Texas A&M University

Capstone Personal Notebook

Diana Canchola

UIN: 530000174

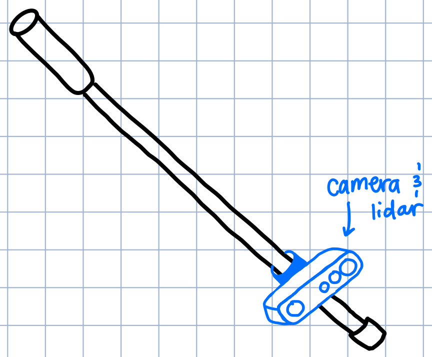
CSCE 483

AUGUST 21, 2024

PROJECT BRAINSTORM

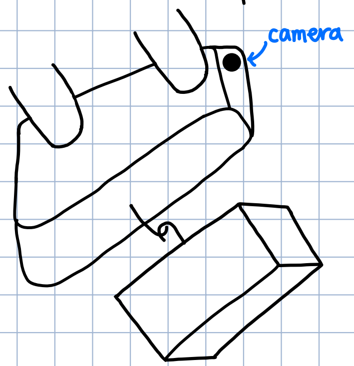
1. BLIND AID

- Attachment to a blind persons cane that will tell them what is in front of them and the distance.
- This information would be sent to an app on their phone or a headpiece to alert the user of the walking stick.



2. DRONE DELIVERY

- Small drone with a way to latch a package
- Takes it directly to the person it is intended for
- Facial recognition enabled to seek person to deliver



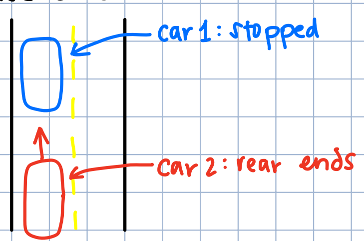
3. SMART PILL HOLDER

- Online app where you can input prescriptions needed for each day
- Based upon the weight it can tell how many pills are in there
- If elder forgets to take them, it will notify the emergency contacts on the app

4. CAR-TO-CAR COLLISION AVOIDANCE

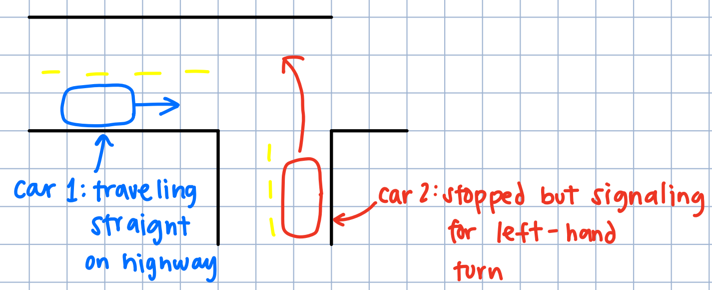
- Enable car-to-car communication about their location, speed, direction
- Can help prevent crashes by alerting drivers of a possible crash or by immediately stopping the cars

use case 1



effect: stop car 2 because it knows car 1's position

USE CASE 2



AUGUST 27, 2024

GUIDE BRAINSTORM

MAIN CONCEPTS

- Placement of sensor
    - have to take weight, size, and depth detection into accaunt
- What kind of sensor
    - camera + depth sensor
    - lidar
    - infrared
- What is the software component? besides writting for the sensors
- Walking sticks are collapsable, how can we keep this while adding wires and sensors
- Software app that alerts you through voice what is going on
- Utilize a SLAM algorithm to keep track of obstacles or objects to alert users
- Handle applies vibrations to the user

AUGUST 29, 2024 PROPOSAL PLANNING

DEADLINES

- need statement
- objective statement
- requirements

NEED

Identify the needs of the project

- Research required
- Identify what is on the market
  - ↳ why this needs to be enhanced
    what are the current limitations

OBJECTIVE

- What is being proposed to meet the need
- Preliminary design objectives ⇒ no details
  - sense & detect obstacle in front of the walking stick
  - alert user of obstacle
- Description of design solution

REQUIREMENTS

- short statements that address the technical needs

Environment
- coding language
- safety
- processor

Performance
- stats on how it should work
- delay time
- processing rates

Functionality
- how it should be used
- expectations on the function

Economic
- price to make ≤ $300
- price to consumer ≤ $25

Energy
- power consumption

Health & Safety
- how it interacts w/ humans
- safety measures

Legal
- intellectual property

maintainability
- Software maintainability
- user availability

manufacturability
- dimensions
- os

Operational
- conditions for operational use
  what the system can withstand

Political
- compliance

Reliability & Availability
- how long will the tool last

Social & Cultural
- accessibility to all cultures

usability
- how user friendly should it be

DESIGN IDEAS CONT.

ACTION: Reach out to Project SEARCH for visually impaired focus group

↓

HAVE NOT GOTTEN A RESPONCE BACK

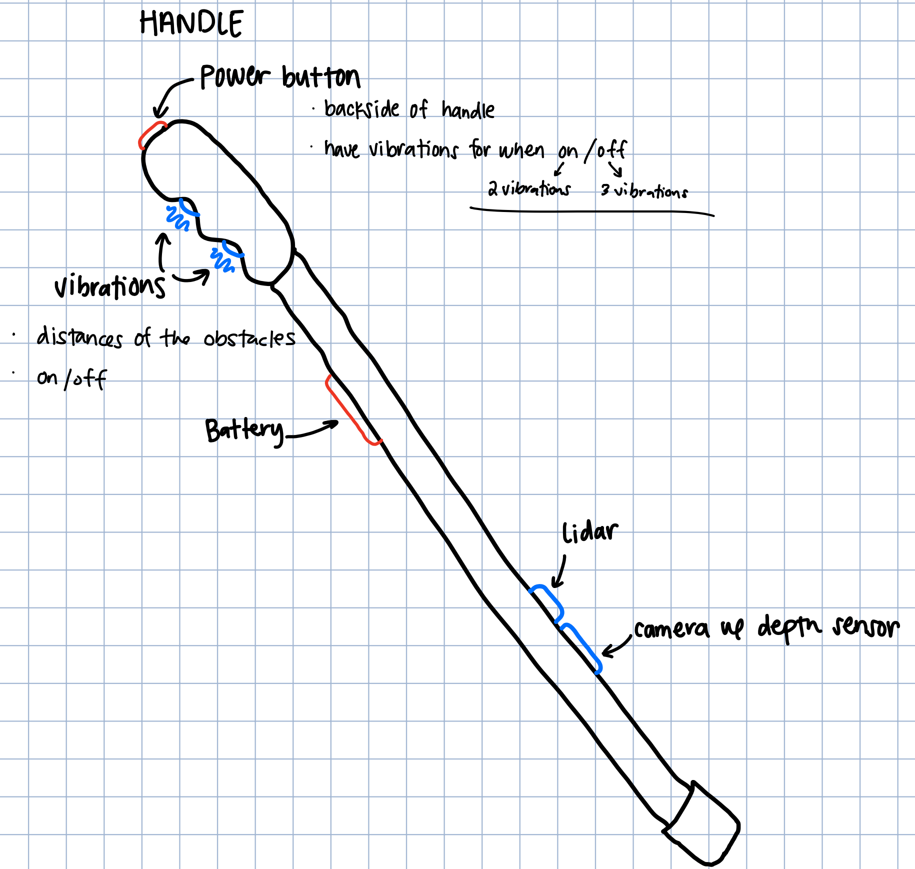
- CONTINUE TO REACHOUT FOR FOCUS GROUPS

  OR

- FIND OTHER ALTERNATIVES
  - SPECIAL ED TEACHERS
  - CSISD PROGRAMS
  - BISD PROGRAMS

September 3, 2024

PROJECT PROPOSAL CONT



NOTES

- object recognition using lidar and camera
  - How far do we go?
    - do we say what the objects are?
      - depends the quality of the camera ability to render the images
- communicate between the sensors and the vibration in the handle
- vibrations should be customized by the user for sensitivity
- battery must last enough for an average trip outside
  - small and commercially available
  - rechargable
- light-weight design with add-ons to keep mobility of walking stick

PROPOSAL NOTES WITH STAKEHOLDERS

NEED STATEMENT

- First sentence should highlight statistics are world wide and not just national statistics
  ↓ numbers are large and can mislead

- Reference where the statistic is coming from => World Health Organization (WHO)

- Give an example of hard environments for the visually impaired
  ↓
  focus on one environment => reduce the scope to a certain environment

REQUIREMENTS

- use more numbers in the requirements
  - performance
  - battery life
  - responce time

- use bullet points instead of numbered requirement statements
  ↓
  don't categorize them / classify the requirements

Weekly Agenda: September 9, 2024

Objectives

- Parts List
- Project Proposal
- Scheduling Breakdown
- Team Breakdown

Parts List

[Parts List Document]
[Purchase Form]

Project Proposal

Section	Team Member Assigned
Team Logo	TBD
Executive Summary	Noah Kilpatrick
Introduction	Jack Couture
Literature and Technical Survey	Jack Letsinger
Proposed Work	Diana Canchola
Engineering Standards	Ryan Wu & Alyan Tharani
References	Everyone
Appendices	Everyone

Scheduling Breakdown

[Schedule Document Link]

Team Breakdown

Team	Leader	Members
Hardware	Jack Letsinger	1. Ryan Wu 2. Alyan Tharani
Software	Jack Couture	1. Noah Kilpatrick 2. Diana Canchola

Meeting 09/10/2024 Notes

- Split up project proposal, going to begin work 9/12/24
- Assigned internal teams, software and hardware subteams
- Began initial development of camera depth sensor code in python, creates array of values

  Actions/Next Steps

  - Create CSV files from the data outputted by the camera
  - Use the CSV files to render images for object recognition
  - Convert python code to c/c++
  - Research micro controllers

Meeting 09/12/2024 Notes

- Created schematic diagrams for the hardware and software components
- Created CSV files from depth camera with better data
- Initial c++ implementation of rendering an image from the csv depth camera files
- Received past project to be used for parts

  Actions/Next Steps

  - Prepare proposal presentation
  - Finish Schedule (Hardware and Software)
  - Begin proposal report

PRESENTATION PLANNING

RUBRIC REQUIREMENTS

1. PROBLEM
   - Research the history of the white cane
     1921: James Briggs invents the white cane (takes walking stick and paints it white so people can easily see it)
     1931: Formal recognition
     1935: Long cane developed for navigation use
     1947: Collapsable
   ★ Development centered around veteran disabilities
     ⋮
     The issue lies in 21st century development

2. NEED
   - Full spatial awareness
   - real-time feedback
   - innovation to familiar device to enhance independence

3. GOALS
   - Detect obstacles
   - Alert users
   - Full mobility

4. LITERATURE
   - unimprovable white cane article I found while researching use cases
   - We Walk
   - Stanford Research project

5. DESIGN CONSTRAINTS
   - size
   - weight
   - Battery life
   - Latency

6. ALTERNATE SOLUTIONS
   - Taken from past notes taken by the team and myself

7. PROPOSED DESIGN
   - Jack L: Hardware Diagram
   - Jack C: Software Diagram
   - I have overall sketch from a past brainstorm class session

8. BUDGET
   - Get links for items
     Walking Stick, Vibration motor, LiDAR, Battery, Battery Charger, Pi

9. SCHEDULE
   - Ask hardware and software leads to add their internal team deadlines

10. PRODUCT IMPACT
   - societal: more independence for millions
   - safety: less accidents
   - environment: commercially available materials

Weekly Agenda: September 16, 2024

Objectives

- Proposal Presentation
- Proposal Report
- Finalize Parts Ordering

Proposal Presentation

Proposal Presentation

Project Proposal

Proposal Report

Section	Team Member Assigned
Team Logo	TBD
Executive Summary	Noah Kilpatrick
Introduction	Jack Couture
Literature and Technical Survey	Jack Letsinger
Proposed Work	Diana Canchola
Engineering Standards	Ryan Wu & Alyan Tharani
References	Everyone
Appendices	Everyone

Ordering Parts

Hardware Shopping List - Amazon

Meeting 09/17/2024 Notes

- For the presentation we must add more detail
  - Dimensions of each part
  - Weight of each part
  - Values describing our goals
  - Design and Validation slide
  - Handle Dimensions

  Actions/Next Steps
  - Must have 90% completion on individual contribution for the proposal report next class
  - Edit the proposal presentation to include the details missing

Meeting 09/19/2024 Notes

- Completed the edits for the proposal presentation
  - Added each part's dimensions and weight
  - Added values for distance of obstacle detection, battery life, latency in seconds, weight of the final design
  - Added two Design & Validation slides to describe how we will test our goals and objectives as well as hardware and software components

  Actions/Next Steps
  - Complete individual portions of the proposal report for each member of the group
    - CV and bio-sketch
  - Complete executive summary of the proposal report
  - Continue working with depth camera and software associated through the weekend as a VM operating 24/7 is fully functional for all members of the team

PRESENTATION NOTES

- have weight of white cane => 0.55lb
- have weight of all the components =>
- have handle options & dimensions
- design Validation

depth camera =>
lidar =>
battery => 44g x2

NOTES

- add more values to the presentation from our requirement statements
    - weight of each component on the actual cane
      ↓
      rq: weight will be less than 2lbs
    - latency will be less than or equal to 0.5sec from obstacle detection to the handle vibrations
    - height will not exceed 53" (current design)
    - Obstacle detection will be w/ 2m
    - Different vibrations at 2m, 1m, 0.5m, 0m

- Design Validation align to our goals/objectives
    - mobility
    - object detection
    - user alert
    - provide steps/tasks on how we will test all the requirements are met
        - weather conditions
        - obstacles
        - how each component will be tested
    - time allotted for integration & test and design validation

PROJECT PROPOSAL : PROPOSED WORK

SUB CATEGORIES

- EVALUATION OF ALTERNATE SOLUTIONS .... 1 PAGE
- DESIGN SPECIFICATIONS .... 3-4 PAGES
- APPROACH FOR DESIGN VALIDATION .... 1 PAGE

Evaluation of alternate solutions

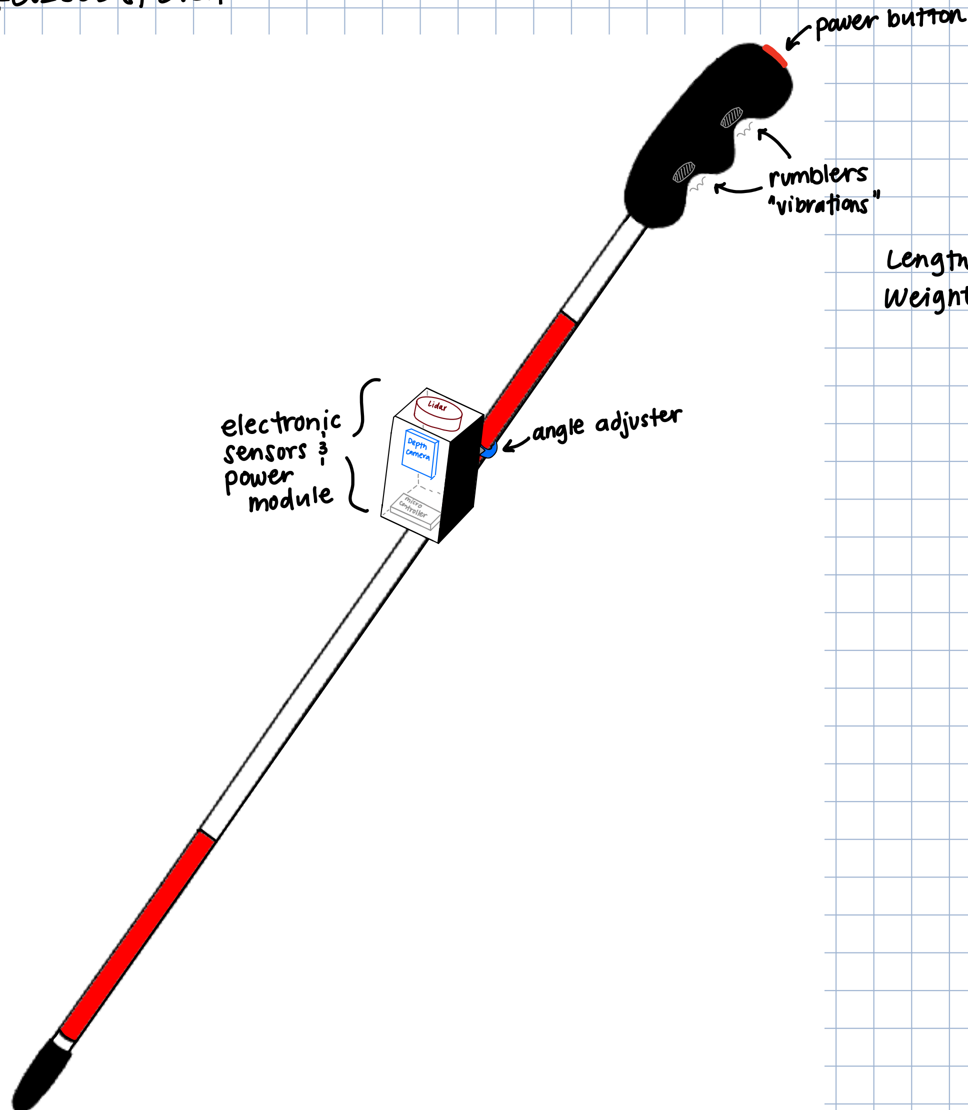
1. Phone Application with audio
2. SLAM Algorithm to add obstacle to database
3. IR Sensor for object detection
4. Bag / Belt attachment with cane
5. Camera object recognition

give brief description
outline pros & cons

Design Specifications

- Description of the proposed solution
- Create a better rendering / sketch of the cane to be added
  describe the design of the sketch
- Add the hardware and software diagrams
  describe each component of the diagrams
- Create a table for each part from the design sketch & hardware
- Could add a sequence diagram to sum it up

REVISED SKETCH



Length = 53"
Weight < 2lbs

Weekly Agenda: September 23, 2024

Objectives

- Weekly Report 1
- Hardware Subteam
- Software Subteam
- First Technical Working Week

Weekly Report

[Weekly Report 1]

Hardware Subteam

- Work towards Raspberry pi / Lidar integration

Software Subteam

- Refine C++ code for video conversion with buffer data

Working Weeks

- Moving forward we will have weekly reports with the professor
  - Tuesday
  - 11:10am
  - 20-30 minute duration
- Tuesday Agenda
  - Weekly Report with professor
  - Subteam focused after
- Thursday Agenda
  - Not obligated by the professor but would be useful to continue to meet
  - This time could be used to provide status, roadblocks or team wide work time

Meeting 09/24/2024 Notes

Hardware

- Alyan - stripped the Lidar wire so that it could interface with the raspberry pi
- Ryan - Soldered the pi pico, configured the Lidar
- Jack - Soldered the pi pico, configured the Lidar, got python running on the

During this hardware meeting we evaluated the materials we were given. We made the choice to strip the wire header of the lidar so it could be used with the pinout, as opposed to having to wait to order whatever 6-pin header was needed. The wire could be disconnected or replaced if needed, and is not permanently attached to the Lidar. Ryan and Jack soldered the Raspberry Pi pico pin headers, allowing it to be used on a breadboard.

Software

- Added everyone to git repo
- Researched documentation for the LiDAR
- Discussed overall software architecture approach

Actions/Next Steps

- get the lidar working with the Pi Pico (In any capacity)
- Receive LiDAR data from hardware team

Meeting 09/26/2024 Notes

- Debugged the Pi Pico for any power distribution issues
  - Checked using a multimeter if 5V was being distributed
  - Debugged the breadboard and Pico by testing if an LED was getting power
  - Started developing a script in python for testing purposes to check if LiDAR was working

Actions/Next Steps

- Debug why depth camera and LiDAR were not working properly when plugged into the Pico rather than directly to the computer.

```python
import matplotlib.pyplot as plt

# Define the packet data (as an example)
data =
b'YY\x13\x00|\x0e\x10\thYY\x13\x00z\x0e\x10\tfYY\x14\x00x\x0e\x10\teYY\x14\x00w\x0e\x10\tdYY\x14\x00u\x0e\x10\tbYY\x14\x00t\x0e\x10\taYY\x14\x00o\x0e\x10\t\\YY\x14\x00x\x0e\x10\teYY\x14\x00g\x0e\x10\tTYY\x14\x00z\x0e\x10\tgYY\x14\x00p\x0e\x10\t]YY\x13\x00k\x0e\x10\tWYY\x13\x00a\x0e\x10\tMYY\x13\x00i\x0e\x10\tUYY\x14\x00r\x0e\x10\t_YY\x14\x00k\x0e\x10\tXYY\x14\x00g\x0e\x10\tTYY\x14\x00a\x0e\x10\tNYY\x14\x00i\x0e\x10\tVYY\x14\x00g\x0e\x10\tTYY\x14\x00h\x0e\x10\tUYY\x14\x00\\\x0e\x10\tIYY\x14\x00c\x0e\x10\tPYY\x14\x00X\x0e\x10\tEYY\x14\x00^\x0e\x10\tKYY\x14\x00f\x0e\x10\tSYY\x13\x00f\x0e\x10\tRYY\x13\x00h\x0e\x10\tTYY\x14\x00f\x0e\x10\tSYY\x14\x00[\x0e\x10\tHYY\x14\x00`\x0e\x10\tMYY\x14\x00b\x0e\x10\tOYY\x14\x00!\x0e\x10\t\x0e'
b'YY\x14\x00\x1f\x0e\x10\t\x0cYY\x14\x00\x18\x0e\x10\t\x05YY\x14\x00\x1a\x0e\x10\t\x07YY\x14\x00\x15\x0e\x10\t\x02YY\x14\x00\x1d\x0e\x10\t\nYY\x14\x00\x1e\x0e\x10\t\x0bYY\x14\x00\x19\x0e\x10\t\x06YY\x14\x00\x1b\x0e\x10\t\x08YY\x14\x00&\x0e\x10\t\x13YY\x14\x00\x1e\x0e\x10\t\x0bYY\x14\x00%\x0e\x10\t\x12YY\x14\x00%\x0e\x10\t\x12YY\x14\x00%\x0e\x10\t\x12YY\x14\x00,\x0e\x10\t\x19YY\x14\x004\x0e\x10\t!YY\x14\x00;\x0e\x10\t(YY\x13\x009\x0e\x10\t%YY\x14\x007\x0e\x10\t$YY\x14\x00I\x0e\x10\t6YY\x14\x00H\x0e\x10\t5YY\x14\x00K\x0e\x10\t8YY\x14\x00J\x0e\x10\t7YY\x14\x00Q\x0e\x10\t>YY\x14\x00U\x0e\x10\tBYY\x14\x00I\x0e\x10\t6YY\x13\x00X\x0e\x10\tDYY\x14\x00S\x0e\x10\t@YY\x14\x00V\x0e\x10\tCYY\x14\x00W\x0e\x10\tDYY\x14\x00R\x0e\x10\t?YY\x14\x00R\x0e\x10\t?YY\x14\x00\\\x0e\x10\tI'
b'YY\x14\x00Q\x0e\x10\t>YY\x13\x00T\x0e\x10\t@YY\x13\x00X\x0e\x10\tDYY\x13\x00W\x0e\x10\tCYY\x13\x00g\x0e\x10\tSYY\x13\x00f\x0e\x10\tRYY\x13\x00X\x0e\x10\tDYY\x13\x00M\x0e\x10\t9YY\x13\x00f\x0e\x10\tRYY\x13\x00f\x0e\x10\tRYY\x13\x00k\x0e\x10\tWYY\x13\x00i\x0e\x10\tUYY\x13\x00o\x0e\x10\t[YY\x13\x00r\x0e\x10\t^YY\x13\x00q\x0e\x10\t]YY\x13\x00o\x0e\x10\t[YY\x13\x00o\x0e\x10\t[YY\x13\x00x\x0e\x10\tdYY\x13\x00h\x0e\x10\tTYY\x13\x00m\x0e\x10\tYYY\x13\x00w\x0e\x10\tcYY\x13\x00z\x0e\x10\tfYY\x13\x00p\x0e\x10\t\\YY\x14\x00w\x0e\x10\tdYY\x13\x00|\x0e\x10\thYY\x13\x00w\x0e\x10\tcYY\x13\x00w\x0e\x10\tcYY\x13\x00}\x0e\x10\tiYY\x13\x00}\x0e\x10\tiYY\x13\x00u\x0e\x10\taYY\x13\x00x\x0e\x10\tdYY\x13\x00|\x0e\x10\th'
b'YY\x14\x00W\x0e\x10\tDYY\x14\x00]\x0e\x10\tJYY\x13\x00P\x0e\x10\t<YY\x14\x00U\x0e\x10\tBYY\x13\x00K\x0e\x10\t7YY\x13\x00O\x0e\x10\t;YY\x13\x00G\x0e\x10\t3YY\x14\x00D\x0e\x10\t1YY\x13\x00M\x0e\x10\t9YY\x13\x00O\x0e\x10\t;YY\x13\x00H\x0e\x10\t4YY\x13\x00M\x0e\x10\t9YY\x13\x00T\x0e\x10\t@YY\x13\x00M\x0e\x10\t9YY\x14\x00
```

```
U\x0e\x10\tBYY\x14\x00S\x0e\x10\t@YY\x14\x00P\x0e\x10\t=YY\x13\x
00L\x0e\x10\t8YY\x13\x00P\x0e\x10\t<YY\x13\x00G\x0e\x10\t3YY\x13
\x00L\x0e\x10\t8YY\x13\x00W\x0e\x10\tCYY\x13\x00O\x0e\x10\t;YY\x
13\x00X\x0e\x10\tDYY\x14\x00L\x0e\x10\t9YY\x13\x00I\x0e\x10\t5YY
\x13\x00L\x0e\x10\t8YY\x13\x00Q\x0e\x10\t=YY\x13\x00Q\x0e\x10\t=
YY\x13\x00N\x0e\x10\t:YY\x13\x00H\x0e\x10\t4YY\x13\x00P\x0e\x10\
t<YY\x13\x00Z\x0e\x10\tF'
b"YY\x13\x00c\x0e\x10\tOYY\x13\x00^\x0e\x10\tJYY\x13\x00]\x0e\x1
0\tIYY\x13\x00Z\x0e\x10\tFYY\x13\x00R\x0e\x10\t>YY\x13\x00U\x0e\
x10\tAYY\x13\x00L\x0e\x10\t8YY\x13\x00G\x0e\x10\t3YY\x13\x00T\x0
e\x10\t@YY\x13\x00O\x0e\x10\t;YY\x13\x00E\x0e\x10\t1YY\x14\x00:\
x0e\x10\t'YY\x14\x00I\x0e\x10\t6YY\x13\x00O\x0e\x10\t;YY\x13\x00
G\x0e\x10\t3YY\x13\x00J\x0e\x10\t6YY\x13\x00I\x0e\x10\t5YY\x13\x
008\x0e\x10\t$YY\x13\x00<\x0e\x10\t(YY\x13\x00?\x0e\x10\t+YY\x13
\x00>\x0e\x10\t*YY\x13\x002\x0e\x10\t\x1eYY\x14\x001\x0e\x10\t\x
1eYY\x14\x00B\x0e\x10\t/YY\x13\x00?\x0e\x10\t+YY\x13\x00=\x0e\x1
0\t)YY\x14\x009\x0e\x10\t&YY\x14\x008\x0e\x10\t%YY\x14\x006\x0e\
x10\t#YY\x14\x004\x0e\x10\t!YY\x13\x008\x0e\x10\t$YY\x13\x003\x0
e\x10\t\x1f"
b'YY\x14\x00\xc7\r\x10\t\xb3YY\x13\x00\xc6\r\x10\t\xb1YY\x13\x00
\xbf\r\x10\t\xaaYY\x14\x00\xb9\r\x10\t\xa5YY\x13\x00\xc8\r\x10\t
\xb3YY\x13\x00\xc9\r\x10\t\xb4YY\x13\x00\xbd\r\x10\t\xa8YY\x13\x
00\xb9\r\x10\t\xa4YY\x13\x00\xba\r\x10\t\xa5YY\x13\x00\xb2\r\x10
\t\x9dYY\x13\x00\xb5\r\x10\t\xa0YY\x13\x00\xb1\r\x10\t\x9cYY\x13
\x00\xa5\r\x10\t\x90YY\x13\x00\x98\r\x10\t\x83YY\x13\x00\x8f\r\x
10\tzYY\x13\x00\x93\r\x10\t~YY\x13\x00\x9a\r\x10\t\x85YY\x14\x00
\x9b\r\x10\t\x87YY\x13\x00\xa4\r\x10\t\x8fYY\x13\x00\xb5\r\x10\t
\xa0YY\x14\x00\xbf\r\x10\t\xabYY\x14\x00\xca\r\x10\t\xb6YY\x13\x
00\xd2\r\x10\t\xbdYY\x13\x00\xed\r\x10\t\xd8YY\x14\x00\xe7\r\x10
\t\xd3YY\x14\x00\x02\x0e\x10\t\xefYY\x14\x00\n\x0e\x10\t\xf7YY\x
14\x00\r\x0e\x10\t\xfaYY\x14\x00\x0f\x0e\x10\t\xfcYY\x14\x00\x1b
\x0e\x10\t\x08YY\x13\x00*\x0e\x10\t\x16YY\x13\x005\x0e\x10\t!'

distances = []

# Define the frame length and start bytes
frame_length = 9  # Assuming each frame is 9 bytes long
start_bytes = b'YY'  # LiDAR frame header 'YY'

# Function to decode little-endian distance from two bytes
def decode_distance(byte1, byte2):
    return byte1 + (byte2 << 8)  # Combine LSB (byte1) and MSB
(byte2)

# Function to decode little-endian signal strength from two
bytes
```

```
def decode_signal_strength(byte1, byte2):
    return byte1 + (byte2 << 8)  # Combine LSB (byte1) and MSB
(byte2)

# Process the data by frames
total_length = len(data)
print(f"Total data length: {total_length} bytes")

# Iterate through the data and decode each frame
for i in range(0, total_length, frame_length):
    frame = data[i:i + frame_length]

    # Check if the frame starts with the correct header 'YY'
    if frame.startswith(start_bytes):
        # Decode distance from bytes 2 and 3 (little-endian)
        distance = decode_distance(frame[2], frame[3])
        distances.append(distance)
        # Decode signal strength from bytes 4 and 5 (little-
endian)
        signal_strength = decode_signal_strength(frame[4],
frame[5])

        # Output the decoded data
        print(f"Frame {i // frame_length + 1}:")
        print(f"  Distance: {distance} cm")
        print(f"  Signal Strength: {signal_strength}")
        print(f"  Reserved/Other Bytes: {frame[6:].hex()}")
    else:
        print(f"Invalid frame starting at byte {i}")
x_axis = [x for x in range(len(distances))]

plt.plot(x_axis, distances)
plt.ylim(0,max(distances)+5)
plt.show()
```

CDR Report

Hardware

- Working breadboard prototype
- Pinout diagram
  - Battery
  - LiDAR
  - Depth Camera
  - 4 Haptic sensors
- CAD model of Handle with components inside
  - 3D renderings and actual handle
- CAD model of electronic sensor housing
  - 3D renderings

Software

- Depth Camera python script
- LiDAR python script
- Compare data of the LiDAR and Depth camera together
- Plots that show the data for both sensors

Meeting Notes

- Pico GPIO pins for adjusting voltages are not well supported, might add resistors to make the vibrations less
  - Create pulses to turn on and off the voltage being supplied to the haptic sensors
- Haptic Sensors pulse pattern could change with the type of object seen by GUIDE
  - Incorporate AI/ML into the software prototype and classify objects using the depth camera
- 3D sensor housing debate on which sensor should go above the other or side by side.
  - Need to purchase a go pro angle adjuster

Weekly Agenda: October 21, 2024

Objectives

- CDR Report
- CDR Presentation
- Notebook
- Peer Review
- Hardware Subteam
- Software Subteam
- Second Technical Working Week

CDR Report

CDR Report

CDR Presentation

CDR Presentation

Hardware Subteam

- Work on 3D model for Handle
- Work on 3D model for electronic housing

Software Subteam

- Begin development on breadboard prototype
- Complete the aggregation of both the depth camera and LiDAR
- Brainstorm how to best use AI/ML to create a “smart” obstacle detection engine

Meeting 10/22/2024 Notes

- Completed CDR Report
- Completed CDR Presentation
- Presented the current progress made on GUIDE
- Discussed the electronic housing 3D model details
    - Fit onto the go pro angle adjuster
    - Insert the sensors into the front face of the attachment
    - The sensors have casings with screw holes that can be added to the design
    - Allow for wires to come in and out of the design
- Breadboard prototype was handed off to software team for direct development on the Pico

Weekly Agenda: October 28, 2024

Objectives

- 3D Modeling
- Software Development

Hardware Subteam

- Print out 3D Model for final handle design
- Test Foam on first handle print to attach handle and walking stick
- Print out first take at electronic housing
    - Test out angle capabilities
    - Add sensors on their and test directly on there

Software Subteam

- Begin development on breadboard prototype
    - Use GPIO pins instead of csv files
- Python Depth Camera Code
    - Extract distance measurements from depth camera
    - Compare this to the distances given from the LiDAR

Meeting 10/29/2024 Notes

- Weekly report meeting with the professor
  - Add/Update requirements on CDR based on added/deleted components in the final design
    - Update the weight
    - Update the response times
    - Add ranges
  - Update the system level description
    - Add a slide after the system level design that goes in depth
      - Include all components in final design
  - Print out handle and the electronic housing
    - Add requirements for angle adjustment using the GoPro angle adjuster
      - Does this inhibit range?
      - Does placement matter?
- Prepare for 3D printed parts on Thursday
  - Purchase foam for gaps and seals to test on the old handle
- Finish the depth camera python code to decode all bytes especially the distance data
  - Find bytes for distance, temperature

Meeting10/31/2024 Notes

- Testing Depth Camera script
  - Bytes of data may be different than first thought, this messes up the decoding portion of the code
- Soldering Haptic Sensors
  - Connections for haptic sensors are weak and need to be solid in order to improve their longevity and avoid disconnections.

Weights

- Walking Stick: 0.5lb
- Handle:
- Electronic Housing:
- Pico 2: 0.13lb
- Battery: 0.26lb
- Buck Converter: 0.006lb
- LiDAR: 0.011
- Depth Camera: 0.022
- 4 Haptic Sensors: Negligible
- Angle Adjuster: 0.19lb
- 3D Prints: 0.21

Total: 1.33lb

Weekly Agenda: November 12, 2024

Objectives

- 3D Modeling
- Software Development

Hardware Subteam

- Start soldering new wires for final prototype
- Attach 3D printed components to the walking stick
- Brainstorm methods to attach two halves of walking stick in the final prototype

Software Subteam

- Python Depth Camera Code
  - Extract distance measurements from depth camera
  - Compare this to the distances given from the LiDAR
  - Finalize method for latency optimization

Meeting 11/12/2024 Notes

- Brainstorming ways to attach handle to the halves and the walking stick
  - We are going to purchase the velcro straps to stick both together
  - Bring saran wrap
  - Bring loctite foam
- Changing 3D model for better fitting handle
  - Decreasing the extrusion between the two grips
- Brainstorming how to fit all the electronics into the handle
  - Where to place all the sensors particularly the battery
    - Influences the size of the wires
    - Influences the pins used on the Pico

Meeting 11/14/2024 Notes

- Reprinted the handle for better fit without compromising the capacity for the components
  - Slightly better fit
  - Electronic sensor housing is more stable
    - Extrusion on connection with angle adjuster
    - Thicker print
  - Final 15 hour print has been finalized
    - Changes include one haptic sensor insert is off
    - Sensor adjuster can be merged into one instead of having a dowel in the middle of the two pieces
- Handle integration
  - Soldered final Pico microcontroller to haptic sensors, buck converter, and wires for the LiDAR and Depth Camera.
- Software for combining LiDAR distances and Depth Camera distances to create an average distance measurement

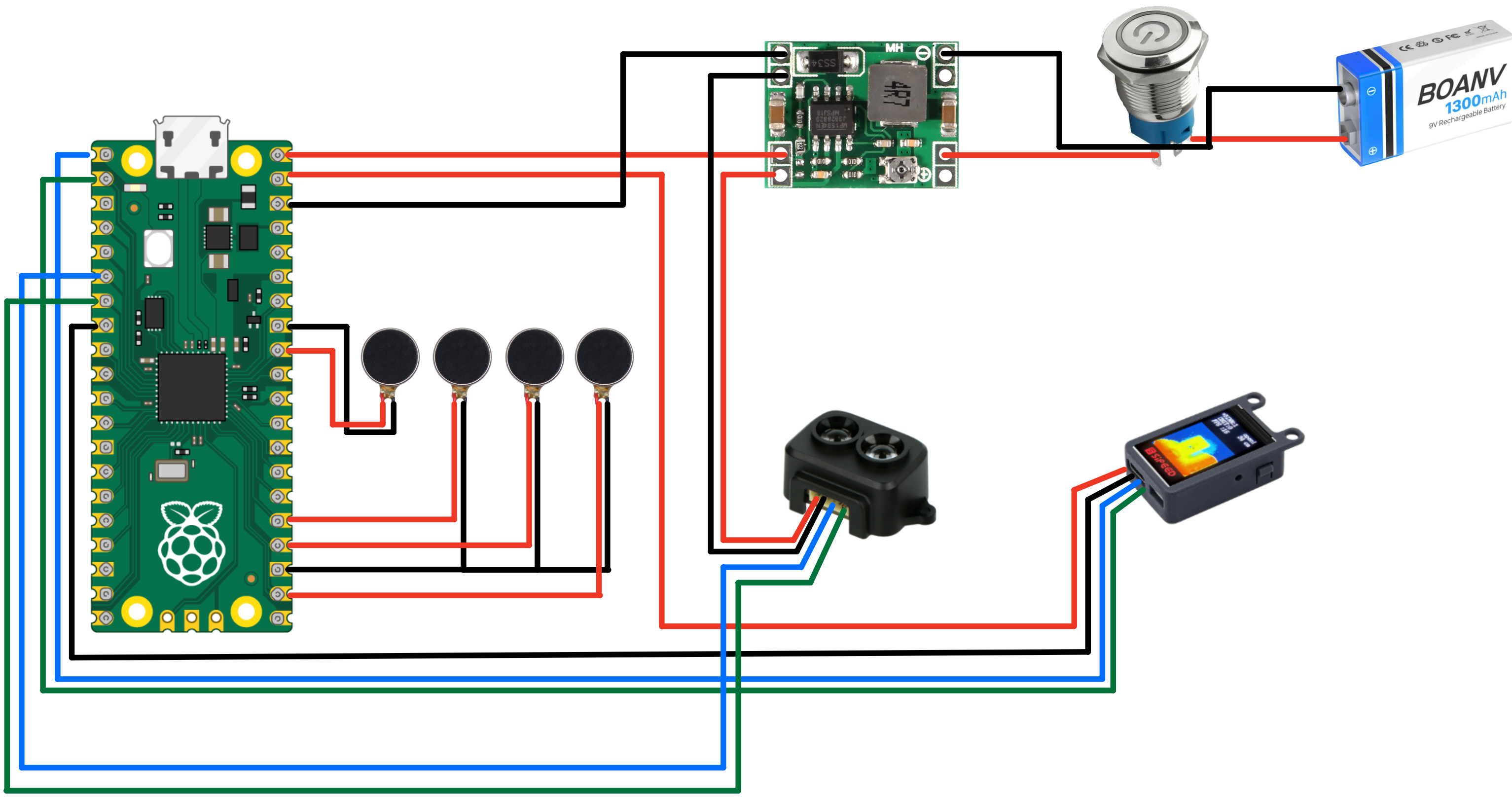
GUIDE FINAL DELIVERABLES

REQUIRED

- Readme
  - documents the list of all contents
- Designs
  - code
  - schematics
  - data
  - data sheets
  - software tools
- Reports
  - proposal - CDR - weekly reports - final report - presentations
- Media
  - pictures
  - movie demo
  - poster
- Team Notebooks

To DO

- Add Weekly Report8
- Hardware
- Software
- Final Reports
- Media
- Team Notebooks


MH
4R7
BOANV
1300mAh
9V Rechargeable Battery