Spring 2017

Augmented Reality Helmet

Kevin Kapko
The University of Akron, kmk91@zips.uakron.edu

Augmented-Reality Helmet

Final Design Report

Group 15

Bradley, Azia
DeLonais, John
Hromco, Brian
Kapko, Kevin

Dr. Forrest Sheng Bao

12-7-2016

Table of Contents

List of Figures

List of Tables

Abstract

Cyclists need a detection method for vehicles to help prevent accidents and a method to display exercise data in order to maximize their workout. An array of sensors paired with a display allows a cyclist to be notified of workout information and any surrounding traffic. A smartphone app calculates the information from the sensors and sends the calculations to the display. This will allow the user to be able to see their exercise information without disrupting their bike ride as well as give some measure of safety to the cyclist when sharing the street with cars.

- Real-time exercise information display
- Field of vision can remain in front of the cyclist
- Incoming car detection

1. Problem Statement

Need

Twelve percent of cyclist injuries are due to being hit by cars [6]. There needs to be a safer method for bike riders to exercise on roads with vehicles while still attaining the best workout experience. Knowledge of a person's heart rate and calories burned while working out can be utilized in order to help optimize a workout while exercising on a bicycle. In addition, an assortment of sensors can help record all pertinent information during a workout. However, this does not allow a user to adjust his routine in the middle of a workout. A helmet that can communicate with the sensors and present the information to the user via a display will help maximize any bike-riding exercise and also create a safer experience for the rider.

Objective

The Augmented-Reality Helmet (ARH) will be modeled after a bicycle helmet and will display information collected from an array of sensors. An app will be developed that can communicate with the helmet. The app will collect user inputs for customizing the workout and send the information wirelessly to the helmet. The helmet will incorporate three major components: a display which will be integrated into the helmet; a wireless module incorporated into a microcontroller that will communicate with the app and send information to the display in real-time; a power system which will run all of the electronics in the helmet for an extended period of time.

Research Survey

Image processing is needed for ARH to understand how to improve the quality of a digitized image. According to U.S. Patent 9235770 [1]**,** An image processing device is provided, the image processing device comprising: an image input unit configured to be input with a frame image of an imaging area imaged by a camera; an image processing unit configured to process the frame image input to the image input unit, and detect an object imaged in the frame image; and an operation frequency determination unit configured to determine a frequency of an operation clock of the image processing unit according to the number of objects detected by the image processing unit, wherein the operation frequency determination unit lowers the frequency of the operation clock of the image processing unit as the number of objects detected by the image processing unit becomes smaller.

U.S. Patent 8364389 [2] describes a system by which sensors attached to a bicycle can feed information to a series of head-mounted displays. This system, when worn, allows an individual riding a bicycle to see pertinent details about the individual's trip. A group of bicyclists riding together, when equipped with this system, can see information about all the people and bicycles in the group.

The ARH helmet will need to display a variety of information in a manner similar to that of the system described in the patent. ARH must also allow the wearer to maintain a relatively clear field of vision while riding a bicycle so as to allow for the wearer's safety.

U.S. Patent 8964298 [3] describes a system wherein a user with a smartphone wears both a wristband and a custom pair of glasses. The wristband communicates with the phone and the glasses to allow for a gesture-based interface with the augmented reality displayed on the glasses. The phone's internet connection and GPS allow for the glasses to display an augmented reality overlay for the user. Together, the phone, wristband, and glasses allow for a hands-free augmented reality experience for the user of this system.

The communication network described in this patent will be very similar to that of ARH. Both systems require a network between a head-mounted display, a phone, and a wristband which will communicate with the phone. The communication of the network must be in real-time or else the system will not operate as intended.

Many displays that match the specifications needed for the ARH require a voltage higher than what can be supplied by batteries. As such, the voltage supplied from a battery must be amplified in order to satisfy the needs of the display. A boost DC-DC converter makes the amplification possible while keeping the current steady. [4]

Multiple sensors are needed to collect data for ARH. In order to detect car distance within a close range to the user's bike, an ultrasonic proximity sensor must be used. Radiation patterns must be understood when evaluating possible sensors. This article gives an overview of some fundamental acoustical parameters that affect the performance of an ultrasonic sensor. When selecting an ultrasonic sensor for a particular application, it is important to consider how the echo will be affected by acoustical fundamentals. This is important due to there being a wide range of sensors available that operate at different frequencies and have different beam angles. Parameters to consider include variations in frequency of sensors, resolution, relative accuracy, target range measurement, and the effective beam angle. [5]

Marketing Requirements

1. Lightweight and stay securely on head.
2. The display will have minimal vision impairment and little display clutter.
3. Low temperature power supply
4. Fast and accurate data streaming
5. Long duration
6. Quick and easy to set up
7. Measure heart rate, speed, distance traveled, and calories burned.
8. Detect traffic approaching from behind the user.

Objective Tree

The four main categories of concern for the ARH can be seen in the Objective Tree in Figure 1.

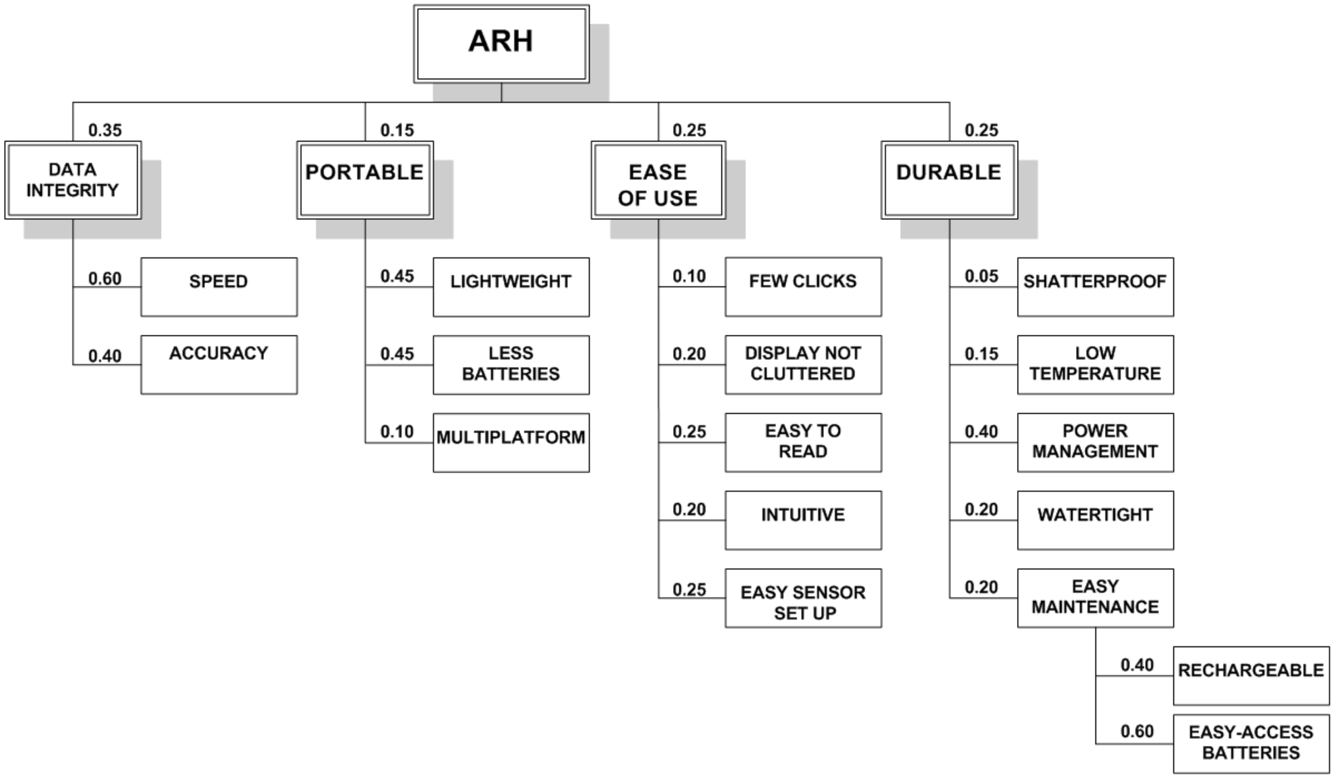


Figure 1 – Objective Tree

Data Integrity ranked highest in terms of the design. Speed prioritized accuracy only slightly since a slow frame throughput can make incoming objects appear farther away than they are in reality, which can create accidents.

Durability and ease of use merited equal importance. The durability section requires the ARH to be operable in any situation in which the user may be riding his bicycle. Several subcategories (temperature, watertight, shatterproof) ensure weather-proofing for mild-weather biking. The remaining categories (power management, easy maintenance) allow the system to be operable throughout the duration of the bike ride.

Ease of use had fairly equal weight throughout the subcategories. Fewer clicks carried little weight because the system should be automated after some minor user customization.

Portability carried the least weight because it had the least risk of bringing harm to the user. The subcategories for portability ensure that the user loses minimal amount of mobility when equipped with the ARH, and that the system can be used by a larger audience.

2. Design Requirements Specification

Marketing Specifications	Engineering Requirement	Justification
5	Functional for at least 1 hour when fully charged.	Excluding rigorous training, an average cardio works lasts 20-60 minutes in order to improve fitness. The system should be operable throughout the entirety of the workout.
4	Display video feed at a rate of no less than 30 frames per second.	When fps drops below that threshold, disorientation can occur which can cause motion sickness. In addition, reaction time may be affected since the sequence of events become garbled.
4	Video latency must not be more than 50ms	Latency exceeding the threshold can endanger the user since the scene the user sees differs from the scene in reality. A pole that appears far from the user can actually be much closer, which can cause accidents.
2	Display exercise information in a manner that does not obstruct more than 20% of the display.	If an excess of the cyclist's vision becomes obscured, he can no longer see obstacles that may appear in his path, which will endanger the cyclist.
8	Detect oncoming vehicles at a range of at least 10 meters and visibly alert the user of the ARH within 250ms.	When traveling on a street with vehicular traffic, the amount of cars that cannot easily be heard by a cyclist increases daily. The threshold gives the rider approximately 1 second notice before a vehicle traveling 35 miles per hour, entering the detection range pulls up beside the cyclist.
4, 7	Gather data from sensors at least once every 3 seconds.	Too much of a time difference will reduce the accuracy of the data displayed to the user. Too small of a time difference will increase power consumption from signal communication and more frequent calculations.

1	Remains on the head when coming to a stop from 16 km/h over 1 seconds across 7 meters	Represents the average cycling speed coming to a full force stop. If the system disengages from the head, it risks vision impairment and possible entanglement of the rider.
3	Does not exceed temperatures of 40°C while at full operation	Many devices have an optimal temperature for operation. When operated outside of the temperature, the device can become damage and give possibly faulty readings. In addition, having a hot system close to the body can cause discomfiture for the user.
1	Helmet of the ARH must not exceed more than 3 kg	Excessive weight on the head can cause unnecessary strain on the neck, which can lead to increased chance of injury and possible health conditions.
6	Attaching sensors to bike should take no more than 5 minutes.	Long setup time can discourage use of a product.

Marketing Specifications

1. Lightweight and stay securely on head.
2. The display will have minimal vision impairment and little display clutter.
3. Low temperature power supply
4. Fast and accurate data streaming
5. Long duration
6. Quick and easy to set up
7. Measure heart rate, speed, distance traveled, and calories burned.
8. Detect traffic approaching from behind the user.

Table 1 – Design Requirements

3. Accepted Technical Design

<u>Block Level 0</u>

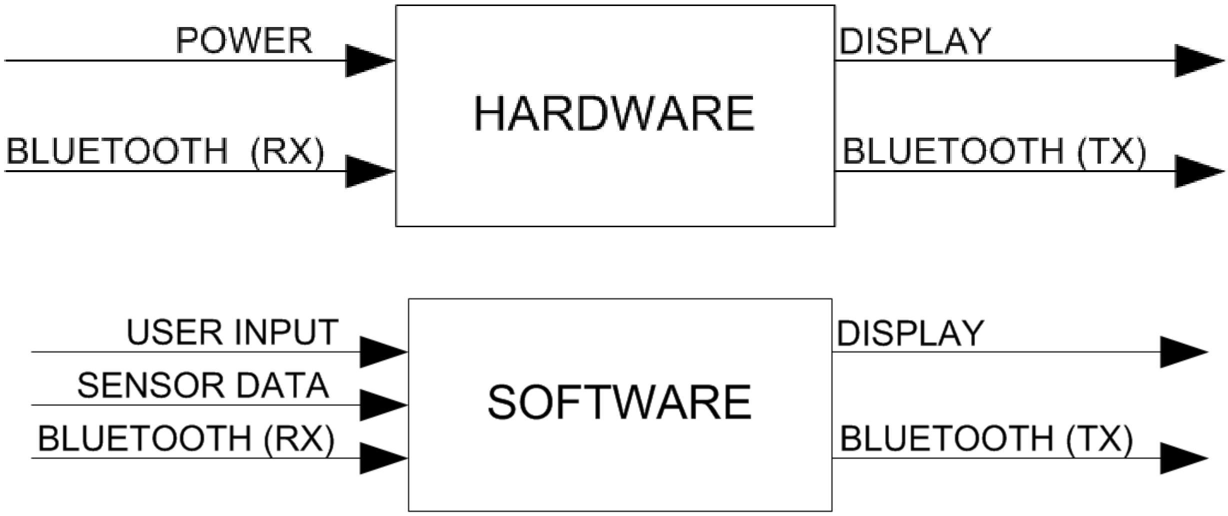


Figure 2 – Level 0 Block Diagram

Theory of Operation: The hardware level 0 diagram for the ARH shows the basic inputs and outputs. One input is power; which would be a DC power supply. The other input for hardware will be the receiving wireless signal into the micro controller. The output will be a display and a wireless signal transmitting back to the phone with exercise summary information.

Module	Hardware
Designer	N/A
Inputs	• Power • Wifi Signal (Rx)
Outputs	• Display • Wifi Signal (Tx)
Functionality	Wirelessly communicates with Mobile device and displays exercise information to the user.

Table 2 – Level 0 Hardware FR Table

Module	Software
Designer	N/A
Inputs	• User Input • Sensor Data • Wifi Signal (Rx)
Outputs	• Display Data • Wifi Signal (Tx)
Functionality	Calculates biking information using sensor data and user input. Transmits the information to the microcontroller. Masks exercise information over incoming video frames and sends the modified video frames to a headset display.

Table 3 – Level 0 Software FR Table

Block Level 1

Hardware

Theory of Operation: The following figures show a further level of detail than the previous figures. It shows a breakdown of sub modules and sub system connections within the multiple system modules. This level 1 diagram is showing a breakdown of sub modules. The corresponding Functional Requirement Tables follow each illustration. Figure 3 shows the level one block diagram. The power supply is going to be distributed to two of the sub modules shown that include the microcontroller 1 and microcontroller 2. The sensor block is a group of sensors that will collect data and send it to microcontroller 2. Microcontroller 2 will then send the data via Bluetooth to the phone. The phone app will then transmit data to the microcontroller 1. A video feed will output from the microcontroller 1 and onto the headset display along with feedback to the phone.

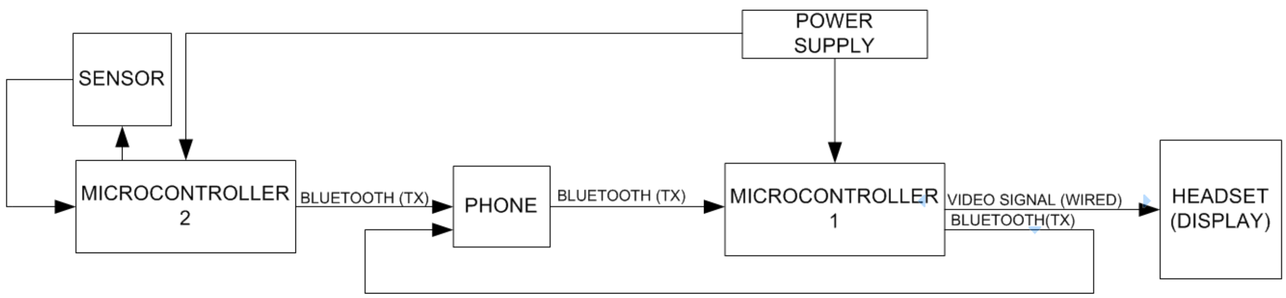


Figure 3 – Level 1 Hardware Block Diagram

Module	Microcontroller 1
Designer	Azia
Input(s)	• Power Supply (DC Supply Battery Pack) • Data Tx by Phone
Output(s)	• Video Signal • Data Tx to Phone
Functionality	The microcontroller will receive sensor information from the phone App. The microcontroller then displays the information to the headset and sends feedback to phone.

Table 4 – Level 1 Hardware: Microcontroller 1 FR

Module	Phone
Designer	N/A
Input(s)	• Sensor Data • Data Tx by microcontroller 1
Output(s)	• Data Tx to microcontroller 1
Functionality	The phone will receive sensor data and then transmit wirelessly data to the Microcontroller 1.

Table 5 – Level 1 Hardware: Phone FR

Module	Microcontroller 2
Designer	Brian
Input(s)	• Power Supply (DC Supply Battery Pack) • Data wired by sensors
Output(s)	• Data Tx to Phone
Functionality	The microcontroller 2 will be wired up to sensors. It will then take the data and transmit the data to the phone via Bluetooth.

Table 6 – Level 1 Hardware: Microcontrolelr 2 FR

Software

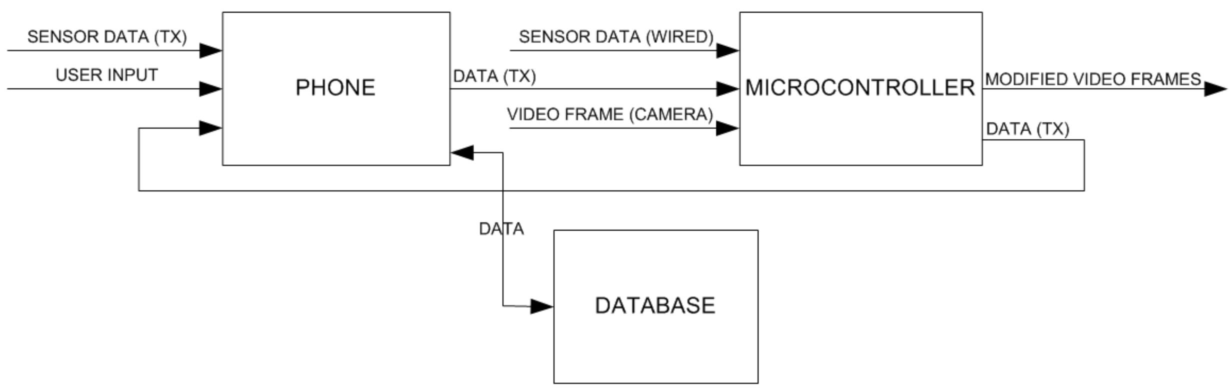


Figure 4 – Level 1 Software Block Diagram

Theory of Operation: The user will input relevant data into the phone application, such as their age, their weight, the weight of their bike, and the diameter of their bicycle tires. This information is then used to calculate various metrics, which are then wirelessly sent to microcontroller 1, which keeps track of the exercise and sends information to the display for the user. At the end of the exercise, microcontroller 1 will send the results of the exercise back to the phone, which then forwards it to a database to keep a record of prior exercises.

Module	Phone
Designer	N/A
Input(s)	● Sensor Data ● User Input ● Wireless Signal (Rx)
Output(s)	● Wireless Signal (Tx) (Microcontroller 1) ● Wireless Signal (Tx) (Database)
Functionality	The phone app will adjust exercise calculations based off of user input. Exercise information will then be generated based off of exercise calculations. Communicates with a database to store and retrieve exercise session information.

Table 7 – Level 1 Software: Phone FR

Module	Microcontroller 1
Designer	N/A
Input(s)	• Video frames • Wireless Signal (Rx) • Ultrasonic Sensor
Output(s)	• Modified Video Frames • Wireless Signal (Tx)
Functionality	Applies a mask of the exercise information from wireless signal to incoming video frames. Also masks a warning to video frame based off of ultrasonic sensor. Sends out augmented frames as display data to headset display.

Table 8 – Level 1 Software: Microcontroller FR

Module	Database
Designer	N/A
Input(s)	• Wireless Signal (Rx)
Output(s)	• Wireless Signal (Tx)
Functionality	Communicates with phone app to store and retrieve past exercise session information.

Table 9 – Level 1 Software: Database FR

Block Level 2

Hardware

Theory of Operation: The level two block diagram goes into the broadest details of the hardware design. The DC supplied battery pack will power the microcontroller 1, microcontroller 2, heart rate, and gyroscope. The Hall Effect and ultrasonic will be powered by the microcontroller 2. The sensors block includes several different types of sensors: the heart rate sensor, which also has the accelerometer and gyroscope built into it, and will measure the user's heart rate; accelerometer will determine the user’s acceleration. The gyroscope will be used to determine the rider’s orientation; the Hall Effect sensor is used to detect magnetic fields. It will be placed on the magnetic bike rim. One rotation would be equivalent to the Hall Effect sum/number of magnets. The Hall Effect sensor will be used to calculate the distance traveled and the speed. Car detection will be approximated using the ultrasonic proximity sensor. The camera will feed into the camera port of the microcontroller and provide video frames for the headset display.

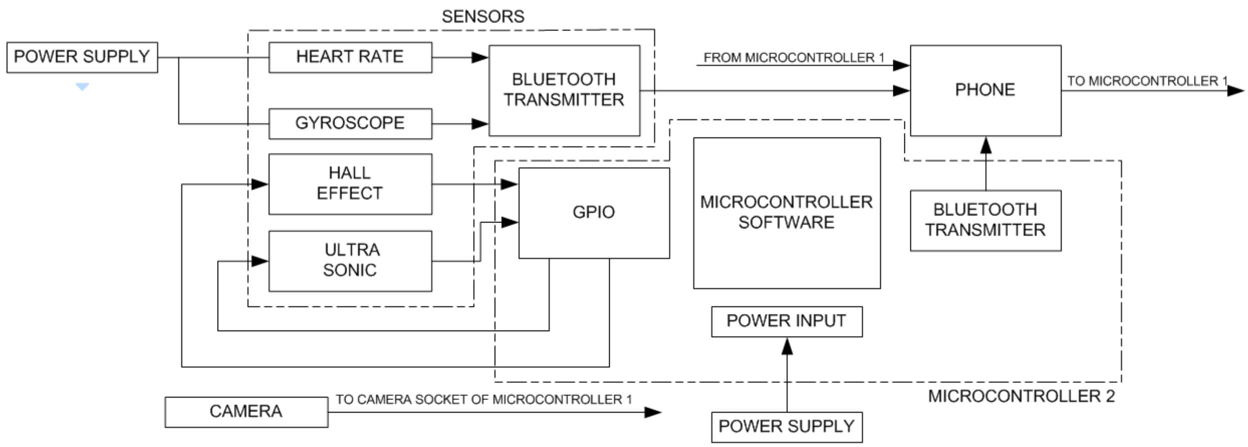


Figure 5 – Level 2 Hardware Block Diagram: Sensors and Microcontroller 2

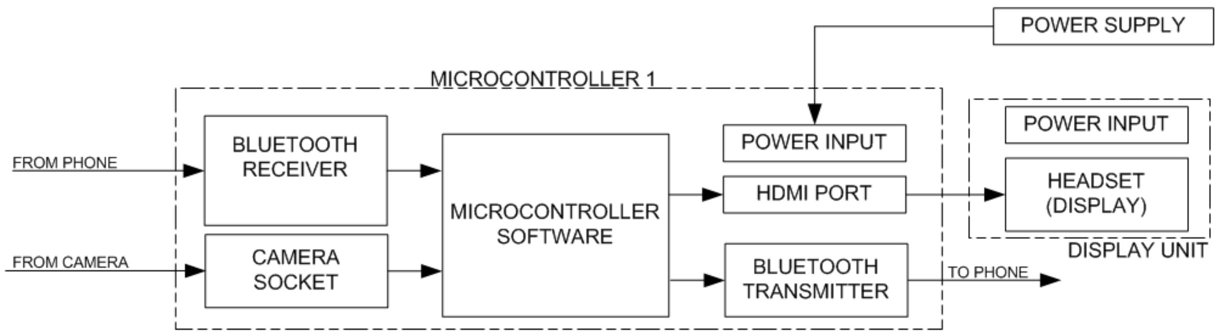


Figure 6 – Level 2 Hardware Block Diagram: Microcontroller 1

Module	Sensor Block
Designer	Brian
Input(s)	• Power supply from microcontroller 2 (Hall Effect & Ultrasonic) • Power supply (Heart Rate)
Output(s)	• Data wired to microcontroller 2 • Data Tx to phone
Functionality	The heart rate sensor will collect the user’s heart rate and speed. Car detection will be detected from the ultrasonic proximity sensor. The Hall Effect sensor determines the RPM.

Table 10 – Level 2 Hardware: Sensor Block FR

Module	Phone
Designer	N/A
Input(s)	• Data Tx from sensors • Data Tx from microcontroller 1 • Data Tx from microcontroller 2
Output(s)	• Data Tx to microcontroller 1
Functionality	The phone will receive sensor data and then the app can adjust exercise calculations based off of user input. Exercise information will then be generated based off of exercise calculations. Communicates with a database to store and retrieve exercise session information.

Table 11 – Level 2 Hardware: Phone FR

Module	Microcontroller 1
Designer	Azia
Input(s)	• Power Supply • Camera • Data Tx from phone
Output(s)	• Headset Display Unit • Data Tx from microcontroller 1
Functionality	The microcontroller 1 will receive the sensor information from the phone App and then display the information on the headset. The camera video feed that will be displayed to the user will connect to the display through the HDMI port on the microcontroller 1.

Table 12 – Level 2 Hardware: Microcontroller 1 FR

Module	Headset/Display
Designer	Azia
Input(s)	• Video HDMI Signal • Power Supply
Output(s)	• Displayed exercise information • Display camera video feed
Functionality	The headset will show video of what is in front of user and simultaneously show the selected exercise information from the phone app.

Table 13 – Level 2 Hardware: Headset/Display FR

Module	Microcontroller 2
Designer	Brian
Input(s)	• Power Supply • Data wired by sensors
Output(s)	• Headset Display Unit • Data Tx from microcontroller 1
Functionality	The microcontroller 1 will receive the sensor information from the phone App and then display the information on the headset. The camera video feed that will be displayed to the user will connect to the display through the HDMI port on the microcontroller 1.

Table 14 – Level 2 Hardware: Microcontroller 2 FR

Software

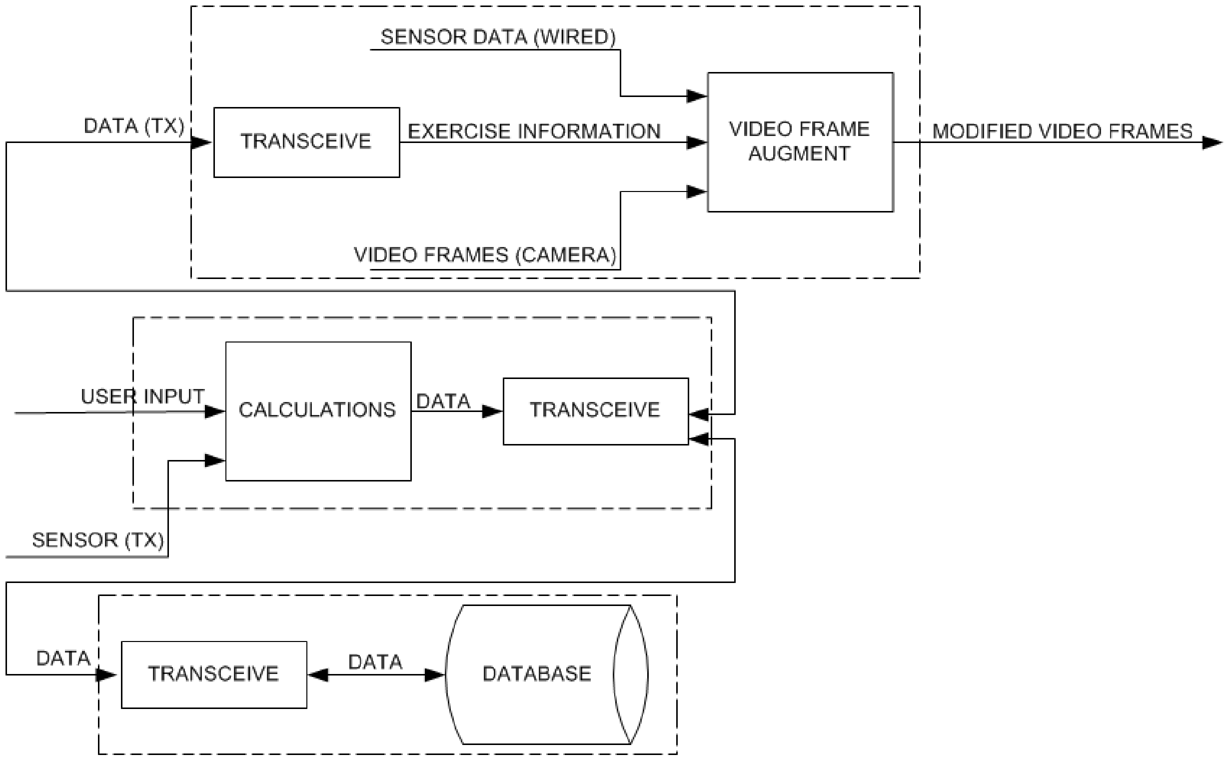


Figure 7 – Level 2 Software Block Diagram

Theory of Operation: The user input from the phone app will be inserted into the above calculations which will output the exercise information. This information will be packed and transmitted via a phone transceiver module to the waiting transceiver of microcontroller 1. Microcontroller 1's transceiver module will receive this data and will then mask the information as well as the ultrasonic sensor warning on top of incoming video frames. These frames are then sent to the display.

Once a session is determined to be complete, the session information will be sent from the microcontroller transceiver to the phone transceiver. The phone transceiver will then send the session information to the database transceiver, which will in turn store the session information in the database. Past session information can be pulled from the database and sent to the phone for reviewing.

Several different types of equations are implemented in the calculations block in order to compute the exercise information. The first equation, used to calculate the amount of tire rotations, is:

$$\gamma = \frac{Hall\ \ Effect\ Sum}{\#\ of\ magnets}$$

The 'Hall Effect Sum' representing the number of times that the Hall Effect sensor gives an active high reading, the '# of magnets' representing the magnets placed on the rim of the tire, and γ representing the total number of tire rotations. The number of magnets remains a variable since the user will input how many magnets they have placed on the tire. γ is then used in the calculation for the total distance that the user has travelled:

$$d = (2 * \pi * r_{tire}) * \gamma$$

The inside of the parentheses calculates the circumference of the tire, where r_{tire} is the radius of the tire as indicated by the user during user customization and d is the total distance travelled. The circumference of the tire is then multiplied by the number of times that the tired rotated in order to calculate the total distance that the bike has travelled since data started being transmitted.

$$s = \frac{d}{t}$$

d is the distance calculated previously, t is the time that has elapsed, and s is the current speed of the cyclist. The next equation does not represent any exercise information that will be displayed directly, but is used in the calculations in the final equation of the calculation block. This is the equation to compute the vertical distance traveled:

$$d_y = d * \sin(\theta)$$

Where d is the distance travelled and θ is the angular orientation in respect with the x-z plane. This equation works under the assumption that the distance that the user travels is not solely horizontal and vertical, but a combination of the two vectors. Thus, in order to find the y distance from the distance that is a combination of x and y, the sin function is used with the angular orientation of the rider which is obtained from the gyroscope. The final equation used in the calculations block (Table 15) is the total number of calories burned:

$$c_e = \frac{(3.509 + 0.2581s^3) * t}{4186.8} + \frac{m_{rider} * g * d_y}{418}, where\ g = 9.8\,{}^{m}/_{s^2}$$

Where s is the speed of the cyclist, t is the elapsed time, m_{rider} is the mass of the rider entered during customization, g is the acceleration of gravity which will be implemented as a constant, d_y is the vertical distance that has been travelled, and c_e is the amount of calories expended.

After all of these calculations have been made, this information will go to a transceiver block, represented in Table 16, Table 17, and Table 18. All of the transceiver blocks are functionally the same and either: concatenate pertinent information to be sent out; parse incoming information into a usable format. How these two methods are carried out depends specifically on the transceiver block. The database block, Table 18, handles all information transactions between the phone and the remote database. Exercise information will either be stored or retrieved from the database depending on incoming signal. This signal will either request a storage of information or a retrieval of information.

The final block, Table 20, handles the actual creation of the frames which will be displayed to the user. The first calculation required for displaying the information is calculating how to mask the exercise information onto the video frames. The exercise information will need to be large enough that the user can read the data, but not so large that the field of view of the user becomes obstructed. This can be realized by making the text seem to appear a set distance away from the user. This requires transposing 3-dimensional coordinates onto a 2-dimensional plane. This technique is called foreshortening, and is represented by the equations:

$$x_p = \frac{x}{{}^{z}/_{d} + 1}, y_p = \frac{y}{{}^{z}/_{d} + 1}$$

x and y represent the actual x- and y-coordinates of the object, z is the distance from the display to the object, d is the distance from the viewer to the display, and x_p and y_p represent the adjusted x- and y-coordinates that are used on the display. This technique of foreshortening allows the masked exercise information to appear at any chosen distance, and will set exactly which pixels need to be manipulated in order to mask the information. After these calculations have been made to determine where the masked exercise and warning information will appear, the images that arrive from the camera need to be converted into a type of data that will allow easier masking of the exercise information. All frames received by the block will need to be transformed from image data into an array in order to allow the data to be manipulated. The pixel locations that need to be changed will then be altered and effectively mask the exercise information onto the video frame. If an incoming object is detected, the warning information will also be masked onto the video frame at this point. Once the masking has finished, the array of pixel data will then be converted back into an image and be sent out to the display.

Module	Calculations
Designer	N/A
Input(s)	• User Input • Sensor Data • Exercise Equations
Output(s)	• Exercise Information
Functionality	Calculates exercise information using sensor data and modified exercise equations.

Table 15 – Level 2 Software: Calculations FR

Module	Transceive (Phone)
Designer	N/A
Input(s)	• Exercise Information (Calc) • Wireless Signal (Rx)
Output(s)	• Wireless Signal (Tx) • Data packet (Database)
Functionality	Receives and unpacks end-of-exercise information from microcontroller, then transmits it to database. Packs exercise information and transmits it to microcontroller.

Table 16 – Level 2 Software: Transceive (Phone) FR

Module	Transceive (Database)
Designer	N/A
Input(s)	• Exercise Session Information (Database) • Data packet
Output(s)	• Data packet • Exercise Session Information (Phone)
Functionality	Sends and receives data packets between the phone app and database.

Table 17 – Level 2 Software: Transceive (Database) FR

Module	Database
Designer	N/A
Input(s)	• Exercise Session Information (Phone) • Wireless Signal (Rx)
Output(s)	• Wireless Signal (Tx) • Exercise Session Information (Database)
Functionality	

Table 18 – Level 2 Software: Database FR

Module	Transceive (uC)
Designer	N/A
Input(s)	• Exercise Information (end-of-exercise) • Wireless Signal (Rx)
Output(s)	• Wireless Signal (Tx) • Exercise Information (Augment)
Functionality	Communicates with phone app to store and retrieve past exercise session information.

Table 19 – Level 2 Software: Transceive (uC) FR

Module	Video Frame Augmentation
Designer	N/A
Input(s)	• Exercise Information • Video Frames • Ultrasonic sensor
Output(s)	• Modified Video Frames
Functionality	Applies mask of exercise information to video frames. If necessary, also applies mask of incoming object warning to video frames as well.

Table 20 – Level 2 Software: Video Frame Augmentation FR

Block Level 3

Hardware

RASPBERRY PI 3

The Raspberry Pi 3 B+ will be powered by a 5 VDC at 2.5A. The power will be connected to the GPIO pins 5 and 6. The Bluetooth transceiver that is located on the Raspberry Pi will be able to receive exercise and safety data from the phone along with send data back to the phone. A Raspberry Pi Camera Board will be connected to the camera socket on the Raspberry Pi. The camera will be able to stream live events of what is in front of the bike rider. Camera feed will then be inputted into the Raspberry Pi so that the frames can be processed and exercise and safety information from the Bluetooth can be displayed. Once the Camera feed is process by the Raspberry Pi, it will output the live feed onto the MyBud that the bike rider is wearing. The Raspberry Pi will then have an HDMI cable to output the display to the MyBud glasses. Figure 1Figure 8 below shows the layout of the Raspberry Pi 3 B+ inputs and outputs.

The Raspberry Pi 3 B+ will be used for Microcontroller 1. The Raspberry Pi 3 B+ will be powered with 5 VDC at 2.5A from BP1. The power will be connected to the GPIO pins 5 and 6. The Bluetooth transmitter that is located on the Raspberry Pi will be able to receive exercise and safety data from the phone along with send data back to the phone. A Raspberry Pi Camera Board will be connected to the camera socket on the Raspberry Pi. The camera will be able to stream live events of what is in front of the bike rider. Camera feed will then be inputted into the Raspberry Pi so that the frames can be processed and exercise and safety information from the Bluetooth can be displayed. Once the Camera feed is process by the Raspberry Pi, it will output the live feed onto the MyBud that the bike rider is wearing. The Raspberry Pi will then have an HDMI cable to output the display to the MyBud glasses. Figure 8 below is the schematic repetition of Microcontroller 1.

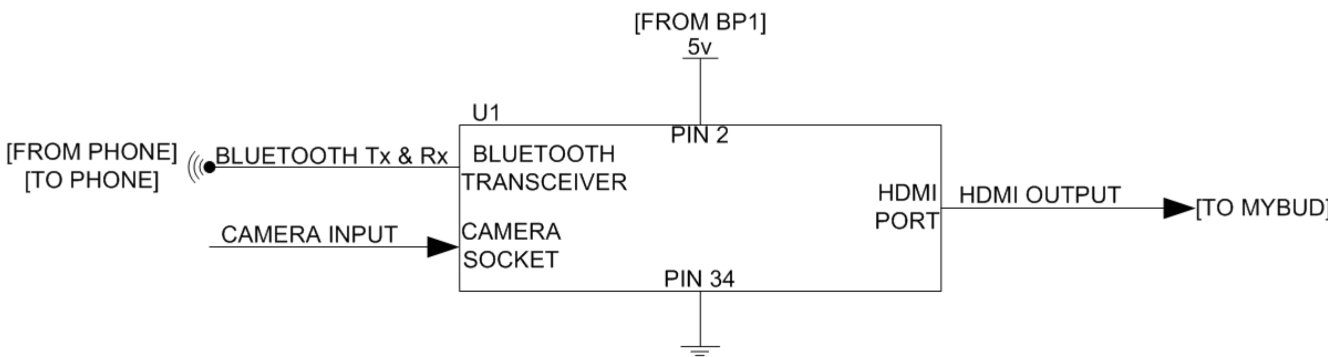


Figure 8 – Level 3 Hardware Block Diagram of Microcontroller 1

Module	[U1] Raspberry Pi 3 B+ (Microcontroller 1)
Designer	Azia
Input(s)	• 5V DC at 2.5A from Battery pack 1 (BP1) • Video data from Raspberry Pi camera board
Output(s)	• Video Signal to MyBud • Data TX to Phone Via Bluetooth
Functionality	The microcontroller will receive sensor information from the phone App. The microcontroller then displays the information to the headset and sends feedback to phone.

Table 21 – Level 3 Hardware: Raspberry Pi 3 B+ (Microcontroller 1) FR

The Microcontroller 2 will also be using a Raspberry Pi 3 B+. With the Raspberry Pi 3 B+ having 40 GPIO pins on it give it the ability to give power to the Hall Effect sensor and the ultrasonic sensor and receive the data from the sensor as well. The Raspberry Pi 3 B+ can also send that data and send it via Bluetooth with its on board Bluetooth transmitter. Table 22 gives a summary of all its connections. [U2] Raspberry Pi 3 B+ (Microcontroller 2) schematic can be found in Figure 9 and Figure 10.

Module	[U2] Raspberry Pi 3 B+ (Microcontroller 2)
Designer	Brian
Input(s)	• 5V DC at 2.5A from Battery pack 2 (BP2) • (GPIO) Hall Effect sensor data • (PWM) Ultrasonic sensor data
Output(s)	• Data Tx to Phone Via Bluetooth • (GPIO) 5VDC to Hall Effect sensor • (GPIO) 5VDC to Ultrasonic sensor
Functionality	The Microcontroller 2 will supply 5 VDC to the Hall Effect and ultrasonic sensor. It will take in the data from the sensors and using the Bluetooth transmitter on the Raspberry Pi, send the sensor data to the phone.

Table 22 – Level 3 Hardware: Raspberry Pi 3 B+ (Microcontroller 2) FR

The Raspberry Pi Camera Board gives the ability to live stream in 720p60, which is in good range to reduce latency. Table 23 gives the functionality for the camera.

Module	Raspberry Pi Camera Board
Designer	Brian
Input(s)	• Power input from Raspberry Pi (Microcontroller 1)
Output(s)	• Live Video feed to the Raspberry Pi (Microcontroller 1)
Functionality	The Microcontroller 2 will supply 5 VDC to the Hall Effect and ultrasonic sensor. It will take in the data from the sensors and using the Bluetooth transmitter on the Raspberry Pi, send the sensor data to the phone.

Table 23 – Level 3 Hardware: Raspberry Pi Camera Board FR

The Hall Effect sensor is used to detect magnetic fields. It will be placed on the rear bike rim. To count rotations, this sensor was the appropriate choice. One rotation would be equivalent to the Hall Effect sum/number of magnets. The number of rotations will allow speed to be recorded to the micro controller. In addition, a magnet will also go on the rear of the bike for the sensor to work effectively. The Hall Effect sensor will be used to calculate the distance traveled and the speed. The connection between the Microcontroller 2 and the Hall Effect sensor is in Figure 9.

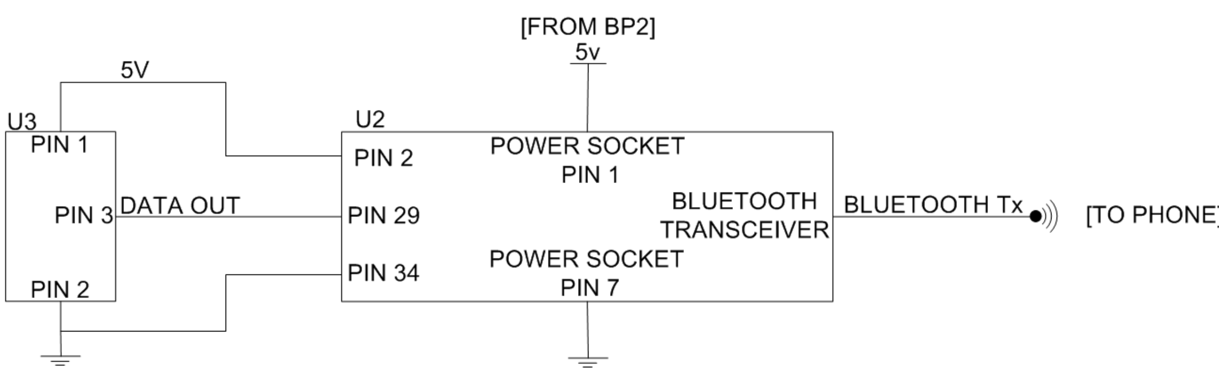


Figure 9 – Level 3 Hardware Block Diagram of Hall Effect Sensor

Module	[U3] US-1881 (Hall Effect Sensor)
Designer	Brian
Input(s)	● 5VDC from the GPIO of the Raspberry Pi
Output(s)	● Data out Wired to Microcontroller 2
Functionality	The Hall Effect sensor is used to detect magnetic fields. The Hall Effect sensor will be used to calculate the distance traveled and the speed.

Table 24 – Level 3 Hardware: US-1881 (Hall Effect Sensor) FR

To detect oncoming vehicles within ten meters, the ultrasonic sensor will be used. The sensors detection range can detect up to 10 meters. Every 250ms, the sensor can take readings. This is sufficient time for our system. The connection between the Microcontroller 2 and the Hall Effect sensor is in Figure 10.

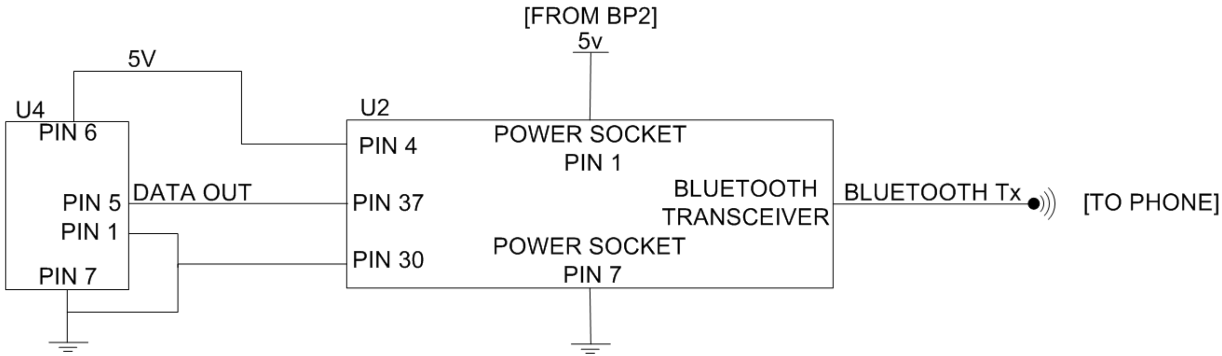


Figure 10 – Level 3 Hardware Block Diagram of Ultrasonic Sensor

Module	[U4] MB1261 (Ultrasonic Sensor)
Designer	Brian
Input(s)	● 5VDC from the GPIO of the Raspberry Pi
Output(s)	● Data out Wired to Microcontroller 2
Functionality	Detect oncoming vehicles within at least 10 feet distance and send alert to user within 250ms

Table 25 – Level 3 Hardware: MB1261 (Ultrasonic Sensor) FR

The MyBud glasses are going to be used as the display. The MyBud glasses will be powered from a DC battery pack on the back of the rider. The battery pack can be seen on right of Figure 11 (the rectangle). A video feed will output from the microcontroller and onto the headset display. Calories, Distance, and Speed will display on the glasses with the video frames simultaneously.

Figure 11 – Picture of the MyBud glasses with power pack

Module	MyBud
Designer	N/A
Input(s)	● Power Supply ● Video output from Microcontroller 1
Output(s)	● Display Video Feed ● Display Exercise Information
Functionality	Receive video output from Microcontroller 1 and display exercise information to rider

Table 26 – Level 3 Hardware: MyBud FR

The bike rider will be wearing a watch like device that will collect the bike rider's heart rate and reference direction that will be used to help calculate exercise information. Figure 12 is the wristband that will house the MBient sensor. The benefit of using the MBient sensor is that it has a low power Bluetooth transmitter on it that can easily send data to phones.

Figure 12 – Picture of the wristband used to house the MBient sensor

The benefit of using the MBient sensor is that it has a low power Bluetooth transmitter on it that can easily send data to phones. The sensor shown in Figure 13 below will attach the watch wristband shown for the rider to place on wrist. The sensor watch has its own battery inside that powers it.

Figure 13 – Picture of the MBient sensor

Module	MBient (Heart Rate and Gyroscope)
Designer	N/A
Input(s)	• N/A
Output(s)	• Heart Rate data via Bluetooth to Phone • Gyroscope data via Bluetooth to Phone
Functionality	The watch will allow the heart rate, acceleration, and orientation is determined most accessibly. The band will contain the sensor on the users watch. The data collected will then wirelessly transmit back to the phone app.

Table 27 – Level 3 Hardware: MBient (Heart Rate and Gyroscope) FR

The Raspberry Pi requires 5VDC and 2.5A to power it. To have ARH work for 1 hour, the capacity using the following equation needs to be calculated

$$Minimum\ capacity\ for\ battery: C = xT\ [A * hr]$$

With x being the current for the device in Ampere and T being the running time in hours to get Ampere-hours. Therefore $x = 2.5A$ (required for Raspberry Pi) and $T \geq 1hr$ (Design Requirements). Below is the calculated minimum capacity.

$$C \geq 2.5 * 1$$
$$C \geq 2.5\ [A * hr]$$

To compensate for lost in capacity due to full charge cycles, C' is calculated in order to have a battery last longer.

$$Charge\ cycle\ compensation: C' = \frac{C}{0.8}$$
$$C' \geq \frac{2.5}{0.8}$$

$$C' \geq 3.125[A * hr]$$

Therefore, the minimum required capacity of the battery will be 3.125Ahr to guarantee a run time of 1 hour. However, it is important to know how a battery performs with constant current discharge. Table 28 below compares three batteries from VRLA Rechargeable Battery, at nominal capacity of 4Ah, 4.5Ah, and 5Ah.

	Nominal Capacity	Constant current discharge with F.V=1.75 at 1 hr
BP4-6	4Ah	2.52A
BP4.5-6	4.5Ah	2.835A
BP5-6	5Ah	3.15A

Table 28 – Battery Comparison

Looking at Table 28 it is shown that the capacity should be around 4.5Ah to ensure it last for the hour run with current running at 2.5A

To ensure safety for the bike rider, two batteries will be used to ensure the bike rider is not connected to the bike by wires. The first battery pack BP1 will be located on the bike riders back and will power the Microcontroller 1 mounted on the helmet. The second battery pack BP2 is located on the rear of the bike will supply power to the Microcontroller 2 that will be collecting data from the sensors and transmitting the data to the phone.

Software

Pseudocode

```
main(){
	while(true){
		if(!Exercise_Stop){
			display(processFrame(image));
		}else{
			display(image);
		}
	}

}
```

If the user is not currently exercising, no information will be masked over the display since there is no information to mask. Otherwise, exercise and warning information will get masked over the incoming video frames.

```
function processFrame(image){
	processed[] = divideFrame();
	processed = maskWorkout();
	processed = maskWarning();

	return arrayToImage(processed);

}
```

When an image is processed, it will be broken into an array that can be manipulated. From that point, it will then attempt to mask the workout and the warning on top of the image.

After the information has been masked onto the array, the array will be converted back to an image and sent to the display.

```
function array maskWarning(array){
	if(car_near){
		displayWarning(array);
	}
}
```

The mask will only be applied to the array if there is an incoming car.

```
function maskWorkout(array, infoNumber){
	switch(infoNumber){
		case 0: showHeartRate();
		case 1: showCalories();
		case 2: showDistance();
		case 3: showSpeed();
		default: showTime();
	}
}
```

The type of information that is masked onto the array will depend on a passed in parameter. This allows the information to be able to cycle through the information that is masked for the display

```
main()
{
	Get_User_Input ();   // User Details, Exercise Start/Stop, View Database Yes/No
	if (Exercise_Start)
	{
		Update_User-Based_Equations();   // Max Heart Rate
		Loop();
	}
	if (View_Database_Yes)
	{
		Display_Database_Results();
	}
}
```

The exercise can only be started after the input has been received from the user since many of the equations depend upon user input. Once the exercise is started, a loop will be entered which will constantly incorporate information gathered from the sensors.

```
Loop()
```

```
{
        while(!Exercise_Stop)
        {
                Get_Sensor_Data();          // Heart Rate, Position, Speed, Angle
                if (Heart_Rate > Max_Heart_Rate)
                {
                        Display_Alert_to_User();
                }
                Update_Equations();         // Velocity, Direction, Calories Burned
                Send_Results_To_Helmet();
        }
        Send_Exercise_Data_to_Database();
}
```

While the exercise has not ended, sensors data will be gathered and used in the equations to calculate the exercise information. The results of the computations will then be sent to the helmet in order for it to be masked onto video frames.

```
Display_Database_Results()
{
        Get_Results_from_Database();
        Show_Results();            // Last Exercise, Last x Exercises, Best Exercise
}
```

It will retrieve previous exercise information from the database and then show that information on the designated section of the phone app.

4. Parts List

Qty.	Refdes	Part Num.	Description
1	U4	MB1261	Ultrasonic Sensor
1	U3	US-1881	Hall Effect Sensor
1		n/a	Heart Rate Sensor
1		n/a	Wristlet for Sensor
2	U1, U2	n/a	Raspberry Pi 3 B+
1		n/a	Raspberry Pi Camera Board
1		n/a	mybud
2	BP1, BP2	n/a	Battery pack with 4.5Ah

Table 29 – Parts List part 1

Qty.	Part Num.	Description	Unit Cost	Total Cost
1	MB1261	Ultrasonic Sensor	$54.95	$54.95
1	US-1881	Hall Effect Sensor	0.86	0.86
1	n/a	Heart Rate Sensor	86.95	86.95
1	n/a	Wristlet for Sensor	9.00	9.00
2	n/a	Raspberry Pi 3 B+		
1	n/a	Raspberry Pi Camera Board		
1	n/a	mybud		
2	n/a	Battery pack with 4.5Ah		

Table 30 – Parts List part 2

5. Project Schedules

Name	Begin date	End date	Resources
SDP1 fall2016	8/28/16	1/16/17	
Project Design	8/28/16	1/16/17	
Preliminary Design Report	8/28/16	10/9/16	
Problem Statement	8/28/16	10/9/16	
Need	9/1/16	9/8/16	Kevin Kapko (Archivist)
Objective	9/1/16	9/8/16	Kevin Kapko (Archivist)
Background with New Design	9/1/16	9/8/16	Azia Bradley (Hardware Manager),Kevin Kapko (Archivist),John Delonais (Software Manager)
Marketing Requirements with New Design	9/1/16	9/8/16	Kevin Kapko (Archivist),John Delonais (Software Manager)
Objective Tree	8/28/16	10/9/16	Brian Hromco (Project Leader)
Remake in Visio	8/28/16	10/9/16	Brian Hromco (Project Leader)
Add new section	8/28/16	10/9/16	Kevin Kapko (Archivist),Brian Hromco (Project Leader)
Redistribute Values	8/28/16	10/9/16	Kevin Kapko (Archivist),Brian Hromco (Project Leader)
Block Diagrams Level 0 w/ FR tables RE-DO in VISIO	9/1/16	9/15/16	
Hardware modules (identify designer)	9/1/16	9/15/16	Azia Bradley (Hardware Manager),Brian Hromco (Project Leader)
Software modules (identify designer)	9/1/16	9/15/16	Kevin Kapko (Archivist),Brian Hromco (Project Leader)
Preliminary Design Presentation 3:20PM	9/15/16	9/15/16	Azia Bradley (Hardware Manager),Kevin Kapko (Archivist),John Delonais (Software Manager),Brian Hromco (Project Leader)
Midterm Report	9/15/16	12/2/16	
Design Requirements Specification	9/19/16	10/2/16	John Delonais (Software Manager),Kevin Kapko (Archivist)
Midterm Design Gantt Chart	9/19/16	10/2/16	Brian Hromco (Project Leader)
Design Calculations	9/15/16	12/2/16	
Electrical Calculations	9/15/16	9/15/16	
Communication	9/15/16	9/15/16	Kevin Kapko (Archivist),John Delonais (Software Manager)
Sensor WIFI Transmittion	9/15/16	9/15/16	Kevin Kapko (Archivist),John Delonais (Software Manager)
Phone Bluetooth to Rpi	9/15/16	9/15/16	Kevin Kapko (Archivist),John Delonais (Software Manager)
Computing	9/15/16	9/15/16	
image processing	9/15/16	9/15/16	Kevin Kapko (Archivist),John Delonais (Software Manager)
Sensors	9/15/16	9/15/16	
Hall Effect Sensor	9/15/16	9/15/16	Azia Bradley (Hardware Manager)
Rear Car Detection	9/15/16	9/15/16	Kevin Kapko (Archivist),John Delonais (Software Manager)
Power Supply	9/15/16	12/2/16	
Current needed	9/15/16	12/2/16	Brian Hromco (Project Leader)
Power, Voltage, Current	9/15/16	12/2/16	
Raspberry Pi Compute Module Current Dr...	9/15/16	12/2/16	Brian Hromco (Project Leader)
My Bud	9/15/16	9/15/16	Azia Bradley (Hardware Manager),Brian Hromco (Project Leader)
My Bud battery Life	9/15/16	9/15/16	Kevin Kapko (Archivist)
My Bud Current Draw	9/15/16	9/15/16	Azia Bradley (Hardware Manager)
Rechargeable	9/15/16	9/15/16	Azia Bradley (Hardware Manager),Brian Hromco (Project Leader)
Voltage Needed	9/15/16	9/15/16	Brian Hromco (Project Leader)
Mechanical Calculations	9/15/16	9/15/16	
Mounting Hall sensor to bike wheel	9/15/16	9/15/16	John Delonais (Software Manager)
Mounting power supply on persons body	9/15/16	9/15/16	John Delonais (Software Manager)
Mounting Rpi on Helmet	9/15/16	9/15/16	John Delonais (Software Manager)
Exercising Calculations	9/15/16	9/15/16	
How many Calories Burned	9/15/16	9/15/16	Kevin Kapko (Archivist),John Delonais (Software Manager)
How many times the Bike wheel turns for RPM	9/15/16	9/15/16	Kevin Kapko (Archivist),John Delonais (Software Manager)
At your current speed, how fast does it take to fini...	9/15/16	9/15/16	Kevin Kapko (Archivist),John Delonais (Software Manager)
Block Diagrams Level 1 w/ FR tables & ToO	9/26/16	10/2/16	
Hardware modules (identify designer)	9/26/16	10/2/16	Azia Bradley (Hardware Manager),Brian Hromco (Project Leader)
Software modules (identify designer)	9/26/16	10/2/16	Kevin Kapko (Archivist),Brian Hromco (Project Leader)
Block Diagrams Level 2 w/ FR tables & ToO	10/3/16	10/10/16	
Hardware modules (identify designer)	10/3/16	10/10/16	Azia Bradley (Hardware Manager)
Software modules (identify designer)	10/3/16	10/10/16	Kevin Kapko (Archivist),John Delonais (Software Manager),Brian Hromco (Project Leader)
Midterm Design Presentations 3:20-5:00PM Part 1	10/20/16	10/20/16	Azia Bradley (Hardware Manager),Kevin Kapko (Archivist),John Delonais (Software Manager),Brian Hromco (Project Leader)
Project Poster	10/27/16	11/7/16	Azia Bradley (Hardware Manager),Kevin Kapko (Archivist),John Delonais (Software Manager),Brian Hromco (Project Leader)
Final Design Report	12/3/16	1/16/17	
Final Design Presentation Part 1 3:20PM-5:00PM	12/1/16	12/1/16	
Final Design Presentation Part 2 3:20PM-5:00PM	12/8/16	12/8/16	
Final Design Presentation Part 3 5:15PM-7:15PM	12/14/16	12/14/16	

6. Design Team Information

Azia Bradley, EE, Hardware Lead
Brian Hromco, EE, Project Lead
John Delonais, CpE, Software Lead
Kevin Kapko, CpE, Archivist

7. Conclusions and Recommendations

As of today, the team has spent a decent amount of time researching current virtual reality headsets on the market and exercise devices. Given enough time, the ARH could be expanded to an exercise virtual reality helmet device compatible with devices on the market today. It is expected by the end of the semester to have a completed design of our headset and hardware and to compare this design with other prototypes. A final design decision should be made by December.

8. References

[1] A. Podeltiva et al.,"Image processing device, image processing method, and computer readable medium", U.S Patent 9235770, Jan. 12, 2012
[2] J. Dorogusker et al., "Systems and methods for integrating a portable electronic device with a bicycle", U.S. Patent 8364389, Jan. 29 2013
[3] J. Haddick and R. Osterhaut, "Video display modification based on sensor input for a see-through near-to-eye display", U.S. Patent 8964298, Feb. 24 2015
[4] D. Ge and Z. Chen, "On-Chip Boost DC-DC Converter in Color OLED Driver & Controller ICs for Mobile Application," 2005 6th International Conference on ASIC.
[5] Donald P. Massay,"Sensors- Articles." Sensors - Articles. N.p., n.d. Web. 3 Oct. 2016.
[6] "DATA & RESOURCES." Pedestrian & Bicycle Information Center. U.S Department of Transportation Federal Highway Administration, n.d. Web. 16 Oct. 2016.

9. Appendices

i.) Hall Effect Sensor

US1881

Hall Latch – High Sensitivity

Melexis Microelectronic Integrated Systems

Features and Benefits

- Wide operating voltage range from 3.5V to 24V
- High magnetic sensitivity – Multi-purpose
- CMOS technology
- Chopper-stabilized amplifier stage
- Low current consumption
- Open drain output
- Thin SOT23 3L and flat TO-92 3L both RoHS Compliant packages

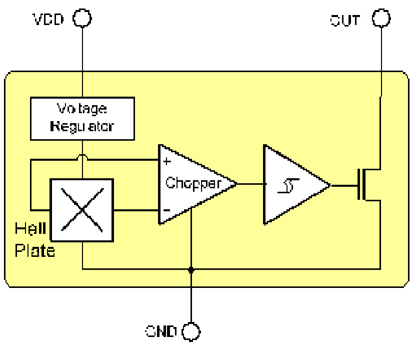
Application Examples

- Automotive, Consumer and Industrial
- Solid-state switch
- Brushless DC motor commutation
- Speed detection
- Linear position detection
- Angular position detection
- Proximity detection

Ordering Information

Part No.	Temperature Code	Package Code
US1881	E (-40°C to 85°C)	SE (TSOT-3L)
US1881	E (-40°C to 85°C)	UA (TO-92)
US1881	K (-40°C to 125°C)	SE (TSOT-3L)
US1881	K (-40°C to 125°C)	UA (TO-92)
US1881	L (-40°C to 150°C)	SE (TSOT-3L)
US1881	L (-40°C to 150°C)	UA (TO-92)

1 Functional Diagram



2 General Description

The Melexis US1881 is a Hall-effect latch designed in mixed signal CMOS technology.

The device integrates a voltage regulator, Hall sensor with dynamic offset cancellation system, Schmitt trigger and an open-drain output driver, all in a single package.

Thanks to its wide operating voltage range and extended choice of temperature range, it is quite suitable for use in automotive, industrial and consumer applications.

The device is delivered in a Thin Small Outline Transistor (TSOT) for surface mount process and in a Plastic Single In Line (TO-92 flat) for through-hole mount.
Both 3-lead packages are RoHS compliant.

3 Glossary of Terms

MilliTesla (mT), Gauss	Units of magnetic flux density: 1mT = 10 Gauss
RoHS	Restriction of Hazardous Substances
TSOT	Thin Small Outline Transistor (TSOT package) – also referred with the Melexis package code "SE"
ESD	Electro-Static Discharge
BLDC	Brush-Less Direct-Current
Operating Point (B_{OP})	Magnetic flux density applied on the branded side of the package which turns the output driver ON ($V_{OUT} = V_{DSon}$)
Release Point (B_{RP})	Magnetic flux density applied on the branded side of the package which turns the output driver OFF (V_{OUT} = high)

4 Absolute Maximum Ratings

Parameter	Symbol	Value	Units
Supply Voltage	V_{DD}	28	V
Supply Current	I_{DD}	50	mA
Output Voltage	V_{OUT}	28	V
Output Current	I_{OUT}	50	mA
Storage Temperature Range	T_S	-50 to 150	°C
Maximum Junction Temperature	T_J	165	°C

Table 1: Absolute maximum ratings

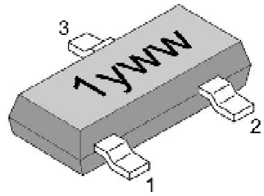
Exceeding the absolute maximum ratings may cause permanent damage. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

Operating Temperature Range	Symbol	Value	Units
Temperature Suffix "E"	T_A	-40 to 85	°C
Temperature Suffix "K"	T_A	-40 to 125	°C
Temperature Suffix "L"	T_A	-40 to 150	°C

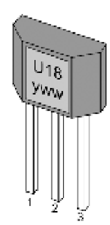
5 Pin Definitions and Descriptions

SE Pin №	UA Pin №	Name	Type	Function
1	1	VDD	Supply	Supply Voltage pin
2	3	OUT	Output	Open Drain Output pin
3	2	GND	Ground	Ground pin

Table 2: Pin definitions and descriptions



SE package



UA package

6 General Electrical Specifications

DC Operating Parameters $T_A = 25^oC$, $V_{DD} = 3.5V$ to 24V (unless otherwise specified)

Parameter	Symbol	Test Conditions	Min	Typ	Max	Units
Supply Voltage	V_{DD}	Operating	3.5		24	V
Supply Current	I_{DD}	$B < B_{RP}$			5	mA
Output Saturation Voltage	V_{DSon}	$I_{OUT} = 20mA$, $B > B_{OP}$			0.5	V
Output Leakage Current	I_{OFF}	$B < B_{RP}$, $V_{OUT} = 24V$		0.3	10	µA
Output Rise Time	t_r	$R_L = 1k\Omega$, $C_L = 20pF$		0.25		µs
Output Fall Time	t_f	$R_L = 1k\Omega$, $C_L = 20pF$		0.25		µs
Maximum Switching Frequency	F_{SW}			10		KHz
Package Thermal Resistance	R_{TH}	Single layer (1S) Jedec board		301		°C/W

Table 3: Electrical specifications

7 Magnetic Specifications

DC Operating Parameters $V_{DD} = 3.5V$ to 24V (unless otherwise specified)

Parameter	Symbol	Test Conditions	Min	Typ	Max	Units
Operating Point	B_{OP}		0.5		9.5	mT
Release Point	B_{RP}	E spec., $T_A = 85°C$	-9.5		-0.5	mT
Hysteresis	B_{HYST}		7		12	mT
Operating Point	B_{OP}		0.5		9.5	mT
Release Point	B_{RP}	K spec., $T_A = 125°C$	-9.5		-0.5	mT
Hysteresis	B_{HYST}		7		12	mT
Operating Point	B_{OP}		0.5		9.5	mT
Release Point	B_{RP}	L spec., $T_A = 150°C$	-9.5		-0.5	mT
Hysteresis	B_{HYST}		6		12.5	mT

Table 4: Magnetic specifications

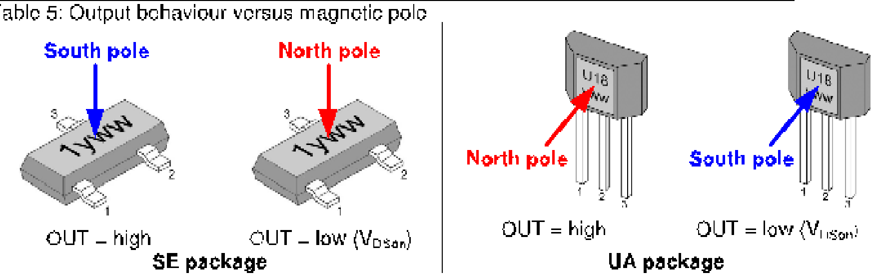
Note 1: For typical values, please refer to the performance graphs in section 11

8 Output Behaviour versus Magnetic Pole

DC Operating Parameters $T_A = -40^oC$ to 150^oC, $V_{DD} = 3.5V$ to 24V (unless otherwise specified)

Parameter	Test Conditions (SE)	OUT (SE)	Test Conditions (UA)	OUT (UA)
South pole	$B < B_{RP}$	High	$B > B_{OP}$	Low
North pole	$B > B_{OP}$	Low	$B < B_{RP}$	High

Table 5: Output behaviour versus magnetic pole

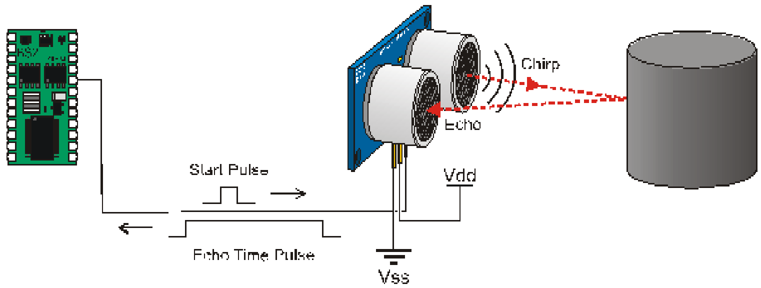


ii) Ultrasonic Sensor

PING))) Ultrasonic Distance Sensor (#28015)

The Parallax PING)))™ ultrasonic distance sensor provides precise, non-contact distance measurements from about 2 cm (0.8 inches) to 3 meters (3.3 yards). It is very easy to connect to microcontrollers such as the BASIC Stamp®, Propeller chip, or Arduino, requiring only one I/O pin.

The PING))) sensor works by transmitting an ultrasonic (well above human hearing range) burst and providing an output pulse that corresponds to the time required for the burst echo to return to the sensor. By measuring the echo pulse width, the distance to target can easily be calculated.



Features

- Range: 2 cm to 3 m (0.8 in to 3.3 yd)
- Burst indicator LED shows sensor activity
- Bidirectional TTL pulse interface on a single I/O pin can communicate with 5 V TTL or 3.3 V CMOS microcontrollers
- Input trigger: positive TTL pulse, 2 µs min, 5 µs typ.
- Echo pulse: positive TTL pulse, 115 µs minimum to 18.5 ms maximum.
- RoHS Compliant

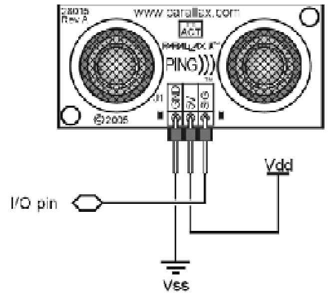
Key Specifications

- Supply voltage: +5 VDC
- Supply current: 30 mA typ; 35 mA max
- Communication: Positive TTL pulse
- Package: 3-pin SIP, 0.1" spacing (ground, power, signal)
- Operating temperature: 0 – 70° C.
- Size: 22 mm H x 46 mm W x 16 mm D (0.84 in x 1.8 in x 0.6 in)
- Weight: 9 g (0.32 oz)

Pin Definitions

GND	Ground (Vss)
5 V	5 VDC (Vdd)
SIG	Signal (I/O pin)

The PING))) sensor has a male 3-pin header used to supply ground, power (+5 VDC) and signal. The header may be plugged into a directly into solderless breadboard, or into a standard 3-wire extension cable (Parallax part #800-00120).



iii.) Mbient Lab Multi-sensor