

Touch-Free Eye Tracker



**Department of Electrical Engineering and Computer Science
University of Central Florida**

Group 29

Timothy Wells - Electrical Engineer
Patrick Gonzalez - Computer Engineer
Mchael Fillinger - Computer Engineer
James Curtis - Electrical Engineer

Sponsors / Contributors / Customers

None
Due date
12/4/2019

Table of Contents

1.0 Executive Summary.....	Page 7
2.0 Project Description.....	Page 9
2.1 Motivation.....	Page 9
2.2 Goals/Objectives.....	Page 9
2.3 Eye Tracking.....	Page 10
2.4 Marketing Requirements.....	Page 10
2.5 House of Quality Diagram.....	Page 11
2.6 Block Diagram.....	Page 12
3.0 Research Related to Project.....	Page 14
3.1 Existing Products.....	Page 14
3.1.1 Tobii.....	Page 14
3.1.2 SR Research.....	Page 16
3.2 Relevant Technologies.....	Page 17
3.2.1 Microcontroller.....	Page 17
3.3 Oculus Rift Research.....	Page 25
3.3.1 Oculus Eye Tracking Based on Depth Sensing.....	Page 25
3.3.2 Oculus Light Field Camera.....	Page 26
3.4 Eye-Tracking Methodology.....	Page 27
3.5 Existing Wi-Fi and Future of Wi-Fi.....	Page 29
3.5.1 Frequency and Range of Current Wi-Fi Networks.....	Page 30
3.5.2 Data Transfer Rates for Current Wi-Fi.....	Page 31
3.5.3 Future of Wi-Fi 802.11ax (Wi-Fi 6).....	Page 32
3.5.4 Comparing and Choosing Amongst Different Wi-Fi Networks.....	Page 34
3.6 Bluetooth Communication.....	Page 35
3.6.1 Frequency and Range of Current Bluetooth.....	Page 35
3.6.2 Data Transfer Rate of Bluetooth.....	Page 36
3.6.3 Consideration of Use of Bluetooth on the Project.....	Page 37
3.7 Zigbee Wireless Communication.....	Page 37
3.7.1 Frequency and Range of Zigbee.....	Page 38
3.7.2 Data Transfer Rate of Zigbee and Power Consumption of Zigbee Devices.....	Page 41
3.7.3 Security of Zigbee.....	Page 43
3.7.4 Consideration of Zigbee Use for our Project.....	Page 44
3.8 Machine Learning / Neural Networks.....	Page 44

3.8.1 How a Neural Network Works.....	Page 45
3.8.2 Activation Functions.....	Page 45
3.8.3 Basic Layout of a Neural Network.....	Page 49
3.9 3-D Printing.....	Page 50
3.9.1 3-D Modeling Software Details.....	Page 51
3.9.2 3-D Printing Machine Prices and Availability.....	Page 51
4.0 Related Standards and Design Constraints.....	Page 53
4.1 Related Standards.....	Page 53
4.1.1 Battery Standards.....	Page 53
4.1.2 Programming Language Standards.....	Page 54
4.1.3 Software and Engineering Standards.....	Page 55
4.1.4 Software Testing Standards.....	Page 55
4.1.5 Health and Safety Standards.....	Page 58
4.1.5.1 Health and Safety Risks of Eye Tracking Technology.....	Page 59
4.1.6 Wireless Communication Standards.....	Page 62
4.1.7 Bluetooth Standards.....	Page 62
4.1.8 Zigbee Standards.....	Page 63
4.2 Design Constraints.....	Page 64
4.2.1 Economic and Time Constraints.....	Page 64
4.2.2 Health and Safety Constraints.....	Page 64
4.2.3 Technological Constraints.....	Page 65
4.2.4 Environmental, Social, and Political Constraints.....	Page 65
4.2.5 Manufacturability Constraints and Sustainability Constraints.....	Page 65
5.0 Project Hardware Design Details.....	Page 67
5.1 Design Architecture Details.....	Page 67
5.2 Infrared LED Emitters and Sensors.....	Page 68
5.2.1 Sensor Array Details.....	Page 70
5.2.2 Sensor Array Previous Iterations.....	Page 72
5.3 Googles Details.....	Page 74
5.4 Converter Board Details.....	Page 75
5.4.1 Converter Board Previous Iterations.....	Page 76
5.5 Main Processor Details.....	Page 78
5.6 Power Supply Details.....	Page 79
5.7 Display Screen Details.....	Page 79
6.0 Project Software Design Details.....	Page 80
6.1 Software Purpose.....	Page 80

6.1.1 Primary Processor Software.....	Page 81
6.1.2 Wi-Fi Functionality.....	Page 81
6.1.3 Code Composer Studio (IDE).....	Page 82
6.1.4 TinkerCAD.....	Page 82
6.1.5 AutoCAD.....	Page 82
6.1.6 PyCharm.....	Page 82
7.0 Hardware and Software Comparisons.....	Page 84
7.1 Sensor Array Options.....	Page 84
7.1.2 Choosing the Sensor Array.....	Page 85
7.2 Goggles Set Requirements.....	Page 85
7.2.1 Choosing the Goggles Set.....	Page 86
7.2.2 Self Built.....	Page 87
7.2.3 Google Cardboard.....	Page 87
7.2.4 Google Daydream View.....	Page 88
7.2.5 Utopia 360.....	Page 88
7.3 Selector Chip.....	Page 89
7.3.1 Choosing The Selector Chip.....	Page 91
7.3.1.1 CD4076B Chip.....	Page 92
7.3.1.2 CD74HC4067M96 Chip.....	Page 93
7.3.1.3 MPC506AU/1K Chip.....	Page 93
7.3.1.4 CD74HCT4067M Chip.....	Page 94
7.3.1.5 CD74HC4067M Chip.....	Page 95
7.4 Analog to Digital Converter Chip.....	Page 96
7.4.1 Choosing The Analog to Digital Converter Chip.....	Page 97
7.5 Microcontroller Unit	Page 98
7.5.1 Choosing the Microcontroller.....	Page 99
7.6 The Main Processor Unit.....	Page 100
7.6.1 Choosing the Main Processor Unit.....	Page 101
7.7 The Display Screen.....	Page 102
7.7.1 Choosing The Display Screen.....	Page 103
7.7.2 Axion ACN-5507.....	Page 104
7.7.3 BW TFT LCD.....	Page 104
7.7.4 Phone.....	Page 105
8.0 Project Testing and Prototype Construction.....	Page 107
8.1 Prototype Sensor Design Test.....	Page 108

8.2 Prototype 1 Converter System Test.....	Page 110
8.3 Prototype 2 IR Concept Test Set Up.....	Page 110
8.3.0 Prototype 2 (Bread-Board Version) Design.....	Page 111
8.3.1 Prototype 2 Goggle and IR Array Setup.....	Page 111
8.3.2 Prototype 2 Bread-Board Design.....	Page 114
8.3.3 Prototype 2 Voltage Divider and Selector Chip Setup.....	Page 116
8.3.4 Prototype 2 Analog to Digital Converter Setup.....	Page 116
8.4 Prototype 2 Main Coding.....	Page 118
8.4.1 Prototype 2 MSP430 Code.....	Page 118
8.5 Prototype 2 PCB Board Design 1.0.....	Page 118
8.5.1 Prototype 2 PCB Board Schematic.....	Page 119
8.6 Microcontroller Testing Procedures.....	Page 121
9.0 Administrative Content.....	Page 125
9.1 Budget.....	Page 125
9.2 Milestones for the Project.....	Page 126
10.0 Project Summary and Conclusion.....	Page 129
11.0 Appendices.....	Page 130
11.1 Bibliography.....	Page 130

List of Figures

Figure 1 Block Diagram	Page 13
Figure 2 Eye Tracker Diagram	Page 16
Figure 3 Eye Tracking Glasses Diagram	Page 17
Figure 4 Head Strapped Eye Tracking Device	Page 18
Figure 5 Block Diagram of MSP430G2x53	Page 19
Figure 6 Registers Used in the MSP430	Page 21
Figure 7 Circuit Diagram of ADC	Page 22
Figure 8 Clock Cycles Graph of ADC	Page 24
Figure 9 Eye Tracking Screen	Page 27
Figure 10 Light Field Camera System	Page 28
Figure 11 Purkinje Images	Page 30
Figure 12 Analysis of Performance Benefits Achieved by Spatial-Reuse	Page 34
Figure 13 Model of the Zigbee Network	Page 39
Figure 14 Zigbee Device Performance with Varying Distances	Page 40
Figure 15 Attenuation Level with Varying Distances between Zigbee Devices	Page 41
Figure 16 Zigbee Architecture	Page 42
Figure 17 Throughput vs Packet Size	Page 44
Figure 18 Graph of Sigmoid Function	Page 46
Figure 19 Graph of Tanh Function	Page 47
Figure 20 Graph of ReLU Function	Page 48
Figure 21 Graph of Leaky ReLU Function	Page 49
Figure 22 Basic Layout of a Neural Network	Page 50
Figure 23 3-D Printer	Page 52
Figure 24 Set of Test Design Techniques	Page 57
Figure 25 Warning Sticker Flow Chart	Page 59
Figure 26 Schematic of the Human Eye	Page 62
Figure 27 Architecture of Low-Rate Wireless Personal Area Networks	Page 64
Figure 28 Project Concept	Page 69
Figure 29 5mm Infrared LED, T-1 3/4	Page 70
Figure 30 IR Array Holder Bracket	Page 72
Figure 31 IR Emitter and Sensor Diagram	Page 73
Figure 32: Previous Sensor Design	Page 74
Figure 33 Goggle Design Concept	Page 76
Figure 34 Diagram of MSP430	Page 77
Figure 35 Selector Chip Circuit and Truth Table	Page 79
Figure 36 Utopia 360 Headset	Page 90
Figure 37 The CD74HC4067M Selector Chip	Page 97
Figure 38 Prototype 1	Page 109
Figure 39 Prototype Sensor	Page 110
Figure 40 Prototype Converter	Page 111
Figure 41 Overview of Prototype 2.....	Page 112
Figure 42 View Through Goggles Prototype 2.....	Page 113
Figure 43.....	Page 114
Figure 44.....	Page 116
Figure 45.....	Page 118
Figure 46 Preliminary PCB Schematic.....	Page 122

List of Tables

Table 1 House of Quality Diagram	Page 12
Table 2 Transfer Rate of Video (Distance).....	Page 32
Table 3 Transfer Rate of Video (Different Video Resolution).....	Page 33
Table 4 Comparisons of Wifi Networks.....	Page 35
Table 5 Comparison of Specifications Between the Recent Versions of Bluetooth.....	Page 37
Table 6 Ranges of Frequencies Zigbee can Operate In.....	Page 39
Table 7 Comparison of Power Consumption Between Wi-Fi, Zigbee, and Green Power....	Page 43
Table 8 Transition Speed Comparison Between Detectors.....	Page 85
Table 9 Comparison of Goggles.....	Page 87
Table 10 Comparison of Selector Chips.....	Page 92
Table 11 Comparison of ADC Chips	Page 98
Table 12 Comparison of Different MSP430 Launchpads.....	Page 100
Table 13 SBC Comparisons.....	Page 102
Table 14 Comparison of Displays.....	Page 104
Table 15 Eye State Value Table	Page 108
Table 16 Eye State Value Table	Page 109
Table 17 Table Showing LDR Positions	Page 110
Table 18 Budget	Page 126
Table 19 Project Milestones	Page 128

1.0 Executive Summary

Eye tracking technology has seen a huge jump in development over the past 20 years. This is largely thanks to the development of machine learning algorithms and neural networks. However most of the eye tracking technology that is available for purchase today is exorbitantly expensive. We are looking to create a relatively cheap alternative. Typically the priciest component of eye tracking technology is the camera. We aim to remove that component altogether and solely focus on light sensors. Utilizing infrared sensors and a bit of machine learning our project will detect eye movement and our application will react accordingly.

Our project will be utilizing 16 Infrared Light Sensors, 8 for each eye in order to detect the gaze of the person using the device. The signal received from the Infrared sensors are converted from analog to digital using an MSP430. A selector chip is used to help manage the amount of inputs we have from the sensors. We will then be using a Raspberry Pi 3 in order to process our machine learning algorithms which processes the data taken in from the Infrared Sensors and determines the gaze of the user. These machine learning algorithms will be trained by a set of inputs created by us in order to help recognize the input data automatically and correctly identify the direction of the gaze of the user. The Raspberry Pi 3 will also be used to implement the devices wireless functionality. The wireless network that we will utilize will be 802.11n which is a 2.4 GHz frequency wireless network which can be picked up by our Raspberry Pi 3.

The biggest constraint our project will be facing is the limited budget we have to work with. The limited budget will affect how effective our eye tracking device will be at accurately determining where the user is looking because the technology we'll be able to purchase for the eye-tracking device won't be the most advanced piece of technology out there. There really aren't any ethical, environmental, political, or social constraints for us to deal with in terms of development. As far as manufacturability goes, as long as we use component are already being

manufactured on a large scale or can be manufactured on a large scale then we won't have any problems on manufacturability constraints. Components that are being manufactured on a large scale are very sustainable typically because of the high demand for these components also demands a high quality product. The health and safety standards will affect our project. Since we are dealing with a technology that will be interacting with the human eye which is very sensitive to radiation exposure we will have to be very judicious in selecting the intensity and brightness values of our LEDs.

We will be following numerous standards put in place by various organizations such as ISO, IEC, and IEEE. These standards cover things like wireless communication, software engineering, software testing, health and safety, portable batteries, and programming languages. The standards covered help establish a baseline of quality and safety that will help guide our project to become an acceptable finished product.

2.0 Project Description

Here in chapter 2 we will be going over our motivation for the projects and our goals and objectives for the development of the project and the final product. We will also be discussing a little bit of the methodology used in Eye-Tracking and some of the Marketing Requirements for our product. We have also included a House of Quality Diagram in order to showcase some of the correlations between the Engineering Requirements and the Marketing Requirements.

2.1 Motivation

Has this ever happened to you, where you try to operate a device thinking that you have highlighted what you want, but the device thinks you want something else? This divide between the user and device is caused by either, limitation of either of the user or device, crude or non-intuitive design, or more importantly a lack of ways the user can interact with the device. In today's technology, there are many untapped or unutilized input sources for a person to further interact with personal devices like a heads up display, or hands free control that allows one to step inside the mind of device. These inconvenience causes the user to have to learn or rely on non-intuitive control systems for interacting with said device. The idea is to allow both device and user to share line of sight of the task that lies in front of both of them. By developing a low-cost, hardware based solution, we hope to make eye-tracking an easier to approach and more viable solution.

2.2 Goals/Objectives

In developing this project, our main purpose is to create a portable fast acting control system that uses the operator's eye position/ state to control another external device with line of sight from inside the device. By using an array of Infra-Red(IR) sensors, we hope to be able to use the data from the sensors to then determine where the user is looking in frame to what the device the user is trying to use. The idea of using an array of sensors as opposed to a camera is so that the device will consume less power and will not need to process as much data. The objectives of this device are as follows:

- Create an array of sensors around each eye
- Convert the data to a digital signal to be read by a microcontroller
- Develop an algorithm for the microcontroller to determine the eye's position
- Use the eye's position to send commands to camera system to move the camera

2.3 Eye Tracking

The concept of tracking the eye is not new in today's technology, but the way one tracks it changes as technology improves. The principle behind the concept of how this system tracks the user's eye in real time is through the use of IR, or Infra-Red transistors. The IR will be set up in an array structure floating around the user's eye measuring minute changes in and around the eye. IR can change their resistance by large margins like from $10M\Omega$, while in dark, to 100Ω , while in the light. By reading the difference in voltage potential and relevant changes in those potentials, one gets both state and transitional information of the user's eye for to interact with the device. These sensors use the light created from a display inside a helmet or mask incomparsin to the same light reflected from the users eye to read changes in positions of the eye.

Why use IR vs things like cameras, well there a few reasons why. One is that the difference in raw data processing between cameras image tracking vs IR. Where cameras have to take in a decent resolution sized image to begin with. Then the camera system would have to process the image to find the eye and the parts that make up the eye as well. All that then comes to together just to find the state of the eye. While for the proposed IR array system, it simply takes state values around the eye and process an array of simple digital signals. Which means that most of the important data is processed outside the main MCU. Segwaying into the second reason is that the difference in requirements between the two system's MCU needs. As stated before camera tracking systems require large memory and very fast processing speeds to get some what real time responses to any change that might occur. While for an IR system, since they only process simple digital signals that could be four or five bits long, most MCU units can easily be used as long as it meets the processing time of a few microseconds to achieve the relative title of real time response.

The user will control a camera arm through the line of sight of the arm. As the user looks up through the arms video display, the arm will respond back by also looking up. Vice Versa for the cases for directional movement.

2.4 Marketing Requirements

1. The device must be comfortable to wear.
2. The device should be easy to use.
3. The power consumption of the device should be low.
4. The Robot Arm must be responsive.
5. Device must be able to transmit video to the screen for our device seamlessly.
6. The device should be dependable.

2.5 House of Quality Diagram

Legend									
+	Maximum								
-	Minimum								
↑	Pos Correlation								
↓	Neg Correlation								
Engineering Requirements	Weight								
Marketing Requirements	-	+	+	+	+	+	-	-	
Comfortable	+	↑↑	↑				↓		
Easy to Use	+	↑↑			↑	↑↑	↓	↑	
Power Consumption	-			↓	↑	↓	↑		
Responsiveness	+			↑↑		↑	↓	↑	
Video Transmission	+			↑↑	↑		↓		
Cost	-	↓	↓		↓	↓	↑↑	↓	
Targets for Engineering Requirements		< 5 Pounds	> 53 cm and < 61 cm	> 1000 kbits/s	> 1 Hour Battery Life	Rotate 90 Degrees Vertically and 360 Degrees Horizontally	< \$500 Total	< 25 degrees	

Table 1: House of Quality Diagram

2.6 Block Diagram

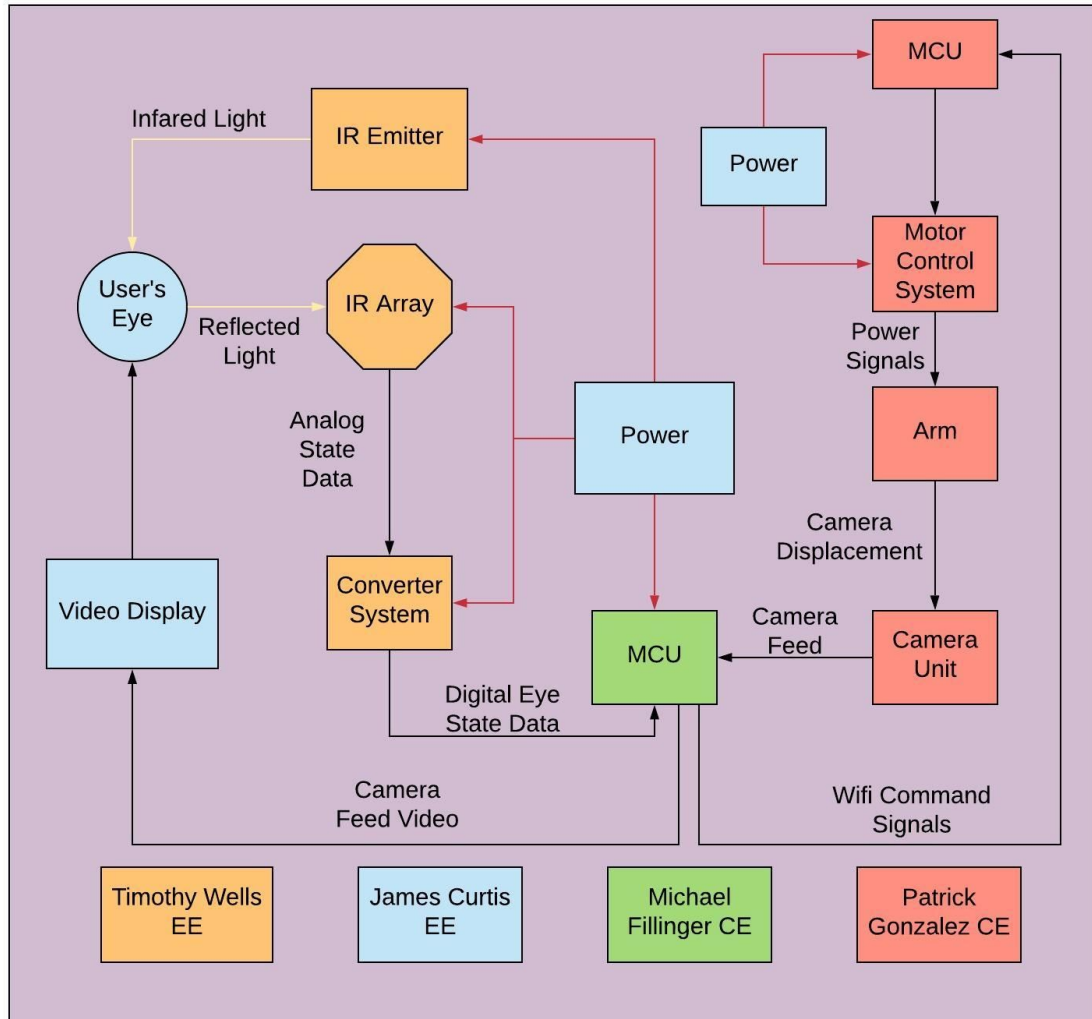


Figure 1: Block Diagram

The device is split into two devices One is the IR goggles, the other is the arm/turret the goggles control. The arm is mostly pre built components while the LDR system will be built from the ground up, besides the MCU and display. The two devices will communicate through wifi between the two MCU.

The signal will start in the IR Emitter. It will emit an infrared light to the user's eye. When it hits the eye it reflects light. Depending on which part of the eye it hits it will reflect more or less light. If it hits the skin it will reflect a high amount of light. If it hits the whites of the eye it will reflect a lower amount of light than if it hits the skin, yet still more light than if it reflected off the pupil of the eye. If the light

reflects off the black pupil of the eye then it reflects the least amount of light. The IR Array will have an array of IR detectors and will figure out which IR's are detecting the *least* amount of light because it is figuring out where the pupil is which will be reflecting the least amount of light.

The IR Array will then send the signals that it has detected to a converter system. This converter system will take the data that the IR Array is gathering and make it readable for the MCU. Once the MCU has the data it will be able to read it and figure out where the eye is looking. Once it reads the data it will send a signal to the camera's MCU through wifi to tell it where to look according to the movements of the eye.

3.0 Research Related to Project

This section of the report will discuss the research regarding eye-tracking methodologies and existing technologies. The research discussed will cover eye-tracking techniques from the past, modern eye-tracking methodologies, existing eye-tracking technology, and future eye tracking methodology and possible technology. Eye tracking has been researched for well over a century, and many of the principles and motivations of eye tracking have not changed. Much of the original use for eye-tracking was for medical purposes and psychological purposes, but in recent years eye-tracking has become a lot more common place for interacting with computers. There are very practical uses for eye-tracking, such as allowing people with disabilities to interact with everyday devices they may not otherwise be able to use and communicate with family and friends that has been denied for them. Eye-tracking is also beginning to be used for entertainment, especially in head-mounted devices such as VR headsets. Our eyes are one of the primary ways we perceive the world around us, and there for knowing what the user is looking at holds great value for a number of purposes.

3.1 Existing Products

As computers continue to become smaller and more powerful, eye-tracking is becoming more feasible for commercial use and at a lower cost than ever before. Older eye-tracking technology tended to be intrusive and uncomfortable, or require expensive and difficult to use hardware. More modern approaches tend to take advantage of modern image recognition software to accomplish the same task. Most of the devices on the market rely on the same key set of principles: an array of illuminators is put around the eye (often employing infrared light), an image sensor captures the reflection pattern created by the lights, and an external processing unit is used to interpret the data. The three products we will look at for existing eye-tracking systems are Tobii, SensoMotoric Instruments, and SR Research. These are all big names when it comes to research in the field.

3.1.1 Tobii

Tobii is a Swedish company founded in Stockholm in 2001. They provide a multitude of products, with both table-mounted (Figure 3.1.1.1) and head-mounted (Figure 2) hardware. As with most modern eye-tracking systems, Tobii utilizes pupil center corneal reflection (PCCR). This works by finding the center of the pupil and the point of reflection created on the cornea and measuring the vector between them. This vector, along with other information

about the eye, is then fed to the processing unit. Algorithms within the processing unit use this information to try and derive the real direction that the eye is pointing.

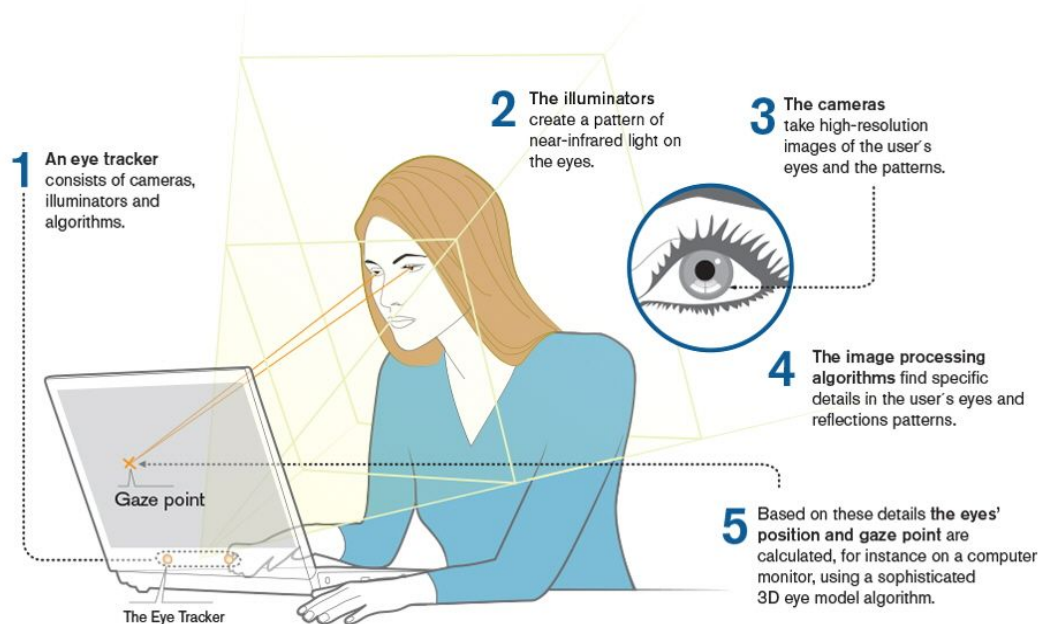


Figure 2 : Eye Tracker Diagram

Tobii offers a range of devices to allow for eye-tracking, but the device most relevant to the product's design would be the Tobii Pro Glasses 2. They work in the fashion demonstrated in figure 2. They are designed for research in capturing natural viewing behavior in real-world environments. To accomplish this task, this device requires highly accurate viewing data, as well as high data quality. The unit uses corneal reflection and dark pupil tracking, and accomplishes this using 4 image sensors. However, due to the device being used primarily for research, it would be rare to see the device being used by the average consumer.

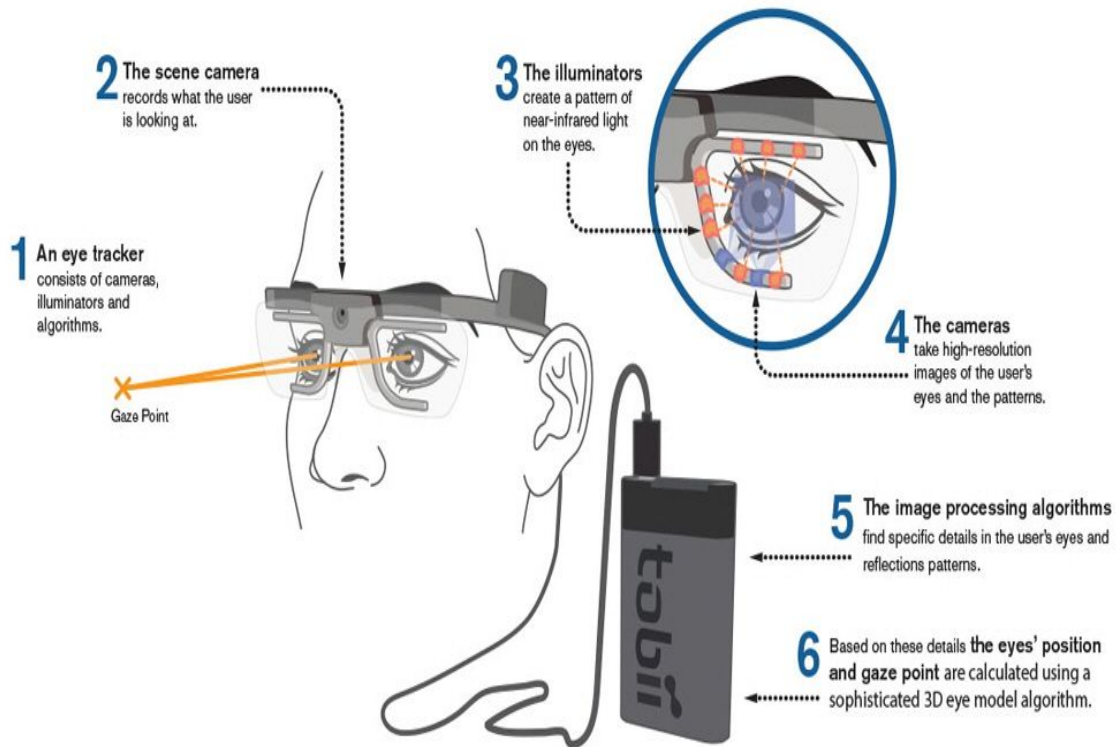


Figure 3: Eye Tracking Glasses Diagram

3.1.2 SR Research

SR Research is another company to provide solutions for eye-tracking for research purposes. SR Research has one head-mounted product, the EyeLink II. This device is meant purely for research purposes, and is far too inconvenient for any kind of consumer use (Figure 4). However, the device has some impressive specifications, with a 500Hz sampling rate and 0.01 degree resolution. The device operates using 2 cameras and a head mount, and the camera feed is then fed back to a processing unit for image processing. It has 2 different tracking modes, one for pupil only, which offers a higher sampling rate and less noise,

and a Pupil-Corneal Reflection mode, which offers more stability in case the device slips or vibrates while in use.

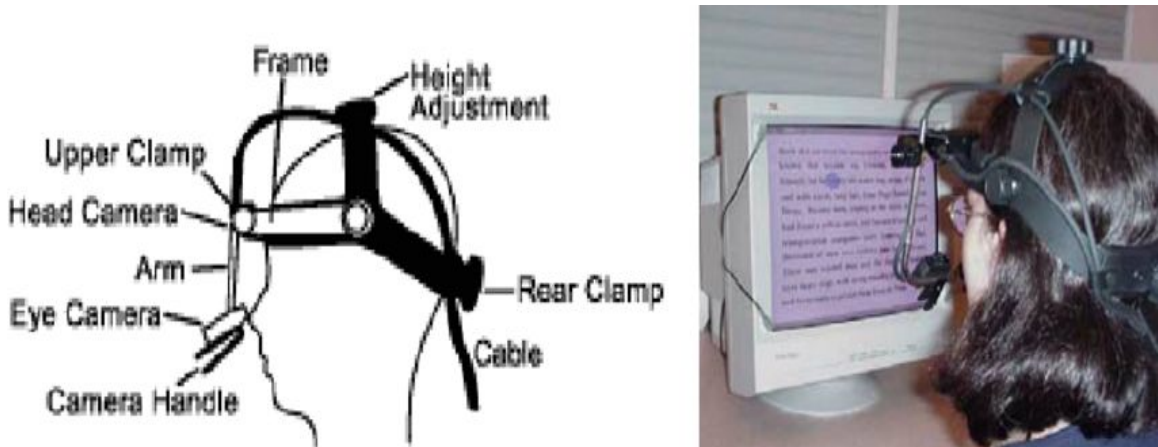


Figure 4 Head Strapped Eye Tracking Device

3.2 Relevant Technologies

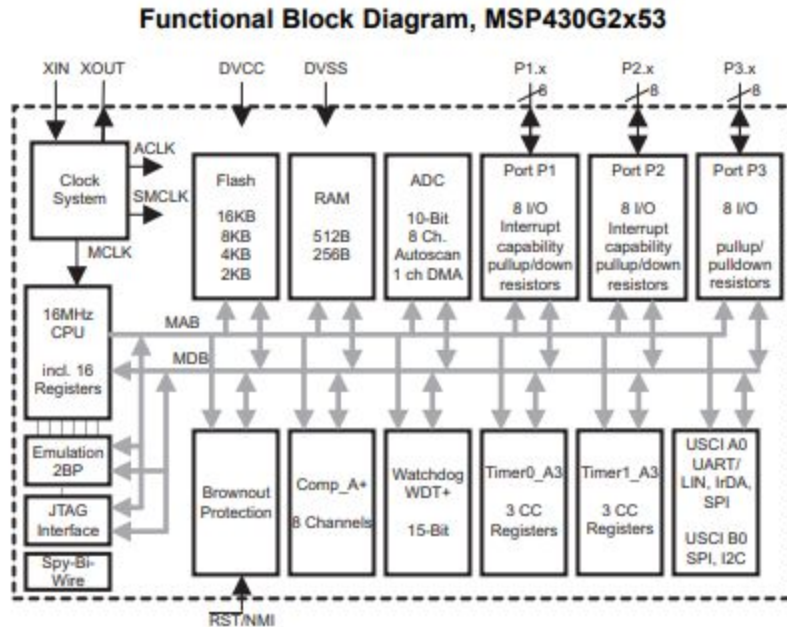
We believe that our proposed method of detecting eye gaze direction is relatively unique, and as such there are few products out there that use the same devices we plan to use to accomplish the task. This section will cover the research done for each part of our system. Our system will consist of a power system, sensor array, a microcontroller, and application software. This section exists as an analysis of the technology we used to ensure that we have created the best product based on what is available on the market, and what the developers of the product are most experienced in working in.

3.2.1 Microcontroller

For the purposes of the product, the team has decided to move forward using the MSP430 family of microcontrollers. The engineers on the team are very familiar with using this microcontroller, and its versatile nature will prove useful for the multitude of tasks that it will be involved in. The MSP430 uses a 16-bit RISC CPU, and the MSP430 is a great choice for offering many different avenues for solutions in devices incorporating mixed signal applications. The MSP430 also boasts an ultralow power architecture, which will be very useful for this design as the device will run almost entirely on a mobile battery power source.

The chip also provides multiple clocking solutions based on what type of application the chip will be used for. There is a lower frequency 32KHz auxiliary

clock driven by a crystal oscillator that would be ideal for ultralow power applications, and a much faster master clock that can operate at up to 1MHz for more intense digital signal processing. For our product, we plan on using an MSP430G2x53 device, as it includes all of the built in devices required for the application intended to run on the device. The functional block diagram of the MSP430G2x53 is shown below in figure 5, from the MSP430G2x53, MSP430G2x13 Mixed Signal Microcontroller datasheet (Rev. J):



NOTE: Port P3 is available on 28-pin and 32-pin devices only.

Figure 5 Block Diagram of MSP430G2x53

The CPU of the MSP430 is designed for use with high-level coding languages like C, and the CPU can also access the full range of memory addresses without the need of a page table, which will enhance data look-up times. The CPU also supports the following features, from the MSP430x2xx Family User's Guide (Rev. J):

- RISC architecture with 27 instructions and 7 addressing modes.
- Orthogonal architecture with every instruction usable with every addressing mode.
- Full register access including program counter, status registers, and stack pointer.
- Single-cycle register operations.

- Large 16-bit register file reduces fetches to memory.
- 16-bit address bus allows direct access and branching throughout entire memory range.
- 16-bit data bus allows direct manipulation of word-wide arguments.
- Constant generator provides six most used immediate values and reduces code size.
- Direct memory-to-memory transfers without intermediate register holding.
- Word and byte addressing and instruction formats.

The CPU uses sixteen 16-bit registers, with twelve of those designed for general use. The block diagram for these registers is shown below in figure 6.

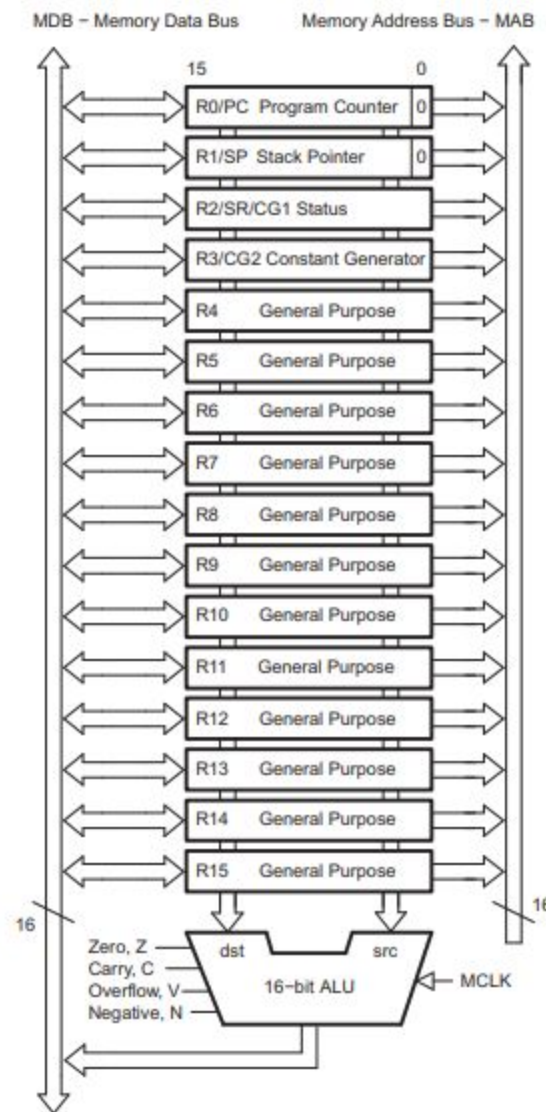
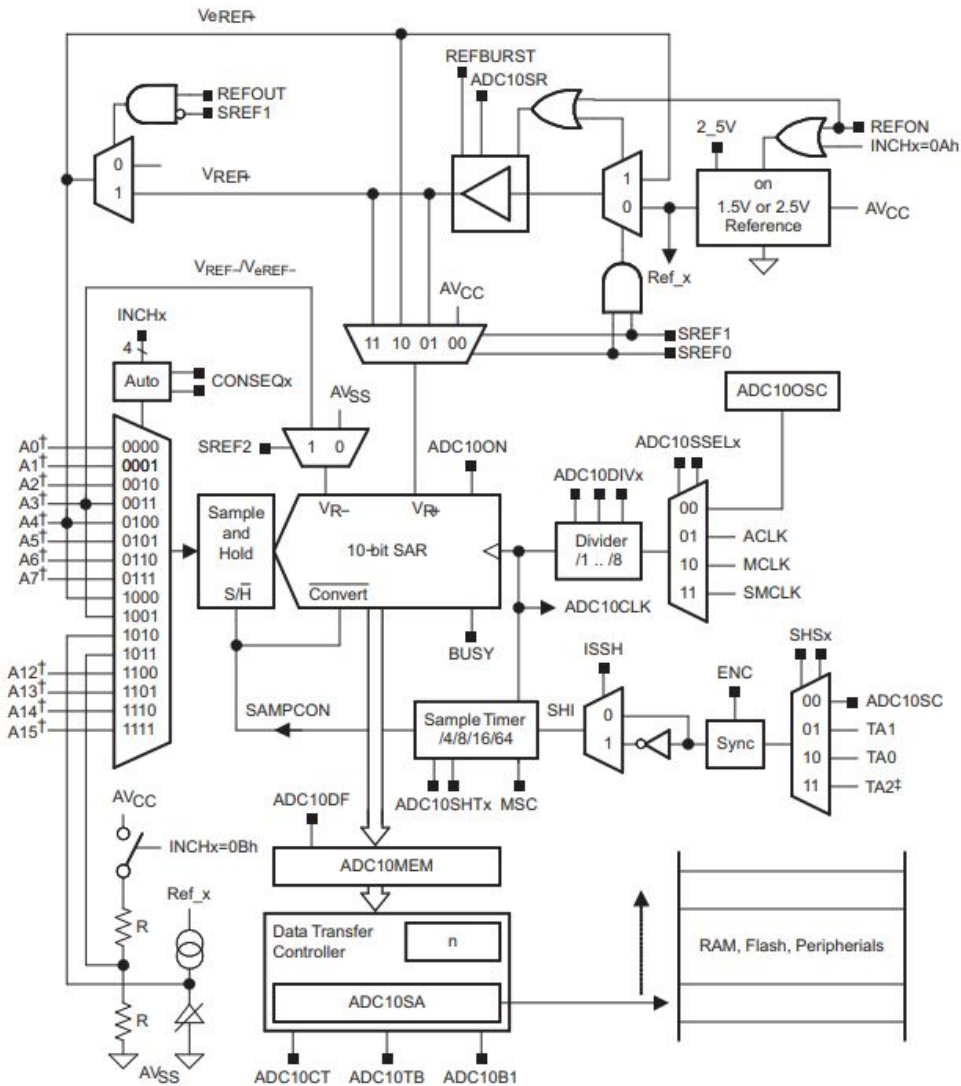


Figure 6: Registers used in the MSP430

Another very important feature of the MSP430 is the built-in analog to digital converter (ADC). The specific model we plan on using in our product is ADC10. ADC 10 is a 10-bit analog to digital converter, which uses a 10-bit successive-approximation-register (SAR) core for a small form factor, fast sampling speed, low power usage, and medium to high resolution. The block diagram showing how the ADC circuit works is shown below in figure 7, from the MSP430x2xx Family User's Guide (Rev. J):



†Channels A12-A15 are available in MSP430F22xx devices only. Channels A12-A15 tied to channel A11 in other devices. Not all channels are available in all devices.
 ‡TA1 on MSP430F20x2, MSP430G2x31, and MSP430G2x30 devices

Figure 7

This is ideal for our product, as the design requires sampling the sensors as fast as possible in an attempt to quickly and accurately determine the position of the eye. Listed below are the features of the ADC10 module from the MSP430x2xx Family User's Guide (Rev. J):

- Greater than 200-kSPS maximum conversion rate
- Monotonic 10-bit converter with no missing codes
- Sample-and-hold with programmable sample periods
- Conversion initiation by software or Timer_A
- Software selectable on-chip reference voltage generation (1.5 V or 2.5 V)
- Software selectable internal or external reference
- Up to eight external input channels (twelve on MSP430F22xx devices)
- Conversion channels for internal temperature sensor, VCC, and external references
- Selectable conversion clock source
- Single-channel, repeated single-channel, sequence, and repeated sequence conversion modes
- ADC core and reference voltage can be powered down separately
- Data transfer controller for automatic storage of conversion results

The ADC10 module has many options that can be configured by the software loaded onto the MSP430. The ADC converts analog signals into a 10-bit digital value that is then stored in the ADC10MEM register. We can then read the value from that register, and send it along a communication line to our processing unit to be used in the primary algorithm. The core uses two programmable reference voltages to define the limits of the ADC conversion. The conversion can be read as a 10-bit binary number unsigned or in 2's complement. The conversion formula is shown below:

$$N_{\text{ADC}} = 1023 \times \frac{V_{\text{IN}} - V_{\text{R-}}}{V_{\text{R+}} - V_{\text{R-}}}$$

The ADC uses a clock for both the sampling period and as a conversion clock. The clock must remain active until the end of the signal conversion, otherwise it will result in an error, and data cannot be used.

The ADC10 has a built-in voltage reference, and two different reference voltages may be selected. It can select between 1.5V and 2.5V as the reference voltage. External references may be applied through pins A4 (positive reference) and A3 (negative reference). When using external reference voltages, the internal reference voltage may be turned off in order to save device power. However, the

internal reference is designed for low-power operation. When the conversion rate is below 50kbps, the device buffer can be set to reduce power consumption by about half.

It is important to consider the sample timing for the ADC unit. The conversion is initiated at the rising edge of the sample input signal. The sample input period can also be adjusted based on the clock, and can be 4, 8, 16, or 64 clock cycle sample periods. The sampling timer sets SAMPCON to high for the selected sample period. When SAMPCON falls from high to low, the conversion begins, which takes 13 clock cycles, effectively allowing the programmer to determine how long each sampling cycle takes, as shown in the example clock diagram below (SHI is the sample input signal):

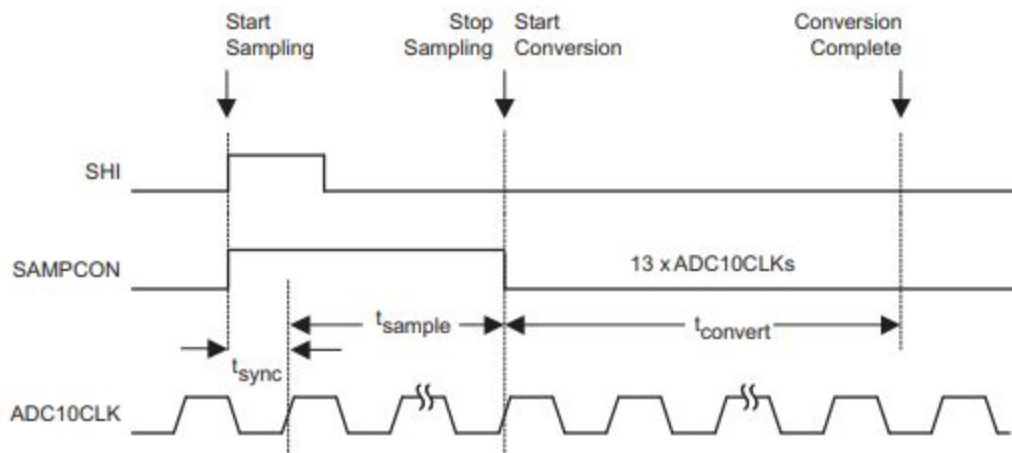


Figure 8: Clock Cycles Graph of ADC

Another important reason we chose to select the MSP430 for our project was the wide range of available communication protocols. The MSP430 supports Universal Asynchronous Receiver/Transmitter (UART), Serial Peripheral Interface (SPI), and Inter-Integrated Circuit (I²C). These are all the methods supported by the MSP430s universal serial communication interface (USCI). The MSP430 also has another form of peripheral communication called universal synchronous/asynchronous receiver/transmitter (USART) which supports UART and SPI as well.

USCI is separated into two modules, USCI_Ax and USCI_Bx. The USCI_Ax module supports UART mode, automatic baud rate detection for local

interconnect network communication, and SPI mode. USCI_Bx supports I²C mode and SPI mode as well. First we will discuss UART mode, as this may be the ideal form of communication with the central processing unit in our product. UART can be used to connect the MSP430 to an external system using 2 pins on the device, UCAXRXD for receiving data, and UCAXTXD for transmitting data to the device. The UCSYNC bit must be cleared for UART to function as it is an asynchronous communication protocol. UART mode includes the following features for communication, from [21]:

- 7- or 8-bit data with odd, even, or non-parity
- Independent transmit and receive shift registers
- Separate transmit and receive buffer registers
- LSB-first or MSB-first data transmit and receive
- Built-in idle-line and address-bit communication protocols for multiprocessor systems
- Receiver start-edge detection for auto-wake up from LPMx modes
- Programmable baud rate with modulation for fractional baud rate support
- Status flags for error detection and suppression
- Status flags for address detection
- Independent interrupt capability for receive and transmit

These wide variety of capabilities offers many diverse ways to approach communication problems that may be encountered when testing communication in our product. Like many of the features of the MSP430, the communication modes are designed to operate under low power usage, allowing us to extend the battery life of our product. One of the advantages of using UART mode is it's inherent asynchronous nature. The device allows the programmer to transmit data at a selected baud rate, and the transmitter and receiver will both operate at the selected baud rate. When only two devices are communicating with each other via UART, no multiprocessor formatting is required to communicate, and if more than two devices are talking, an addressing system may be put in place. Another feature of UART is automatic baud rate detection. This works by preceding a data frame with a synchronization sequence. This sequence consists of a break and a sync field. This automatic rate detection requires a specific UART configuration, where the data is in 8-bit form, least significant bit first, no parity, and single stop bit. This mode does not allow for addressing. The synchronization is based on the measurement between the first falling edge and last falling edge of the pattern. The automatic baud rate detection mode can be implemented in a full-duplex system. However, the device cannot transmit data while setting up the synchronization, and if a framing error occurs, the data transmitted during the time of that frame will be corrupted.

The USCI also supports SPI mode on both the USCI_Ax module and the USCI_Bx module. The MSP430 connects to external devices for this mode using either three or four pins: UCxSIMO, UCxSOMI, UCxCLK, and UCxSTE. According to the user's guide, the SPI mode is selected when the UCSYNC bit is 1 and the proper SPI mode is selected using the UCMODEx bits. The following SPI features are available on the MSP430, according to [21]:

- 7- or 8-bit data length
- LSB-first or MSB-first data transmit and receive
- 3-pin and 4-pin SPI operation
- Master or slave modes
- Independent transmit and receive shift registers
- Separate transmit and receive buffer registers
- Continuous transmit and receive operation
- Selectable clock polarity and phase control
- Programmable clock frequency in master mode
- Independent interrupt capability for receive and transmit
- Slave operation in LPM4

The architecture of SPI is commonly referred to as Master/Slave architecture, hence the wording used above. In our case, the MSP430 would act as the master device, as there is no need to receive input from the primary data processing unit.

3.3 Oculus Rift Research

To do any research in a goggle or helmet based eye tracking systems one must mention the this or otherwise there would not be justice if not to mention the fact of the strides by Oculus. With all their strides in eye tracking they showed that there is no real simple answer to the problem. For instance, most would think that the simple answer would be to use a camera but in this case as shown from Oculus below that just won't cut it. Oculus goes as far as to re-invent the idea of cameras to try and make the camera work in this case of the close environment of the goggle display unit.

3.3.1 Oculus Eye Tracking Based on Depth Sensing

One great example of Oculus VR eye tracking in terms of this project would be that of their patent on the product called the Methods and Devices for Eye Tracking Based on Depth Sensing, published on April 20, 2017, publish number:

US 2017/0109562 A1. The system described by the patten says that it is able to configure the distance between the device and the surface of the eye through a depth profiler, as shown in figure 9 below. In basic terms the system is a Charged Coupled Device, or CCD, made of IR receivers behind a display. In even more simpler terms they rebuild a camera with the added feature of a screen on top of the part of the camera that takes the picture. In terms of this project, they are close cousins in design whereby ours are an array around the eye, measuring magnitude in changes in around the eye, this one goes even further to reinvent the wheel to solve the problem.

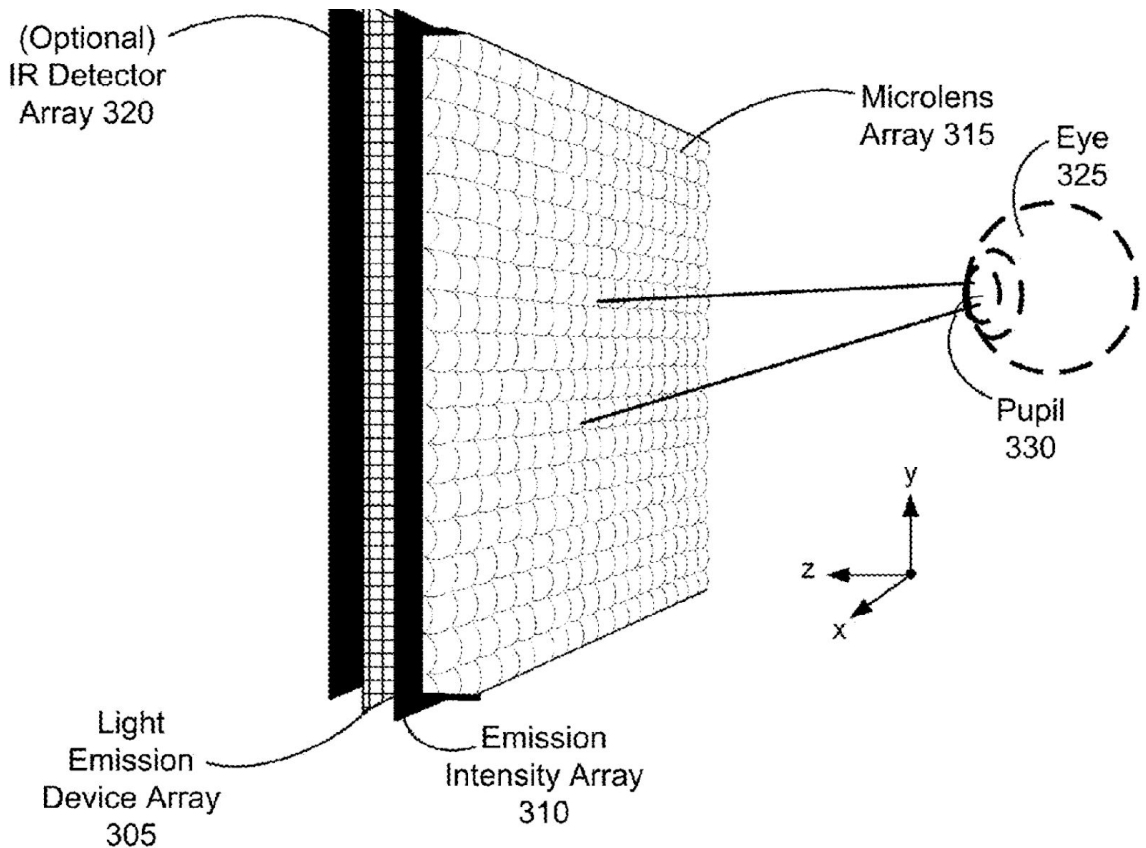


Figure 9 Eye Tracking Screen

3.3.2 Oculus Light Field Camera

Another example from the Oculus VR company in terms of eye tracking is their patten on Eye Tracking using a Light Field Camera on Head-Mounted Display, published on November 6, 2018 patent number US 10,120,442 B2. As seen

below in figure 10, this camera is able to be placed off to the side of the eye but still able to see the whole eye due to the unique nature of the light field camera. This camera as the patent describes it “captures a plenoptic image of the user’s eye”. Which it continues to say that a “plenoptic image includes light intensity data and the direction data of the captured light rays”. Basically this camera can take indirect 3d model images of the eye. By finding the normal to the center of the iris plane of the model it can track the line of sight indirectly from the sensor. This showed how much redesigning of the concept of the camera based tracking system is needed to just be able to solve the problem in this manner of using cameras for the problem. How this relates to our concept of eye tracking system is that our system runs on intensity but with fixed directions to measure while this system allows for extreme flexibility that goes way beyond our scope of the project at this current point in time

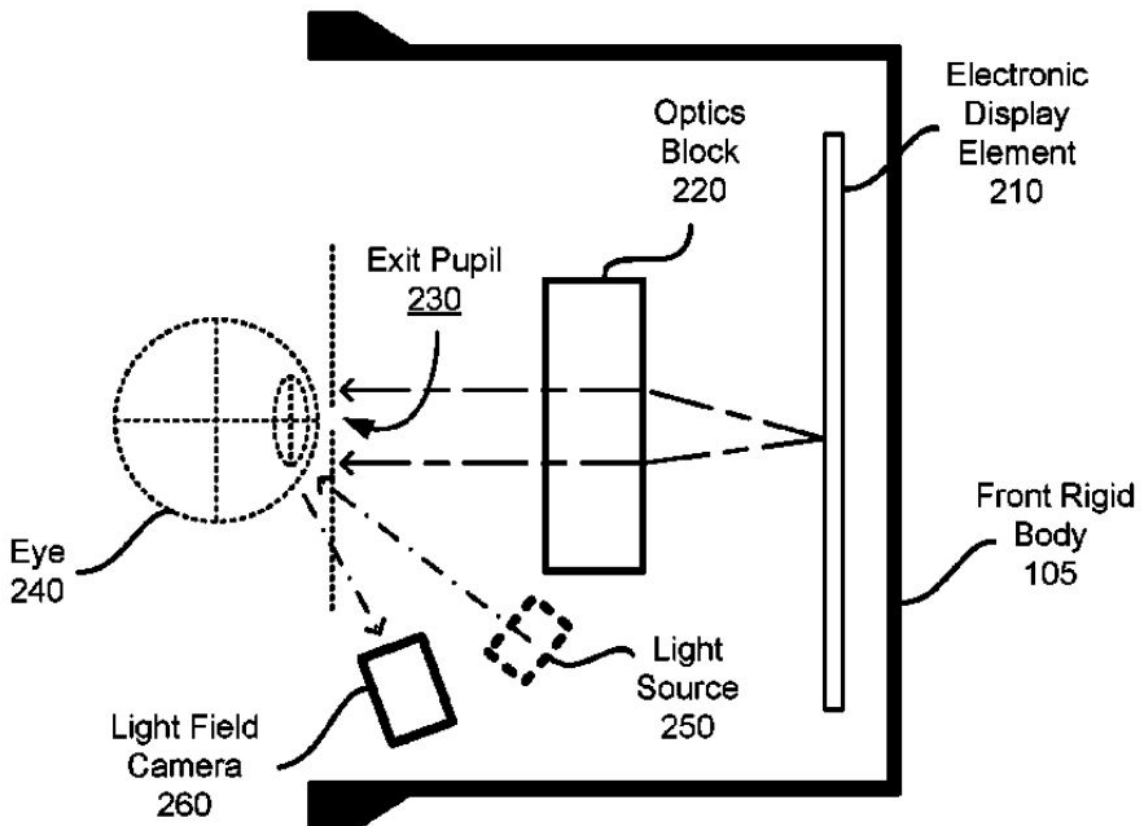


Figure 10: Light Field camera system

3.4 Eye-Tracking Methodology

Eye tracking methodologies have been the subject of academic research over a long period of time, as early as the beginning of the 20th century. Early eye-tracking relied primarily on electro-mechanical devices, like

Electro-OculoGraphy. This methodology was popular in the 1970's [Duchowski]. This method relies on voltage differences on the skin surrounding the eye, in the range of microvolts. It also only measures eye position relative to the head, as opposed to where the eye is looking in space, often referred to as point-of-regard. Another method utilizing a contact lens can very accurately determine where the eye is pointing in reference to the user's head, but this method is very intrusive and may be uncomfortable for the subject to use. As imaging devices became more commonplace, a method known as Video-OculoGraphy or Photo-OculoGraphy became popular. In early use of this methodology, photos or videos would be recorded of the subjects eye movements, and would then later be examined to determine where the subject was looking. As computer image processing became more available, this process would be automated through the use of image processing algorithms.

Most modern forms of eye-tracking stem from Video-OculoGraphy, in the form of technology that takes advantage of the pupil's position, and light reflection off of the corneal lens. By using two points of reference, the eye position can also be determined independent of the head, making this method useful for determining exactly where in space that the gaze is directed. The reflections created on the surface of the eye are called Purkinje Images, and there are at least four of them visible on the surface of the eye, as shown in Figure 11. In modern eye-tracking, these reflections are created using infrared light sources, because it does not affect what the user sees, but can easily be measured by infrared image sensors.

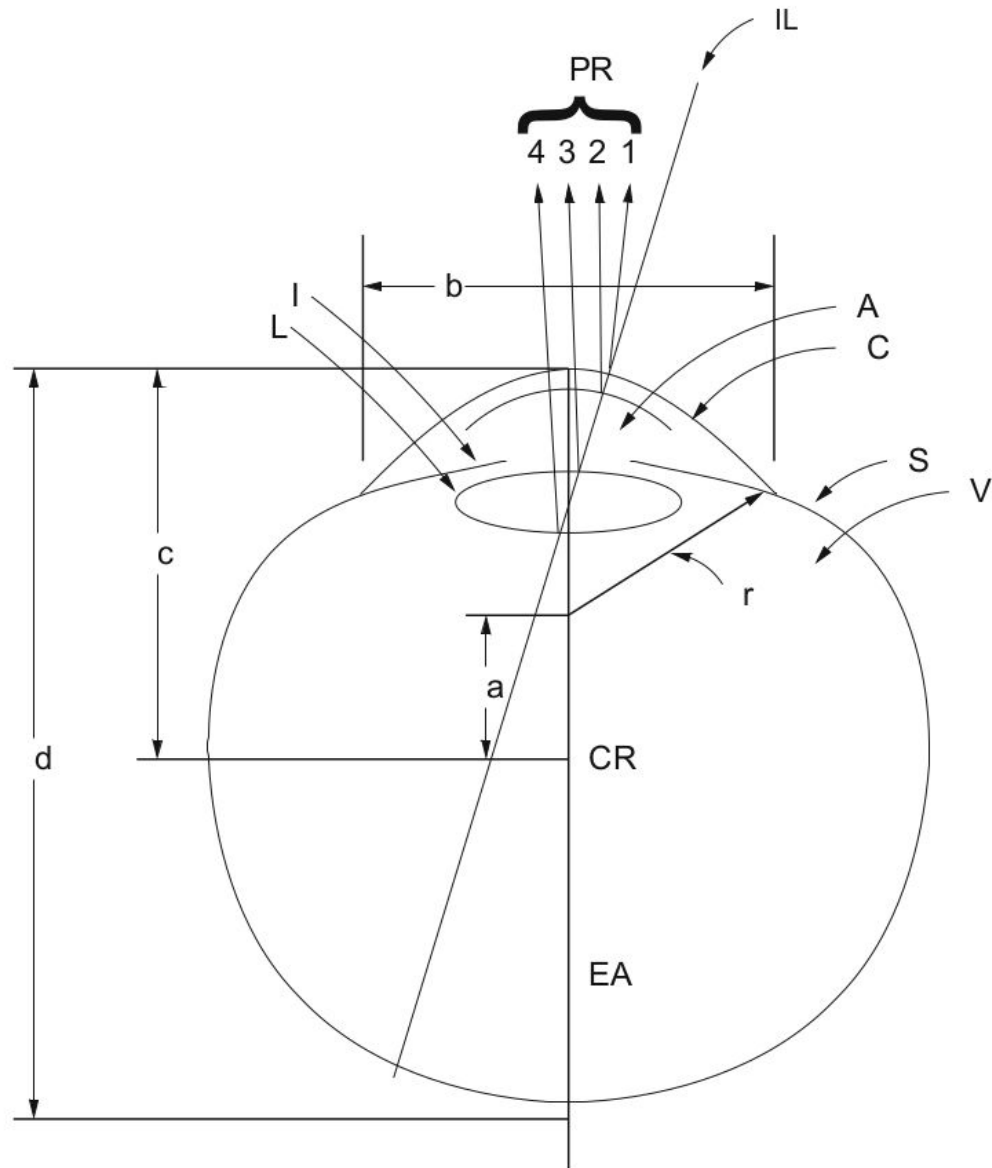


Figure 11: Purkinje Images

3.5 Existing Wi-Fi and Future of Wi-Fi

Wi-Fi was first invented in 1997 and it has developed a great deal in its over 20 years of existence. It is now one of the most ubiquitous or perhaps the most ubiquitous form of wireless communication around today. In this section we will

discuss some of the specifications of Wi-Fi and we will discuss our use of it for our project.

3.5.1 Frequency and Range of Current Wi-Fi Networks

Today there are 2 frequencies that are being used for communication with Wifi and devices that can connect to Wifi. One of these frequencies is set at 2.4 GHz. The other frequency being used is set at 5 GHz. Both of these frequencies have their pros and cons. The 2.4 GHz frequency has a larger range of communication than the 5 GHz frequency. The actual range of each of these Frequencies varies based on numerous factors. A typical home network using one wireless router with a frequency of 2.4 GHz has a range of about 150 feet indoors and about 300 feet outdoors. The range for a typical 5 GHz frequency home network is about 70 feet indoors and about 140 feet outdoors. 5 GHz frequency Wifi networks have about half the range of a 2.4 GHz frequency Wifi network, but the 5GHz frequency has a higher data transfer rate. Another notable thing about these frequencies is that certain devices are only compatible with some Wifi network types and not others. For example many older legacy devices like the Nintendo 3DS are not capable of picking up signals from a 5GHz frequency Wifi network.

3.5.2 Data Transfer Rates (Speed) for Current Wi-Fi

TABLE II. TRANSFER RATE OF VIDEO FOR DIFFERENT LOCATIONS

Performance	Video 2160p (4k video)					
	Point 1		Point 2		Point 3	
IEEE 802.11	n	ac	n	ac	n	ac
Elapsed time	01:49.1	01:58.9	01:52.7	01:53.5	40:01.5	-
Maximum transfer rate (Mbps)	44.1	55.4	28.9	48.3	11.2	-
Average transfer rate (Mbps)	19.2	19.4	16.7	16.6	942.8	-

Table 2: from "Performance Comparison from IEEE Wi-Go 802.11ac and Wi-Fi IEEE 802.11n" [10]

This table shows the data transfer rate between 3 devices at different ranges from the access point over a similar amount of elapsed time shown above. The first device is on the third floor of a building with a distance to the access point of less than 10 meters. The second device is located on the third floor as well with a distance between 10 and 35 meters from the access point. The third device is located on the second floor of the same building and has a distance between 35 and 50 meters from the access point. Point 1, Point 2, and Point 3 refer to the first device, the second device, and the third device respectively. In Point 1 and Point 2 we can see that 802.11ac and 802.11n perform similarly. In Point 3 802.11ac can't be picked up by the device and 802.11n runs at an average data transfer rate of 942 Kbps.

The data rate of 802.11ac varies with range. "Up to a distance of 17.1 meters the throughput was above 500 Mbps (Megabits per second) with a PER (package error rate) lower than 1%. With proper antenna orientation, the maximum range achieved was of 24.3 meters, with a throughput of 90.1 Mbps." [9] We also have

to be mindful of compatibility between our device and the various Wifi network types that are commonly available today. We might make our device pick up only either 2.4 GHz Frequency or the 5 GHz frequency. Depending on the development of the newest Wifi standard, we may even switch from one of the more common network types today to the newest Wifi standard that should be deployed soon.

TABLE I. TRANFER RATE OF VIDIO

Performance	Video resolution type					
	2160p (4k Video)		1440p (2k video)		1080p	
IEEE 802.11	n	ac	n	ac	n	ac
Elapsed time	16.3	13.2	03:46.8	03:40.4	02:02.8	01:56.4
Maximum transfer rate	42.5 Mbps	55.4 Mbps	37.6 Mbps	42.4 Mbps	42.7 Mbps	55.5 Mbps
Average transfer rate	30.2 Mbps	17.6 Mbps	6.4 Mbps	6.8 Mbps	14.3 Mbps	36.8 Mbps

Table 3: from "Performance Comparison from IEEE Wi-Go 802.11ac and Wi-Fi IEEE 802.11n" [10]

The Table above shows the max and average data transfer rate over the elapsed time shown above of both 802.11ac and 802.11n Wi-Fi Networks for Video Streaming. For 4k video 802.11n has a higher data transfer rate than 802.11ac. For 2k video the average transfer rates are about the same. For 1080p video the average transfer rate for 802.11ac is higher.

3.5.3 Future of Wi-Fi 802.11ax (Wi-Fi 6)

802.11ax has been designed with a little bit of a different goal in mind from 802.11n and 802.11ac. 802.11ax aims to mainly increase the average data transfer rate instead of mainly increasing the raw max data transfer rate. 802.11ax has received the name of High Efficiency WLAN as a result of this. Through the use of OFDMA, 802.11ax can handle multiple clients on a given access point much better than previous Wifi networks. This will be of great use in areas in which a large number of clients are connected to a single access point. A few examples of this are airports, hospitals, stadiums, libraries, and classrooms.

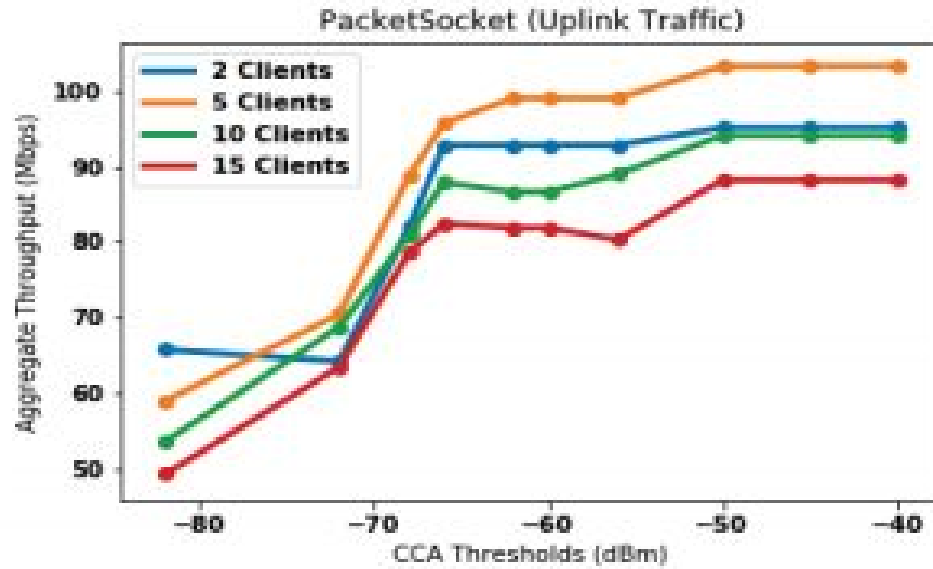


Figure 12: from “How much can we reuse? An empirical analysis of the performance benefits achieved by spatial-reuse of IEEE 802.11ax”

“Significant improvement (50-79%) in aggregate throughput is achieved using dynamic CCA threshold adjustment. Nearly optimal aggregate throughput is observed for CCA values between -66 dBm and -50 dBm.” [11]

3.5.4 Comparing and Choosing Amongst Different Wi-Fi Networks

Wifi Network Types	Frequency	Bandwidth	Theoretical Max Speed AKA Data Rate	Realistic Speeds AKA Data Rate (Dependent on Range)	Method of Transmission
802.11n (Wifi 4)	2.4 GHz and 2.5 GHz	20 MHz And 40 MHz	Up to 540 Mb/s	30 Mb/s	Orthogonal Frequency Division Multiplexing (OFDM)
802.11ac (Wifi 5)	5GHz	40 MHz 80 MHz 160 MHz	Up to 6 Gb/s	100 Mb/s	Orthogonal Frequency Division Multiplexing
802.11ax (Wifi 6)	2.4 GHz and 5 GHz Possibly 6 GHz	20 MHz 40 MHz 80 MHz 160 MHz	Up to 9 Gb/s	N/A	Orthogonal Frequency Division Multiple Access (OFDMA)

Table 4: Comparisons of Wifi Networks

The table above compares all of the Wi-Fi Networks under consideration for our project. Here we compare the Frequencies, Bandwidth, Theoretical Max Speed, Realistic Speeds, and Method of Transmission. For our project we are planning to use the best overall 802.11n Wi-Fi network. Wi-Fi has the highest data transfer rates out of all of the common forms of Wireless Communication (Wi-Fi, Bluetooth, and Zigbee). This higher data transfer rate will allow us to stream video with the best quality out of all of the forms of wireless communication. Currently the 802.11n Wi-Fi network is the most commonly available Wi-Fi network for use. It also has a solid data transfer rate of about 30 Mb/s realistically which will be able to handle video streaming.

3.6 Bluetooth Communication

Bluetooth was originally developed over 20 years ago in 1994 by Dr. Jaap Hartsen. Bluetooth has become a mainstay in most devices today. You can find bluetooth communication in cell phones, cars, tablets, video game consoles and much more. Bluetooth communication is used by millions and possibly billions of devices worldwide for wireless communication. It is used mainly by portable devices for wireless communication. Any kind of file can be transmitted using Bluetooth be it photos, music, videos, documents, etc. Bluetooth has been mostly used as a low power alternative for wireless communication to Wi-Fi. Bluetooth also has a shorter range of function than Wi-Fi and is used mostly to transfer data that does not require a huge amount of throughput for good performance. The latest version of Bluetooth is Bluetooth 5.1. We will discuss Bluetooth 5.1 a bit later, but for now we will discuss the big update to Bluetooth that happened recently which was the version update to Bluetooth 5.0. Bluetooth 5.0 was released 6 years after Bluetooth 4.0 in late 2016. Bluetooth 5.0 offers a slew of enhancements to its specifications. In this section we will be discussing the specifications of Bluetooth and possible implications on our project if we decide to use it.

3.6.1 Frequency and Range of Current Bluetooth

There are three distinct classes of devices that are equipped with Bluetooth. Class 1, which is the most powerful class has a range of up to 100 meters. Class 2, which is the most common class found today has a range of about 10 meters. Class 3, which is the least common class today has a range of less than a meter. The table below shows a comparison of the specifications between the various Bluetooth versions.

Feature	Bluetooth Classic	Bluetooth 4.x	Bluetooth 5
Radio frequency (MHz)	2400 to 2483.5	2400 to 2483.5	2400 to 2483.5
Distance/range (meters)	Up to 100	Up to 100	Up to 200
Medium access technique	Frequency hopping	Frequency hopping	Frequency hopping
Nominal data rate (Mb/s)	1-3	1	2
Latency (ms)	< 100	< 6	< 3
Network topology	Piconet, scatternet	Star-bus, mesh	Star-bus, mesh
Multihop solution	Scatternet	Yes	Yes
Profile concept	Yes	Yes	Yes
Nodes/active slaves	7	Unlimited	Unlimited
Message size (bytes)	Up to 358	31	255
Certification body	Bluetooth SIG	Bluetooth SIG	Bluetooth SIG

Table 5: Comparison of Specifications Between the recent versions of Bluetooth

As seen in the table above Bluetooth 5.0 doubles the effective range of wireless communication from the previous version of Bluetooth 4.x. The max range of Bluetooth 5.0 sits at about 200 meters while the previous version of Bluetooth had a max range of 100 meters. In Bluetooth 4.x the range of the connection indoors was about 10 meters. Bluetooth 5.0 quadruples the range with a range of 40 meters indoors. This massive increase in range indoors means that there are less access points needed. This results in a reduction of consumption power needed for Bluetooth devices. This is critical to making Bluetooth communication a primary form of wireless communication for the Internet of Things (IoT) and it could even lead to Bluetooth replacing Wi-Fi and Zigbee for many IoT applications. Bluetooth 5.0 comes with a new connection type. Bluetooth 5 Coded is the new connection type that has been added and its goal is to provide long-distance connections with a lower bit rate. The primary objective of this connection is a broader range over speed.

3.6.2 Data Transfer Rate of Bluetooth (Speed)

The speed of Bluetooth 5.0 has been a major upgrade in comparison to the previous version. As you can see from Table 5, the maximum speed of Bluetooth 5.0 is 2 Mb/s and the previous version Bluetooth 4.x is about 1 Mb/s. Although

Bluetooth 5.0 has doubled its throughput when compared to its previous version of Bluetooth 4.x, the actual time it takes to complete a data transfer is about 1.7 times faster. The reason for this is that the gap between packets that are sent has not been reduced. This means that we are not sending data at a higher frequency, we are just able to process more data at once. As shown in Table 5, the size of the messages that we are able to send has increased to 255 bytes in Bluetooth 5.0 from 31 bytes in Bluetooth 4.x. This allows us to transfer data with more substantial information such as URLs or telemetry data. [23]

3.6.3 Consideration of Use of Bluetooth on the Project

Although Bluetooth has come a long way, we will most likely not be using it for the project. We are planning on utilizing video streaming for our project and Bluetooth does not provide a powerful enough data transfer rate in order to justify its use. If we were even able to get streaming to work the overall quality of the video would be very poor. This would be unacceptable and we would not be delivering a quality product. We are also using wireless communication to transfer data from our Infrared Light Sensors to our MSP 430. Bluetooth could be used for this purpose, but it looks as if Zigbee can do the same with less power consumption and more security. If this isn't the case with Zigbee then we will consider using Bluetooth for transferring the data received from our Infrared Light Sensors.

3.7 Zigbee Wireless Communication

Zigbee is a form of wireless communication based on the IEEE 802.15.4 architecture and specifications. Zigbee is used as a low power and highly secure alternate form of wireless communication to Bluetooth and Wi-Fi. Zigbee is typically used in devices that require a long battery life or that require a good amount of security. Devices that use Zigbee includes low power radios, medical data collection devices, Thermostats, Light Sensors, etc. These devices are considered to be IoT (Internet of Things) Devices. These devices communicate data between each other and that then affects how they function. Zigbee is leading the way in terms of use in these IoT devices because of its security and low power consumption. By the year 2020 it is projected that about 8 billion devices around the world will have some form of wireless communication AKA be an IoT device. For this reason Zigbee is exceedingly relevant and it is why we will be discussing it. In this section we will discuss many of the specifications of Zigbee and whether or not it will be of use for our project.

We will be going over the network of Zigbee here. Below in Figure X is a model of the architecture used by Zigbee for its network. The technical components of the Zigbee network model are the Coordinator, Router, and End Device. Every Zigbee network must have a coordinator. The coordinator manages the overall network and it provides security control of the network. The router in a Zigbee

network acts as an intermediate node between the coordinator and the end device. The router handles traffic between the end device and the coordinator. The end device is something that is most familiar to most people. An end device can be a smart light bulb, a thermostat, a sensor of some sort, and many other kinds of devices. The end device handles traffic between itself and its parent node and it can enter a low power mode or sleep in order to conserve power.

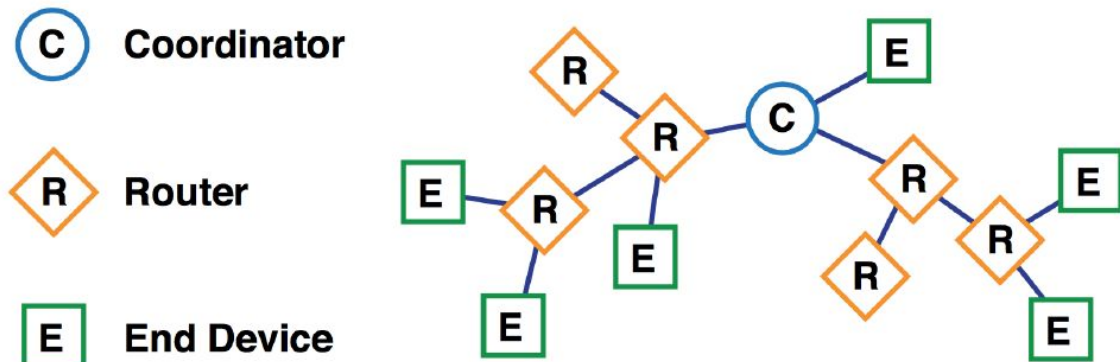


Figure 13: Model of the Zigbee Network

3.7.1 Frequency and Range of Zigbee

Zigbee has various modes of frequency that it can operate in. Table X shows all the different frequencies that Zigbee operates in. These frequency modes of operation also vary depending on region.

Band	868 MHz	915 MHz	2450MHz
Region	EU, Japan	USA	Worldwide
Channels	1	10	16
Data rate	20 kbps	40 kbps	250kbps

Table 6: Ranges of Frequencies Zigbee can operate in

“ The access mode for all of these frequency bands is Direct Sequence Spread Spectrum (DSSS). The 2450 MHz band employs Offset Quadrature Phase Shift Keying (O-QPSK) for modulation while the 868/915 MHz bands use Binary Phase Shift Keying (BPSK).” [26] As shown in table 6 the data rate for Zigbee varies depending on the region. In the USA the frequency band of operation for Zigbee is 915 MHz and it has a data rate of 40 kb/s. The data rates shown above in Table 6 are under optimal performance conditions. We will discuss more realistic data rates in the next subsection.

The range of most Zigbee devices is between 10 and 100 meters. This range is dependent on environmental factors and the Zigbee protocol that is being run. The range for Zigbee devices is about 10 meters indoors and the theoretical max of 100 meters is an outdoor environment in which the devices maintain line of sight. We looked at the paper “Testing Communication of Range Zigbee Technology” [29] in order to help evaluate the realistic range of Zigbee devices. In this paper an experiment was conducted to assess the communication range of Zigbee devices. The Zigbee devices used in this paper were the XBee Module Series 1 and the XBee Module Series 2. In a room to room attempt to communicate between devices there was a sharp dropoff in the number of packets received from one device to the other. These are the conditions that this room to room testing was conducted under.

- At 0 meters there were no barriers
- At 5 meters the barrier was a one Curtain wall
- At 10 meters the barrier was one Curtain wall and one Structure wall
- At 15 meters the barrier was two Curtain walls and one Structure wall

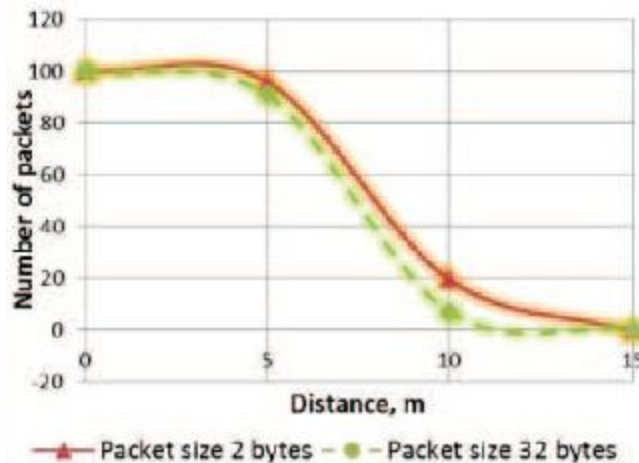


Figure 14: Graph Showing Zigbee Device Performance with Varying Distances Room to Room

Figure X in the previous page shows how the performance of the Zigbee Device varies with distance. Unfortunately the quality of the the text graph is very poor so we will explain all the various parts of the graph in detail. The Vertical axis of the graph is the Number of packets being sent at a given time and the Horizontal Axis is the distance between the transmitting and receiving device. The Green Line on the graph indicates that the size of the packets being sent is 32 bytes. The Red Line on the graph indicates that the size of the packets being sent is 2 bytes. The number of packets being sent begins to drop off at a distance of around 5 meters. At 10 meters, the number of packets being sent has decreased

dramatically from about 100 packets to about 10 to 20 packets. At 15 meters, the Zigbee devices fail to establish connection. It is pretty clear to see from this graph that the effective range of these devices is about 10 meters and anything beyond that requires line of sight between the devices.

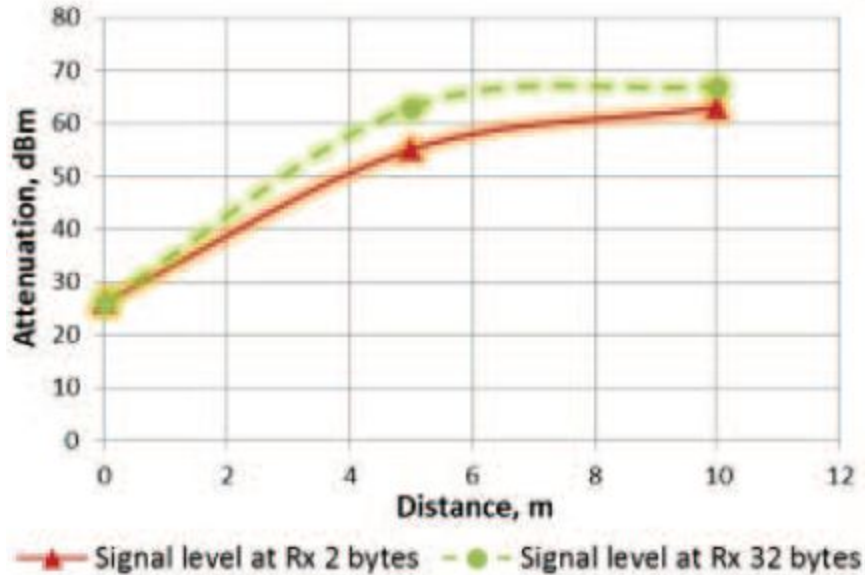


Figure 15: Graph Showing Attenuation Level with a Variation in Distance Between Zigbee Devices

This graph is comparing the Attenuation level between the transmitting Zigbee device and the receiving Zigbee device. Once again because of the poor quality of the graph we will be going in detail breaking down the details of the graph. The vertical axis is for the Attenuation levels and the horizontal axis is for the distance between the receiving and transmitting devices. The red line on the graph indicates that the signal level at Rx is 2 bytes and the green line on the graph indicates that the signal level at Rx is 32 bytes. Attenuation is the strength of the signal being transmitted between the transmitting device and the receiving device. Attenuation is measured in dB (decibels). The higher the value for our Attenuation then the more distorted our signal is. At 0 meters, the Attenuation level for both signal levels is about 25 dB. At 2 meters, the attenuation level for the signal level held at 32 bytes is a little bit above 40 dB and the attenuation level for the signal level held at 2 bytes is little bit below 40 dB. At 4 meters, the attenuation level for the signal held at 32 bytes is slightly below 60 dB and the attenuation level for the signal held at 2 bytes is 50 dB. At 6 meters, the attenuation level for the signal held at 32 bytes is about 65 dB and the attenuation level for the signal held at 2 bytes is slightly below 60 dB. After 6 meters the attenuation levels for both the signal held at 2 bytes and 32 bytes do not increase enough to be of note. According to these attenuation levels, the appropriate range of use for these devices is about 6 meters.

3.7.2 Data Transfer Rate of Zigbee (Speed) and Power Consumption of Zigbee Devices

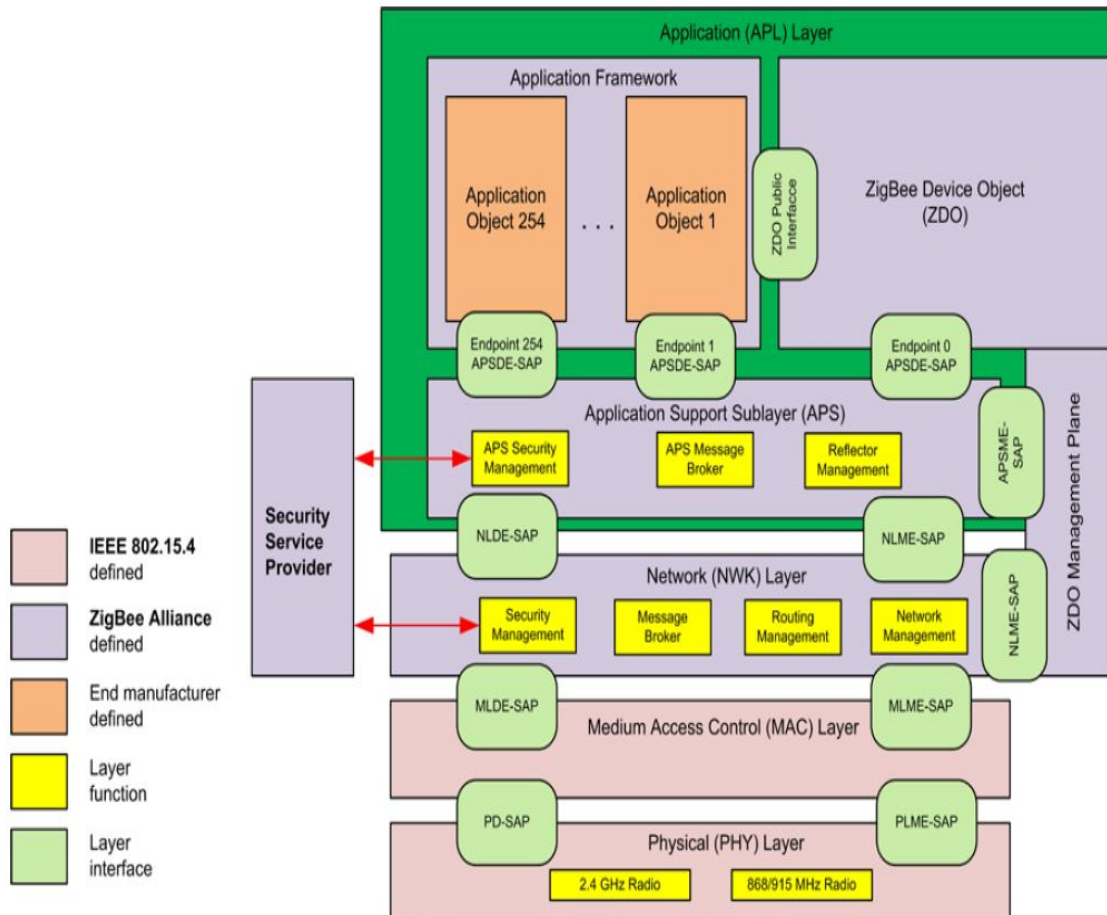


Figure 16: Zigbee Architecture

The figure 16 shows the architecture for Zigbee devices. This architecture is based off of the architecture for IEEE’s 802.15.4 Low-Rate Wireless Personal Area Networks. Everything above the NWK Layer in Figure X is a part of the application layer shown in Figure X2. Zigbee builds the NWK Layer and Application layer on top of the PHY and MAC Layers shown in Figure X2 which is a part of the IEEE 802.15.4 Architecture.

Technology	Power use per hour	Device	Comparison
Wi-Fi	2 kilo Joule	Laptop	
ZigBee	20 milli Joule	Set top box	1/100,000 compared to Wi-Fi
Green Power	200 micro Joule	Light switch	1/100 compared to ZigBee

Table 7: Comparison of Power Consumption Between Wi-Fi, Zigbee and Green Power

Table 7 shows the power consumption of devices using different forms of wireless communication. Zigbee communication uses 100,000 times less power than Wi-Fi communication and Green Power uses 100 times less power than Zigbee Communication. Green Power is a feature of Zigbee 3.0. Zigbee 3.0 is the latest version of Zigbee. With Green Power, devices using just double A batteries can last years.

To help assess the speed of Zigbee, we looked at a paper that focused on testing the speed of Zigbee using WSN (Wireless Sensor Networks). The WSN system is built based upon modularity design. In the system there are two types of nodes sensor nodes and coordinator nodes. The sensor nodes detect temperature and humidity of the environment. The coordinator node is used to collect the data readings from the sensor nodes and sends the data to the user. The Zigbee based device used was operating in the 2.4 GHz frequency band with a theoretical max data rate of 250 Kb/s. This experiment was performed in order to see how baud rate affects the latency of communication. "The baud rate is defined as processing data rate in serial communication of Xbee module" [28]. The baud rate is set at various values such as 9600, 19200, 38400, and 115200 b/s (bits per second). The maximum throughput was 19.2 Kb/s while the baud rate was set at 115200 b/s, which is a far cry from the max theoretical rate of 250 Kb/s. The results from the experiment can be seen in Figure 17. Obviously the higher the baud rate, then the more throughput the experimenters were able to attain. The baud rate in the Figure X below is denoted by the shapes along the lines of the graph. So the X denotes 9,600 b/s, the square denotes 19,200 b/s, the diamond denotes 38,400 b/s, the triangle denotes 57600 b/s, and the circle denotes 115,200 b/s.

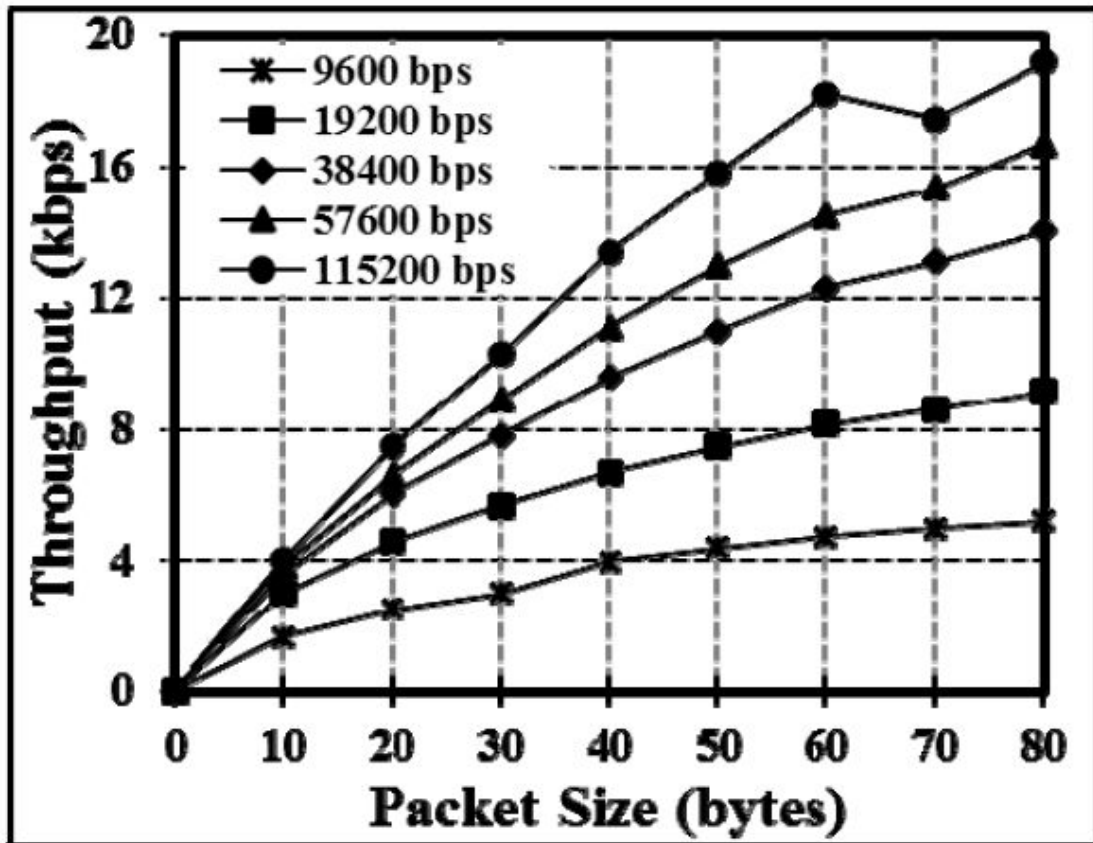


Figure 17: Throughput vs Packet Size Utilizing Different Baud Rates Graph of the Results from the WSN Experiment

3.7.3 Security of Zigbee

Since Zigbee is one of the few wireless communication forms that touts its security capabilities, we will delve into just how secure it is and if it is secure then why so. The security used in Zigbee is based on symmetric-key cryptography. Symmetric-key cryptography is when two users use a single key to encrypt and decrypt plain text. “Zigbee uses the highly secure 128-bit AES-based encryption system” [27]. Zigbee assumes an “open trust” model between layers in its architecture. This means that the layers in the protocol stack as shown in Figure X trust each other and do not have any security measures between layers. The cryptographic protection only applies between devices, but not between layers. The reason Zigbee does this is to keep a low cost. Zigbee devices use a network key and link key to communicate. A network key is a 128-bit key shared by all devices in the network and is used for broadcasting communication. A link key is shared between two devices and is a 128-bit key. “The Zigbee standard implements strong security features. The AES algorithm for data encryption and

data authentication provides Zigbee encryption with sufficient robustness, but the security depends on the secrecy of the encryption keys, which may be breached during the keys' initialization or distribution.” [27]

3.7.4 Consideration of Zigbee Use for our Project

We will consider integrating Zigbee into our device. The immense benefits of the low power consumption would be very helpful in keeping the battery life for our project long lasting. Zigbee would be used to help send information from our Infrared Detecting Sensors to our device that will be wearable. This data should be of a small enough size so that Zigbee can handle it. The integration of Zigbee into the device would be of low cost relative to Bluetooth and Wi-Fi. Although these qualities are great, the poor range of Zigbee may be a reason we do not use Zigbee. Our wearable device is obviously designed to be portable so the distance to an access point can vary a great amount.

3.8 Machine Learning/Neural Networks

Our proposed device does not operate like most standard eye-tracking devices. While special consideration has been given to both past and modern eye-tracking methodologies, it is apparent to the team that our device will not operate on the same kind of algorithms. Most eye-tracking algorithms require detailed images of the eye to be captured. This is due to the fact that that most modern eye-tracking relies on using the center of the eye, combined with a reflection off of the cornea, to create a vector by which the angle of the gaze is calculated.

Our goal is to not become dependent on using expensive image capturing sensors, or need to write overly complex algorithms to determine simple eye position. To solve the problem of the complex algorithms usually required to determine eye position, we propose a new method: machine learning. This would still use the information given to us by past methodologies, specifically the dark-pupil tracking technique. Because of the nature of the dark-pupil tracking, each of the different eye positions we would want to observe should present a unique pattern in the sensor levels we observe. Normally, we would need to manually observe these different patterns, and from that derive an algorithm that could then interpret our findings to produce an eye position. Instead, we propose training a neural network to pick up those patterns, instead of a human observer.

3.8.1 How a Neural Network Works

The basic way to think about a neural network is in three layers: the input layer, the hidden layer, and the output layer. These layers are made of nodes, called neurons. Each layer is fully connected to the next with a weight attached to each connection. In the hidden layer, a nonlinear function is applied to each value before being sent to the next layer. These nonlinear functions are applied so that the Neural Network can follow nonlinear patterns.

3.8.2 Activation Functions

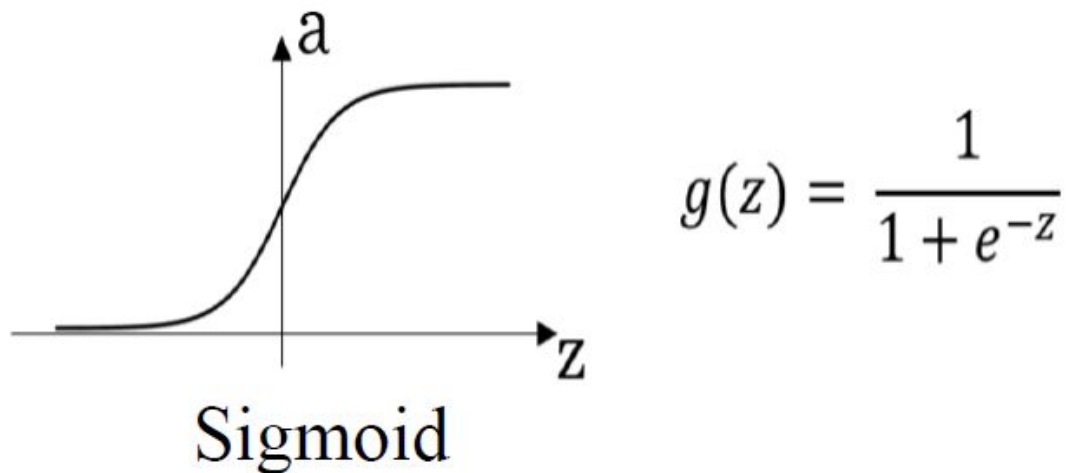


Figure 18: Graph of Sigmoid Function

This is the sigmoid activation function. The sigmoid function has a good way of interpreting the firing rate of a neuron in a neural network. The interpretation works as follows, the value of 0 refers to a Neuron not firing at all and a value of 1 refers to a Neuron firing at its maximum frequency. However this activation function is not very popular nowadays because of 2 major drawbacks.

1. Sigmoids saturate and kill gradients. When the Neuron activates at either 0 or 1 there almost no gradient in the sigmoid function. This leads to the whole backpropagation process failing because when this gradient is multiplied through then this effectively makes it so that no signal will flow through the neuron to its weights and recursively to its data.
2. Sigmoid outputs are not zero-centered. The data being received by neurons in later layers would not be zero-centered. This leads to the gradients on the weights of the neural networks being always positive or always negative through the process of backpropagation which adversely affects the gradient descent for the Neural Network.

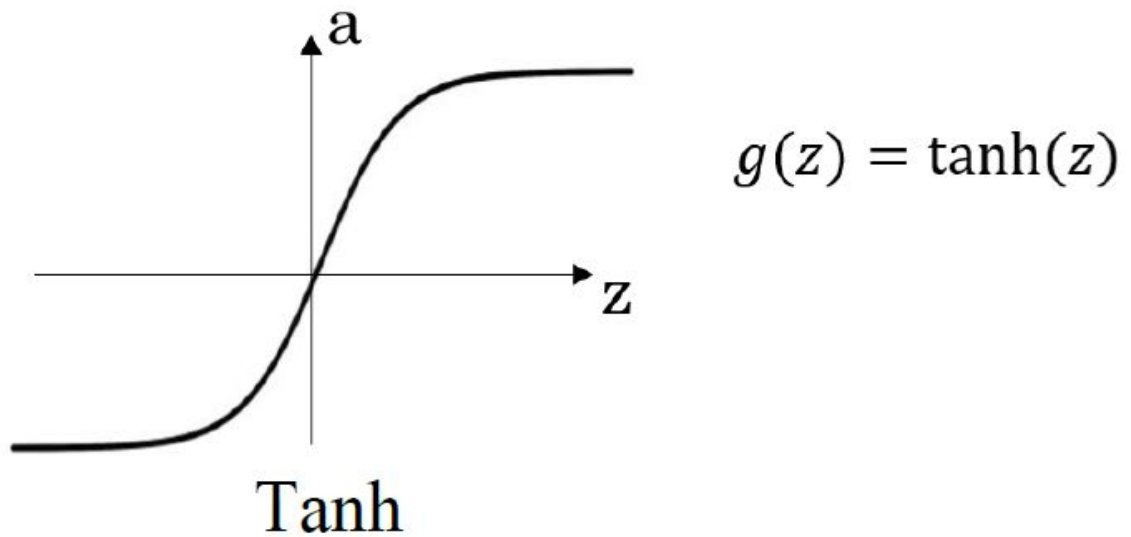
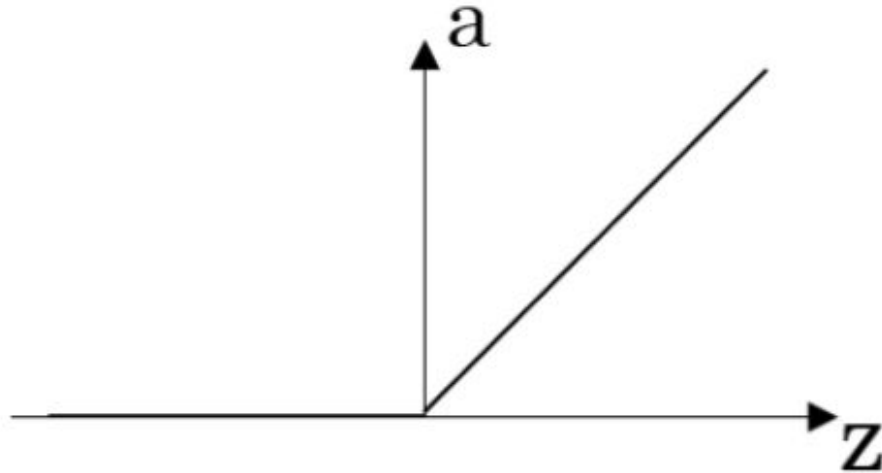


Figure 19: Graph of the Tanh Activation Function

The image above is the Tanh activation function. The Tanh Activation function is much like the sigmoid function except that It forces a real number into the output range of $[-1, 1]$. Unfortunately the Tanh function also has the same problem of gradients that saturate. However it does not suffer from the problem of its output not being zero-centered. As a result the Tanh function is always preferred when compared to the sigmoid function.



ReLU

$$g(z) = \max(0, z)$$

Figure 20: Graph of the ReLU Activation Function

The image above is an example of the of the Rectified Linear Unit (ReLU) Activation Function. This activation function has become very popular over the past few years. Like the other activation functions it has various pros and cons

Pros:

- It has been found to hasten the convergence of stochastic gradient descent when compared to the sigmoid and tanh functions.
- The ReLU function can be implemented by thresholding a matrix of activations at zero. These computations are much less intensive than the ones required by the tanh and sigmoid function which require exponentials and other things.

Cons:

- ReLU functions are fragile during training and particular neurons can “die”. This means that certain neurons will never activate because of the updates to the weights. This typically occurs when a large gradient is passing through a ReLU neuron. This can result in about 40% of the neural network being “dead” meaning that 40% of the neurons in your network will never activate.

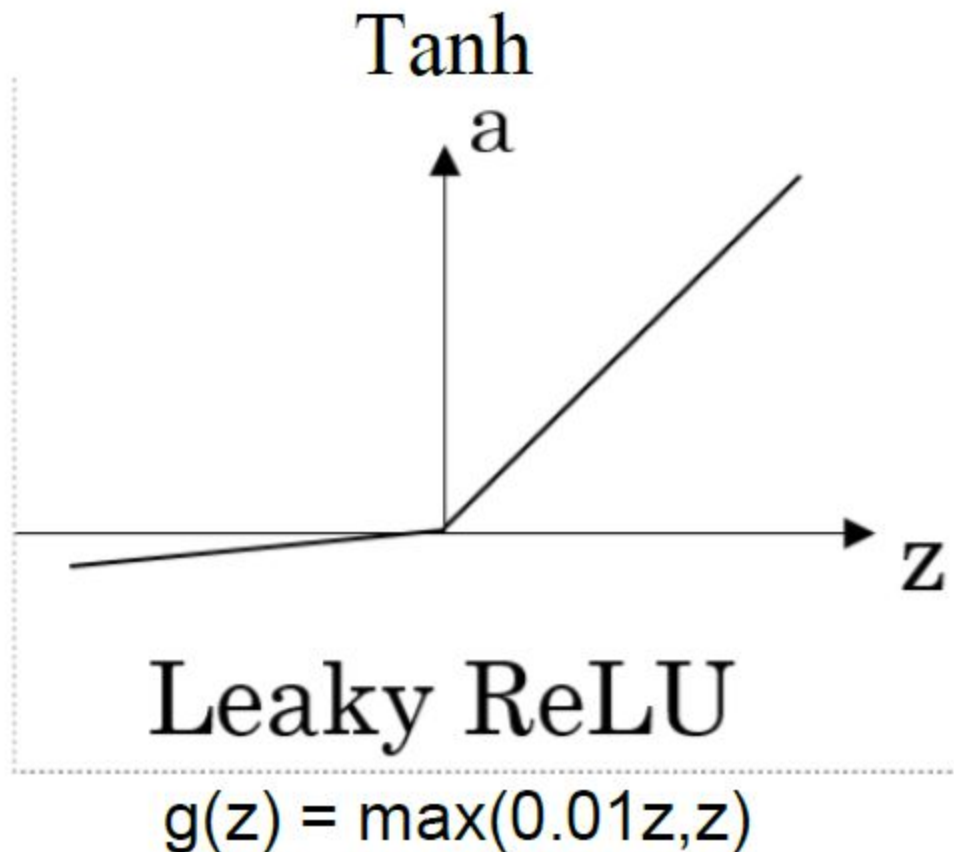


Figure 21: Graph of the Leaky Rectified Linear Unit (ReLU) Activation Function

The Leaky ReLU activation function is basically an attempt to fix the problem with the regular ReLU function. For when x is less than zero there is a slightly negative slope. The negative slope is typically about 0.01. For when x is greater than zero the Leaky ReLU function behaves exactly the same as the original ReLU function. There is no consensus as to whether this function actually solves the problem with the ReLU function. There have been cases that indicate that the Leaky ReLU function helps alleviate the "dead" neuron problem, but there also have been many cases in which it has not done much to alleviate this problem. There are many other proposed activation functions but since they are not as popular as these four that we have discussed we will not be considering them for our project. For the neural network that we plan to implement in our project we will be using the Leaky ReLU activation function as it shows the most promise out of all the functions we have discussed.

3.8.3 Basic Layout of a Neural Network

A diagram of the entire Neural Network is shown below, to help the reader get a better understanding.

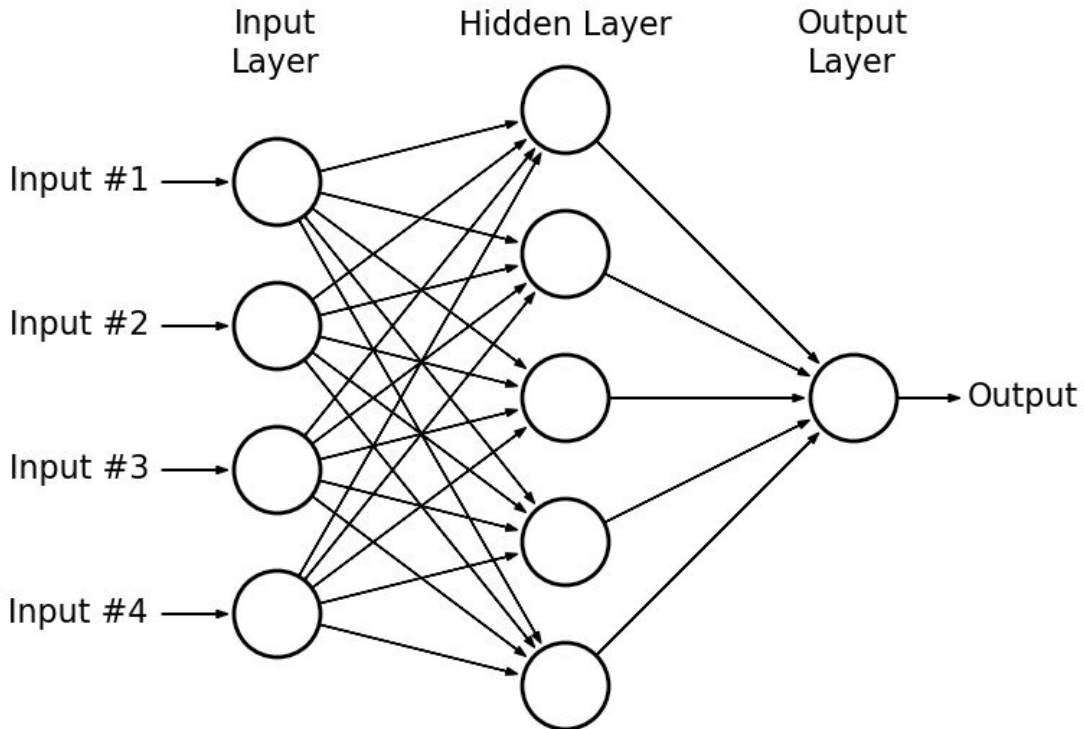


Figure 22: A basic layout of a neural network

The first step in creating a functional algorithm from a neural network is a process called training. An input is given to the neural network with a label attached. The label represents the true answer of data. For example, you could input a picture of a face into the network with the label 'face'. There will be a class of possible answers in the output layer that represent the different possible answers. A probability will be assigned to each of the possible answers, and the error between what the network output and the 'true' label will be used to feed back into the network to change the values of the weights. This process is referred to as back propagation. In theory, if you repeat this process over and over again with a large selection of inputs, called epochs, the answer that the computer gives will become more correct. In essence, the network is building its ability to recognize an underlying pattern between all of the data that it is given.

However, this does not mean that the network will perform perfectly. There can be several problems encountered when creating a neural network. The first problem would be that we ideally would want a diverse set of testing data. In order for the network to be able to find the underlying pattern, many different forms of the same input are required. This is so the network is able to separate the noise from the real underlying pattern. By using too much of the same type of input, it will not be able to separate consistent noise

Another problem that can be encountered is in the accuracy of the system. We hope that by creating a rather simple class of output labels, we will be able to avoid this issue, but in larger neural networks accuracy becomes a huge issue. If the algorithm is only able to detect the correct output 95% of the time, that means that we would expect it to be wrong one out of every twenty passes. This would be unacceptable for our intended use of the algorithm. But by using the power of statistics, we may be able to mitigate this issue. If our product is able to take samples fast enough, we could take the average of multiple samples before sending the output to its final destination. This would dramatically improve the accuracy of our final answer, making it much more viable for use in our product.

3.9 3-D Printing

3-D Printing has been called the new “Industrial Revolution.” It has indeed impacted the industry in a big way. The biggest thing that 3-D Printing immediately impacts is the Labor cost that goes into manufacturing things. Also customizing a design is incredibly easy because of the software that is used in order to model the objects being printed. The software being used is CAD software. Due to the easy customization of any design, small unique production runs are possible at a much cheaper price than usual with standard manufacturing. This feature of course rolls over into medium production runs. Medium production runs become much more cost effective for similar reasons. Standard manufacturing is cost effective only in huge production runs. Depending on the product being manufactured this usually results in production runs in the thousands.

3.9.1 3-D Modeling Software Details

The reason we are choosing to use 3-D Printing is for its quality in manufacturing and the ability to adjust a design with ease. The fact that the design for the manufacturing process is done in software is the reason for the ability to easily adjust a design for a product. The software that we plan to use for the 3-D Modeling is TinkerCAD. TinkerCAD is free. We will be using TinkerCAD because of how beginner friendly it is. AutoCAD will be used as an alternative if the design made from TinkerCAD isn't the design desired. Since TinkerCAD is designed for beginners it's designs are limited. AutoCAD is a software made for use by professionals which makes it more difficult to use but it does have the limitations that TinkerCAD has and is a good alternative in this sense. AutoCAD is also incredibly expensive but that isn't much of a factor because most of the computers in the Engineering building already have AutoCAD on them for use.

3.9.2 3-D Printing Machine Prices and Availability



Figure 23: 3-D Printer

The prices for a 3-D printing machine can range from \$400 dollars to upwards of \$6000 dollars. This cost is entirely too much for our projected budget. Thankfully there is a 3-D Printer available for use on the UCF Campus. The use of the 3-D printer isn't free though. It costs \$7 dollars per cubic inch of the object being produced. This is well within our projected budget as we intend to use the 3-D Printer in order to manufacture a chassis for the Infrared LED Sensors. This

Chassis must be made in order to make something to place the Infrared LED Sensors in an equidistant fashion and in a structure that will hold them firmly.

4.0 Related Standards and Design Constraints

Standards are published documents that establish a certain set of guidelines or requirements for developers to follow. Standards are important because they help simplify many aspects of development that would otherwise become incredibly complex without them. Some standards are enforced by the government while other standards are just guidelines that can be helpful to follow. Standards also help with the compatibility, safety, quality assurance, and the stability of the product. In the following sections we will be covering many of the standards that impact our project.

4.1 Related Standards

Here in this section of standards we will be going over all of the standards that can possibly have an impact on our project. The standards we will be covering are Portable Rechargeable Battery Standards, Wireless Communication Standards, Bluetooth Standards, Health and Safety Standards, Software Engineering Standards, Software Testing Standards, Programming Language Standards, Standards for Exposing the Human Eye to Radiation, and the Standards for Zigbee. Most of these standards that we discuss are maintained by IEEE, ISO, and IEC.

4.1.1 Battery Standards

The document we will be using for reference for battery standards is called the “IEEE Standard for Rechargeable Batteries for Portable Computing.” This standard goes over controls for the design and manufacturing of Lithium-Ion (Li-ion) and Lithium-Ion Polymer (Li-ion polymer) rechargeable battery packs. These standards go over how to design, test and evaluate a cell or battery pack. How to mitigate failures in the battery systems is also talked about in detail in this standard. Here we will mainly be going over how to mitigate failures in the battery system.

In the event of a short circuit, the battery pack will limit the output current even if the battery pack is removed from the device. The battery pack must have 2 independent methods to limit current. One can be at the cell level and the other can be a connector structure to minimize the possibility of an external short circuit. [20] Overheating Temperature protection will be included to help prevent any damage that can occur to the battery under these conditions. Battery Packs should include at least one thermal protection circuit or a device independent of

internal cell mechanisms. [20] When temperature limits are exceeded the battery pack will shut down or take any other protective action. A max charging voltage and current must be set before the manufacturing process. The charging system should be designed to limit the charging voltage and current to these maximum values. Upper limits on current discharging must be set as well. The battery pack must have at least one overcurrent protection circuit that is designed to limit the current below the upper limit that has been set. The battery pack must be durable to many different kinds of mechanical stress including press, bend, twist, drop, vibration, and shock. The orientation of the battery packs in the device should be placed in a way that mitigates these stresses as much as possible.

4.1.2 Programming Language Standards

“C is a general-purpose programming language featuring economy of expression, modern control flow, and data structure capabilities, and a rich set of operators and data types.” [4] The standards of the C Programming Language are maintained by ISO/IEC JTC1. The C programming language standards build by ISO help specify various things about the C programming language. These specifications include “the representation of C programs, the syntax and constraints of the C language, the semantic rules for interpreting C programs, the representation of input data to be processed by C programs, the representation of output data produced by C programs, and the restrictions and limits imposed by a conforming implementation of C.”[5] These standards were put in place to help promote “portability, reliability, maintainability, and efficient execution of C language programs on a variety of computing systems.”[5] These standards will help in all of the areas mentioned above for the project. We hope to use the C programming language to develop some of the software needed to program the microcontroller if the granular efficiency of assembly language is not necessary.

In the standards for the Python programming language there are many guidelines for the style of the text to be written for Python code. The layout of the code is covered in the standards. The layout of the code in this case refers to indentation, maximum line length, blank lines, imports, source file encoding, and many other things. The code layout is spelled out in order to help the readability of the code and the maintainability of the code. How String Quotes properly work and how whitespace is dealt with in expressions and statements is also answered here. The utilization of comments is explained in the standards. Comments are very important to use in order to understand code written from another person or to be able to pick up where you left off with the code. Overall comments are useful to help quickly understand code. Naming conventions are spelled out in the document in order to have some consistency between everyone’s code.

Assembly Language has been the primary programming language for microcontrollers for their entire existence. Assembly Language offers benefits over other programming languages because it is incredibly efficient since it is a very low level language. We will more than likely be using Assembly to program our own Microcontrollers and so understanding the standards put in place for Assembly Language will be beneficial in order to program the microcontroller. IEEE has created a standard for Assembly Languages in 1985 titled “IEEE Standards for Microprocessor Assembly Language”. Although the document is not current, the standards are still helpful for many of the Assembly Languages in use today. In the document many of the most commonly used set of instructions are defined. Conventions and the coding format for assembly language programming are established. Many of the mnemonics commonly used in assembly language are also defined in this document such as ADD, SUB, MUL, and so on. There are many aspects of assembly language that the document does not establish a standard for. The reason for this is because of the fact that many manufacturers make their own slight variations to the instruction set of the assembly language that is compatible with their microcontroller.

4.1.3 Software and Engineering Standards

The “ISO/IEC 90003:2014, Software Engineering Guidelines for the Application of ISO 9001:2008 to Computer Software” covers standards that apply to software engineering. In these standards quality management systems requirements are covered. These requirements refer to the responsibility of a developer to produce a product that meets the customer’s requirements. Processes must also be kept in place for constant improvement of the product that are in line with the customer’s requirements. These standards are applicable only to software of this variety. A software that is “part of a commercial contract with another organization, a product available for a market sector, used to support the processes of an organization, embedded in a hardware product, or related to software services.” [8] Our software will be embedded in a hardware product therefore these standards are applicable to our software.

4.1.4 Software Testing Standards

The “ISO/IEC/IEEE 29119-4, Software and Systems Engineering - Software Testing” document goes over some techniques for testing software. We will be using the standards set in this document to help test our own software. The testing techniques discussed in these standards are classified as structure-based, specification-based, or experience-based.

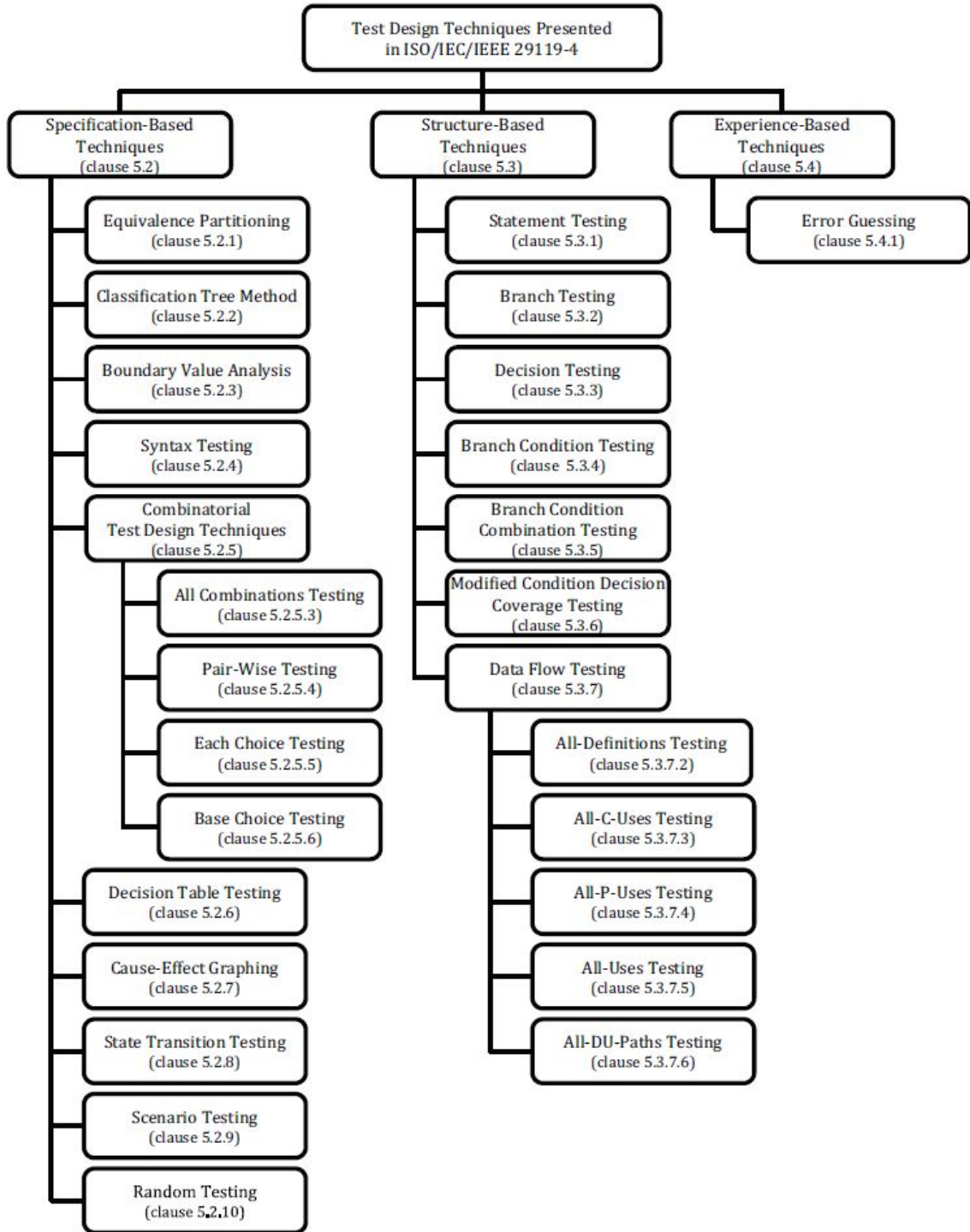


Figure 24 - The Set of Test Design Techniques Presented in ISO/IEC/IEEE 29119-4

The classification system for test design techniques is shown in the figure above. Once we have decided which testing process to go with, we should then use

many of the different testing techniques listed below the classifications. Under the Specification-Based Techniques we have Equivalence Partitioning, Classification Tree Method, Boundary Value Analysis, Syntax Testing, Combinatorial Test Design Techniques, Design Table Testing, Cause-Effect Graphing, State Transition Testing, Scenario Testing, and Random Testing. Under the Structure-Based Techniques we have Statement Testing, Branch Testing, Decision Testing, Branch Condition Testing, Branch Condition Combination Testing, Modified Condition Decision Coverage Testing, and Data Flow Testing. Under the Experience-Based Techniques we only have one testing method which is Error Guessing. Each of these classifications are good at finding particular defects.

We use terms like “black box” and “white box” for specification-based techniques and structure-based techniques respectively. For specification-based techniques we don’t know all of the inner workings of the process and therefore classify as such. For structure-based techniques we know most or all of the workings of the process and therefore classify it accordingly as well. For experience-based techniques we are just guessing and checking the result to see if it is the desired result.

Specification-Based Testing Techniques treats software as a blackbox. It only views the inputs and outputs of the software. This is why Specification-Based Testing is also called “Black Box Testing.” One thing to note is that specification-based testing can be used on anything that is a specification. This includes testing hardware which can be useful in our case. Requirements, specifications, models, or user needs are used as the main source of information to design test cases for specification-based testing techniques. [17]

Structure-Based Testing Techniques also called “White Box Testing Techniques” use the software itself in order to create test cases. Since the inner workings of the software are all known, we can create test cases that work for the software. This is the reason structure-based testing techniques is also referred to as “White Box Testing Techniques.” Once again these Testing Techniques can be used for any kind of system that we have knowledge of the inner processes. [17]

For Experience-Based Testing Techniques the knowledge and experience of the tester is used as the primary source of information for test case design. Experience-Based Testing Techniques are used when there is no specification or the specification does not provide enough information to derive a test case from it. Experience-Based Testing Techniques can be used in conjunction with Structure-Based Testing Techniques or Specification-Based Testing Techniques.

4.1.5 Health and Safety Standards

Health and safety standards could easily be argued for the most important standards. Safety standards are important because they could determine a human beings life. Meaning if they are not done right then they could kill a person. For example, if we have our headset that should not be allowed to get wet while it is on someone's head. Because if it is it would electrocute them. This example is not necessarily true. It is a hypothetical example. Going back to the example, it would be dreadful if the device did not have a warning on the device about making sure to not go out in the rain while wearing the headset because the wearer would die from too much electricity coursing through their body.

To determine how to put these warnings on the device or what not it would be helpful to follow the picture below.

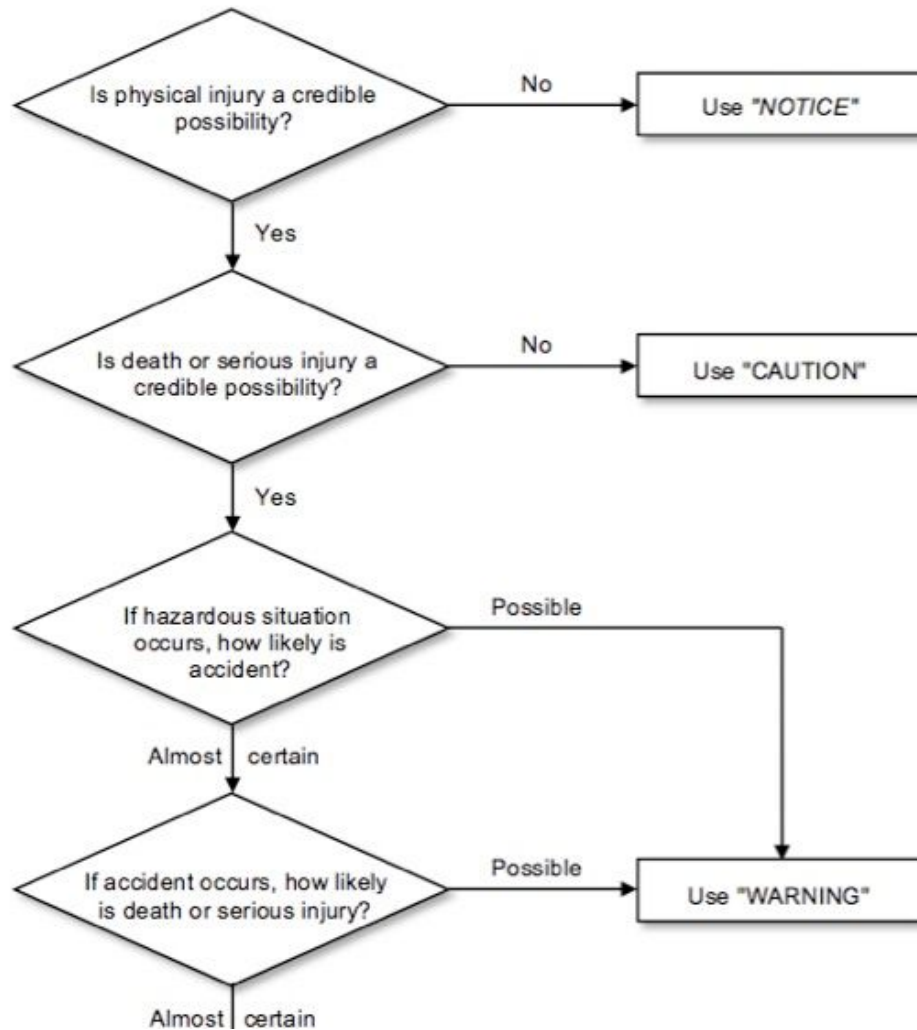


Figure 25: Health and Safety Warning Sticker FlowChart

This standard of using “Notice”, “Caution”, and “Warning” is a good way to get the customer aware of any danger that could happen to them if they use the device in a wrong way. If the block diagram is followed through then warning would be the most dangerous warning. If “Warning” is used that would mean death could occur. “Warning” could also be used for making the user aware of using the device in a way that could cause serious injury and not just death. “Caution” would be used if physical injury is a mere possibility. It does not mean that it is a warning of death, for that is what “Warning” is for. The last one is “Notice”, which is the lowest form of warning the user. It does not mean there could be death, serious injury, or even physical injury.

These warnings, caution, and notices will be good for the project because it could prevent injuries and possibly even death. When making the device, going through this block diagram would be a helpful way of determining how to make the user aware of possible injuries or possible deaths. This will ensure the device to be more safe.

4.1.5.1 Health and Safety Risks of Eye Tracking Technology

A great amount of precautions must be taken into account when dealing with technology that attempts to track the eye through light. Certain wavelengths of light can be damaging to the Human Eye such as Ultraviolet Light, Blue Light, Near-Infrared Light and even Infrared Light with a sufficient amount of concentration. Although the eye is much more sensitive to Ultraviolet Light and Blue Light and is more likely to be damaged by these wavelengths of light, it can still be caused harm by Infrared Light. To date there hasn't been instances of Infrared LEDs causing damage to the Human Eye in IR sensors. However LEDs have been getting more and more powerful in intensity and brightness over the last few years. Since we will be using Infrared Sensors to help track eye movement, we will also be using an LED that emits Infrared Light that we can sense. We must ensure that the Infrared Light that is emitted by the LED is not damaging to the human eye. Another thing to consider is the potential harm this radiation does to the skin.

The standards on safety when it comes to potential damage from radiation exposure to LEDs on the Human Eye were established in 2006 by the IEC. The document that holds these standards is the IEC 62471: Photobiological Safety of Lamps and Lamp Systems. This document not only goes over the potential dangers radiation exposure to the human eye but also to skin. As we know skin is adversely affected by prolonged exposure to Ultraviolet Light, however for the wavelengths of light that we will be dealing with in our project the concentration and exposure levels of Near-Infrared Light and Infrared Light that the user will be subjected to in our project is well below the safety standards put in place by the IEC in clause 4.3.8. Near-Infrared Light has a wavelength of about 700 nm to 1400 nm. The IEC standards committee developed a risk classification grouping

for continuous-wave, pulsed lamps, and LEDs. There are 4 risk classifications. One being the Exempt Group. Any device that does not pose infrared radiation hazard for the within 1000 seconds, retinal thermal hazard within 10 seconds, and near-infrared retinal hazard without a strong visual stimulus within 1000 seconds belongs to the Exempt Group. [12] Any device that does not pose an Infrared Radiation hazard for the eye within 100 seconds, retinal thermal hazard within 10 seconds, and near-infrared retinal hazard without a strong visual stimulus within 100 seconds belongs to Risk Group 1 (Low Risk). [12] Any device that does not pose an Infrared radiation hazard for the eye within 10 seconds, retinal thermal hazard within 0.25 seconds (aversion response), and near-infrared retinal hazard without a strong visual stimulus within 10 seconds belongs to Risk Group 2 (Moderate Risk). [12] Any device that exceeds the limits for Risk Group 2 is in Risk Group 3. [12] It should be noted that most Infrared LEDs fall well within the Exempt Group or the Risk Group 1 as set by the IEC.

The maximum allowed ocular exposure to infrared radiation, E_{IR} , over the wavelength range 780nm to 3000nm is defined as follows: For exposure times $t \leq 1000$ seconds, the limit depends on the exposure time (t) itself shown by [Equation 1](#).

$$E_{IR} = \sum_{\lambda = 780}^{\lambda = 3000} E_{\lambda} \cdot \Delta\lambda \leq 18000 \cdot t^{-0.75} [W \cdot m^{-2}] \quad (\text{EQ. 1})$$

For exposure times $t > 1000$ seconds, the limit becomes a fixed value shown by [Equation 2](#):

$$E_{IR} = \sum_{\lambda = 780}^{\lambda = 3000} E_{\lambda} \cdot \Delta\lambda \leq 100 [W \cdot m^{-2}] \quad (\text{EQ. 2})$$

These equations are used to find the limits of hazardous radiation that one can expose to the human eye for a given amount of time depending on the energy intensity of the radiation. These equations go over all of the light wavelengths from 780 nm to 3000 nm. Equation 1 is used for exposure times of less than or equal to 1000 seconds and Equation 2 is used for exposure times that are greater than 1000 seconds. “ E_{λ} is the spectral irradiance in $W - m^{-2}$, $\Delta\lambda$ is the source spectral bandwidth in nm and t is the exposure duration in seconds.” [12] To help with Eye Safety, Intersil has developed a calculator that determines through the equations above if the LED will be safe for the user. This will be used in our project to ensure the safety of the user. The total irradiance can be found on most datasheets for whatever LED you are using.

The wavelength of the Infrared LEDs that we are using have a wavelength of about 940 nm. The Spectral Bandwidth of the Infrared LEDs that we plan to use is about 45 nm. The time that the eye will be exposed to the Infrared LEDs will be greater than 1000 seconds. Therefore some values will be fixed from Equation 2 given in the figure above. The user will be using the device for possibly longer than 1000 seconds and so this is the reason for setting the time in the equation to be greater than 1000 seconds. Unfortunately the spectral irradiance is not given in the datasheet provided by Everlight. [30] Without knowing the spectral irradiance it is not possible to calculate the maximum allowed ocular exposure to infrared radiation. Therefore we will be just using a comparison of another Infrared LED that has some similar specifications to calculate the maximum allowed ocular exposure.

Human Eye

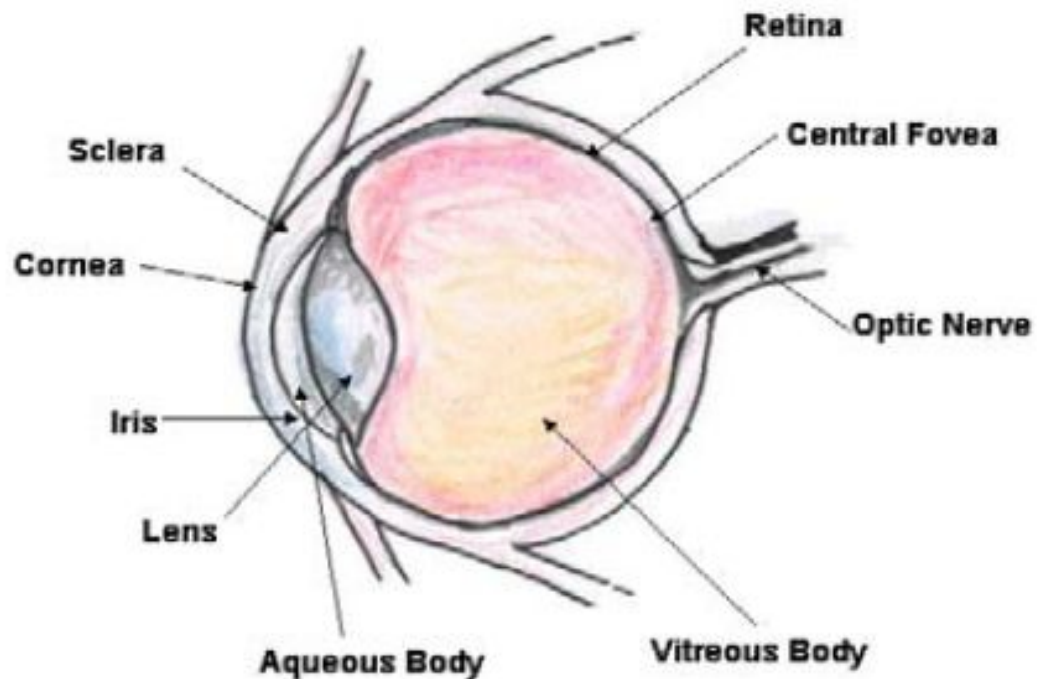


Figure 26: Schematic of the Human Eye

This figure of the human eye shows a schematic of the human eye. “The eye contains two lens elements - the cornea and the lens. The cornea provides a fixed focusing capability, allowing for adjustment for near and far objects. Combined these elements focus illumination on the retina where the various photo-stimulated cells (Rods and Cones) convert the light energy into electro-biological stimuli that can be processed in the visual cortex of the human

brain.” [12] Every single one of these different parts of the eye can be damaged through sufficient exposure to radiation.

4.1.6 Wireless Communication Standards

There are many different types of wireless communication available for use in devices today. The most common form of wireless communication comes through Wireless Fidelity (Wifi). Wifi was first invented in 1997 at the University of Hawaii. With the invention of Wifi a committee called 802.11 was created to help create standards for this form of wireless communication. Eventually this committee went on to become a part of IEEE while still being under the same name. 802.11 is still in charge today of creating, maintaining, and amending the standards for Wifi. The most up to date standards for Wi-Fi

There are also different networks available for Wifi. The most common networks you’ll see today are either 802.11ac or 802.11n. 802.11ax is the newest Wifi network and it’s also the most flexible and fastest. However 802.11ax has yet to be fully implemented and deployed, but it’s worth talking about since it is not very far off from being implemented as the new standard. Most projections are looking at late 2019 for the new Wifi standard to be implemented and deployed in devices. Each of these networks also have different data transfer rates or speeds. . Our project will be fully built towards the end of the Spring 2020 Semester. Its possible that 802.11ax is readily available for use by the time we finish our project.

4.1.7 Bluetooth Standards

The standards on Bluetooth Communication were originally maintained by IEEE in the document named 802.15.1. The standards for Bluetooth are now handled by an organization called the Bluetooth SIG (Special Interest Group). The Bluetooth SIG maintain the quality of Bluetooth communication along with many other standards. In this section we will be delving into the original standards for Bluetooth because the standards currently being maintained by the Bluetooth SIG requires a membership to access them. In order to become a member you must submit documents proving that you have formed a company and we are unable to verify ourselves as a company. Certain membership levels also require a substantial amount of annual payment which far exceeds our projected budget for the project. For this reason we will be going over the standards established by IEEE.

“The original goal of the IEEE 802.15.1 Task Group was to achieve a level of interoperability that would allow the transfer of data between a WPAN device and an IEEE 802.11 device.”[22] This was shown to be unfeasible but this document shows some mechanisms that allow better coexistence with IEEE 802.11b class

of devices. Bluetooth communication operates in the unlicensed ISM band of 2.4 GHz. “During typical operation, a physical radio channel is shared by a group of devices that are synchronized to a common clock and frequency hopping pattern.” [22] There is one device that is considered the *master*. The device that is considered the *master* provides a synchronization reference. All of the other devices in the group are considered slaves. A group of devices that are synchronized in this fashion are called a piconet. In the Bluetooth technology, this is the fundamental form of communication.

4.1.8 Zigbee Standards

The standard we will be going over for Zigbee is the IEEE Standard for Low-Rate Wireless Networks also known as a 802.15.4 Network standards. “This standard defines the physical layer (PHY) and medium access control (MAC) sublayer specifications for low-data-rate wireless connectivity with fixed, portable, and moving devices with no battery or very limited battery consumptions requirements. The standard provides ultra low complexity, ultra low cost, ultra low power consumption and low data rate wireless connectivity among inexpensive devices.” [24] Low-rate wireless personal area network (LR-WPAN) are used by applications with limited power and low throughput requirements.

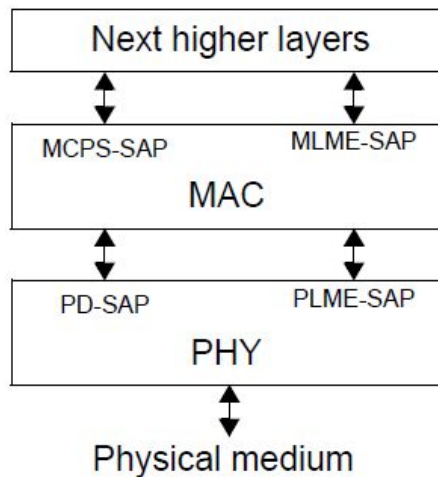


Figure 27: Architecture of Low-Rate Wireless Personal Area Networks

The Figure 27 shows the architecture of low-rate wireless personal area networks which includes Zigbee. The next higher layers are made up by a network layer which allows for configuration, manipulation, and message routing, and it consists of an application layer.

4.2 Design Constraints

A Design Constraint is a rule or limitation placed on the development of a product which may or may not be self imposed. These constraints can be helpful or detrimental to the development process. These constraints narrow the possibilities that can be taken with the development. This can hinder creativity, but it can also help simplify the selection process. In this section we will discuss the various design constraints that will affect the development of our project.

4.2.1 Economic and Time Constraints

Our budget for the project is a big constraint. Our projected budget is around 500 dollars. This project does not have a sponsor so it will be self funded by the group. This leaves us with very little wiggle room in terms of mistakes we can make when building the project. Most of our budget will be going towards getting a decent display screen. We will also be spending money on a microcontroller and the main CPU. Our main CPU will most likely be the Raspberry Pi 4. These budget constraints will not allow us to obtain the most advanced piece of technology for each aspect of the project unfortunately. Our budget also affects the access we have to certain standards. Many of these standards cost upwards of 150 dollars in order to gain access to them. Due to this we will only be using standards that are free to access through the UCF Library or just through regular access.

The amount of time we have to spend working on the project is an additional constraint that must be taken into consideration. 3 out of the 4 members of the group have jobs and all of the members of the group have other classes to tend to as well. Our schedules between class and work make it very difficult for all of us to meet up at the same time. We really had to rely on meetings between just 2 or 3 of us in order to get things done. We were at least able to have weekly meetings every Tuesday, which helped smooth out any miscommunications or misunderstandings we might have had with one another. We must ensure that we keep a consistent workload throughout the semester in order to get all work done without having to crunch when deadline for an assignment approaches.

4.2.2 Health and Safety Constraints

A great amount of precautions must be taken into account when dealing with technology that attempts to track the eye through light. We will adhere to the safety standards set by the IEC in the document 62471: Photobiological Safety of Lamps and Lamp Systems in order to ensure that our product is safe for use. Our device can only be of a certain intensity and brightness according to the IEC if we intend the user to use it for prolonged periods of time. A warning will more than likely have to be put in place to make the user aware that prolonged use can

result in harm to the human eye. We'll have to calculate the amount of time that is safe for the user to use our device.

4.2.3 Technological Constraints

Our budget limits us from using the most advanced technology for our project. This hinderance will be noticeable when it comes to the sensitivity of the Eye Tracking Device. Our device will more than likely not be able to pinpoint exactly where the user is looking. It will only be able to tell the general direction in which the person is looking. With this lack of accuracy our choices of application for the Eye Tracking Device will also be limited. Many of the ideas for application we have come up with require pinpoint accuracy from the device in order to work properly. We will have to go with an application that only requires the knowledge of the general direction the user is looking in order to accomplish its goal. Also since we are using an Analog to Digital Converter, some data will inevitably be lost in that process resulting in worse accuracy. The wireless communication standards will impact our project greatly. One very important aspect to our project is streaming video from a camera to a display screen that the user will see. The stability, smoothness, and quality of the video stream will be dependent on the Wifi standards in place today. Obviously the higher the rate of data transfer, then the better our stream will be. We have to ensure that the speeds of the Wifi standards that are common today will be enough to stream video with a decent amount of quality for the user. We are looking to make our device fully wireless so the range of the Wifi signal will also affect our project as well.

4.2.4 Environmental, Social, and Political Constraints

There aren't any Environmental Constraints to worry about in our project. However there are some social and political implications to Eye Tracking Technology. Using cameras there are algorithms that can track where a person is looking at any given time. This data can be used in Social Studies and this can be interpreted as a violation of privacy in the political realm. Thankfully our project doesn't have to deal with any of these constraints because of our implementation of the eye tracking. Our device uses IR sensors to detect eye movement which can only be used if the user voluntarily uses the device.

4.2.5 Manufacturability Constraints and Sustainability Constraints

Due to Manufacturability Constraints our project should be using components that are simple to manufacture. Also the cheaper the component, then the better as long as it stays above the standards put in place. Sustainability will also play a big role in the components we choose. The display screen we choose will have to last a sufficient amount of time to ensure a quality product. The MSP430 and

the Raspberry Pi 4 have a good amount of sustainability that will satisfy those constraints. Also the MSP430 and Raspberry Pi 4 are already being manufactured on a mass scale so they will be perfect for meeting any possible manufacturing constraints. The battery that we choose to power our wireless device will be the most difficult decision to make for our project. The battery must have a solid battery life in order to have a good amount of sustainability. Our goal is to have at least an hour of battery life with the device. The battery must also be under a large manufacturing already or if we decide to design our own battery then all of the specs of the battery must be conducive to it being easily manufacturable.

5.0 Project Hardware Design Details

To begin with, the project is broken into two halves for the hardware, and the software. The hardware is further broken down into five components, sensor array, goggles, converter board, main MCU, and output devices. In this section we will be delving into each of these five components in detail. The layout of this section is broken down to the component concept used in this project. These sections then further break down each component section down to the concept, iterations of the section component.

Furthermore the choosing of these hardwares is also discussed with the reasoning behind why we chose certain chips, photonic detectors, and or power supplies. The reasoning has to do with our constraints and standards as discussed earlier in the paper as well with our engineering requirements.

5.1 Design Architectures Details

The concept of our project was to create a device that could link human vision with that of the device or the machine. The task at hand is to build an interface between user and device. The idea we came up with is shown in the below image 28 where the main device is the goggles and main MCU as acting for the concept interface and the possible methods this system could control. The goggles house an array of sensors and converter board that generates state values of the eye's state. Then this system transfer the data to the main MCU that then calculates the desired output for the system. The system is designed to run off the movement/ direction of the user's line of sight to operate devices like displays or camera based devices like turrets on robot tanks

It is crucial to see that in designing this product the chips and hardware that we use have to be small and light enough to fit on a person's head or pocket as seen in the picture below. If the parts are enormous then it won't fit and wearing the head gear would be uncomfortable and probably cause damage to the user's neck if it is too heavy. Therefore the architecture design must be efficient to where it can fit on someone's head and be comfortable.

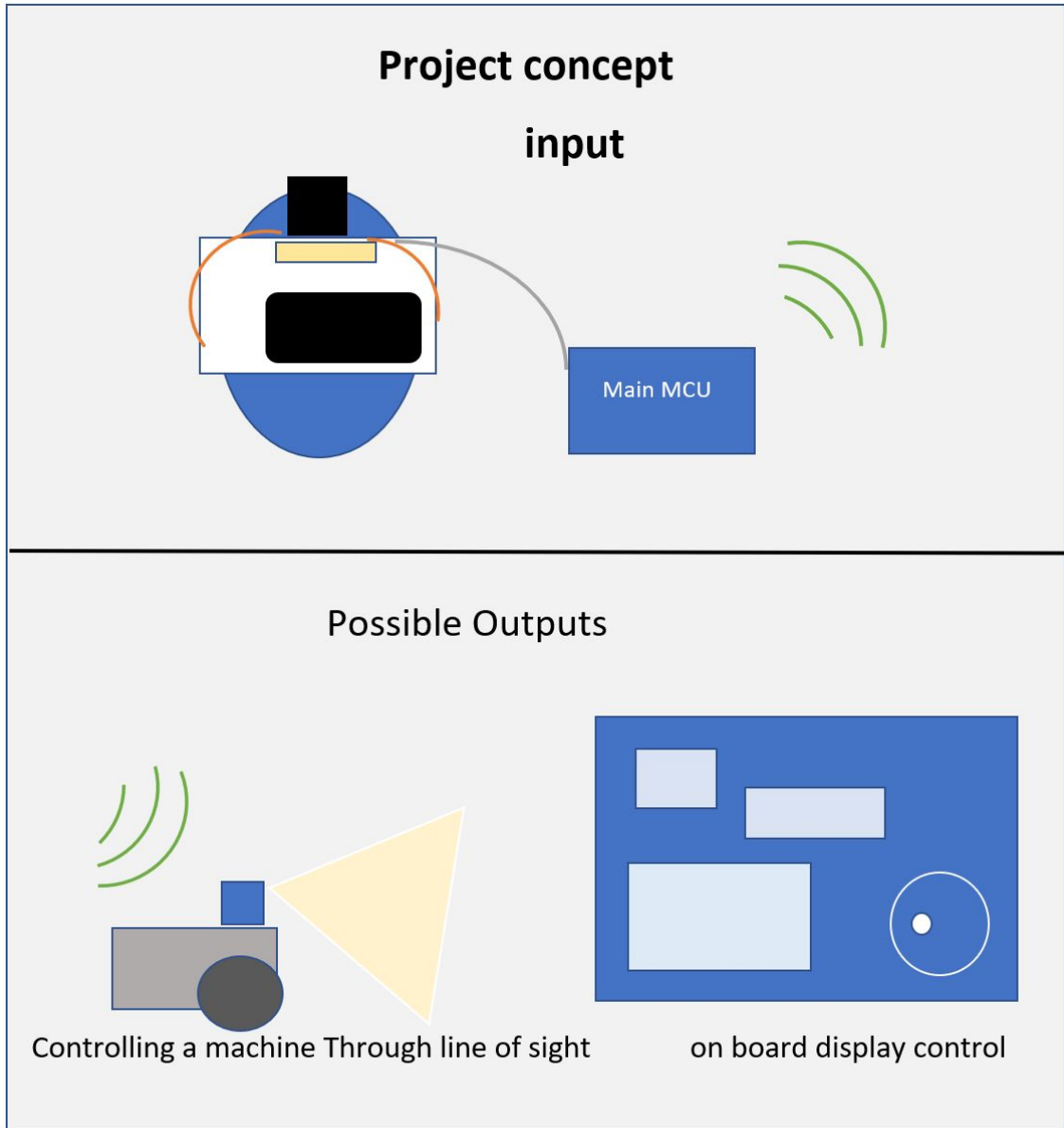


Figure 28 - Project concept

5.2 Infrared LED Emitters and Sensors

The Infrared Light Emitting Diodes we are using are created by Everlight. The diodes we are using are 5 millimeters in length. They are highly reliable. The picture below displays the LEDs that we will be using.

IR333/H0-A(A&A)



Figure 29: 5mm Infrared LED, T-1 3/4

The maximum wavelength emitted by the LEDs are at about 940 nm. The radiant intensity of these Infrared LEDs is dependent on the amount of amps in the current running through it. [30]

Parameter	Symbol	Condition	Min.	Typ.	Max.	Units
Radiant Intensity	E_e	$I_F = 20$ mA	5.6	7.8	--	mW/sr
Radiant Intensity	E_e	$I_F = 100$ mA	--	35	--	mW/sr
Radiant Intensity	E_e	$I_F =$ mA	--	350	--	mW/sr

There is one Additional Condition. The pulse width is 100 μ s . The duty cycle is less than or equal to 1%.

Table X: Radiant Intensity Values Dependent on Current

Unfortunately there isn't a value for the spectral irradiance from the datasheet provided by Everlight. [30] Because of this we are unable to apply the equations given in Chapter 4 of Standards in Section 1.5.1. We are unable to assess the safety of the infrared light exposure through these equations. After looking through many datasheets for various Infrared LEDs it would seem that none of the other LEDs have Spectral Irradiance given from the manufacturer. We will have to continue looking for ways to calculate the Spectral Irradiance for the Infrared LED in order to gauge its safety.

5.2.1 Sensor Array Details

To start off the sensor is an array of IR receivers built in to a crescent design around both eyes of the user or operator. This design is due to the fact that human eyes are roughly link and most of the key points of change in the eye happen at the outer edge. Plus as a side note with this design it allows for better vision since there will be no components in between the user's eye. This design is slip into three components one and two are the left and right IR sensor arrays witch connects to the holder/ goggles the operator uses.

For the left and right and left IR arrays, they are sub-components that are basically the same, just mirrored to each other. The arrays are simple holder brackets and multiple individual IR receivers as shown in image 30. The holder bracket set to the sides of the face with a crescent half ring that hovers about a centimeter in front of the eye socket. The crescent half ring would house multiple shielded holders for each IR receiver as shown in image 31. These shielded holders are in basic terms as the name implies holders for each IR receiver but cover the sensor from any direct IR signals and only allowing the reflected IR signals from the users eye. The IR receivers are simply the transistor and wires that lead back to the converter board. Each side's IR receivers share the same power line while each sensor have their own data line that run back to the converter board. The holder brackets as a side note also carry the wires back out of the way of the user to the converter board.

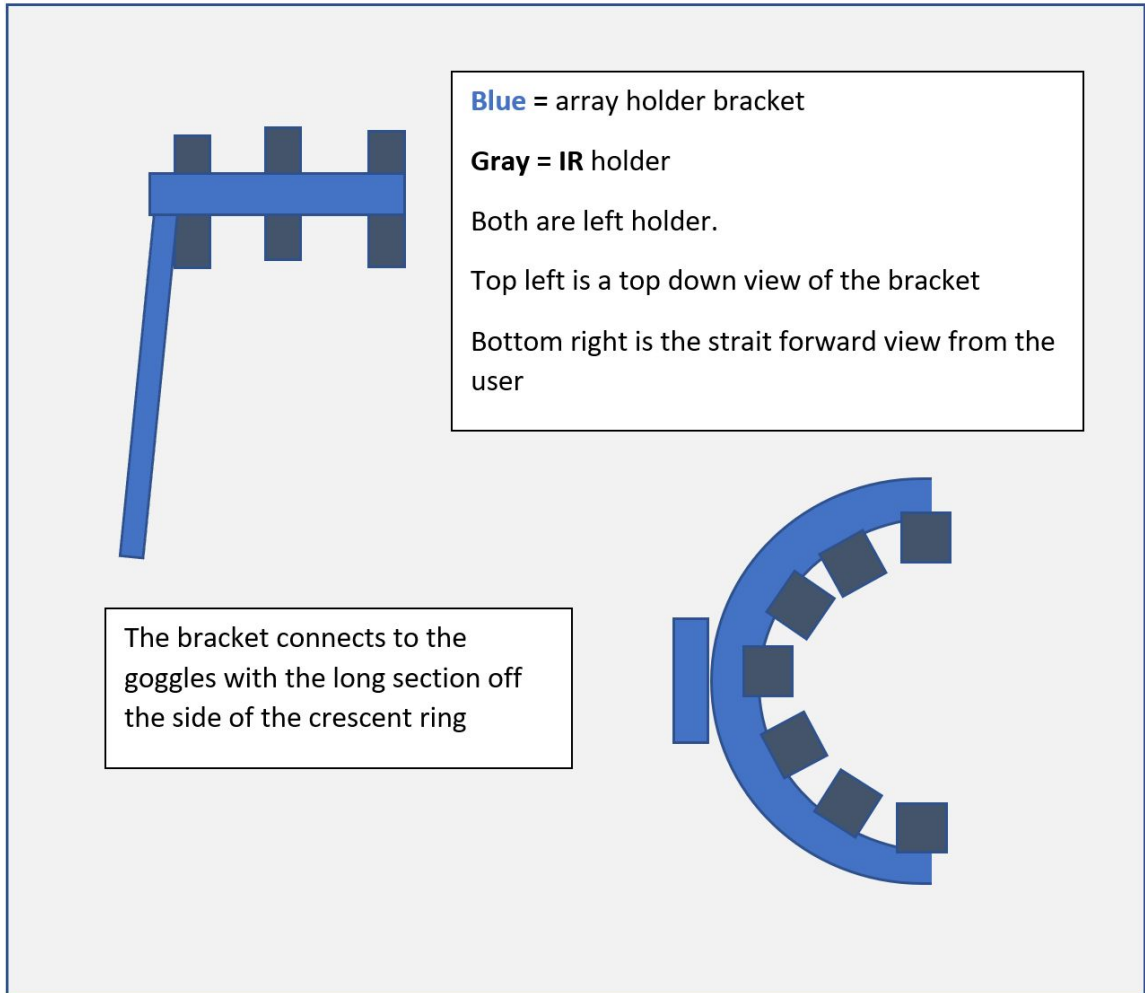


Figure 30: IR array holder bracket

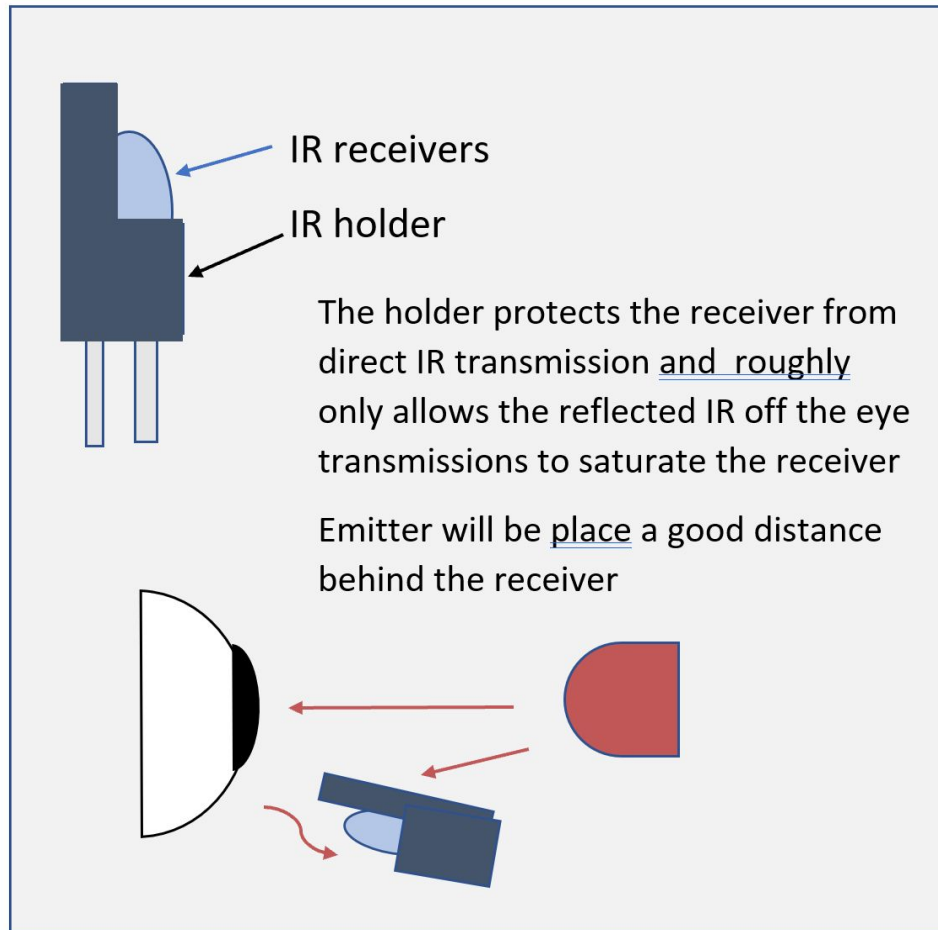


Figure 31: IR Emitter and Sensor Diagram

5.2.2 Sensor Array Previous Iterations

For this project this component has gone through a few iterations before landing on the design above. The sensors first started out with LDR, or Light Dependent Resistor, as a way to detect the changes in the eye's state. As seen in image 32 These LDRs would function by taking light from the display screen by a reference LDR facing screen and the LDR facing the users eye to pick up the reflection of the display screen off the surface of the user's eye. After that the system in comparison to the current sensor array was basically the same. Both use the same concept of holders but that the LDRs required a second, or reference sensor in paired with each data sensor. The reasons that we change from LDR to IR were as listed.

1. LDRs take time to change values while IR are much quicker. Like LDRs take about 0.2 seconds to reach steady state value.

2. LDRs are depended on the display screen light levels which could be harmful to the operator's eye with the display set at a very high level so that the sensor could work, while IR run on their own emitters which are very unlikely to harm the users eye since human eyes are not affected by IR wavelength of light.
3. LDRs are not as stable due to slow change rate and an ever changing light source of the display screen that is playing whatever at the time, while IR can run a constant stream of IR waves from the emitters due the fact that the whole system is roughly a closed environment, depending on the fit of the goggles we use.

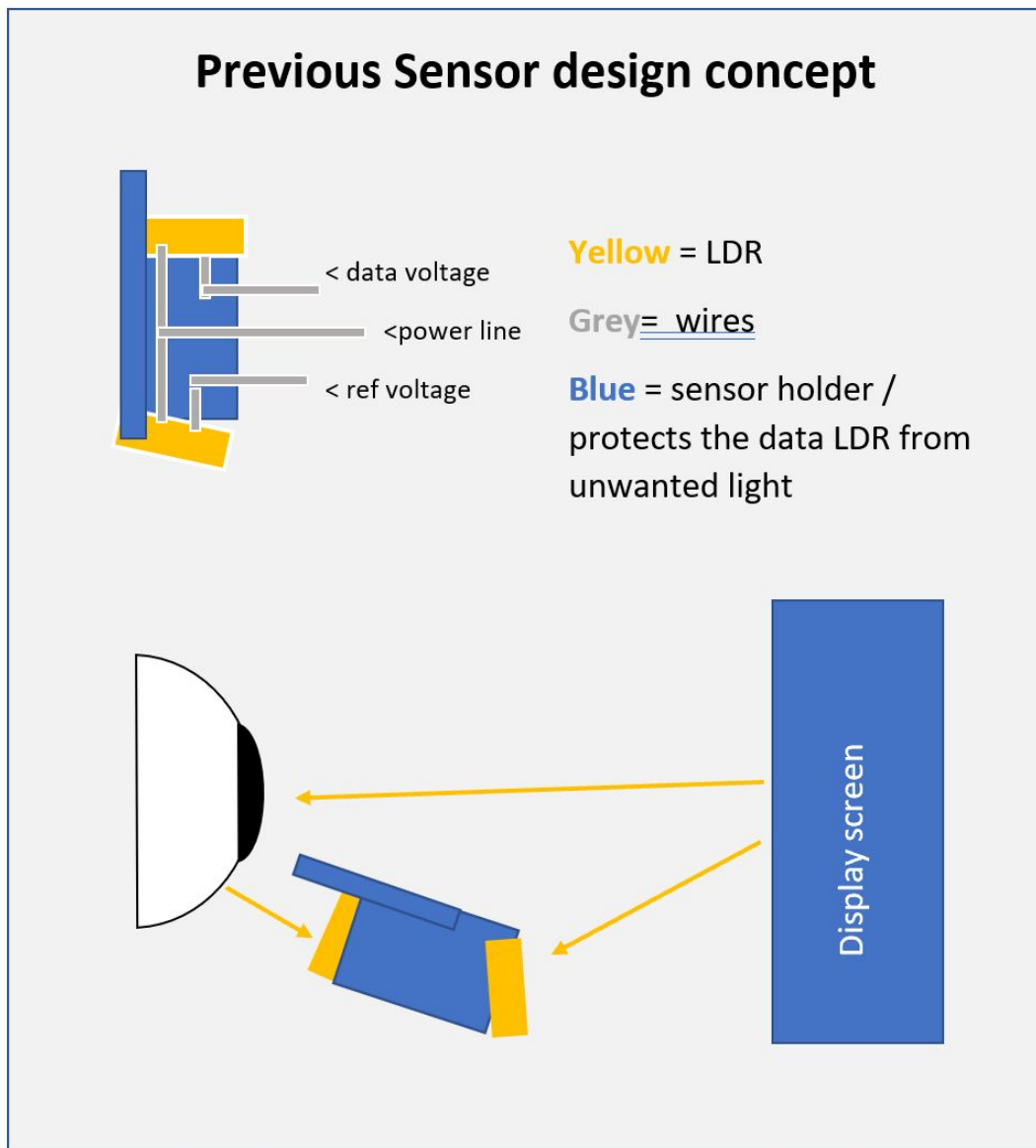


Figure 32: Previous sensor design

5.3 Goggles Details

For the goggles, they are the main apparatus that holds the display, the IR emitter array, and the converter board. This component is as shown in figure 5.3.1 is the main component that holds everything that the user interfaces with. For now this device is a rough concept product as the design shows. The goggles use in the main structure which will be an over the counter model for comfort and easy for construction.

The basis of the build are that the left and right array connected to the sides of the harness and the converter passed for now on the outside of the unit so it does not block any of the display or the IR sensor components. If possible the converter should be able to fit inside and if this is so it will be tested and then used in this configuration. The converter however will pass a data line or a ribbon cable down from the goggles to the main MCU unit that will then process the data from the arrays. The main MCU unit will include these abilities besides doing whatever output based task selected by the wearer, the system will feed back to the goggles display for the wearer to use. The display for the system in question that will be used will be that of a cell phone or smartphone.

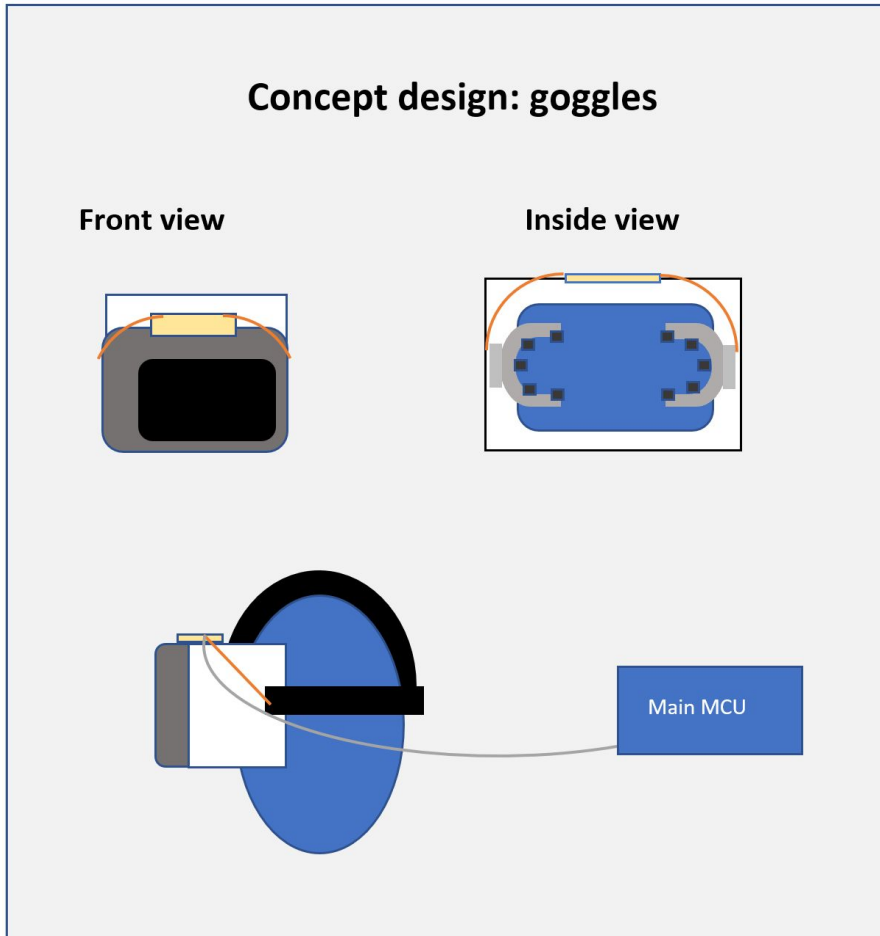


Figure 33: Goggle Design concept

5.4 Converter Board Details

Moving on to the converter board, this is the system's analog to digital converter system that takes the state value of each IR receiver and converts them to a simple 8-bit state value for the MCU to read. The converter system is divided into three parts a voltage divider to generate the state value, a selector for choosing which IR sensor is being polled, and the analog to digital converter that converts the state value to usable data for the main device. This can be seen in the figure 5.4.1 where the board is divided into two sections due to limitation of preventing lines from crossing to start off the power from the converter runs out through the IR then runs back to the first component, the voltage divider.

After the divider generates a voltage difference finally it then runs through the converter which compares the voltages from the divide to a set reference voltage. For the converter we have chosen is this product which is the MSP430 due to that it is very cheap, that it has a built in converter, and that it gives us more control when the MCU polles in the converter system. Basically the main

MCU can tell the MSP430 when to send the data and when to process the next state value.

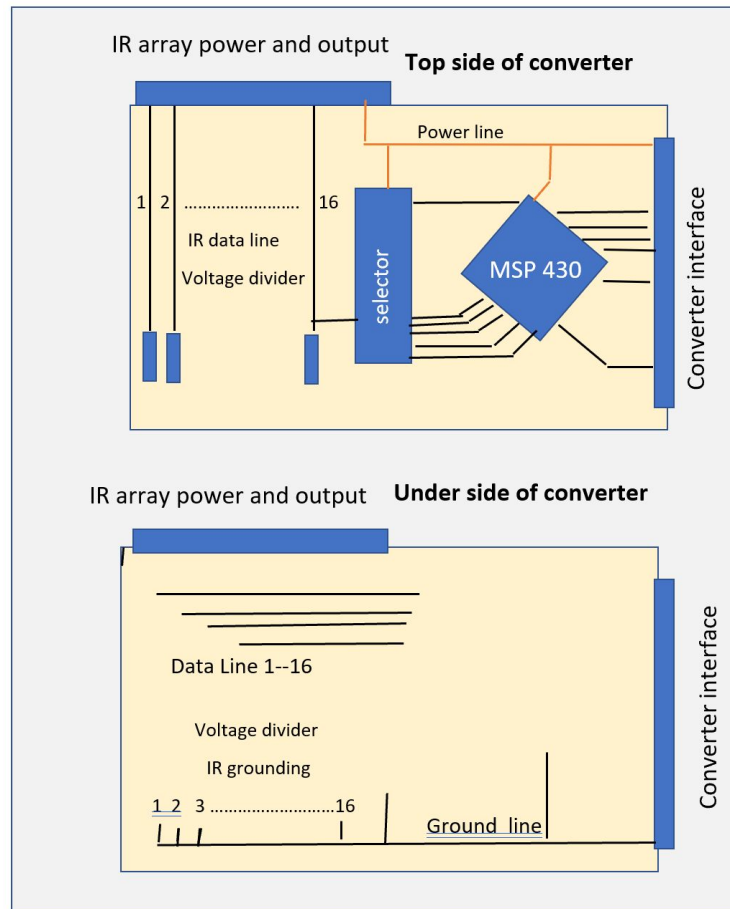


Figure 34: Diagram of MSP430

5.4.1 Converter Board Previous Iterations

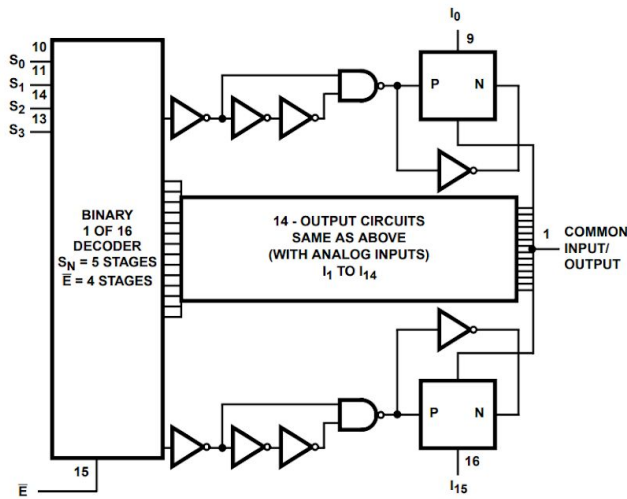
For this project this component has gone through a few iterations before landing on the design above. Since this is the main hardware of the build this was under much debate on most aspects of the system layout.

1. First was the idea of how we go about converting the voltage levels to data for the system to read. The base concept that was first discussed to use a flash converter like chip for each data line and build some sort of selector to choose which one of the many data lines to be passed through the system to the main MCU unit. The flash converter was first to drop the idea of using the MSP430's voltage comparator system as the analog to digital converter due to the fact that the MSP430 is very cheap, like 30

cents or so and is a system that is compatible with many MCU boards like Pie and so on.

2. Second of the changes was the idea of how the system ideas was of how the system was selecting the data line in question after the idea of the MSP430 was set as the converter, the system was at first set to have as many MSP430 as there was data lines but that was shot down a week later to just one and it communicating to a system or chip that would pass the voltages to the MSP430 individually. After further discussion it was decided upon later that we could use the device that is from Texas instruments which had the very chip needed known as CD74HC4067 as seen in figure 5.4.1.1. This chip allows the MSP430 to cleanly choose one data voltage to be passed on command with the added channel block to prevent data voltages from intersecting each other.
3. Third the designs that were not used was the placement order of the selector and voltage divider in terms to the MSP430/ analog to digital converter. The idea was either to have the selector before the voltage divider or after. If the selector is before than it allows the divider to be a simple one line and negate any resistance of the selector but it does cause slight delay to the circuit due the fact the circuit is breaking and reconnecting connections of the circuit constantly. The other was to place the selector after the voltage divider to allow the data voltage lines to be stable when the system polls for the information/ voltage level from the sensors but this comes at the cost of space on the board since that would mean that there will be a simple voltage divider for all data lines of each sensor.

Functional Diagram



TRUTH TABLE

S0	S1	S2	S3	\bar{E}	SELECTED CHANNEL
X	X	X	X	1	None
0	0	0	0	0	0
1	0	0	0	0	1
0	1	0	0	0	2
1	1	0	0	0	3
0	0	1	0	0	4
1	0	1	0	0	5
0	1	1	0	0	6
1	1	1	0	0	7
0	0	0	1	0	8
1	0	0	1	0	9
0	1	0	1	0	10
1	1	0	1	0	11
0	0	1	1	0	12
1	0	1	1	0	13
0	1	1	1	0	14
1	1	1	1	0	15

H= High Level
 L= Low Level
 X= Don't Care

Figure 35: Selector Chip Circuit Diagram and Truth Table

5.5 Main Processor Details

For our project the system required a main processor unit to take said data from the sensor array and process it with visual information for the operator to receive from the display screen in the goggles unit. The requirements for the system were that this unit needed to be multicore for parallel processing for things like the video display feed, the sensor array calculation for the system, and any other

output we think to tie in to this system to demonstrate the system capabilities. For now we currently decided that the main processor unit to be for this lab result will be that of the Raspberry Pi 3B or that of the Raspberry Pi 4. Reason for this decision was that these devices do meet the standard we need but mainly due to lack of funds and currently in possession of said unit.

5.6 Power Supply Details

There are many different ways to do the power supply for this eye tracking device. First thing that is needed is to figure out which design is best is to come up with a budget for it. Once the budget is made the next thing is to find out is how much voltage is needed and how much current will be needed. There may be more than one power supply depending on what the application is. If the application needs a power supply then a power supply will need to be made for that. If not then there will need to be a power supply for only the MCU. The power supply may be simply just batteries.

5.7 Display Screen Details

For this section of the project we went over many kinds of ways of getting a display for goggle sets. Due to our limited budget, limited access to, and unwilling ability to damage proper display goggle sets like an Oculus VR set thus the search for a replacement display for our project goggle set began. The goggles set that will be used during this project won't come with a pre-built in screen due to the difference in price. Where goggle sets without a display range up to \$79.99 from the Google Daydream View while the low price range for goggle sets with a display are like \$279.00 for the Samsung HMD Odyssey headset.

The plan for the display screen used in our project is to provide the operator of the project device a simple game feedback loop for operation of the system. As the operator looks on the screen the system will read user eye position then change the depiction on the screen depending on the output response from the eye position in relation to the screen.

6.0 Project Software Design Details

For our project, we intend to create the software for the MSP430 using the C coding language. Code Composer Studio is the . The software will handle configuring the Analog to Digital signal conversion, as well as configuring the chips communication protocol. The chips primary functionality is to read the inputs from the sensor array and relay that information to the central processing unit to interpret the data using advanced algorithms. In the following sections we will go into more details of the different software that we intend to utilize for the project.

6.1 Software Purpose

The embedded software serves two primary purposes for our project: to accurately capture the signal being produced by an array of sensors, and to reliably transmit that data to another unit for processing. The first functionality will be realized through configuring the built-in ADC provided in the MSP430G2x53 family. The ADC10 supports 10-bit resolution for digital conversion, which should provide enough accuracy for the algorithms running on the central processing unit. The proper settings for the ADC will be configured using the library provided by Texas Instruments for the appropriate MSP430 family.

The second primary functionality, communicating the data, will be realized through the use of one of the communication protocols provided in the MSP430 family. The protocols provided through the microcontroller's universal serial communication interface are universal asynchronous receiver-transmitter (UART), serial peripheral interface (SPI), and inter-integrated circuit (I²C). For the purposes of our products design, we will utilize the I²C protocol for device communication. I²C provides numerous benefits, including low power usage, device addressing capabilities, up to 400kbs data transmission speeds, and programmable clock frequency.

The embedded software for our device will be written entirely in the C programming language. This is how most embedded software and firmware is written. By using C, the programmer has complete control over how the device will function, as well as access to the libraries and functions written by Texas Instruments. The programmers on the team are fluent in the C programming language as well.

6.1.1 Primary Processor Software

The software written for the central processing unit will not be written in C, unlike the embedded software for the product. The software for the central processing unit will be written in Python. Python is a powerful language, and an excellent selection for our device for multiple reasons.

The primary reason that we will be using Python in our central processing unit is because python is a versatile, object-oriented programming language that supports a wide variety of open source tools to use in the project. Its nature as a scripting language is also very helpful for development, as it allows for rapid prototyping and easy to use code. Our team collectively has experience in working in Python as well, which is part of what led to the decision to use it.

The other primary reason we decided to use Python was for our neural-network portion of the software. Python is a popular choice among data scientists and researchers, so there are many open source libraries for us to use to create a proper model for our final product. One of the many libraries available to us for this purpose is PyTorch. PyTorch is the currently in use version of an older library called Torch that was based in the Lua programming language. PyTorch is based on this older library, and is used in general machine learning and neural network creation. It was created primarily by Facebook AI researchers, and is used widely among data scientists. One of the most important features of PyTorch is that the computing can be accelerated by GPUs, which will greatly speed up the training process if the team has access to high-end computing equipment.

One of the primary deep learning libraries that our team intends on using for this project is Keras. Keras, like many other deep learning libraries, is an open source library written in Python. Keras has a very user friendly application programming interface (API). This makes the library easy to work with from the programmer's point of view, and it will enhance the speed at which we are able to prototype and tune our model to better suit our needs for the project. Keras was designed in such a way to be very versatile, and can handle many different types of neural networks and types of data to process. This is especially important to our team as the data we intend to use for our neural network is proprietary to our project.

6.1.2 Wi-Fi Functionality

The microcontroller we will be using is either the Raspberry Pi 4 Model B or the Raspberry Pi 3 Model B+. Both of these microcontrollers are capable of picking Wi-Fi network frequencies of 2.4 GHz and 5 GHz which means they will be compatible with both the 802.11n Wifi network and the 802.11ac Wifi network.

We intend to connect our main CPU (Raspberry Pi) with the MSP430 wirelessly. This wireless connection will be used to send information about the direction the user is looking to the Raspberry Pi. This information will then be utilized in an application of some sort.

6.1.3 Code Composer Studio (IDE)

Code Composer Studio is an Integrated Development Environment (IDE) that we will be using when coding the MSP430. Code Composer Studio was developed to support the development of code for Texas Instruments' Microcontroller and Embedded Processors. Code Composer Studio includes an optimizing C/C++ compiler, source code editor, project build environment, debugger, profiler, and many other features. [15] Code Composer Studio includes all of the debugging benefits from the Eclipse IDE because it is based on Eclipse. Code Composer Studio is only able to be run by 64-bit Operating Systems. The Operating Systems that are supported as of 10/30/19 are macOS, Windows, and Linux.

6.1.4 TinkerCAD

TinkerCAD is a free 3-D Modeling software made for 3-D Printing. This software is made for beginners at 3-D Modeling. We hope to utilize this software to help create some of the models to 3-D print for our project. TinkerCAD was bought out by Autodesk. TinkerCAD was made in 2010. The goal of this software was to make 3-D Modelling accessible to all. None of the members of our group have any experience with any kind of 3-D Modeling software. For this reason we hope to use TinkerCAD throughout the entirety of the project.

6.1.5 AutoCAD

AutoCAD is a 2-D and 3-D modeling software that is incredibly versatile. AutoCAD will be used in the event that TinkerCAD cannot be used to fulfill our desires in design. AutoCAD does not have the limitations that TinkerCAD has. The downside to AutoCAD is it's steep learning curve so we would like to avoid using this software if at all possible. We will stick to TinkerCAD for as long as possible. An AutoCAD subscription for a year costs about \$1610 dollars. Thankfully it is free for University Students and it is available on computers in the Engineering department.

6.1.6 PyCharm

The main development tool for our team to code the Python script with will be PyCharm. PyCharm is a Python development environment created by JetBrains. PyCharm is a tool intended to speed up the development of writing Python code by providing the user a large suite of tools to enhance the programming

experience, and make the process of writing code to PEP8 standards much easier. By using this tool, our development process will be much more productive, while still maintaining widespread Python programming standards. PyCharm provides automatic code completion, meaning that as you write code in the development environment, the editor will learn when you write certain code, and upon detecting a similar pattern, it will offer you suggestions about what the editor thinks you want to write next. This helps mitigate the issue of “boilerplate code” (pieces of code that are required by the language to be repeatedly written), and assist in dealing with long variable names, which are useful for making code easy to read, but slow down productivity due to taking longer to write.

Another important feature of PyCharm is the built-in tools specifically designed to assist in scientific programming. There is built-in support for NumPy, a popular library that enhances Python's ability to quickly process numerical data, which is useful in the application of neural networks. PyCharm also comes with built-in support for Anaconda, which refers to itself as the “world's most popular science platform”. Like many of the other sources of software we intend to use, Anaconda is free and open source. On the website for Anaconda, they claim that using Anaconda will allow the user to quickly build and deploy neural networks, and offer the following advantages:

- Collect data from files, databases, and data lakes
- Manage environments with Conda (all package dependencies are taken care of at the time of download)
- Share, collaborate on, and reproduce projects
- Deploy projects into production with the single click of a button

We are using Anaconda in our project because multiple members of the team have experience in working with Anaconda. It allows the user to easily install any library from a large selection, and gives users a way to install libraries outside of the standard Anaconda database of packages.

7.0 Hardware and Software Comparisons

During this section we will go over each and every component and the reasoning for each component. In the beginning of each component section we will go over the idea of why said component is needed. Then followed by the list of related components of the concept component needed for this project's construction. Lastly after each section there will be a brief discussion my the reasoning of the selected device of choice from the list of components presented above. The component sections are the sensor array type, goggle set, selector chip, analog to digital converter, main processor, and the display unit.

7.1.1 Sensor Array Options

During the process of this project we contemplated how we would track an eye ball's pupil. We decided to shine light on the eyeball then have an array of light detectors form around the eye and detect the light. Since the eye's pupil reflects a smaller amount of light then the skin and whites of the eye surrounding it the detector array will simply pinpoint where the lowest amount of light is being reflected and that's where the eye will be looking.

In order to accomplish this the type of light detector that we use will be crucial in making the process work. If the light detector is bad that will make the whole product will be fail to work.. If the light detector is good then it can give the product a chance to operate properly as well. We did research and came up with two possible light detectors. The first is the LDR. The LDR is good and reliable product that works in other devices. It also has a good price. The second is the IR detectors. IR detectors seem to work better in these situations and have a higher price than the others.

Part	Transition speed	Requirement	Price	Shop
LDR	About 3ms	Light intensity from display screen		RadioShack on 2001 S Orange Ave Orlando
IR	About 10 μ s	Infrared light from transmitter		RadioShack on 2001 S Orange Ave Orlando

Table 8 Transition Speed Comparison Between Detectors

7.1.2 Choosing the Sensor Array

Choosing the type of light detecting to make our sensor array took considerable consideration. For instance if we chose the LDR ones then it would be cheaper. But they would also be slower. If the sensors are slower than they will detect light slower. Thus if the light being detected is slower than the amount of time it takes to find out where the eye is looking will be slower. Furthermore, if the sensor is too slow then theoretically someone could be looking one way then for a split second look another way and then look back at their original spot and then the sensor would not be able to detect that because it happened so fast. However, if the sensors are fast enough then they will be able to detect every movement the eye makes no matter how fast it is. The faster the light detector is the better the product will be in tracking the movement of the eyeball.

The LDR was good and fast, however the IR detectors were faster and having said earlier the faster the better. In fact the speed of the light detectors will put a cap on everything else. It won't matter how fast our selector chip is or how fast our analog to digital convertor chip is if we do not have a fast sensor array. So for this light detector we wanted the fastest we could get. Since the IR is faster than the LDR we chose the IR even though it is more expensive.

Not only is the IR more expensive, we will also need IR emitters where as if we chose the LDR's we would simply use the display screen's light to shine off the user's eye and into the light detector. So in choosing the IR we will need to buy emitters where as with the LDR we will not, which means more money spending. However these sensor arrays are arguably the most important part of the product and therefore need to be the best.

The speed of the IR vs the LDR had us decide on the IR over the LDR there is also another factor as well. The LDR can gather up dirt and is not as reliable and long lasting as the IR. The IR is more durable than the LDR. The more durable the piece is, the more we will choose it over the other piece. Therefore the fact that the IR detectors are faster and more durable are the reasons why we chose it over the LDR detectors even though we will have to add IR emitters and the cost of the IR is more than the LDR.

7.2 Goggles Set Requirements

For this component section, the choice for head wear/ harness for our project varied from product to product. For this we wanted a set that would be mostly a firm mounting point for our system but with low cost due to the limits of funds. For the design of the goggles set used for this project it didn't matter as long as the

structure of the set was strong enough to hold the sensor array. So this mostly meant the use of plastic, or cardboard if no suitable set is found, for the main structure of our project. In the long run the system structure only has to last for the term of the project since this device is only a proto type, and the only real components that matter are the converter board, Sensor Array, and main processor. Another condition of the goggle set that is needed for this project is the ability to be reconfigurable for the system of the converter board and sensor array to fit inside the goggle set along with the user. The goggles set would be of no use if we can't fit these three things comfortably together as a unit.

Product	Structure	Straps	Plus or Negative	Cost	Shop
UTOPIA3 60	Solid plastic	T shape	Adjustable focus Leather seal	\$36.82	Walmart
Google Cardboard	Cardboard	Single	Not durable and cheaply made	\$18.79	Walmart
Self built	EVA foam	Any	Takes time to make	Un-known	Any-where
Google Daydream View	Hard plastic	Single	Can't be modified easily	\$79.99	Walmart

Table 9 : Comparison of Goggles

7.2.1 Choosing the Goggles Set

The goggle set could be the most important part for the user. When you are using the eye tracker you don't want to be uncomfortable. In fact, someone might possibly wear an eye tracking headset that is comfortable but does not have good tracking capabilities, but no one would wear an eye tracking headset that is good at tracking the eye but is so uncomfortable it will give them a headache. This is the reason the goggle headset is so important because it can determine whether or not someone will even wear the device.

In deciding on a headset there are a couple of things our team is looking for. Here they are:

1. Comfortability. If the headset is not comfortable the user will not want to wear it. Comfortability includes the strap it has, the cushions inside the

helmet, and the weight of the helmet. These three factors were considered in determining the comfortability of the headset.

2. Price. If the unit is too expensive then the user would not be able to purchase it.
3. Customizable. The headset can't just be any headset. The head set needs to be customizable for us to turn it into the eye tracking headset that we are designing.

7.2.2 Self Built

The self built head set would be a headset that we make from scratch. Meaning we go out and buy the strapps. We buy the frame to put the screen in. We buy the cushions on the headset to make it comfortable. We buy all the stuff that goes into designing a headset. Then we put it all together into a workable headset for an eye tracking system.

This headset, or rather method, would not work, or rather is not practical, because it would consume our time. It would take too much time to look for individual parts of a headset, design it, and then finally put it all together. Plus, it would probably cost just about the same, maybe even more. This is the reason we did not decide to make our own headset.

7.2.3 Google Cardboard

The google cardboard is a makeshift cardboard helmet. This one seems great because it is very light, which will be comfortable for the user and the strap was good as well. However, it did not have any cushions in the helmet. Therefore the helmet was only partially comfortable since it had good weight and strap but had bad cushioning.

The google cardboard price was only around \$18, which in our research online of headsets that price is very low. We remembered that the cheaper the product it is to make the less the consumer will have to pay. The less the consumer has to pay the more likely they will use the device.

The google cardboard is made of cardboard and therefore easily customizable. We could easily cut things out we don't want or glue other cardboard in places where we do want it. However, the fact that the headset is made of cardboard and the way to customize it would be glueing things together would make it easily breakable and not as durable as a headset made of metal or even plastic. Another fact to consider with this headset is that cardboard would not be safe to have surrounding the electrical circuits and chips that we would be using in the unit.

The google cardboard may seem good to use for an eye tracking headset, but it is not. It is not good to use because the comfortability is only partially comfortable and it is not durable or customizable. To include the consumer would not want to purchase something that will not last.

7.2.4 Google Daydream View

The google daydream view is a lot better in durability than the google cardboard. It also seems more comfortable too. However there were some flaws in it that made us decide not to use it for our eye tracking product as well. The first is that the price was too much. In order to have a good headset the price needs to be low and the google daydream view was too expensive. The second flaw is that it is not easily customizable. This is of course necessary for us since we are taking a headset and customizing it for the users. Therefore, we decided to not choose this headset because it could not be easily customizable and the price was too high.

7.2.5 Utopia 360

The Utopia 360 is comfortable for any user. The strap that it comes with is adjustable and can fit on multiple sized heads. It is not too heavy too straining of the neck. It has numerous cushions for the user to wear it comfortably. The Utopia 360 meets all the requirements of the comfortability factors as discussed above.

The Utopia 360 price is good as well. It is not as low as the google cardboard, but it is twice as cheap as the google daydream view. So it does not have the best price, but it is still not too high.

This headset is also very customizable. The front of it has lots of room to put our IR detectors and IR emitters in between the screen and the eyeball. The Utopia 360 therefore meets all the requirements that were needed for choosing the headset. It is comfortable, the price is in the good range, and it is easily customizable for us to turn it into our eye tracking headset. This is why we chose the utopia 360 to use in this design.

Below is a picture of the headset we have chosen. It meets all the requirements and as seen below it has enough padding to be comfortable and enough room near the eye to fit all the hardware that it needs to fit inside for proper use.

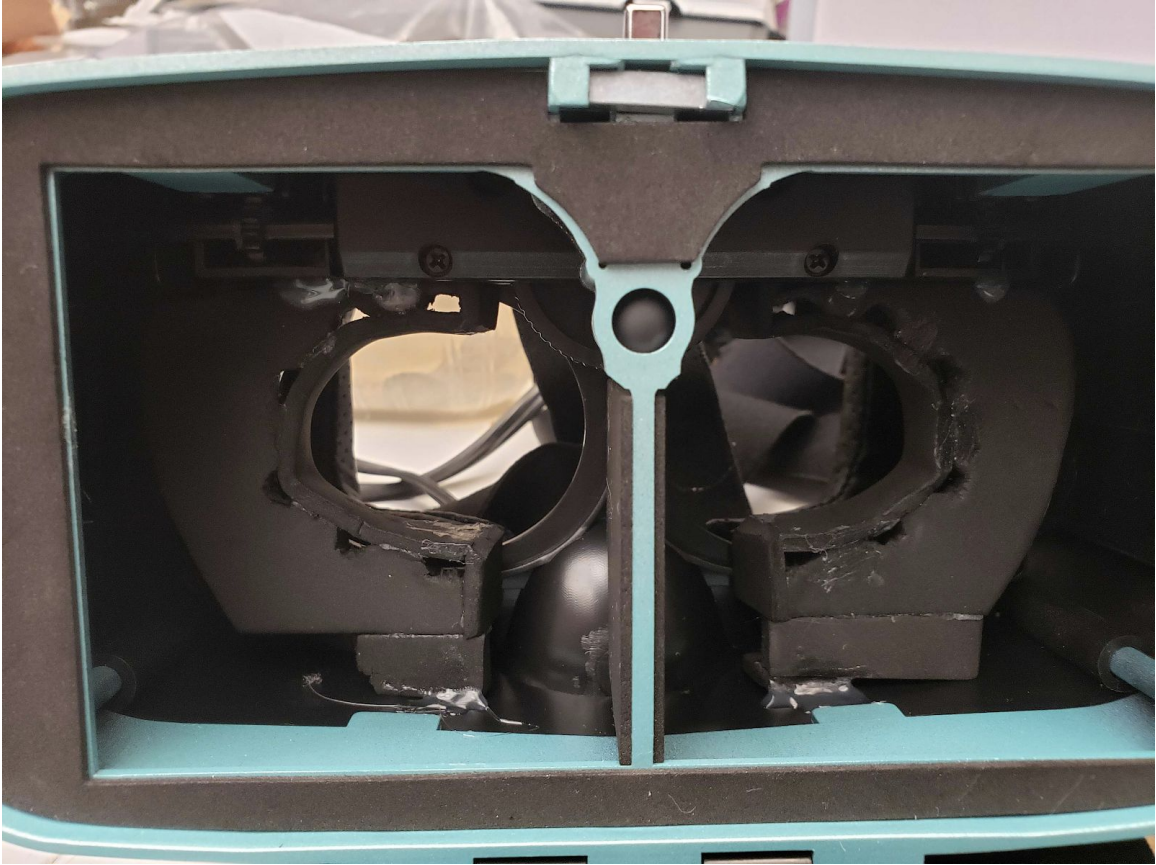


Figure 36: Utopia 360 Headset

7.3 Selector Chip

For the choice in selector chips there are many in concept that could be used for the purpose of this project. Be it of a relay chip to that of an inverted multiplexer, there was a little to much room for choice. The idea of why this project devilmnt device needed a selector chip is to help limit the amount of out data lines for the system.

The concept comes from the fact that each sensor outputs multiple lines of data, depending on the system's analog to digital converter's bit accuracy output, for the system to read and sense that there is an array of sixteen individual sensors, in simple terms there will be to many data bits to pass for any system to read. In practice this component selects one sensor analog data line then passes it to the converter to change it to the digital data line bus the system will read for our calculation of the user's eye state. The requirements that this component needs for this project device are described below.

1. This switching device needs to be fast at switch lines as to not slow down the process of the system calculations.
2. This switching device needs to be able to handle as much individual input lines to one output line.
3. This switching device needs to cause as little resistance on the analog data lines as possible as to not corrupt the data.

With these restrictions we came up with the list of component selector chips in the table below. The switching device we went with was bidirectional multiplexer chips for this project. These chips are broken down into variables like resistance capacitance, configuration or how the inputs are connected to the outs and how fast the system is at changing lines between the ins to the outs.

During the search for the selector chip we found out that the somewhat mechanical switching chips, like the CD74HC4067, physically break the connection of not in use lines temporarily as a means to prevent the unwanted inputs going through. Where chips like CD4067B use transistor gates to prevent the transfer but this can leak more than comparison to that physically breaking the line. Another thing discovered was that the delay times between the transistor gate multiplexer vs the physical breaking line multiplexers, the transistors are a bit slower. However this increase in speed for the temporary breaker multiplexer does come at the cost of added resistance to the chip but not much.

Most multiplexers have very low resistance in comparison to that of the rest of the system, like a few hundred ohms for the selector vs the tens of thousands of ohms from the voltage dividers and that of the analog to digital converter system. So the difference in resistances were not to much of a deal breaker in comparison to the delay time between them because the system will be slowed down by the delay time, plus a little more for safety margins to make sure the line is stable to read, multiplied by the number of input receivers used in the system.

Chip	Configuration	Resistance	Delay	Cost per unit	Place ordered
CD4067B	16 to 1	125 Ω	220ns	\$1.34	digkey.com
CD74HC4067M	16 to 1	160 Ω	51ns to 49ns	\$0.93	digkey.com
CD74HC4067M96	16 to 1	160 Ω	51ns to 49ns	\$0.80	digkey.com
MPC506AU/1K	16 to 1	1.5k Ω	200ns to 250ns	\$14.49	digkey.com
CD74HCT4067M	16 to 1	180 Ω	60ns to 55ns	\$0.93	digkey.com

Table 10: Comparison of Selector Chips

7.3.1 Choosing The Selector Chip

In choosing the selector chip this took a lot of consideration. For instance, if we picked a really cheap one then the delay might be really high. If the delay is extraordinarily high then it will not do what we need it to do. What we need it to do is select which IR the MCU is going to be gathering data from. This product does not look at all the IR detectors and analyzes them at the same time, instead it analyzes each one individually. Therefore, it will select an IR detector and gather the data from it and determine if the eye is looking towards it if the light its deteting is low. The problem if we choose a cheap selector chip is that the time it takes to switch between the IR detectors will be slower. If it is slow then it will not read through the Array of IR's fast enough and a glitch output of where the eye is will occur.

There are quite a few factors in deciding which selector chip we should use for our design. Not only does it need to have a short delay between choosing between the IR detectors, it needs to have a low resistance. If the resistance is high it will use up more of the power. If it uses up more power then it will drain the battery faster which will lead to a shorter battery life. This is why the resistance should be lower. Therefore, three key factors are at play in deciding which selector chip to choose from.

The three factors are as stated below:

1. Price. The cheaper the selector chip the cheaper the whole product will be for sold for in theory.
2. Speed delay. The speed delay that the selector chip can determine the quality of the product as a whole as was explained in the paragraph above.
3. Resistance. The resistance of the selector chip can determine the power used by the product and as a whole determines its quality as explained above.

7.3.1.1 CD4076B Chip

The first chip, CD4076B, that was looked into seemed to be the better product. It had really low resistance. However, it had a slow delay speed and was quite expensive. The second chip we looked at had a significantly better speed delay and was 40 cents cheaper. Both were good signs. However it had a higher resistance. We had to choose to have a better battery life or better quality in detecting the eye and being cheaper. Since the resistance wasn't too great of a difference and therefore only a slight change in battery life, where as the delay was greatly different, we chose the one with better delay and better price.

This is where the third chip came in. The third chip had all the same qualities as the second chip except it had a better price. We were going to get this one but then realized this one was bigger. The third was bigger than the second. This was an issue due to the fact that if the chip doesn't fit into what we are using it for then it is entirely useless. Since the third chip was too big in size we could not choose it and therefore stuck with the second chip, the CD74HC4067M.

You may notice a fourth chip on the table. That one was considered, however if you look it has bad resistance, price, and delay speed. The fifth chip was considered also but since it had better speed than the first chip, yet the price and resistance were not as good as the chip we chose and therefore we did not choose that chip either.

The first chip, CD4076B, seemed really good at first. It's resistance is very low. In fact it is the lowest out of all of the other selector chips. The resistance being low is great because if the resistance is low then less power is needed to be used to power the unit. If there is less power to power it then that means it costs less money for the user. That is a good thing for the users. And this chip is the best in this specific factor in deciding which selector chip to use.

The price is sufficient, but not great. At first glance it looks like it is very cheap, however comparing it to the other chips the price is the second most expensive. Now if we build the device to be long lasting then the price someone would spend in using power to power a cheaper chip would end up being more than the

difference in the price between this chip and the least expensive chip in the table shown above.

The last factor to consider is the speed delay that it has. This is the number one priority because the speed delay is a huge factor in the quality of the device. The speed delay of the specific chip, the CD4076B, is very high compared to the lower ones. This one is 220ns and the other ones are as low as 49ns. The difference between that is almost 200ns. That's not much considering how little a nanosecond is. Another way to look at it is that for this chip to select one chip and then move to the next another chip could have already looked at 4 and be halfway to move on to the next one. Anything that can make this faster is better.

This chip was not selected because even though it had a low resistance and the price wasn't bad, it had a bad speed delay and that is a very important factor.

7.3.1.2 CD74HC4067M96 Chip

The third chip on the table, the CD74HC4067M96, had a better price and speed delay than the previous discussed chip, the CD4067B. This third chip has slightly higher resistance at 160 Ohms. This isn't too big of an issue. In fact, it doesn't even matter. The difference is just 35 Ohms. 35 Ohms won't consume that much more power and therefore this chip won't consume that much power than the first chip.

The price of this chip is the cheapest chip there as shown on the table and up for consideration for the device. The cost is at only 80 cents. This is great because if the chip is cheaper than we can make the product cheaper as well. The price of this chip is 80 cents. The price of the other chip was one dollar and 34 cents. The price difference is 54 cents. Therefore since we only need one chip we could make the device 54 cents cheaper and have the same profit. That is not much of a difference and people will not care if they have to spend an extra 54 cents or pay an extra 54 cents. If there were more chips than the price would increase and it would matter more to the users.

The best thing about this chip and what makes it better than the first chip is the speed delay. This chip's speed delay is only 51ns to 49 ns. Which is roughly 4.5 times faster than the first chip. This means that when the selector chip has gone through it twice, this chip has gone through it nine times. That is something worthy to note and something worthy to make a decision off of.

7.3.1.3 MPC506AU/1K Chip

This fourth chip, the MPC506AU/1K, must be a good chip, otherwise there would be no point in making it and selling it and buying it. However it does not do what

we want or need. The resistance is not just higher than the previous ones but is almost ten times higher than the other ones. That is absolutely terrible. This means that it will consume ten times as much power as the other chips which would burn through the users batteries and money very fast.

This chip also has a bad speed delay. The speed delay is right around the speed delay of the first chip. This could mean its speed isn't too bad. Or it could mean that the first chip's speed delay is terrible too and it is a great choice that we did not decide to go with it.

The price of this one is probably the worst out of all of it. It is fourteen dollars and 49 cents. That is about fourteen dollars more than the other chips. The previous chip was only 54 cents of a difference, which really isn't too big of a deal. This chip has a fourteen dollar difference, which is a 28 times difference. This is significant and something to note down and decide which chip to pick based off of that. The fact that the chip is very expensive and has high resistance are the reason why we did not decide to choose this selector chip.

7.3.1.4 CD74HCT4067M Chip

This fifth chip, the CD74HCT4067M, is a good chip. The resistance, price, and speed delay are all very good. However there are reasons why we did not choose it and those reasons are best seen if you compare this chip to the third chip on the table, the CD74HC4067M96,

This chip, the fifth chips of the group, has a resistance of 180 Ohms. 180 Ohms is a good resistance. It is not great, but it is not great either. It is not low or high. It is a good balance and would have been a fine selector chip for us, however the resistance of the third chip is better because it has a resistance of only 160 Ohms.

This chip's price is not bad either. It is cheaper than the first chip and under one dollar. We would not have to make the final price of the product more to compensate for the selector chip like what we would of had to do for the fourth chip. However when you compare it to the third chip it is not as acceptable. The third chip has a price of 80 cents while this one has a price of 93 cents. This chip's price is low, but it is not as low as the third chip.

This chip's speed delay is very quick, which is exactly what we are looking for in a selector chip for this device we are designing. It has a speed delay of anywhere between 55ns and 60ns, however it is not as fast as the third chip. The third chip is only 4ns to 11ns faster than this chip's speed delay, but why would we pick this one if everything in the other one is better. One might say 13 cents isn't much, or 20 Ohms isn't much, or 4ns to 11ns isn't much, but what is the point in picking this one if the other one is better in everything else.

7.3.1.5 CD74HC4067M Chip

One might be wondering why the third chip was the best yet this chip, the CD74HC4067M, is the one selected. The third chip's resistance and this chip's resistance is the same and so is the speed delay. The only difference is the price. The third chip's price is only 80 cents, yet this chip's price is 93 cents. If the only difference is 13 cents it would make sense to just go for the cheaper one, however we did not and the reason is crucial.

The reason we did not go with the third chip is that upon trying to use it we realized that it is too big to use. The third chip was not able to fit in the headset the way we needed it too. This of course isn't just a problem, but a problem that immediately takes the third chip out of the running. We were then going to just go with the fifth chip since it obviously was the second place winner, however we then came across this chip, which is the same chip as the third chip yet smaller and slightly more expensive.

Since this chip's price went up it was still the same price as the fifth chip and it still had better resistance and speed delay than the fifth chip. Since there was no other chip better than this chip, the CD74HC4067M, as discussed and explained in the sections previous to this, we decided to use this chip.

Below is a picture of the chip that we decide to use for our project. As you can see it is small enough to use for our design and as shown and discussed above it is the best option that we had as well.



Figure 37: The CD74HC4067M Selector Chip

7.4 Analog to Digital Converter Chip

The decision on analog to digital converter is both harder than that of the selector while also being easier at the same time. In practice there are hundreds of converter chips/ builds used into the modern days electronics such as it being in things like the converter for joy-stick controllers. The idea is that we need a converter that can output as much data bits from the analog signal as possible. During the selection of this chip to be used for our project, we found that the converter for the game controller's joy-sticks basically ran what we needed for the system IR array since they run quick, with high resolution of the location of the position of the joy-stick. With the concept of the controller's converter in hand we went out looking for similar designs of chips that would do about the same for us during this project. This meant that the converter needs to be about a twelve bit converter with quick response times.

One great example of this component in practice is the ADC, or analog to digital converter of the MSP430, or to be more exact the FR6989 version of the microprocessor unit. This component has a built in analog to digital converter that

operates for the functions of the MSP430 like light detection and or joy-stick position monitoring.

Converter	Resolution	Cost	Shop
AD7829BRZ	8-bit	\$11.99	mouser.com
MSP430FR6989	12-bit	\$8.30	Digikey.com
TLC0820AIDW	8-bit	\$5.98	mouser.com
MCP3008	10-bit	\$3.75	adafruit.com
ADS10151DGSR	12-bit	\$2.76	Digikey.com

Table 11: Comparison of ADC Chips

7.4.1 Choosing The Analog to Digital Converter Chip

The converter chip is important. We could not just choose any chip and use it. First it needed to be compatible with what we are using it for and then it needs to be able to ship to us on time and then it needs to have a good cost and lastly and most importantly it needs to have a good resolution. If the chip has a bad resolution then the quality for what we are doing will drop significantly. There were also quite a bit of chips out there, hence there are five on the table that we considered before deciding on the MSP430FR6989.

The first Analog to digital converter chip was the AD7829BRZ. This chip was expensive and had bad resolution. We did not pick this chip due to those factors. The TLV0820AIDW chip was a lot cheaper, which was really good. However, it still had a bad resolution. The MCP3008 chip has a better resolution and is cheaper and the ADS10151DGSR chip is the cheapest and has 12-bit resolution, which is one of the best resolutions there is available. The ADS has just as good resolution as the MSP 430, which is good, and is cheaper, which is great. Therefore one might question why we chose the MSP430 chip over the ADS if they have the same good quality as each other but the ADS is shown to be cheaper of the two.

The reason the MSP430 was chosen over the ADS chip was because the MSP430 chip was already owned and available with no need to spend additional funds. Meaning the price would be zero and the fact that we have it on hand, the time it took to get to us would be zero as well. If we chose the ADS then it would have had to deal with the shipping time with the possibility of delays to getting to us, which could have created a very large problem, and of course we would have paid for it as well, where as the MSP430 was already in possession. The reason the MSP430 and the ADS analog to digital converter chips were chosen over the other analog to digital converter chips was because they had better resolution.

7.5 Microcontroller Unit

To make the choice as to what microcontroller we will use, it is important to compare the functionality of each unit. For our purposes, we need to look at the built in Analog to digital converter, the GPIO pins on the device, the available storage on the device, and the price per unit. We have decided to move forward with the MSP430 family of microcontrollers, due in large part to the amount of features on the controller. Most of the devices in the MSP430 family are capable of analog to digital conversion, as well as supporting numerous IO pins. They also boast impressive low power usage functionalities, as well as providing various solutions on how to transmit the data that will be received from the sensor arrays.

This line of Texas Instruments controllers aren't the only devices on the market, but due to the teams relative experience using the devices, as well as the plethora of resources available for the device at no cost to the team, we have decided that this choice is the most optimal. Other devices that could potentially solve the problems presented by the design are a number of arduino controllers. However, resources for those devices are much more limited, and the overall experience of the team with those types of devices is much lower. Below is a table of a few devices from the MSP430 family, with our choice bolded:

MCU	GPIO Pins	ADC Channels	NVM (kB)	Aprx. Price (USD)
MSP430FR2111	12	8	4	0.49 1ku
MSP430G2131	10	8	1	0.49 1ku
MSP430G2333	24	8	4	0.65 1ku
MSP430G2553	24	8	16	0.90 1ku
MSP430FR2153	44	12	16	1.05 1ku
MSP430FR6989	83	16	128	3.75 1ku

Table 12: Comparison of Different MSP430 Launchpads

7.5.1 Choosing the Microcontroller

The Microcontroller known as the MCU is the backbone of how the logic is going to be implemented onto the PCB. There are many considerations to make when it comes to exactly which microcontroller would be best suited for our product. We need enough pins to interact with all of the peripherals on the board, as well as setting up lines for communication.

Another important consideration to make is the number of ADC channels available on the device. Ideally, we want at least 8 channels, for the number of sensors per array device. The resolution is also important, however the lowest resolution of any of the listed devices is 10-bits, which should be more than is required for our products design. The amount of non-volatile memory is also an important consideration for our device. The code that has been written for our products functionality has not yet been finalized, however there will need to be enough space available to load all of the code, as well as enough ram to run the functional code.

As can be seen above, a number of chips from the MSP430 chip family were considered for the design. We decided to go with the MSP430G2553 for a number of reasons. Primarily, it covers all of our design requirements for the microcontroller. It has a 10-bit ADC, 16KB of non-volatile memory, and enough GPIO pins to interact with all necessary peripherals. However, you may also see that the MSP430GR2153 and the MSP430FR6989 also cover these requirements.

The reason that these devices were not chosen is because they are overly complex for the functionality required. Using either of these devices would result in most of the pins going unused, as well as most of the on-chip functionality. However, it is possible that we still may want to consider using these chips in our final product. While they are certainly much more powerful than what we require, there remain concerns that the chip selected may not be quite fast enough to accomplish the necessary tasks. Primarily, we require that the chip quickly gathers and sends the data to the primary processing unit, as that will determine the viability of our product. Ideally, the end user will not notice any undue latency when it comes to operation of the final product.

If the clock speed, sampling speed, or data transfer rates on the chip we selected are not suitable for this operation, we may need to select one of the more powerful chips listed above. Even considering these factors, it is also important to keep the final budget in mind. It may be less expensive to use two microcontroller units rather than one very expensive one. But in that case, it is possible we may run into synchronization issues when it comes to communication between the board and the primary processing unit.

The MSP430FR2111 is a promising chip for our purposes. It features a 16-bit RISC architecture, with a core clock speed of up to 16MHz, similar to many of the other MSP430 chips. It can operate on voltage ranges from 3.6V to 1.8V, which makes it possible to use with a large variety of different power sources. It also supports low power usage, at 3V it uses 120 $\mu\text{A}/\text{MHz}$, and in standby mode it uses 1 μA , even while supporting a real-time clock. Similar to our choice, it provides an 8-channel, 10-bit SAR core analog to digital converter, but only has a 1.5V internal reference. Unfortunately, it only has 3.75KB of nonvolatile memory, and only supports 12 I/O pins, which will not be enough for our design.

The MSP430G2131 is another alternative chip that we explored the possibility of using in our product. The chip provides many of the same functionalities of the other chips in the MSP430 family, and is designed for low-power, low weight operations. While it does support the ADC functionality we require, it does not support the amount of on-chip storage we require for our project.

7.6 The Main Processor Unit

For this project we need an integrated processor board to be capable of performing multiple operations like the eye track function, video processing and outputting video to the display unit. The primary function of this processor is to take the input data from the sensor arrays, and use an algorithm to then determine the position of the eye. Once given the relative position, we can then feed that data into whatever program we want.

The requirements for the processor unit are as follows: it must be fast enough to run the algorithm quickly as possible, so the user does not encounter any undo lag when using the device, it must be able to quickly encode video and transmit the data over a wireless connection, and it should accomplish these tasks while drawing as little power as possible. The unit our team decided to use for our product is the Raspberry Pi 4.

Name	Price (USD)	CPU clock speed	Size (in)
Rock64	44.95 (pine64.org)	1.5GHz	2.2x3.4
VIM2 v1.4 Basic	79.99 (khdas.com)	1.5GHz	2.3x3.2
Raspberry Pi 4 model B	47.69 (amazon)	1.5GHz	2.2x3.3
ASUS Tinker Board S	89.40 (amazon)	1.8GHGz	4.5x6
ODROID-XU4	49.00 (hardkernel.com)	2GHz	2.3x3.3

Table 13: SBC Comparisons

7.6.1 Choosing the Main Processor Unit

There are many choices of single board computers (SBCs) on the market. With such a wide variety available for use in the product, it is not a simple or direct choice for the team. Our primary motivation in picking the Raspberry Pi 4 Model B was for its large variety of documentation and development resources available to the team. This device will be integral for our product to function. One of the primary factors in realizing we would need an SBC for processing was by studying other similar products currently on the market. Most of these products work by having a headset or surface mounted peripheral capture the data, and then sending that data to an external processing unit. For our purposes, the Raspberry Pi will act as that external processing unit.

The other boards listed above would also make great alternatives to the Raspberry Pi, if the need arises. All of the boards above fit the required criteria for acting as our external processing unit. So at that point, our justification boils

down to the price, the size of the board, and the ease of development on said board.

The ASUS Tinker Board S is a very powerful option, with many peripherals. It could reasonably do everything that we require the board to do, along with any additional features we could possibly want to add in the future. However, the board is much larger than any of the other alternatives, and comes at a quite prohibitive cost of \$89.40.

The Rock64, which can be found at pine64.org, is a promising alternative, should we run into any issues using the Raspberry Pi. It comes in a similar size to the Raspberry Pi, and comes at roughly the same price (however the price shown does not include shipping). Our reason for not selecting it as our primary processing unit is that it does not have the same level of community support and available documentation as the Raspberry Pi does.

The VIM2, which can be found through Khadas, would also make an excellent selection. Khadas provides much of the documentation we would require, even if it does not have the same level of community support as the Raspberry Pi. It also has a large amount of RAM and storage, as well as built in AP6356S Wi-Fi, and Bluetooth 4.1. However, these additional features come at a premium, and puts it out of our desired price range.

Our last consideration for a possible SBC was the ODROID-XU4, available from hardkernel.com. This still remains as a possible alternative, should we encounter any unexpected issues in implementing the Raspberry Pi. The primary reason that this device was not selected was because the Raspberry Pi edged it out in terms of budget considerations, and development tools available.

7.7 The Display Screen

For the choices of display screens we are limited in our selection due to the scope of the requirement of needing to be able to fit inside of the goggle set. The display itself does not need to be the top of the line high resolution screen since this project is not testing on the premise of this project is for the device to read and understand the operator's eye position and state, rather than being an entertainment device. This is only the real constraint to this project.

Screen	Resolution	Screen size	price	Weight	shop
Axion ACN-55 07	Unknown	5 inches	\$70.00	1.42 pounds	Amazon
BW TFT LCD	320 x 240	3.1 inches	\$20.00	.4 pounds	Amazon
Phone	720×1280	4.7 in to 6.5 in	N . A .	.3 pounds	N . A .

Table 14: Comparison of Displays

7.7.1 Choosing The Display screen

Choosing the display screen was almost easy to choose. However there was still some considering that needed to take place. There were a couple of requirements that we needed in a display screen.

1. Functional. It needs to work well and be able to do the tasks that we set forth for it to do and what we are wanting it to do, which is displaying what the camera is seeing so that the eye can see it and then the device can actually track the movement of the eye.
2. Price. Price is always a concern. If the products display screen is too costly then no one will want to purchase it because the product will be too expensive.
3. Size. The product cannot be too large due to its need to fit inside of the headset.
4. Weight. The weight of the product is important because if the device is too heavy it could put a strain on the user's neck and cause damage. Or if the screen is so heavy it could be too heavy for a user to even wear it at all. Therefore, the device needs to be light and compact enough for a user to strap it to their head.

When we were choosing the headset these were the three things we were considering. We had to find a screen display that fit all of them. Now some of the screens were better in some areas than others and so we needed to then decide which requirements we wanted more than the others.

7.7.2 Axion ACN-5507

Choosing this one would have been a mistake. The reason is because it fits none of the four requirements discussed above. First the screen would not be functional for what we are needing it to do. Or it might be. We did not spend too much time worrying about if it would be functional because it did not fit any of the other requirements such as price, size, and weight.

The price of the television screen known as TV is \$70.00. That is way too expensive for our design. If the screen display alone is over \$70.00 then the product, in order to make a profit, will need to be more than that. The goal would be to create a device as cheap as possible and that does not include a screen display that costs \$70.00, which is way too much.

The TV's size is also way too big to fit inside the headset. The TV is five inches. Those five inches also do not include the stand it is on and the thickness of the TV. The stand would keep it from being able to easily fit into the headset. The headset could be adjusted so that it would be able to fit, however it would be better to find a screen that fits the headset without causing extra work to make the screen fit it. The TV is also quite thick. This is bad because due to the thickness it would not fit in the headset without modifications. Therefore due to the stand and the thickness of the TV it does not fit the requirement of size as discussed above.

The last requirement it does not meet is the weight of it. The weight is 1.42 pounds. That much weight would make it too heavy for someone to strap to their head and not get a neck strain after a minute. Therefore due to the fact that this TV is not functional for our purpose, the price is too much, the size is too large, and the weight is too much gave us the ability to clearly decide that this TV will not work for our device.

7.7.3 BW TFT LCD

This component would not work due to the fact that it does not meet all the requirements that our screen display needs to meet. For instance, this screen would be very difficult to make it functional for our uses. The price of the object is very low. This is exactly what we need because the price we are looking for needs to be low. This screen is only \$20.00, which is not too expensive yet still a good price for it to work with our product.

This product though does not meet the size requirement because once again it has a stand. The stand would cause us to take the headset we chose and modify it so that the TV fits the headset. This would cause a lot more work and possible

choosing a different headset. This is not what we want, so therefore this is a huge negative if we chose this display.

This TV does have a good weight with it only being .4 pounds. This is a whole pound lighter than the previous one. So if this screen was compared with the other screen already discussed than it has better prices and better weight, but it still does not meet the functionality and the size requirements. Therefore this screen device will not work, even though it is better than the other screen display already discussed.

This display was also considered because it can be used for a camera when you back up. This might be useful for us because we are using a camera in our application and therefore if this device might work well with our function. However, this TV uses wires to connect to the camera. This might not work for us because having too much wires will cause the user to be constrained by wires, this is not what we want. We want the user to be as comfortable as possible and if we have wires running everywhere then it will cause the user discomfort.

7.7.4 Phone

The phone is the one we decided to use because it meets all the requirements for the screen display that we need. A Phone is able to work for our device. A Phone is able to display the stuff we want it to display from the camera. Phones are usually expensive. However, it does meet the price requirement because we will be using one of our own phones. Therefore, the phone is actually costing us nothing because we already have it. This is great, because there is nothing cheaper than something being free.

The phone is also a great size. It is slim. The phone's thickness is usually never thicker than a quarter of an inch. This is great because it can easily fit inside the headset. A Phone also does not have a stand. The TV's we found and already discussed had stands that would require us to change the headset or modify our headset so that the screen would be able to fit into the headset. With the phone that is not the case. The phone has no stand. Therefore, we do not need to modify the headset or get a new one that would be able to hold a screen with a stand.

The Phone is also light in weight. It is actually lighter in weight than any of the other devices we looked into. The two other TV screens looked into were 1.4 pounds and .4 pounds. The Phone is lighter than both of those. This is great because since the phone is so light the user can easily strap it to their head and wear for quite some time before getting any strains in the neck. If the phone were as heavy as the first screen it would be dangerous to consider because it could cause the device to not be usable because it would cause too much stress on the user's neck.

This is why we decided to use the phone as our screen display. The phone is functional for our product. It can actually work for us. The phone is already owned so therefore it will cost us nothing. The phone has a good size and will easily fit inside the headset. The phone has a good weight and therefore the user will be comfortable wearing it.

8.0 Project Testing and Prototype Construction

During this part we started off roughly as first to prove the concept of eye tracking through the use of an array of sensor around the user's eye. At this point in the project we were set to use LDR before we changed over to the IR concept. As seen below in the image it was the first attempt of eye tracking with the sensor. When testing the system state values of the human eye it brought up rough constant data of each point around the eye. As shown in table15 vs table16 we can see a simple difference between eye states and how we can read the current eye states.

To explain the table roughly follows the law out of the sensors in the array around the eye. To put it in simple terms the sensor returns a value depending on the amount of reflectance from the surfaces of the eye. The data showed that the eyelid reflects the most as it should since if it didn't they would mostly be useless. Next highest reflectance from the eye was the whites of the eye which the system gave it as a medium level in comparison to the eyelids and pupil/iris of the eye. Lastly the lowest reflectance found by the system was that of the pupil/iris, which makes sense since that's the part of the eye that takes in light and absorbs the energy of light to process for us to see. This means that the system basically looks for low points in the array in comparison to the standard look forward state of the eye.

It was after testing that we decided that we should change to IRs due to the fact that IRs are faster at changing state and are both safer and more stable to work with than the LDR system we were working with before.

**Eye State Value Returns for
Straight Forward**

High	High	High
Low/Medium		Low/Medium
Low/Medium	Low/Medium	Low/Medium

Table 15: Eye State Value Table

Eye State Value Returns for Look Left

Medium	High	High
Low		Medium
Low	Medium	Medium

Table 16: Eye State Value Table

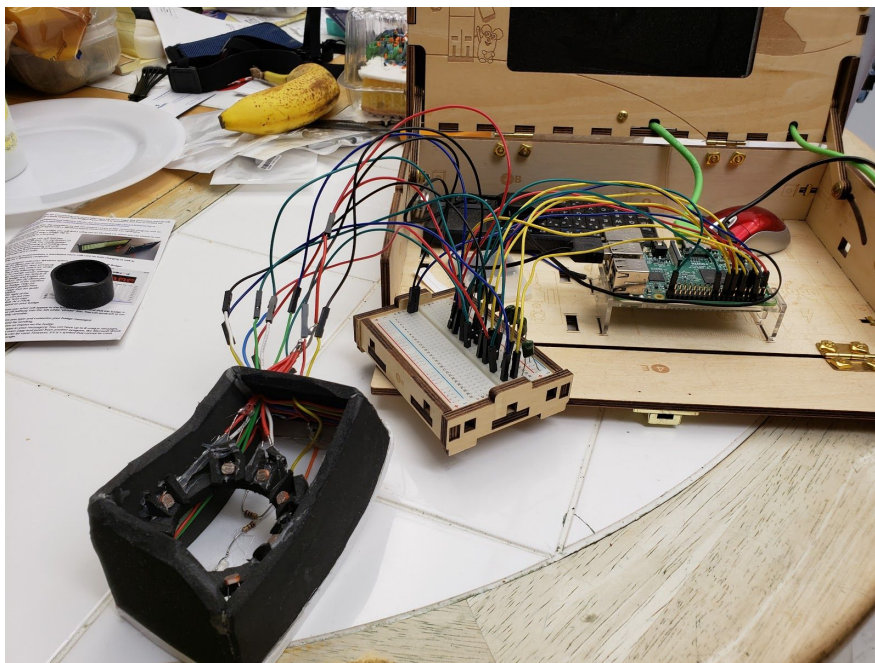


Figure 38: Prototype 1

8.1 Prototype Sensor Design Test

For this project the system was first tested with the concept of the LDR used from the previous iteration designs of the sensor in part 5.2.1. As seen in the figure 7.1.1 below the LDR are set in a full ring holder design instead of the full due to the fact that this was just a proof of concept to see if it was even possible, which thankfully it is. The sensor prototype as the name implies it is a prototype and was quickly thrown together to test the idea of tracking the eye with the use of an array of sensors instead of a camera. The prototype is an array of eight LDRs arranged a simple box like coordinate system as seen in table17. At the time the system was not given a proper display but that of a reflective white plastic sheet

covered with red LEDs to project light onto the eye for testing the LDRs. The system in total was basically a stepping stone to the real thing and gave a basis for the real product to be made.

LDR Sensor Setup of Prototype Sensor

LDR 1	LDR 2	LDR 3
LDR 4		LDR 5
LDR 6	LDR 7	LDR 8

Table 17: Table Showing LDR Positions

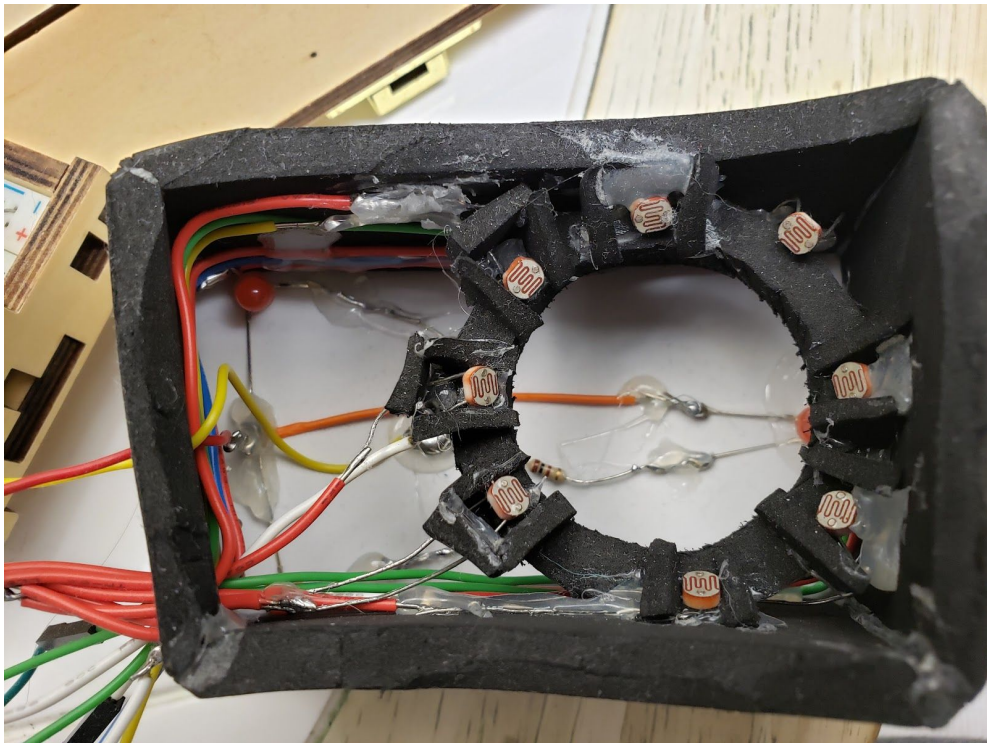


Figure 39: Prototype Sensor

8.2 Prototype 1 Converter System Test

At first the system didn't really have a proper converter board to speak of beside a special kind of digital to analog converter, as seen in the figure 7.2.1 below,

that the Raspberry mark three of the time of testing used to create the analog sensor. The system is a voltage divide of sorts in that the Pi is reading the voltage levels of a capacitor in time to a certain level in a time base table system that the Pi uses for started analog light sensors of LRDs. The system plugged each sensor individually from the array into the converter then to the Pi. The only thing shared between the array sensor was the power as a common source while also including their grounding technically as well. This system required too much of the Pi's timing which caused many errors for the system which promoted the call for a full proper converter to be built for the system in further iterations

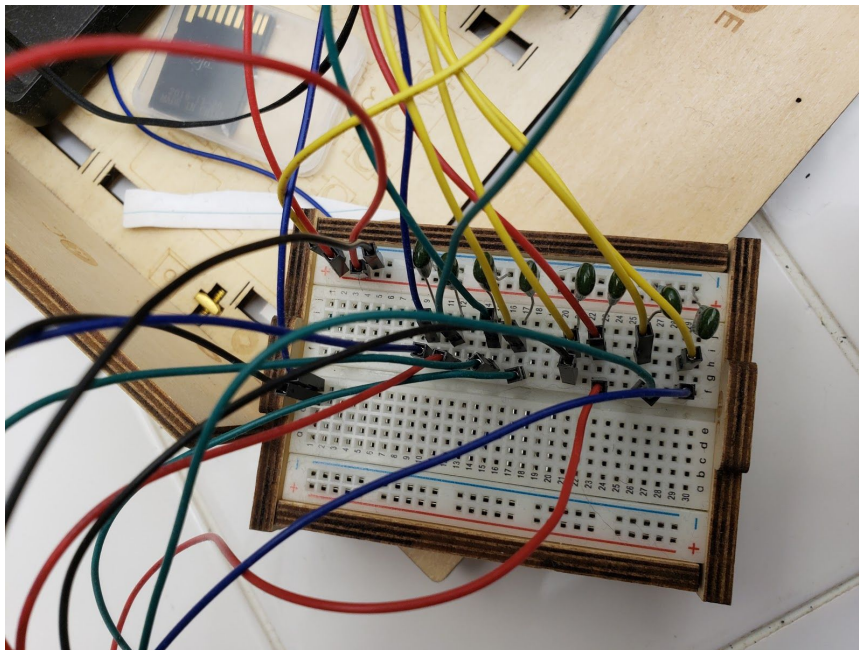


Figure 40: Prototype converter

8.3 Prototype 2 IR Concept Test Set Up

For this device we took the principles of Prototype two and started to build and test each component separately. The idea was to perfect the device step by step to alleviate the problem that occurred during the development of the first prototyped tracking device. Most of this procedure was focused on the converter board than that of the Main processor/ coding since that could be adjusted at any time. The converter board testing was broken down to the voltage divider and selector chip, then the analog to digital converter component process.

8.3.0 Prototype 2 (bread-board version) design

During the time of the projects second Prototype design concept process we came up the bread-board version lay-out as seen in the image41 below. This project prototype bread-board version lay-out is broken down into rough three main components, the input or the goggle/IR-array, the concept bread-board converter system, and the main processor unit.



Figure 41 Overview of Prototype 2

8.3.1 Prototype 2 goggle and IR array setup

For this section we procured and modified the goggle set UTOPIA360 for our main structure of the project. The goggles set was modified by having the internal lens for the 3D effect removed for ease of the program of the display. Then the internal vidivider for the 3D effect assembly was also removed with the lens for room for the IR sensor array. The construction of the IR array was built from the

material EVA foam due to the ease of use of the material and its quick construction time when working with said devices.

8.3.1 Prototype 2 Goggle and IR Array Bread Board Design

For our project we built the new concept prototypes main input system the IR array/ goggle set as seen in image42 and image43. In the first image one can see the internal side of the goggle set that the use will not see due to being slightly outside the line of sight of the user of the set. This image showed the internal lay-out of the array of IR receivers and their respective transmitter. How we set it up was the arrays in the goggles where first split between to sets of six for each side of left and right. However both sides had their common ground lines connected as well the transmitters grounding to one line to help lower the count of wires for the system since each receiver needs its own line. For the wiring of the goggles set's IR system there are two main bundles of wires that flow out from the goggle set's head strap points. In each bundle holds their set of IR receiver lines plus one of the common ground line, left side, or the IR transmitter power line, right side.



Figure 42 View Through Goggles Prototype 2

For what we see in image one sees what the user of the system will be looking at during the operation of the system. The set up of the arrays due block a little of

the outer peripheral view of the user due to the limitation of the construction. The IR array/ array holders were roughly built as temporary design that could scrapped at any time. The IR array also ran into a limitation of quantity of IR receiver at the time of construction,twelve instead of sixteen. The IR array is set up with the receivers inside the EVA foam structure ring while the IR transmitter is on the outside.



Figure 43

We plan on remaking the arrays during the next or final prototype of the system to improve the input of the system. At first we didn't want to cut heavily into the structure as seen in the current design. However we plain on to cut into the main eye hole plate of the goggle set to allow the IR receivers to be placed both closer and more out of the way of the user since the points already outside of the user line of sight by the design of the goggles. Another update to the design for the system is that when the PCB board is made, the board will be placed inside of the goggles with the IR array as to limit both travel distance for the input lines and all outside wire management. Chances are that we will have to completely just replace the plate with a new 3D-printed version of the plate with the edits made to it, to accommodate the IR array.

8.3.2 Prototype 2 Bread-Board Design

During development of the project's second prototypes main system concept design board we came up with the bread-board setup seen in the image below. To explain what the components are in the image they are broken into the voltage divider, the data selector chip, the buffer, and the data converter. For the whole system it was set up to run five volts and common ground between all components.

First in the image at the top left the component is the voltage divider circuit of the system. For this component section the system runs a five volt line that connects to twelve individual lines of a ten thousand ohm resistor for each IR sensor and a four seventy ohm resistor that leads to the IR transmitters. In between each of the ten thousand ohm resistor and IR receiver lines are the analog data tap lines for the system to lead off to the selector chip buffer. The reason we decided to use ten thousand ohms for the voltage divider for each IR receiver line was for a few simple reasons. One was that for most of the circuits PCB board components use this kind of resistor for the simple circuit requirements like loads and power needs of the selector chip and the MSP430. Also there is a dedicated ground lines that takes the common ground of the goggle set and the unused input lines of the selector lines of the selector.

Second in the image, at the top right component is a pre assembled selector chip or cd74HC4067 multiplexer. This will be replaced by surface mount version of the chip during the construction of the PCB board phase of the project. Basically this component runs the analog lines from each of the to the sixteen multiplex lines and feeds them to one output signal line. One or neither of the lines are selected by the selector chip through another set of five bit input data lines that come from the MSP430. The output signal line and input selector lines of the selector chip pass through a buffer bread-board due the fact of the limitation of the pre-cut wire taps use to connect between the MSP430 and selector lines

Third in the image, at the bottom component is the MSP430fr6989 launch board that is used in this project as the analog to digital conversion and communication line between the input and the main processor unit. For now during the bread-board phase this component supplies the power to the system however during the PCB board phase it will get its power from the main processor unit like other common usb device like key-boards. This component is in consent loop of taking a command from the main processor unit for the new data from the one sensor. Then it polls the selector chip for that sensor the main process unit wants. After receiving the analog data it converts it to digital and sends it out to the main process unit.

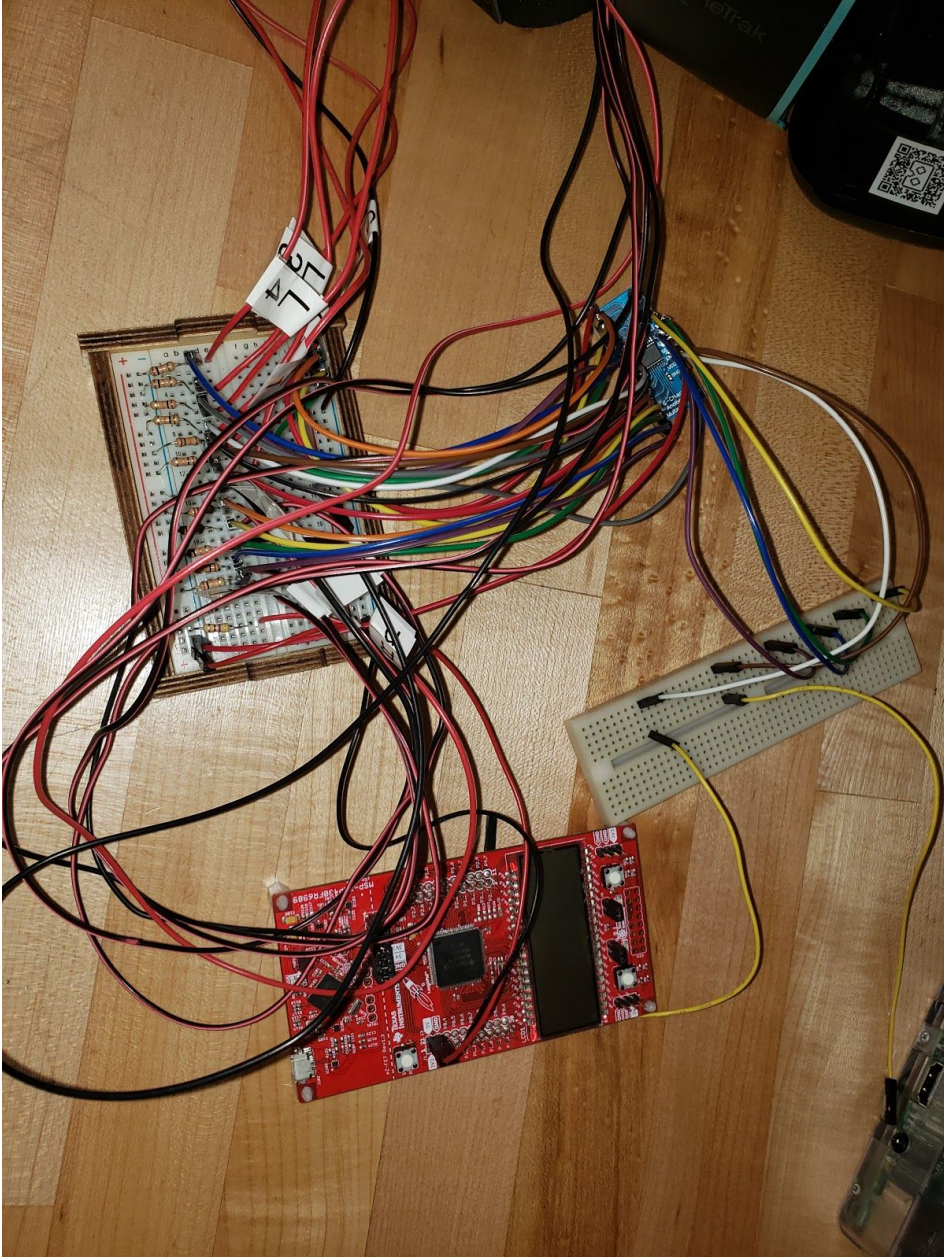


Figure 44

8.3.3 Prototype 2 Voltage Divider and Selector Chip Setup

For this component section of the project was assembled on a small breadboard assembly where it is divided into two sections the voltage divider and the selector chip. These two components are connected in this test section due to the fact the selector chip needs an input of some sort to function and it would just be as easy to just set up the voltage divider of the system. The voltage divider was set up for twelve IR sensors lines due to the fact of limited quantity of said IR sensors from Radioshack.

The selector chip feeds off the voltage divider by a bus line that taps each voltage divider line that runs off the IR sensor lines. The unused input lines of the system are then grounded for safety of the selector chips life. The output of the chip is given a load resistor that leads back to ground to stabilize the output voltage of the selector chip. During the test we ran lines from a Raspberry Pi 3 trigger lines to the selector chip's selector lines for the actual test of the selector chip. The Raspberry Pi was done to be able to test the selector chip more consistently to that of what the projects setup since the chip will communicate to a MSP430 in about the same way.

The premiss of the operation of this section is that the selector chip gets feed a five bit address line, four select lines and one block line for emergency if the system does want any sent at the time. Then the system wait about eighty nanoseconds, the rough safety margin for the selector chips line change time, for the system to change from each data to the next. This whole process repeats forever as long the system is active and calling for the next IR sensor

8.3.4 Prototype 2 Analog to Digital Converter Setup

To test the analog to digital converter set up for the microcontroller, we needed to connect an analog device that would output a wide range of signals to test the full resolution functionality on the chip. For our initial testing phases, we used an MSP430FR6989 chip with an analog joystick attached. This was to represent that we could both capture the data across multiple analog to digital conversion channels, and that we could accurately transmit that data to another device. While this is not the same chip we plan to use in the final product, the code for setting up the analog to digital converter will be very similar across all MSP430 microcontroller devices. The picture below shows the exact setup we used for this testing.

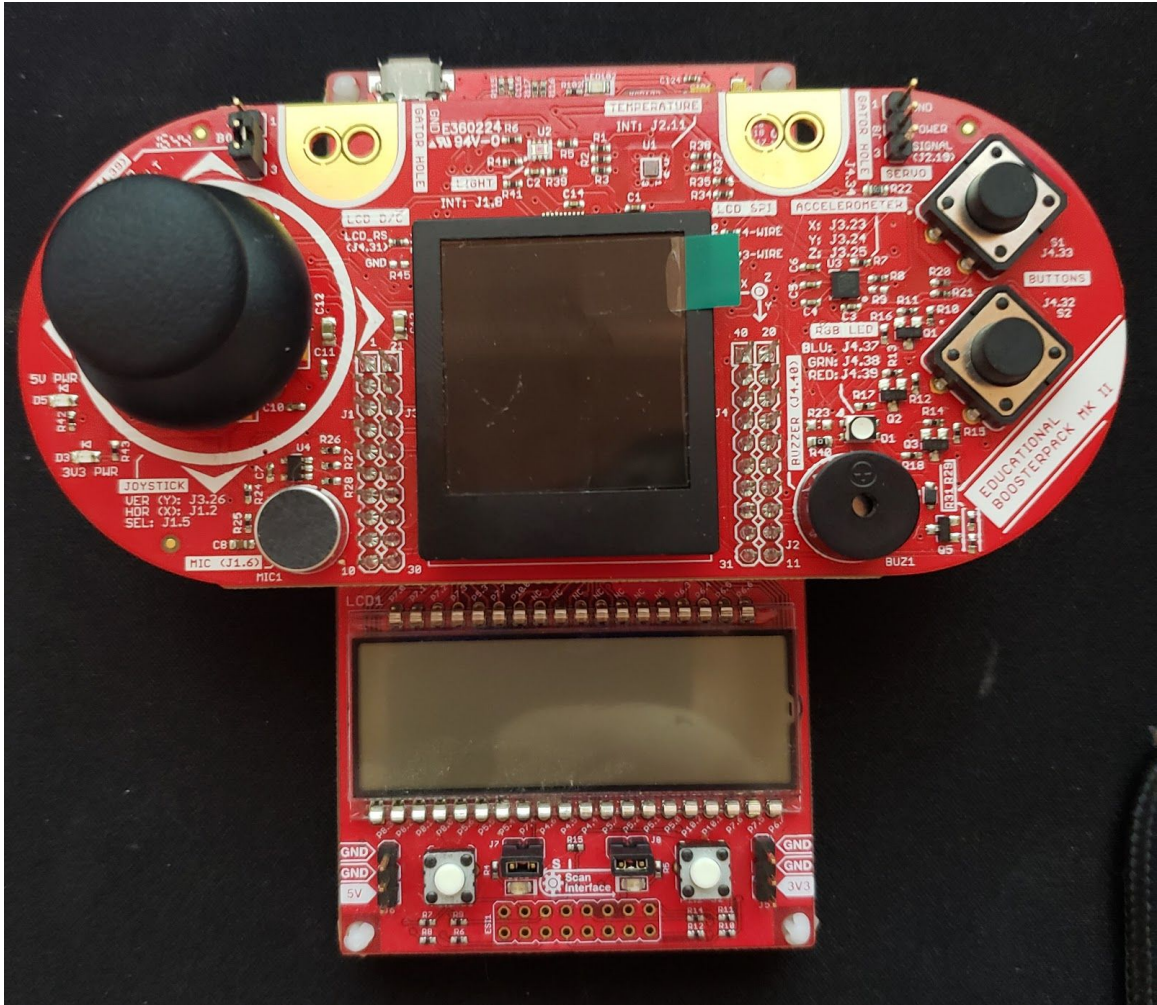


Figure 45

The signal was read by the MSP430 on 2 different channels, one for the X-axis signal, and one for the Y-axis signal. These were then converted into 12-bit numbers, as the analog to digital converter on the MSP430FR6989 uses 12-bit resolution. Our final product will likely be using 10-bit resolution. This number was then sent to a PC using the UART communication protocol. On the PC, the data was read from a COM port using the Tera Term program.

The next step is to convert the code used here into code that will be used on the chip we will use in our final design. This is a relatively simple process that requires minor modifications to the existing code. For example, the code needs to be converted from ADC12 to ADC10. This requires little more than just changing a few of the predefined names currently used in the initialization code to those provided by Texas Instruments through free libraries.

The initialization code comes in multiple parts. The first step in initializing the analog to digital converter is to divert the the pins to pick up analog signals. After that, the next step is to turn on the analog to digital converter through the ADC control register, but the programmer must be sure to turn off the bit for enabling conversion (ENC) while they are in the process of configuring the analog to digital converter. After that is done, the programmer must then determine the appropriate sample and hold time for the conversion. After that, the programmer must select a trigger, and then determine the resolution and data format (2's complement or unsigned).

8.4 Prototype 2 Main Coding

For this section we will go over the main processor coding operation for the system. This is where the main operations of the calculation for both eye track and output function of the device is made. The system code is forever locked in a loop of polling the converter board it's data and then sending back video feed to stimulate more new data for the system.

8.4.1 Prototype 2 MSP430 Code

For this section we will go over the concept of what the MSP 430 will be performing for this project's device. This system will be an intermediate component of the system to both convert the analog to digital data and command the selector chip to change the selected data line.

8.5 Prototype 2 PCB Board Design 1.0

During this section we will go over the design of the PCB board used in this project. This board is roughly divided into four sections of the sub components, the voltage divider, the selector chip, the MSP430, and the power/ grounding of the system. The PCB board will be two sided for the ease of avoiding the multiple bus lines to prevent any lines from being crossed.

The design of the voltage divider is basically a multi parallel linear voltage divider between the IR receiver sensor and a simple ten thousand ohm resistor that leads to ground. In between each line of IR receiver sensor and resistor there is a voltage line tap that pulls through to the other side as to be able to cross over the other voltage divider lines to the selector. The way this section gets power is that a common power line feeds up to each IR sensor and is carried down the system as previously stated.

Moving on to the selector chip and the placement of the chip. It is placed on the top side along with the divider and has the input data line divider two sets, one for the front side of the chip while the other for the back. This is due to the pre-design lay-out of the chip to cause the division of the data lines. The front data lines refer to the lines from the selector chip facing the divider while the back data lines are the ones that face opposite to the divider or face toward the MSP430. The front data line will pull the lines back to top of the board as to free up the path for the back data lines under the board to line up to their proper pin since front and back data lines share the same horizontal row. Feeding from the MSP430 will be the five bit data select line to the selector chips corresponding lines.

The way the component obtains power is that it comes from the same power line from the IR sensor with a simple set of resistors to create a constant flow for the chip. Finally the output of the chip will have a load resistor that pulls from the back to the ground parallel to the input of the MSP430's ADC pin.

8.5.1 Prototype 2 PCB Board Schematic

In this section we will discuss the design of our PCB board schematic. We used Autodesk EAGLE to aid in the design of our PCB board schematic as well as creating layouts for our future PCB. From the Autodesk website: "Autodesk EAGLE is an electronic design automation (EDA) software. Enabling printed circuit board (PCB) designers to seamlessly connect schematic diagrams, component placement, PCB routing, and comprehensive library content". We decided to use this software for our project because the software is free to use for the team, and some of the team members have previous experience in using the software for PCB schematic design, board design, and board fabrication. The parts for the board were imported using .blx files provided by the manufacturer. Those files were then converted to usable libraries in EAGLE using UltraLibrarian, a free to use website.

The first part of the schematic was the MSP430FR6989. This part will act as the primary driver for all on-board functionality. The chip serves multiple purposes on the board. The first purpose is to act as the analog to digital converter for the array of sensors. This is because the microcontroller has a built in analog to digital converter that will be handled by the embedded software. Once this information is processed by the microcontroller. The second functionality of the microcontroller is to control which channel the selector chip is currently feeding information through. The idea behind the selector chip is to use one device to handle all 16 channels that will need their data processed for the product's functionality. By using the 4 selector channels, this device will act like a digital multiplexor. By cycling through the different channels, we can quickly read all of the sensors in the array, while keeping the size of the board to a minimum. The final functionality of the microcontroller is to send the data back to a central

processing unit along provided communication lines. That is why there is a 10 pin connector in the middle of the diagram.

The second largest part of the diagram is the selector chip itself. The selector chip serves one of the most important functionalities for our project. It is the gate through which all of our relevant signals will pass. The best way to think about the selector to chip is to abstract it as a 16 to 1 multiplexor. The selector chip is mostly passive in our design, and its primary purpose is to quickly cycle through all of the available channels. Due to the fact that this part will be passive in nature and that there is a certain amount of delay associated with reading analog signals, it is doubtful that this part will be a bottleneck in our design.

The next part of our design is the voltage regulator, TPS62237DRYR. This part was found using the Texas Instruments WEBENCH Power Designer. We decided to use a balanced approach when it came to the DC to DC power conversion in our project, as we want both a good efficiency and a low cost, but don't want to sacrifice too much of one for the other. This part will account for the natural change in voltage that occurs in batteries as the battery loses energy during use of the product. The device is designed to take an input voltage range between 4 to 6 Volts, and provide a constant voltage output of 3.3 Volts, and output a current of up to 500 milliamps. This part is not as important as the previous two parts in our design, as it is possible that the priorities of the board design may change, and we may end up favoring certain aspects, such as the size on the board, more.

The next set of parts for the board will be the way we handle I/O. This aspect is important to have in the design, but the way of solving this problem is not as important. In order to keep the design simple and easy to work with, we have decided to use standard Molex adapters to handle the input and output lines. These are standard connectors that most engineers have experience in working with. These connections will handle all signals and power lines coming and going from the board. These include the 16 signal channels coming from our set of sensor arrays, as well as communication lines to the central processing unit, and input power from our power supply unit. These lines will also be how the programmer is able to embed the code onto the MSP430

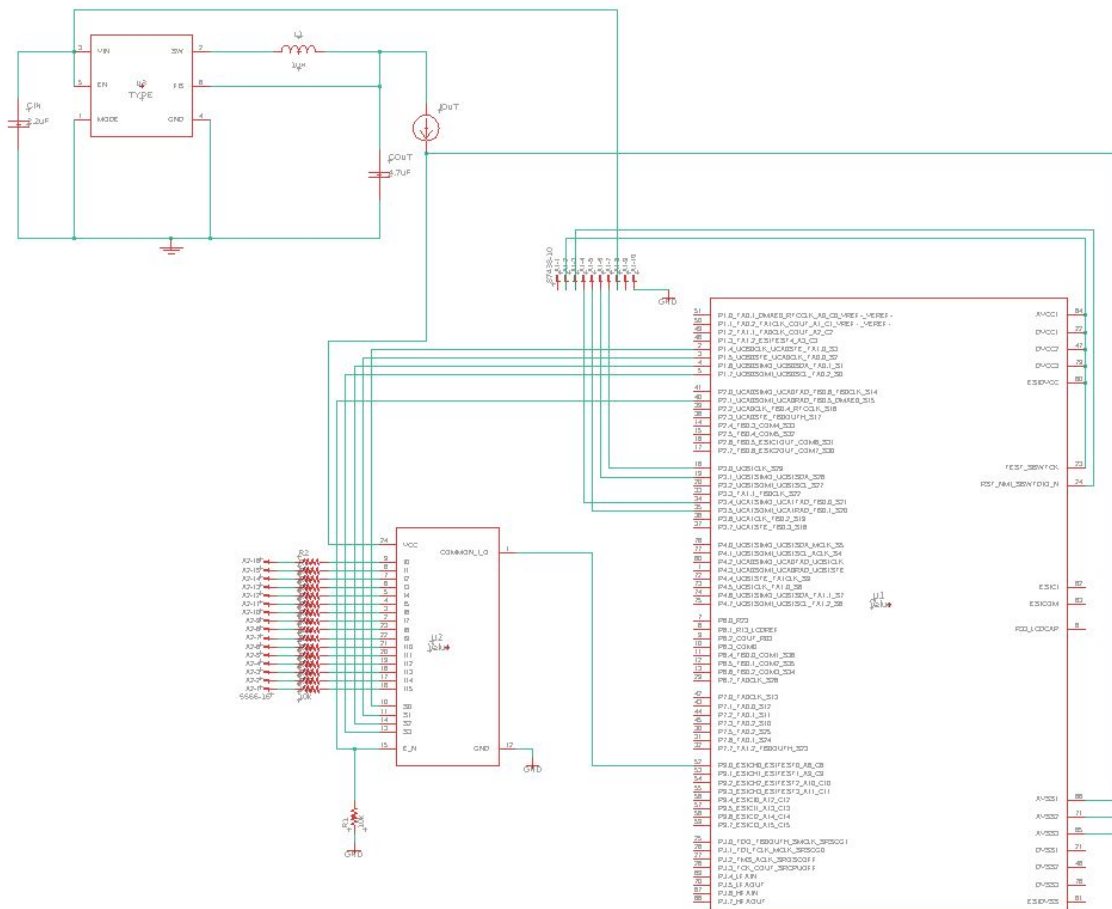


Figure 46: Preliminary PCB Schematic

8.6 Microcontroller Testing Procedures

To ensure that our microcontroller is suitable for our product's applications, there are a series of different functionalities that need to be tested. The first part is testing that the analog to digital converter is suitable for our needs on the project. We will be using a selector chip to choose which sensor we would like to read from, and the MSP430 provides 8 channels for conversion, which is convenient since there are 8 sensors per eye.

To test the analog to digital conversion, we must first create code to initialize the converter. For testing purposes, we used the MSP430FR6989, due to its availability in the laboratory that we used for embedded system testing, however the differences between the code required for the MSP430FR6989 and what is required for the MSP430G2553 are small and only require small adjustments, such as 10-bit vs 12-bit. The code is heavily commented to show exactly how the ADC has been initialized. In practice this code is used to set up the ADC to read from a sensor providing a constant signal voltage. This is then read into a value

which is then transferred to another system to be read into whatever program the user requires the data to be used for.

Another important section of the code is how we initialize the UART communication between the MSP430 and the main system on which the data is intended to be processed. While it has not been finalized, in our testing code, we have used UART to communicate with the primary data processing unit. This is simply due to the need for testing the other peripherals, as this form of communication does not affect the final result as long as the data can still be transmitted at a speed acceptable for the design of the product. Initializing communication protocols on the MSP430 is relatively simple, as the user's guide gives an in-depth explanation on how to configure the different parameters after the clock has been initialized.

Below is a simple block diagram showing how the code on the MSP430 will read the data from the sensor array and send it to the primary data processing unit.

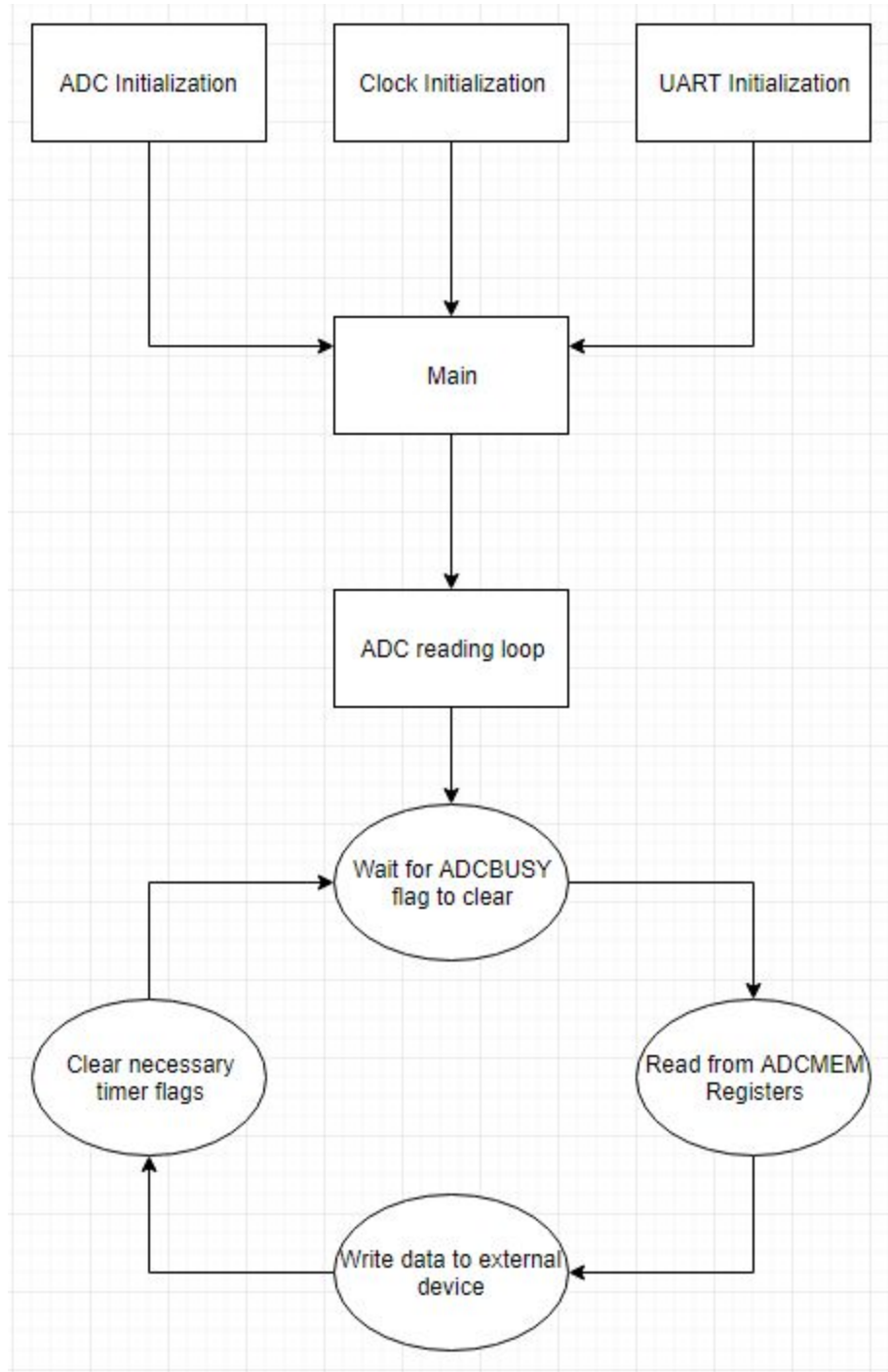


Figure 47: Microcontroller code diagram

As can be seen from the code diagram block above, the process occurring on the microcontroller is relatively simple. This is a basic overview of the code, and not a completely representative diagram. What isn't shown here is that another part

of the code will be sending signals to the selector chip about what sensor will be polled next. For each array of sensors, a voltage signal will be measured, and then a packet of signals will be sent to the primary processing unit.

9.0 Administrative Content

Here in Chapter 9 we will be discussing some of the administrative content of our project. This refers to the roles that we filled by each person in the project, the budget for our project, and the milestones we plan to reach given a deadline.

9.1 Budget

Here is a table of the estimated prices of our major components for the device that we are designing and building.

Estimated Budget

	Quantity	Cost
IR	16 or more	\$0.20 Each
Camera (Web Cam)	1	\$50.00
Display screen (24 inch HDMI LCD Screen	1	\$145.00
Development board (Raspberry Pi 4)	1	\$100.00
Robot arm (LewanSoul Robotic Arm)	1	\$200.00
Total		\$400.00

Table 18: Budget

The budget for the device we are designing is extremely important. The reason it is important is because the consumer, the person buying the product, will not want to spend a lot of money. For instance, someone will not buy an eye tracking device if it costs a billion dollars. Someone might buy one if it costs five hundred dollars. The price of the product depends on the how much it costs to build the product and later to market and distribute to consumers. As you can see above all the products will collectively cost about four hundred dollars for the physical building of the product. But later the cost of labor should be discussed as well.

Four hundred dollars is not a bad price. There are three reasons why we wanted the product not to be so expensive.

1. The cheaper the product the cheaper we can sell it.
2. Also the cheaper the product is to produce hopefully there will be larger margin for making profit with the consumers.
3. The cheaper the product the less we have to spend from our own money. We are college students so we don't have a lot of money to spend on the product.

The important thing we had to keep in mind is to not make the device too cheap. If we make it too cheap it won't be good at tracking the eyeball. For instance, If we buy the cheapest goggle headset then it won't be as comfortable for the users due to differences in individuals wearing the device. If we buy the cheapest selector chip out there than it will switch between the IR sensors slowly and it won't work fast enough to track the eyeball. Everything has a price. Quality has a price. We want the product to work well, that has a price. So even though we want the device to be cheap and good quality, we have to find the balance. Because we could make it super cheap but with terrible quality or expensive and good quality.

After much research and comparing we found that the price shown in the table above in a row, Total, which is four hundred dollars, is a good balance between it not being too expensive and it not being bad quality.

9.2 Milestones for the Project

Here is a table that projects our milestone goals for the Senior Design I and Senior Design II. The first column shows the goal that is to be accomplished and the second column shows the date of when the goal should be accomplished

Project Milestones

Divide and Conquer	9/20/19
Research and Writing	10/11/19
Table of Contents	10/11/19
Write	10/18/19
60 page Draft Senior Design I Documentation	10/21/19
100 page Submission	11/15/19
Have all Major Components Ordered	12/2/19
Final Document Completed	12/2/19
Build Arm and Camera	1/17/20
Build Display	1/17/20
Design LDR Array	1/24/20
Make Converter System	1/24/20
Build Power Unit	2/7/20
Build MCU	2/14/20
Test	2/21/20
Redesign and Finalize	3/27/20
Presentation	4/17/20

Table 19: Project Milestones

This section will be walking through the above milestones and explaining why they are set that way. Up until December 2nd the goals are the assignments and their due dates. For example the divide and conquer assignment was due on September 20th and therefore that was the date we set to have it done by.

The second semester, Senior Design II, starts in the year January 2020. We all have different assignments. For instance, someone is building the power unit and another individual is building the MCU and programming it. Each part of the device will take a different amount of time. For instance the MCU will probably take the longest, so we gave the person working on that part more time to finish it. Where as putting together the Arm and the Camera won't take that long so the person working on that should finish it up really quick in comparison to the other items.

We also want the whole device to be built by the end of February 2020 so that we have plenty of time to test it. What will most likely happen is we will test it and find some problems depending on the testing process. At that time we will work to overcome these issues and discover better ways to have the device work and perform. The next two months will be dedicated to improving the device and fixing any bugs that we find in it through testing it through several different situations and environments.

April 17th of 2020 is highlighted because that is an important date. That date is the date we will be presenting are findings and the documentation that supports are findings. The product needs to be built by then and operational. If it does not work we will not be able to graduate. So no matter what at that time we need to have the device ready to work by that date and that is why it is important to the group.

10.0 Project Summary and Conclusion

This project could change the world. It was a game changer when touch screens, smart phones, tablets, and computers came out. You could control a device by just the touch of your fingers. With our eye tracking device you do not have to have devices all these devices such as touch screens, smart phones, tablets, or computers. You could control the device without ever lifting your finger. All you would need to do is simply move your eyes.

The way this is done is not a simple process. The plan to accomplish this state of the art device is through tracking the reflection of light off of your eyes. To start there is the headset that must be put on that covers the eyes. Once the headset is on there will be tiny IR emitters that will shine light on your eyes. Then the IR sensors will detect the light shining off of your eye and skin surrounding them. The fact that makes this all work is that the human's skin, whites of their eyes, and their pupils all reflect a different amount of light. The pupil emits the least amount of light

Since the pupil emits the least amount of light we use that fact to detect where the user is looking, i.e. where the pupil is pointing. To do this we send the information the light detectors are collecting and send it to our MCU, which then determines where the least amount of light is being detected and that is where the eye will be looking at.

For our project we will use that information that the MCU now has and have it control a camera. However, it could one day be used for many more circumstances or conditions as envisioned in the first paragraph.

11.0 Appendices

This section is used to show the bibliography

11.1 Bibliography

[1] Wifi on Steroids: 802.11AC and 802.11 AD Published by IEEE

[2] IEEE Standard for Information Technology - Telecommunications and information exchange between Local and metropolitan area networks - Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Published by IEEE

[3] IEEE 802.11ax: High-Efficiency WLANs Published by IEEE

[4] The UNIX time-sharing system: The C programming language Published by Nokia Bell Labs

[5] International Standard Programming languages - C Published by ISO

[6] PEP 8 -- Style Guide for Python Code

<https://www.python.org/dev/peps/pep-0008/>

Author: Guido Van Rossum, Barry Warsaw, Nick Coghlan

[7] What are Standards? Why are they Important?

<https://beyondstandards.ieee.org/general-news/what-are-standards-why-are-they-important/>

[8] IEEE Standard Adoption fo ISO/IEC 90003:2014, Software Engineering Guidelines for the Application of ISO 9001:2008 - Published by IEEE

[9] Measurement Based Study of the Performance of IEEE 802.11ac in an indoor environment - Published by IEEE

[10] Performance Comparison of Wi-Fi IEEE 802.11ac and Wi-Fi IEEE 802.11n - Published by IEEE

[11] How much can we reuse? An empirical analysis of the performance benefits achieved by spatial-reuse of IEEE 802.11ax - Published by IEEE

[12] Eye Safety for Proximity Sensing Using Infrared Light-Emitting Diodes - Published by Renesas

[13] Understanding Radiation Safety of High Intensity Light Emitting Diodes - Published by IEEE

[14] IEC-62471: Photobiological Safety of Lamps and Lamp Systems - Published by IEC

[15] Code Composer Studio (CCS) Integrated Development Environment (IDE)
- <http://www.ti.com/tool/CCSTUDIO> - Published by Texas Instruments

[16] Raspberry Pi 4 Documentation -
<https://www.raspberrypi.org/documentation/>

- [17] INTERNATIONAL STANDARD ISO/IEC/IEEE 29119-4 - Software and Systems Engineering Software Testing - Part 4: Test Techniques - Published by IEEE
- [18] Duchowski, Andrew T. "Eye tracking methodology." Theory and practice 328.614 (2007): 2-3.
- [19] Texas Instruments "MSP430G2x53, MSP430G2x13 Mixed Signal Microcontroller datasheet (Rev. J)" SLAS735J, April 2011 (Revised May 2013)
- [20] IEEE Standard for Rechargeable Batteries for Portable Computing - Published by IEEE
- [21] Texas Instruments "MSP430x2xx Family User's Guide (Rev. J)" SLAU144J, December 2004 (Revised July 2013)
- [22] 802.15.1 IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Published by IEEE (Standards for Bluetooth)
- [23] Bluetooth 5: A Concrete Step Forward toward the IoT - Published by IEEE
- [24] IEEE Standard for Low-Rate Wireless Networks - Published by IEEE (Standard for Zigbee)
- [25] Zigbee Alliance
<https://zigbee.org/zigbee-for-developers/zigbee-3-0/>
- [26] Performance Evaluation of Wireless Sensor Networks Based on Zigbee Technology in Smart Home - Published by IEEE
- [27] Security Analysis of Zigbee -
- [28] ZigBee Based Wireless Sensor Networks and Performance Analysis in Various Environments - Published by IEEE
- [29] Testing of Communication Range in Zigbee Technology - Published by IEEE
- [30] Everlight Technical Data Sheet 5mm Infrared LED, T-1 3/4 - Published by Everlight