements and producing new ones that would define our revised project additions while meeting our customer requirements.

To maintain 12 blocks for our team, the block diagram was completely reworked, combining the harness/ enclosure, battery/ regulators, and removed the thermal sensor blocks. This allowed for the addition of Bluetooth modules, magnetometer, and laser sensors. Interfaces and their respective properties were then defined. The methodology research and selection section of our project specification document now lacking information on our new Bluetooth, ToF lasers, and magnetometer components, we completed the necessary work to provide adequate documentation. Finally, individual block descriptions and a process memo were added.

For this term’s block checkoff, Jacy Barr will be demonstrating the haptic feedback block, Samuel Lee will be demonstrating the Bluetooth module block, Alwin Sudhana will be demonstrating the ultrasonic sensor, and Sean Sylwester will be demonstrating the ToF. laser sensor.