SCRATCH

Shot Consultation and Refinement Applied Through Computer Hardware

Group 17

Luke Ambray	*Computer Engineering*
Mark Nelson	*Electrical Engineering*
Goran Lalich	*Electrical Engineering*
Mena Mishriky	*Electrical Engineering*

Introduction

- SCRATCH is a billiards training tool that can be used to improve the performance and consistency of players
- This will be done via training drills with accurate data measurement and feedback
- This data will be measured by a camera and sensors placed both on the user as well as the cue stick
- In addition to this training system, the true accuracy and user feedback system will be tested to its limits with the goal of enabling visually impaired individuals to play pool.
    - To accomplish this final point, the SCRATCH team will work in conjunction with VISION (Group 14)

VISION's Logo

Motivation

- Create a system that improves performance of billiards players at all skill levels ranging from beginners to experts.
- Allows everyone to enjoy the sport more
- Creates an environment where visually impaired individuals can enjoy the game in social environments

Goals and Objectives

- HUD Goals
  - Provide basic visual pre-shot information to inform the user where to aim and shot strength
  - Provide auditory instructions to impaired individual for aiming of cue stick
  - Use of camera on wearable HUD will enable cue ball impact point feedback
  - Display will be used to provide post-shot feedback for strength and impact point

- Cue Stick Goals
  - System can collect and transmit data regarding the orientation and force applied to the cue stick
  - System can collect and transmit button input data to control system

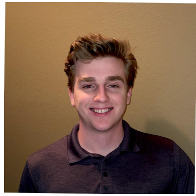
Goals and Objectives (Continued)

- Glove Goals
    - Read angle of user relative to tables edge to align shot
    - Locate ball position and direct user to correct distance for optimal shot
    - Transmit angle and position to control system

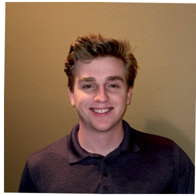
- Central Control Unit (CCU) Goals
    - System receives and analyzes cue stick motion data and camera shot impact placement
    - Once analysis is complete, feedback is sent to the HUD
    - System controls UI via button inputs on cue stick, and gives information to HUD display
    - System receives and uses data from VISION team to direct the impaired individual

Specifications

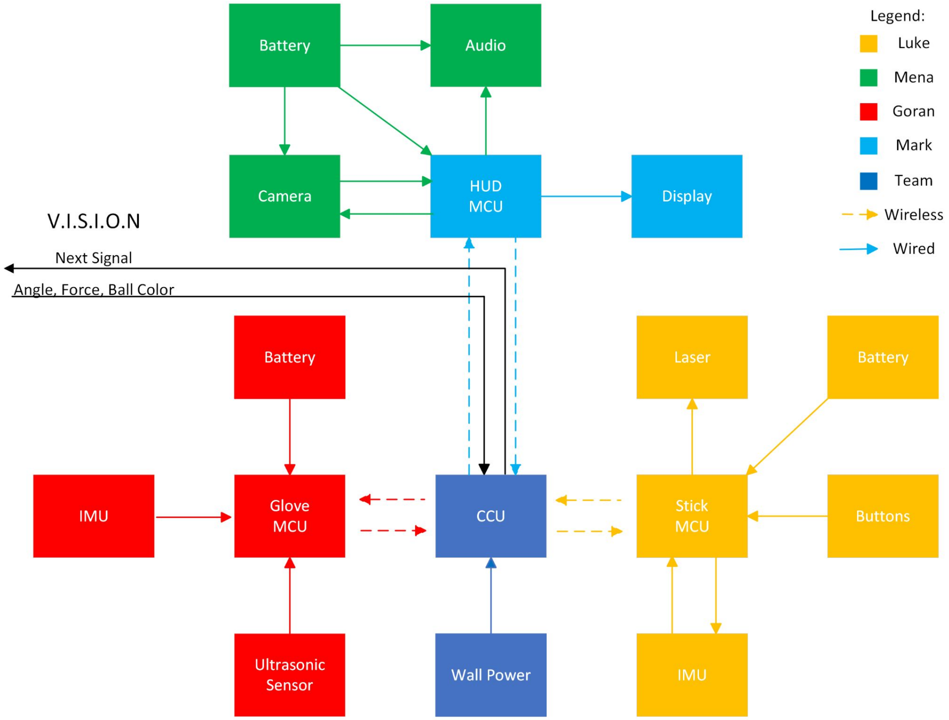
HUD should be lightweight	< 1500 grams
HUD should be able to provide post-shot feedback after a shot attempt is completed	< 8 seconds
Sampling user aim and providing audio feedback should be done on predetermined intervals.	< 2 seconds
Cue stick sensors should be able to determine the orientation of the cue stick within an acceptable margin of error	< 5 degrees
Cue stick should be able to determine the speed of the cue stick within an acceptable margin of error	< 1 meter per second
Electronics should not reach uncomfortable temperatures	< 100 degrees Fahrenheit
Central control unit should be able to respond to button inputs from the cue stick with minimal latency	< 500 milliseconds

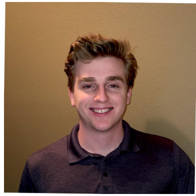


System Descriptions



Hardware Block Diagram

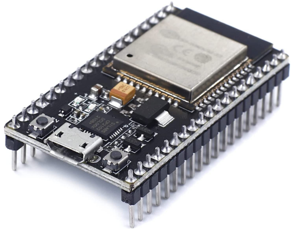




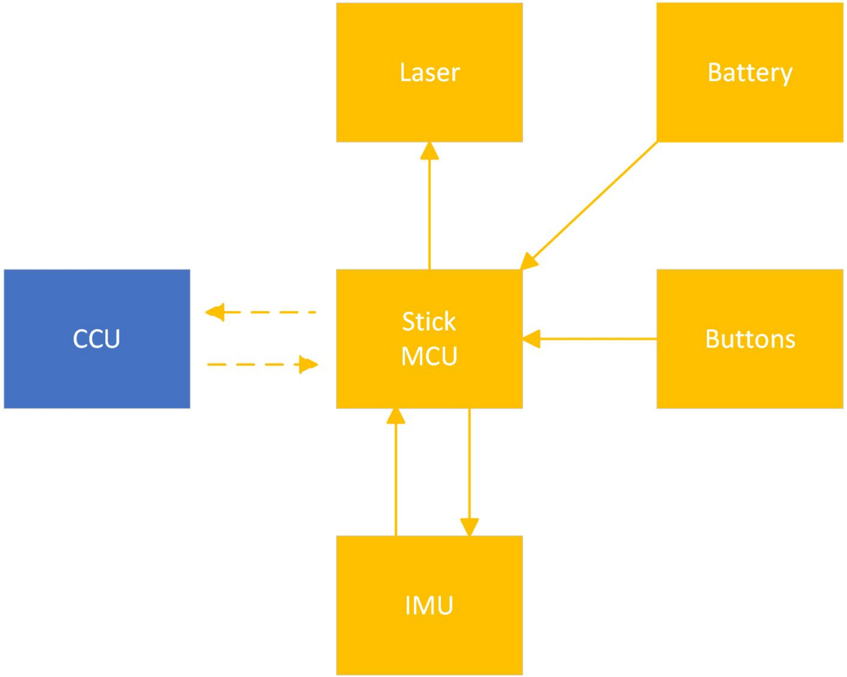
Microcontroller

- Three microcontrollers needed:
  - Independent module used in cue stick PCB
  - Development boards used in HUD and glove
- Selection criteria:
  - Integrated Bluetooth
  - Low power consumption
  - Easy to program
- Selection:
  - Development board: Teyleten Robot ESP-WROOM-32 Development Board (left)
  - Module: ESP-WROOM-32 (right)

Microcontroller	ESP-WROOM-32
Processor	Dual-core Xtensa/Tensilica
Wired Connectivity	SPI, I2C, I2S, UART
Wireless Connectivity	802.11 b/g/n, Bluetooth 4.2, BLE
Max Power Consumption	1.65 W



Cue Stick Block Diagram



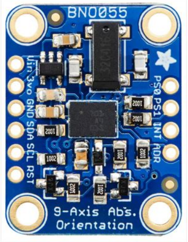
Cue Stick Description

- The cue stick is the primary means by which the user will interact with SCRATCH
- Inertial measurement unit (IMU) is used to determine the orientation and speed of the cue stick during a shot attempt
- Buttons are used to navigate the program menus displayed on the HUD
- Mounted laser will aid the HUD's camera in the determination of the point of contact

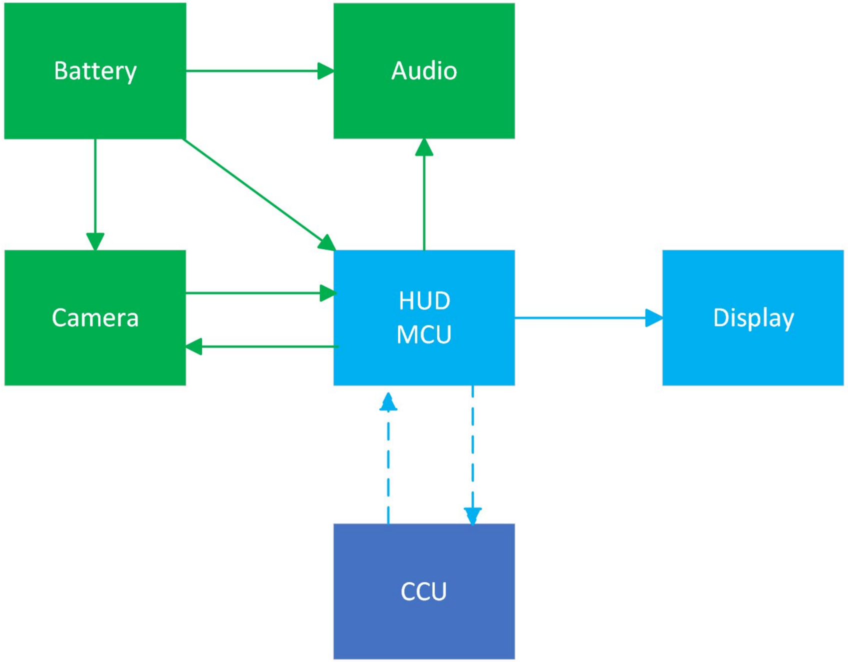
Inertial Measurement Unit (IMU)

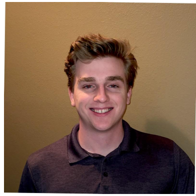
- Purpose: determine the orientation and speed of the cue stick during a shot attempt
- Also used in glove to determine shot angle
- Selection criteria:
  - Small form factor
  - Compatible with ESP32
  - Low power consumption
- Selection:
  - Sensor: Bosch BNO055 (left)
  - Breakout board for testing: Adafruit BNO055 Absolute Orientation Sensor (right)

Board	BNO055
Accelerometer Range	±2 g, ±4 g, ±8 g, ±16 g
Gyroscope Range	±250, ±500, ±1000, ±2000°/sec
Connectivity	I2C/SPI
Power Consumption	36.9 mW



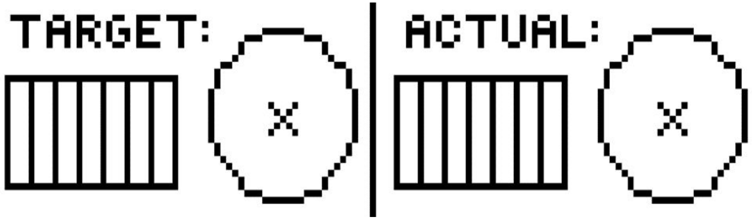
HUD/Camera/Speaker Block Diagram

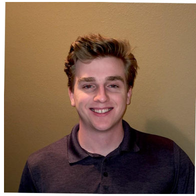




HUD Description

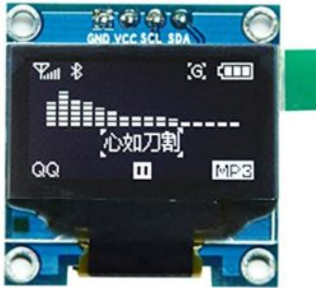
- The wearable HUD consists of three subsystems all of which are worn on the user in a similar fashion to glasses
- The first system is the actual HUD, this is accomplished using two mirrors and a display to place the image clearly in front of the user's eye for information access at all times leading up to the shot
- The information consists of pre-shot directions, post-shot feedback, and UI control



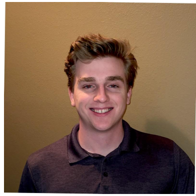


Display

- Purpose: Display pre and post-shot information concisely and clearly
- Selection Criteria:
  - Small size and low weight to minimize negative ergonomics
  - 3.3V Input
  - I2C Communication Protocol
  - Ease of programming
- Selection:
  - HiLetGo 128x64 OLED LCD I2C Display



Display	HiLetGo LCD Display
Resolution	128 x 64
Voltage	3.0 - 5.0
Cost	$7.29
Communication Protocol	I2C

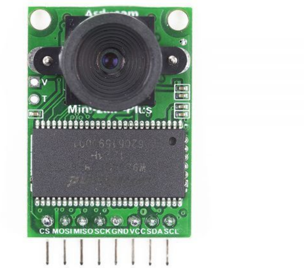


Speaker Description

- The purpose of the speaker system is to provide pre-shot guidance for the visually impaired user
- Guidance comprises of auditory commands that instructs the user to move their body
- The commands are stored on an SD card and sent to the DAC via I2S
- Physically the system is composed of:
    - The I2S port on the ESP32
    - The MAX98357a DAC for conversion of I2S data to analog
    - A simple 3W speaker for audio output
    - An SD card to store the auditory commands
- Command selection is done depending on input from the CCU

Camera Description

- The purpose of the camera subsystem is to provide post-shot feedback to the non-impaired user regarding point of impact between the cue stick and the cue ball
- This is done through capturing an image right before the user makes contact with the ball, then performing minimal computer vision to locate the position of the laser in the image (laser is mounted on the stick)
- Due to the low power requirement of the HUD, the image is sent to the CCU for computer vision tasks
- The Camera subsystem consists of:
  - 2MP ArduCAM Mini camera
  - EP32 MCU
  - Raspberry Pi (CCU)

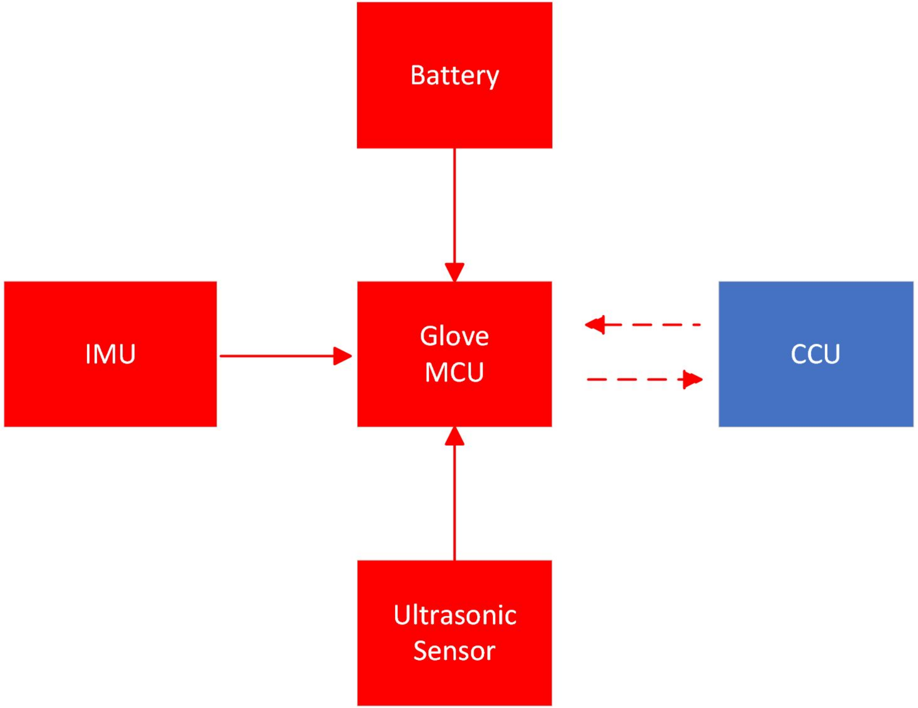


Camera

- Purpose: take a picture used for point of impact determination
- Selection Criteria:
  - Small size to fit in HUD
  - Low power
  - Low cost
  - Ease of programming
  - Customizable resolution for accuracy/latency trade off
- Selection:
  - ArduCAM Mini 2MP Camera

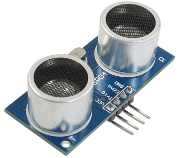
Camera	ArduCAM 2MP
Size	34x24mm
Power	0.35W
Cost	$26
Interface	SPI and I2C

Glove Block Diagram



Glove Description

- The glove is a key component for guiding visually impaired users to the correct location to take a shot
- The IMU is used to determine the angle of the user with respect to the side of the pool table
- The ultrasonic sensor is used to determine the distance from the glove to the cue ball
- The users current angle and distance from the ball will be sent to the CCU to be compared with the desired values
- Speaker system will tell user how to adjust in order to match desired values of angle and distance

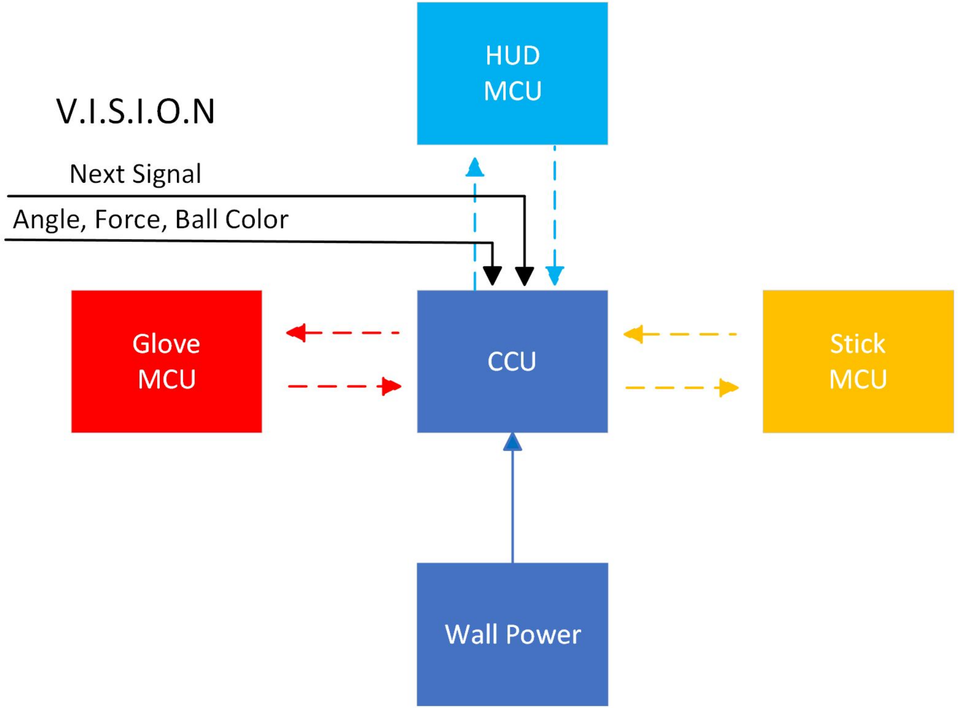


Ultrasonic Sensor

- Purpose: determine the distance of visually impaired users hand from the ball
- Selection Criteria:
  - Powered by 3.3V
  - Small Size
  - Low Power Consumption
  - Able to measure distance up to 90cm
- Selection:
  - RCWL-1601

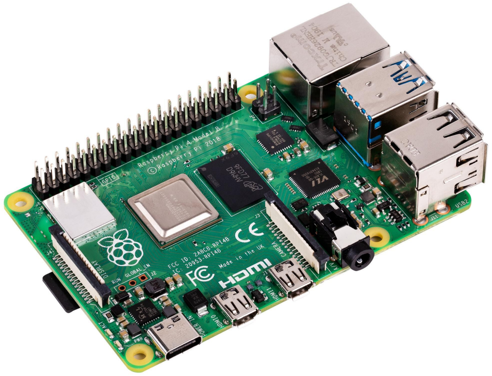
Sensor	RCWL-1601
Distance	2cm to 450cm
Voltage	3V-5.5V
Dimensions	18mm x 40mm
Current Draw	2.2mA

Central Control Unit (CCU) Block Diagram



Central Control Unit (CCU) Description

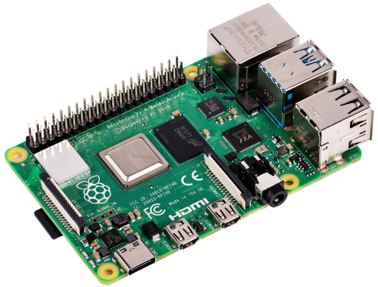
- The central control unit (CCU) connects to all the other subsystems, as well as the VISION team's hardware
- Responsible for interpreting shot data from cue stick, determining what to display on HUD, and using image data to determine point of contact
- For this system, we will be using a Raspberry Pi 4
- As you can imagine, this module has the major state machines of this project



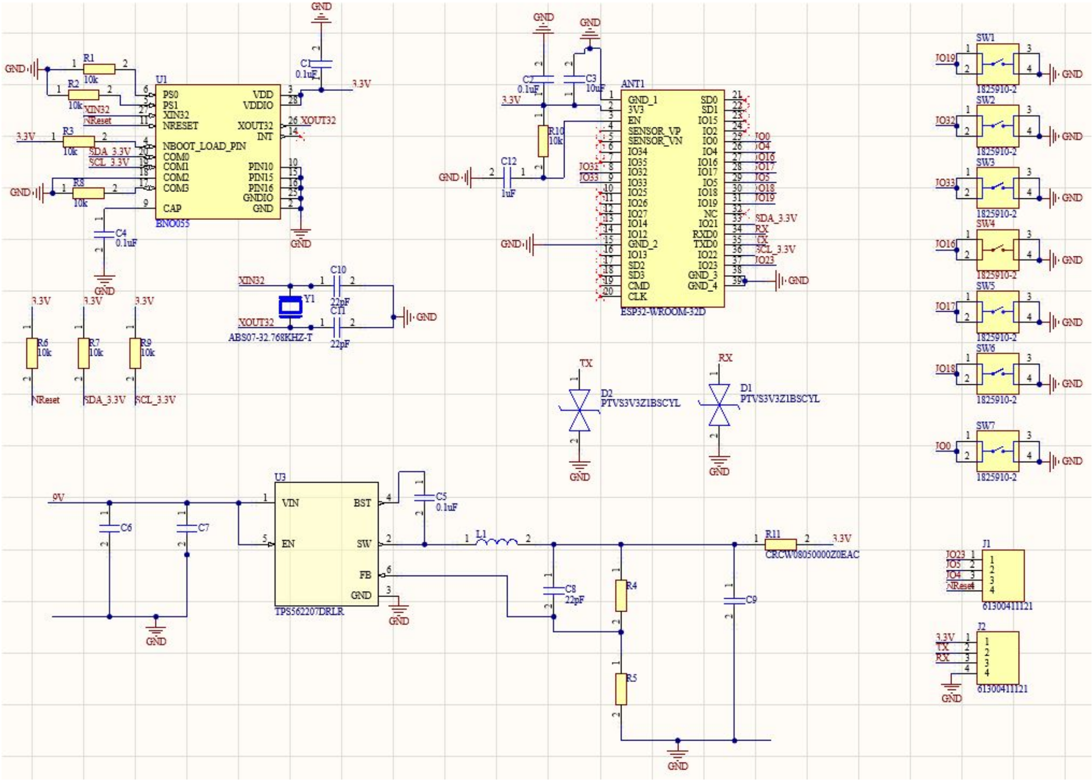
Microprocessor

- Purpose:
  - Interface with various subsystems
  - Perform computer vision
- Selection Criteria
  - Easy to program
  - Integrated wireless communication
  - Processing power to communicate with systems concurrently
  - Ability to perform computer vision
- Selection: Raspberry Pi 4 Model B 8GB

Board	Raspberry Pi 4 Model B
Processor	Quad-core Broadcom BCM2711 (1.5 GHz)
RAM	8 GB LPDDR4 3200 MHz
Wired Connectivity	USB, SPI, I2C, I2S, UART, Ethernet
Wireless Connectivity	802.11ac. Bluetooth 5.0, BLE

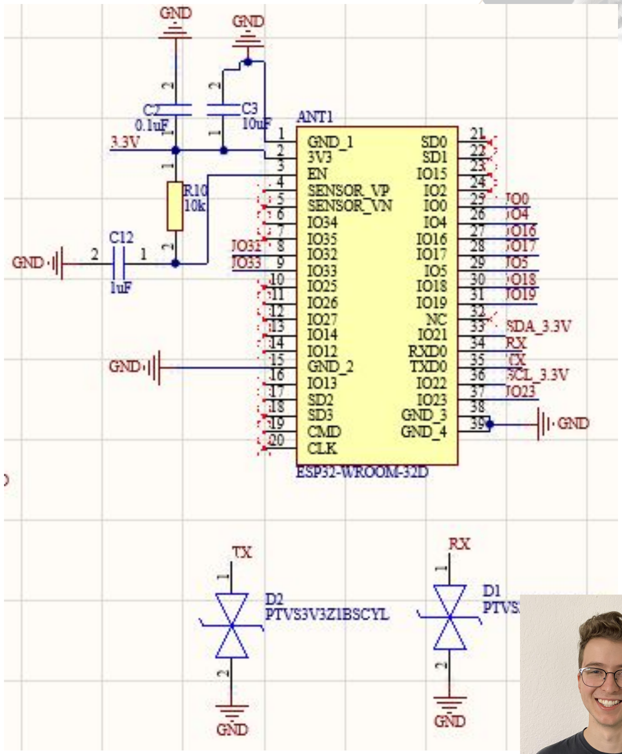


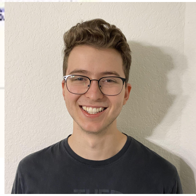
PCB Schematic



Microcontroller Schematic

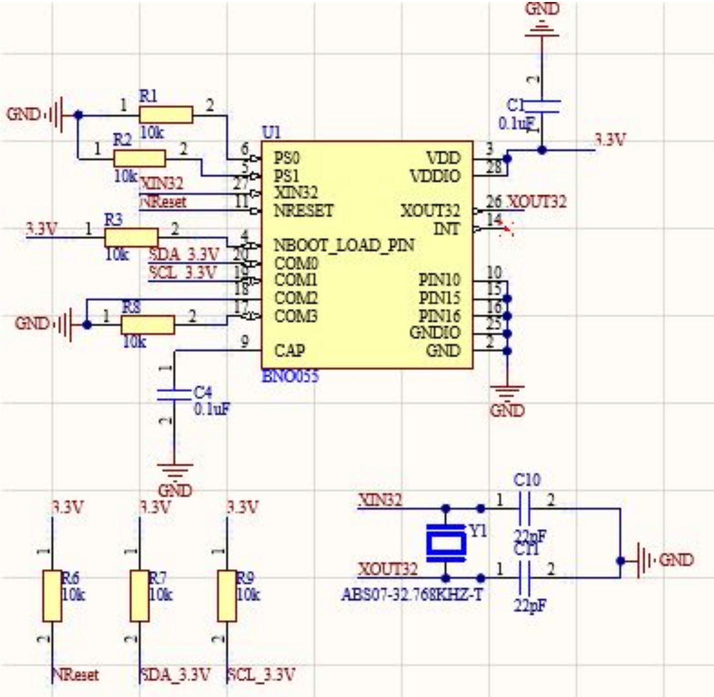
- On the PCB we are using an ESP32 as the main control system
- The microcontroller will control the connection to both the IMU and the push buttons
- TVS diodes are used for ESD protection on the Transmit and Receive lines that we will use to program the device
- This device takes the 3.3V from the power regulator circuit





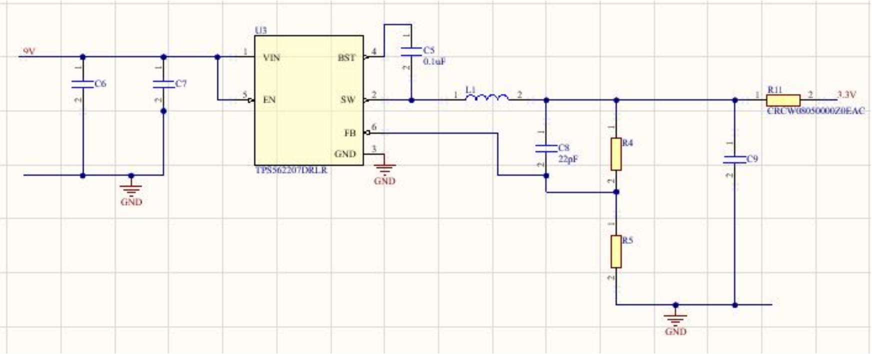
IMU Schematic

- We are using the BNO055 Inertial Measurement Unit on our PCB
- This device communicates with the ESP32 via I2C, These lines are pulled up to 3.3V to communicate with the MCU
- The IMU has an internal clock that it can run off of but an external crystal provides much better timing for this system



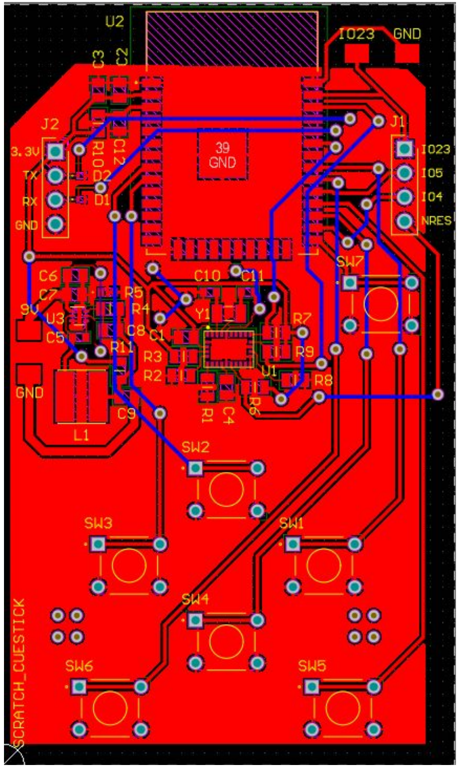
Power System Schematic

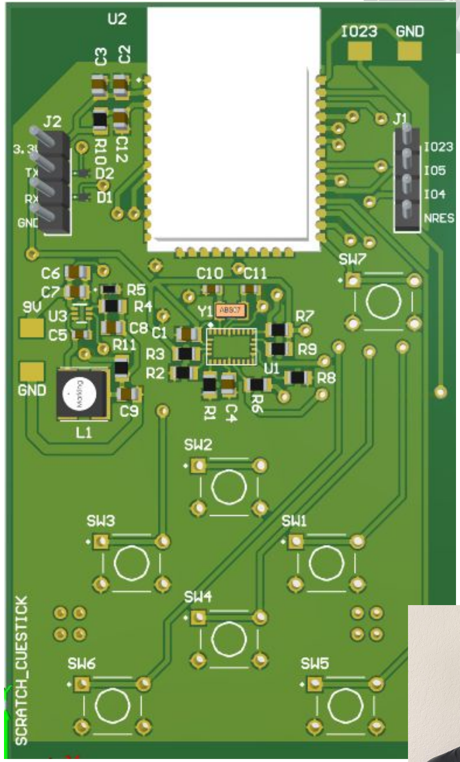
- This power system will convert our input of a 9V battery to 3.3V which will be used as the input to both our MCU and IMU
- We are using the TPS562207DRLR which is a Buck converter that combined with the external circuitry makes a switching regulator
- This system follows the layout recommendations given in the datasheet for the component



PCB Layout

- This is a two layer board designed in Altium that follows basic PCB guidelines
- Layout recommendations followed for devices
- Dimensions of the board are limited since this will be mounted on the rear of a cue stick
- Push buttons will be set up in a configuration to navigate the practice tool

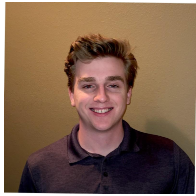




Battery

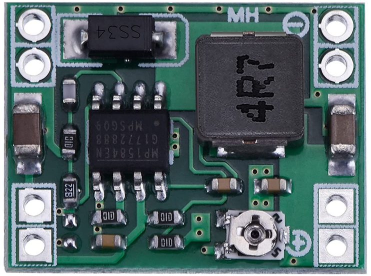
- Purpose: Supply power to main PCB and all peripherals
- Selection Criteria:
  - Rechargable
  - >= 5V output
  - Small Size and Weight
  - Easily replaceable
- Selection:
  - Standard 9V Rechargeable Battery

Dimensions	48mm x 25mm x 15mm
Weight	45 g
Output Voltage	9V
Cost	$1.00-$2.00

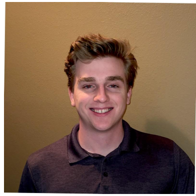


Buck Converter

- Purpose: For peripherals not powered by PCB, convert power from 9V input to required output
- Selection Criteria:
  - 9V in input range
  - Adjustable to both 3.3V and 5V output
  - Small size and weight
  - Low Cost
- Selection:
  - eBoot Mini MP1584EN DC-DC Buck Converter

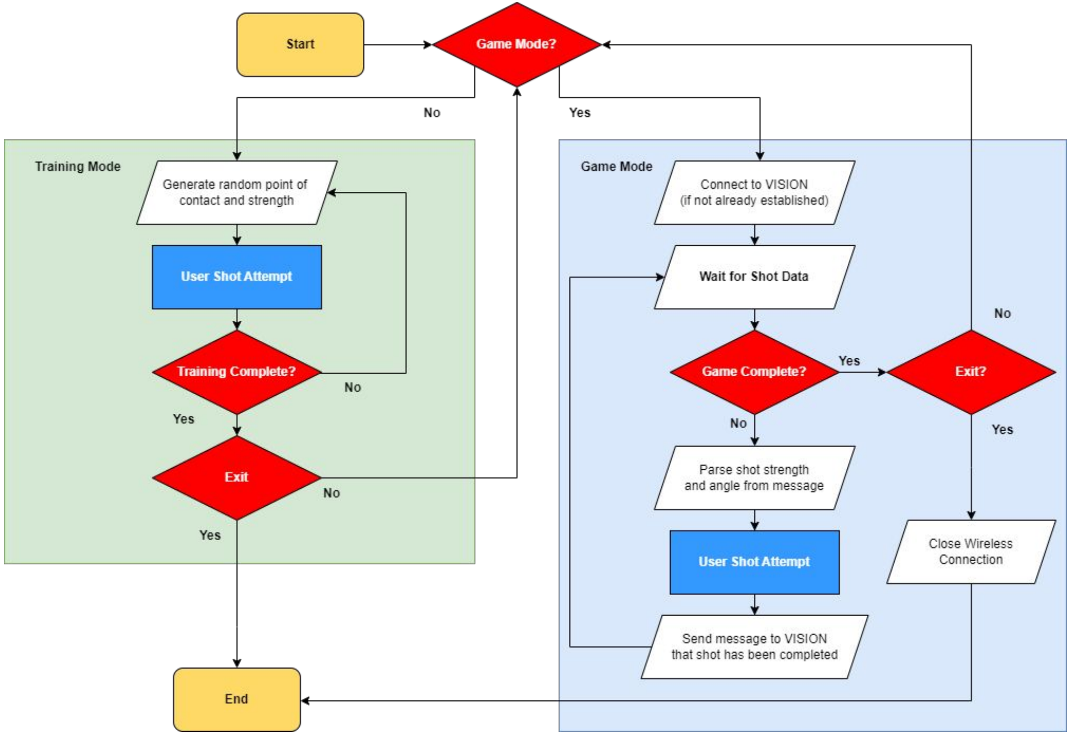


Dimensions	22mm x 17mm
Weight	45 g
Input Voltage	4.5V - 28V
Output Voltage	0.8V - 204V
Maximum Current Output	3A



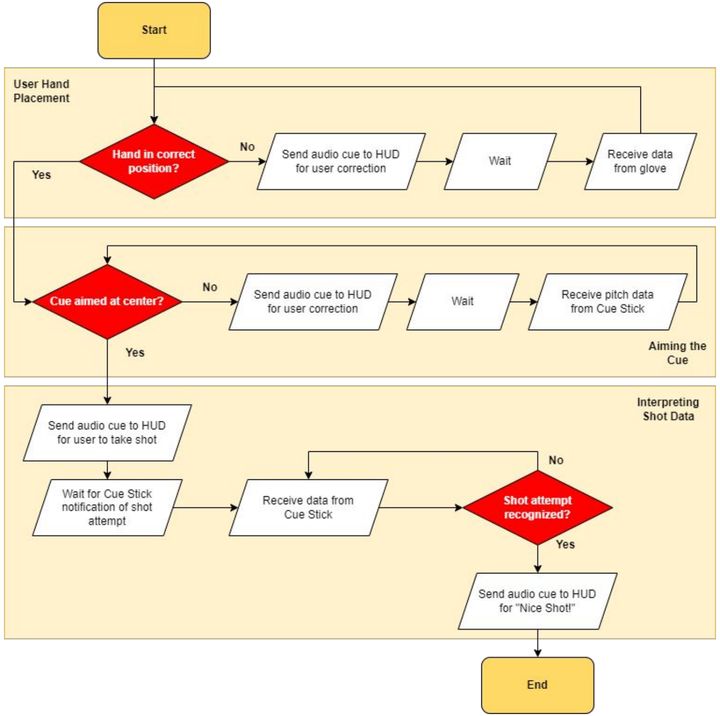
Software/Operation

Normal Operation



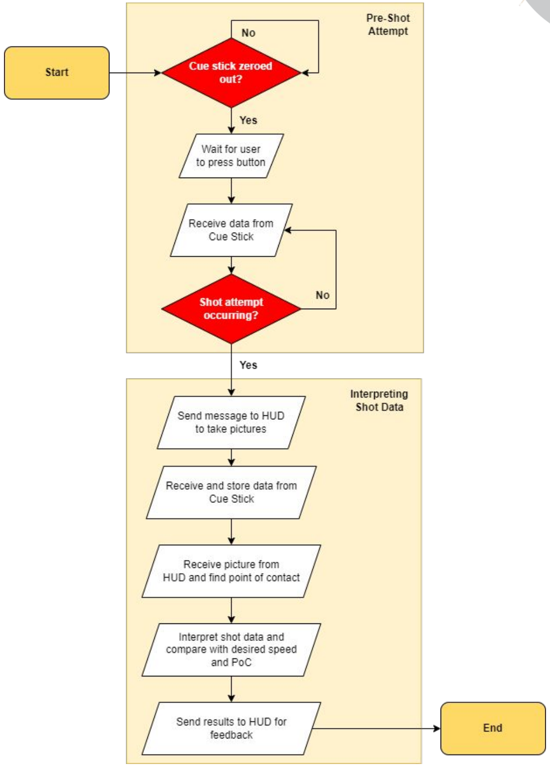
User Shot Attempt (Impaired)

- Peripherals used:
  - Glove
  - Cue Stick
  - HUD speaker
- Glove Procedure:
  - Perform sweep of the glove to match desired angle
  - Measure distance until it meets the desired distance from the cue ball
- Stick Procedure:
  - Wait for cue stick to be level with table
- HUD Procedure
  - On CCU notifications, play appropriate audio files to guide the impaired user's body motion



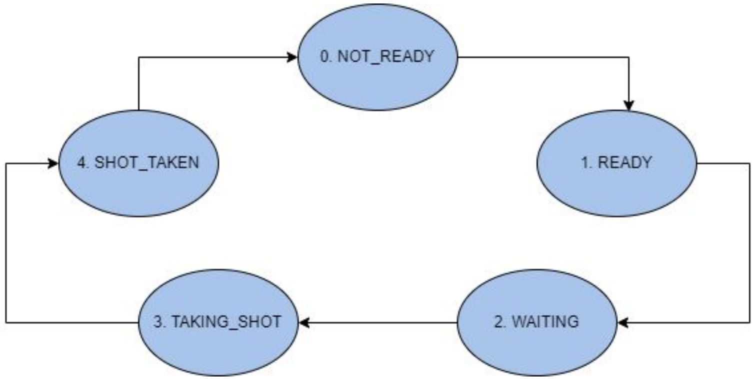
User Shot Attempt (Non-Impaired)

- Peripherals used:
  - Cue stick
  - HUD display
  - HUD camera
- Cue stick orientation and acceleration is reset
- Pre-shot information is displayed on the HUD
- HUD's camera takes a picture at the trigger provided by CCU / Cue Stick
- Picture is sent to the CCU via BLE after the shot attempt is completed



Cue Stick Finite State Machine

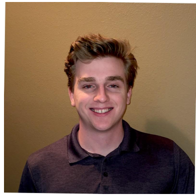
0. NOT_READY
    - Stick is not stationary, user is not ready
1. READY
    - Stick is stationary, user is not ready to take shot
2. WAITING
    - Stick is stationary, user is ready to take shot
3. TAKING_SHOT
    - Stick is not stationary, user is taking a shot
4. SHOT_TAKEN
    - Stick is stationary, user in done taking a shot



Point of Contact Recognition

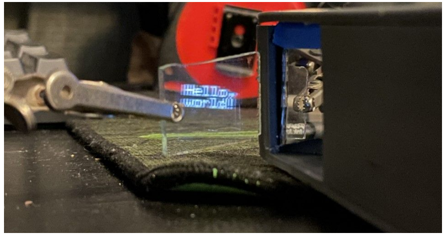
- Hardware:
  - ArduCAM mini camera
  - ESP32
  - CCU (Raspberry Pi)
- Procedure:
  - CCU sends a trigger in the form of a state variable
  - ESP32 sends I2C command to camera to start capture
  - ESP32 polls camera for capture done signal
  - ESP32 reads camera data byte by byte and transmits to CCU
  - CCU assembles image and performs Computer Vision to determine point of impact
    - Mask the image to isolate the cue ball based on color
    - Zoom in on cue ball, and remask to isolate green contours
    - Find contour center
    - Send point of impact coordinates to HUD for display

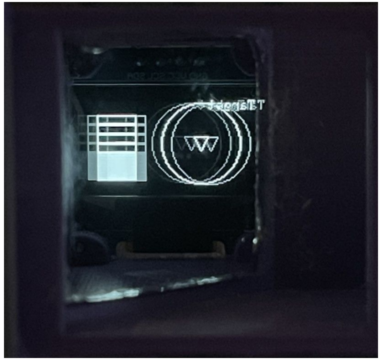
Prototyping

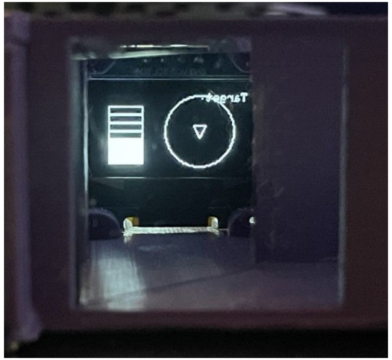


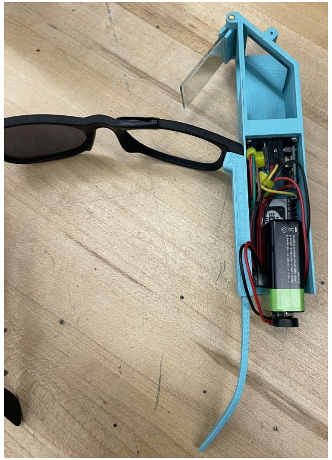
HUD Display Prototyping

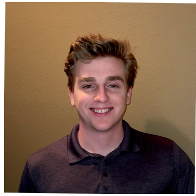
- Starting with a very simple box to verify the design and iron out image quality, the HUDs design has progressed nearly to its final form
- The image quality has been improved by using a first surface mirror and clear reflective film to remove double imaging effects and maximize light reflection



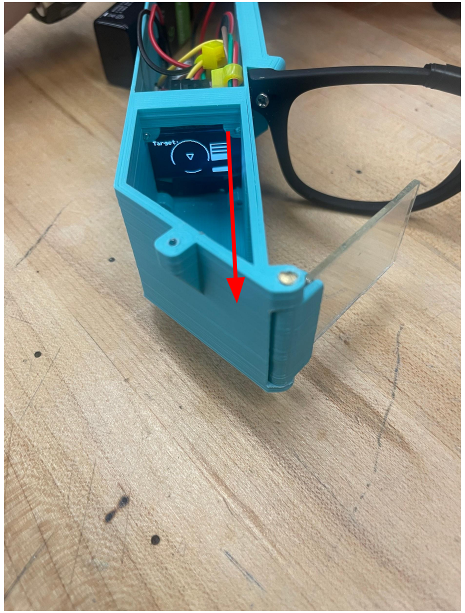




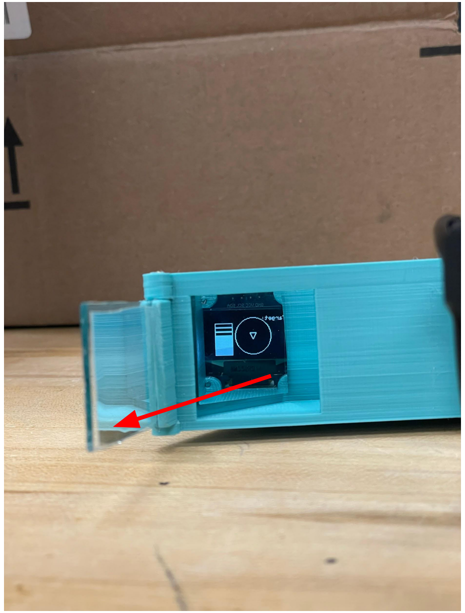




HUD Display Optical Path

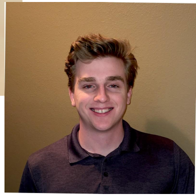


1st Step



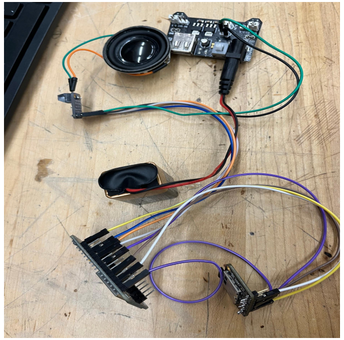
2nd Step

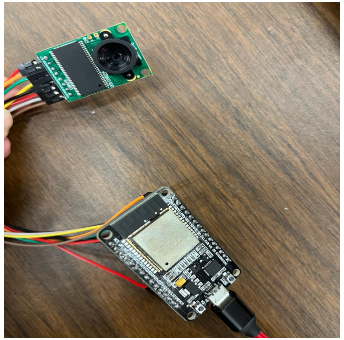
3rd Step



HUD Speaker and Camera Prototyping

- The SD card, as well as the DAC-Speaker combination, have been wired to an ESP32 development board and tested individually
- After that, the audio system was assembled with the ESP32 development board, wired, and tested as well
- Additionally, the camera dev board has been wired to the ESP32 development board and tested
- The next step is fitting the system in the HUD enclosure

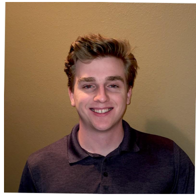




Power System Design

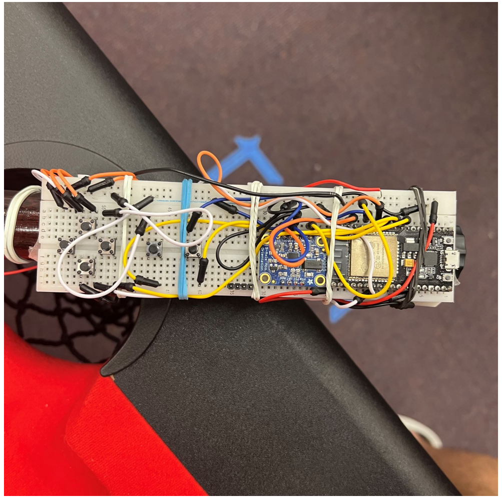
- All components require either 3.3V or 5V
- System design is to supply each ESP or PCB board with 5V and power peripherals through ESP Dev Board
- To exceed battery requirements of at least 30 minutes of operation, a battery with capacity of at least 439 mAH is necessary
- The chosen battery to use was a 9V, 600 maH D cell battery

Battery Requirements			
Component	**Voltage Req**	**Power Draw (Peak)**	**Power Draw (Typical)**
ESP32	3.3 or 5	240	80
Display	3.3	0.78	0.43
Camera	3.3-5	70	20
DAC/Speaker	3 to 6	40	16
Total		351	117
Total (With 1.25 Tolerance)		439	146



Cue Stick Prototyping

- Testing began by connecting the IMU to the development board and reading out raw IMU values
- Button functionality was then tested separately
- Code for IMU and buttons combined and finite state machine was added
    - Mounted to cue stick for further testing
- After FSM was finalized, BLE connectivity was added
- Next step: replace development board with custom PCB

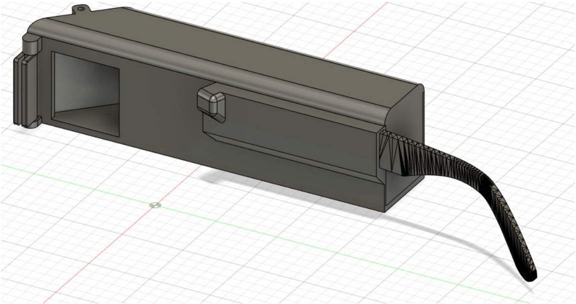


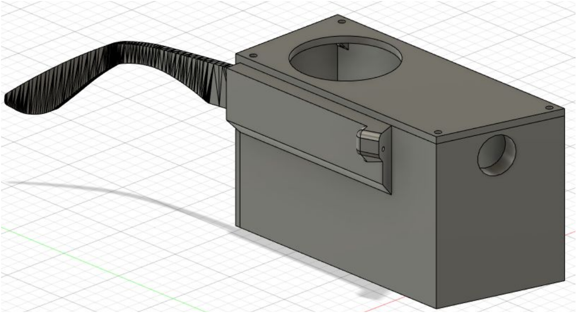
Glove Prototyping

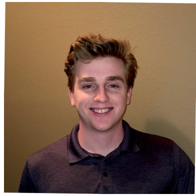
- Started out with testing different devices to determine angle of user, this includes 3-axis magnetometer and IMU, the IMU was chosen due to accurate angle measurements
- Testing ultrasonic sensor with MCU to determine distance to cue ball, then eventually combining both ultrasonic sensor and IMU into one system that communicates with the ESP32
- Finally getting these values to send via BLE and developing mechanical housing for system

Mechanical Design - HUD

- Each subsystem will require its own housing with its own mounting and component protection challenges
- The wearable HUD consists of two sides
  - The right side is for projecting the information in front of the user’s eye and housing the ESP32 which drives all the other peripherals
  - The left side will house the camera, speaker, and the speaker’s digital to analog converter
  - The challenges associated with this design are weight, ergonomics, and wiring the two sides together

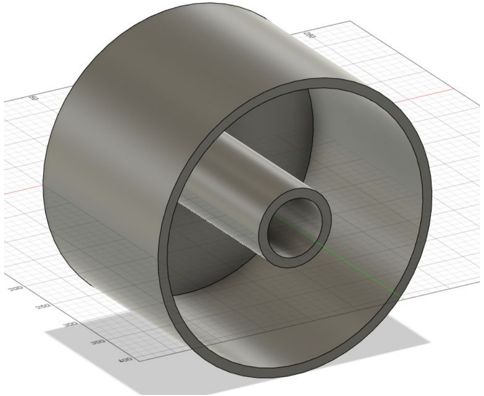


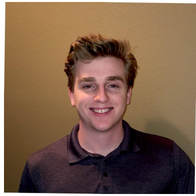




Mechanical Design - Cue Stick

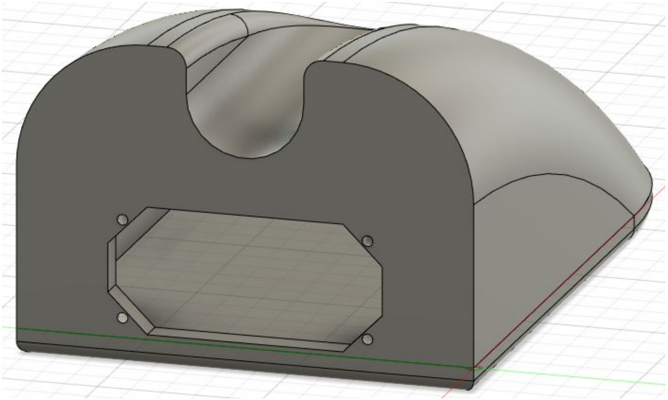
- The cue stick's housing will be mounted and secured at the end of the stick
- This will house the PCB which controls a laser, an IMU, and user input buttons
- The major challenges of this design will be related to minimizing weight and securing the housing

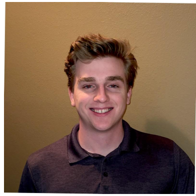




Mechanical Design - Glove

- The glove will house a ESP32, IMU, Ultrasonic Sensor, 9V battery and a regulator circuit
- It needs to fit in the palm of the users hand and have a guide channel in place for the cue stick to be aimed through for the visually impaired user
- The major challenge with this design will be ergonomics





Administrative Content

Financing / Budget

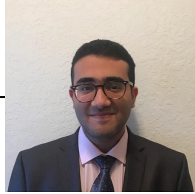
Component	Quantity	Unit Cost	Total
ESP-32 Development Boards	3	$5.96	$17.88
ESP-32 Module	3	$8.95	$26.85
Raspberry Pi 4 8GB	1	$174.90	$174.90
Raspberry Pi PSU	1	$7.99	$7.99
Raspberry Pi Case	1	$11.59	$11.59
Display	1	$7.30	$7.30
3D Printing Materials	1	$25.00	$25.00
Mirror	1	$10.00	$10.00
ArduCAM camera	1	$25.99	$25.99
microSD card (512MB)	1	$4.95	$4.95
microSD card breakout board	1	$2.95	$2.95
MAX98357A DAC	1	$5.95	$5.95
Speaker	1	$1.95	$1.95
PCB and Components	1	$30.00	$30.00
Laser	1	$35.00	$35.00
Ultrasonic Sensor	1	$4.00	$4.00
Adafruit BNO055	2	$29.95	$59.90
Pool Cue	1	$13.97	$13.97
		Total	**$466.17**

Progress

Research	95%
Design	70%
Prototyping	80%
Integration	30%
Testing	50%
Total	70%

Distribution of Work

Name	HUD	CCU	Stick	Glove	Mechanical Design	Wireless Communications	PCB Design
Luke		X	X			X	
Mark	X				X		
Goran				X			X
Mena	X	X				X	



Final Steps/Current Issues

- HUD
  - Fix the integration of all HUD components (camera, speaker and display)
- CCU
  - Adjust the computer vision system for green laser
  - Add Glove angle comparison software on CCU to communicate to HUD speaker
  - Merge subsystem receivers and code on CCU
- Glove
  - Complete final mechanical housing
- Cue Stick
  - Fix IMU on PCB and ensure working condition