SMART GPA

JENNIFER CASTILLO – ELECTRICAL ENGINEERING
JORGE DARDON – COMPUTER ENGINEERING
DANIELA OTERO – COMPUTER ENGINEERING
NICHOLAS ZOLLO – PHOTONICS ENGINEERING

Group 25
Spring 2020
Senior Design 2

SMART GARAGE PARKING AID

What is the SMART GPA?

- The SMART GPA is a garage parking aid system designed for personal home use
- The system measures the distance from the wall to a car in real time using laser triangulation system
- Colored light indicators will display based on the current distance measurement
- The system will include a touch display which can be used for configuration, personalization, and tuning

Motivation

- Idea came to mind when one of our members hit the wall of a parking spot at a UCF garage
- Getting a close parking spot in an enclosed space can be difficult and risky to do, so we want to make a product that makes this easier by showing a distance indicator
- While products for doing this already exist and are employed in parking garages, we want to add convenience features for a personal home environment
- We want to make coming home and parking after work relaxing and comfortable

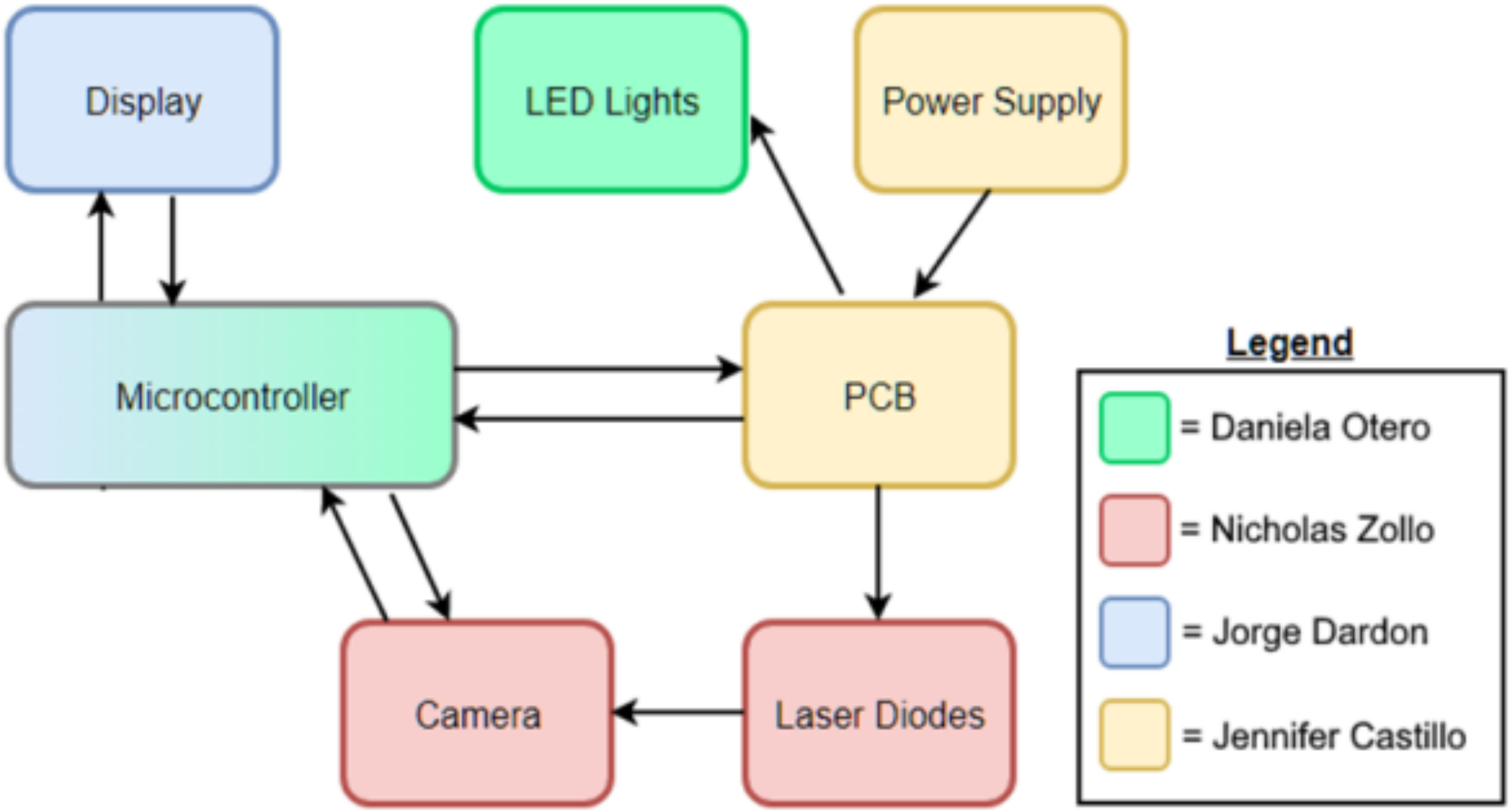
Project Goals & Objectives

- Make the system simple and convenient to use
- Make distance measurement accurate and fast enough to precisely track a car coming towards the device
- Allow all maintenance and customization of the system to be done directly by the user with the touch display
- After main system completed, add additional features to improve home user experience and make product more unique (Cancelled due to coronavirus)

Specifications

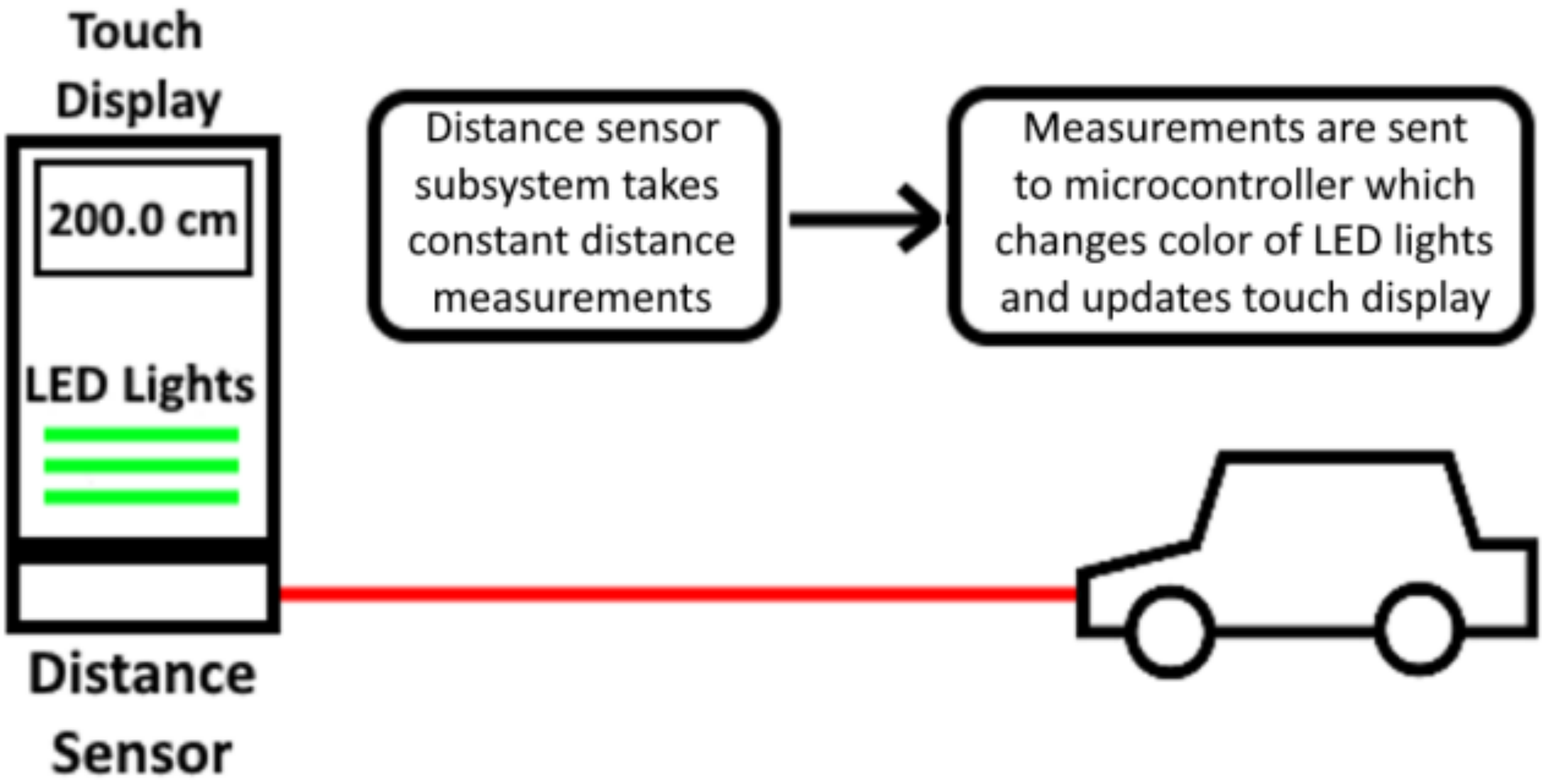
Parameter	Specification
Operating Range	1 to 10 feet
Measurement Rate	$\geq$ 8 Hz
Measurement Accuracy	Error $\leq$ 10 cm
Output Power of Laser	$\leq$ 5 mW
Output Power of LEDs	$\leq$ 15 W
Input Voltage	$\leq$ 12 V
Enclosure Size	1.5 m x 0.2 m x 0.5 m

Project Block Diagram



Subsystems & Mechanisms

Use Case Diagram

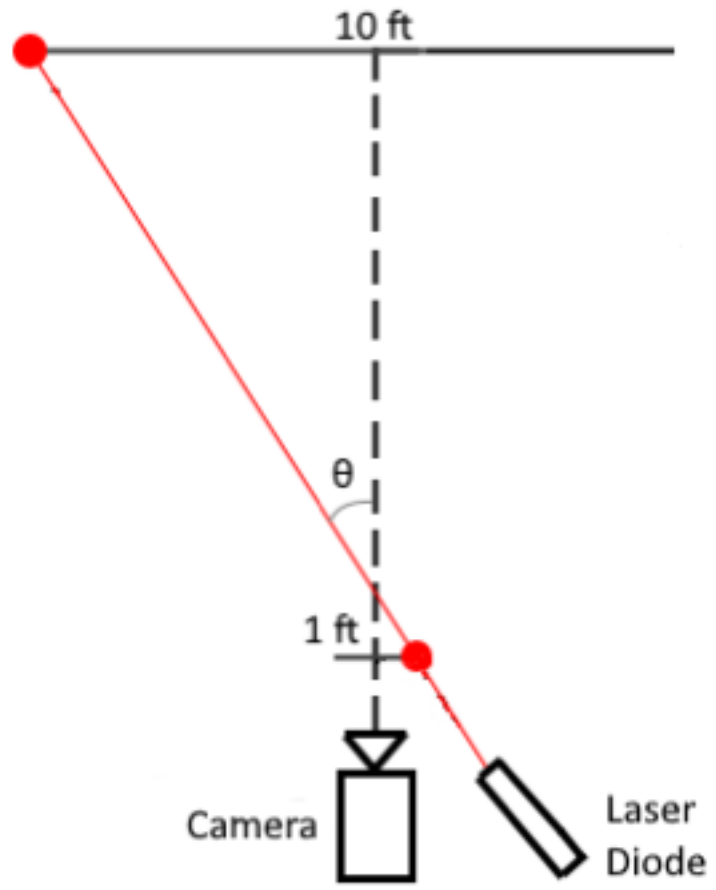


Distance Sensor Method Comparison

	Ultrasonic Sensor	Laser Time of Flight Sensor	Phase Shift Laser Rangefinder	Laser Triangulation
Optimal Range	Up to 10 m	10 to 5000 m	Dependent on modulation frequency	Dependent on optical setup
Accuracy	< 1 cm accuracy	Dependent on speed of electronics	Dependent on modulation frequency	Dependent on camera resolution
Max Response Time	20 ms	20 ns	20 ns	1 / (camera fps)
Required Components	Ultrasonic transducer	Laser, high speed photodiode	Laser, modulator, microcontroller	Laser, camera, micro-controller
Ease of Implementation	Simple	Difficult; requires precise components	Difficult; requires high performance components	Medium; requires image processing
Cost	< $50	Possibly >$200	> $1000	< $100

Laser Triangulation Distance Sensor

- For the SMART GPA's distance measurement, we are using a basic laser triangulation by setting a laser diode at a specific constant angle from a camera and observing the position of the laser dot on the camera to observe the distance
- This method was chosen as it works well for a variety of ranges, including our specified range of 1 to 10 feet
- Significant image processing work is required to identify the position of the laser dot consistently and quickly



Distance Calculation Equation

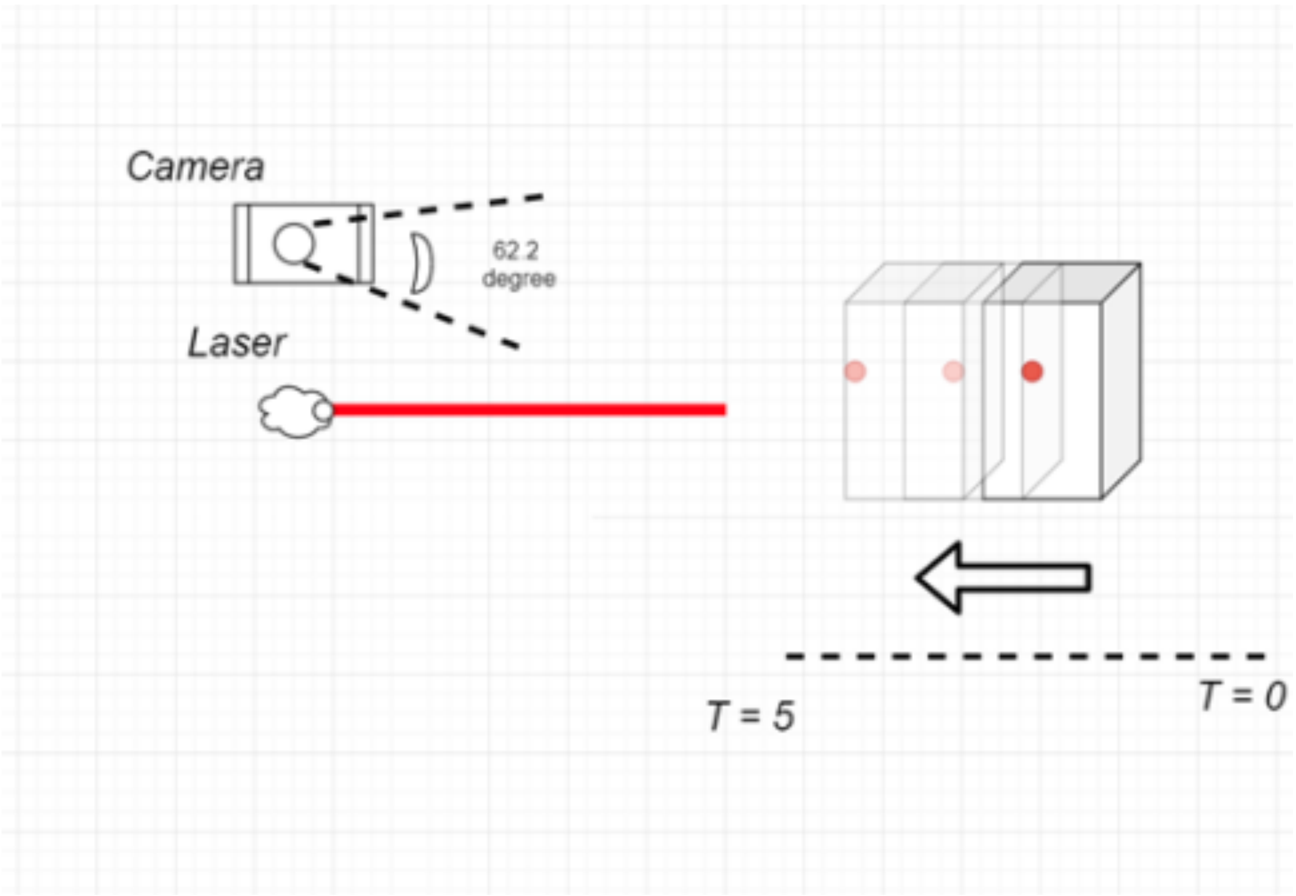
$$z = \frac{d}{\left(\frac{2x}{W} - 1\right) * \tan\left(\frac{HFOV}{2}\right) + \tan(\theta)}$$

- z = Final measured distance
- x = Pixel position of laser dot on camera

- d = Distance between laser diode and camera
- W = Horizontal resolution of the camera
- HFOV = Horizontal angular FOV of the camera
- Θ = Angle between laser diode and camera

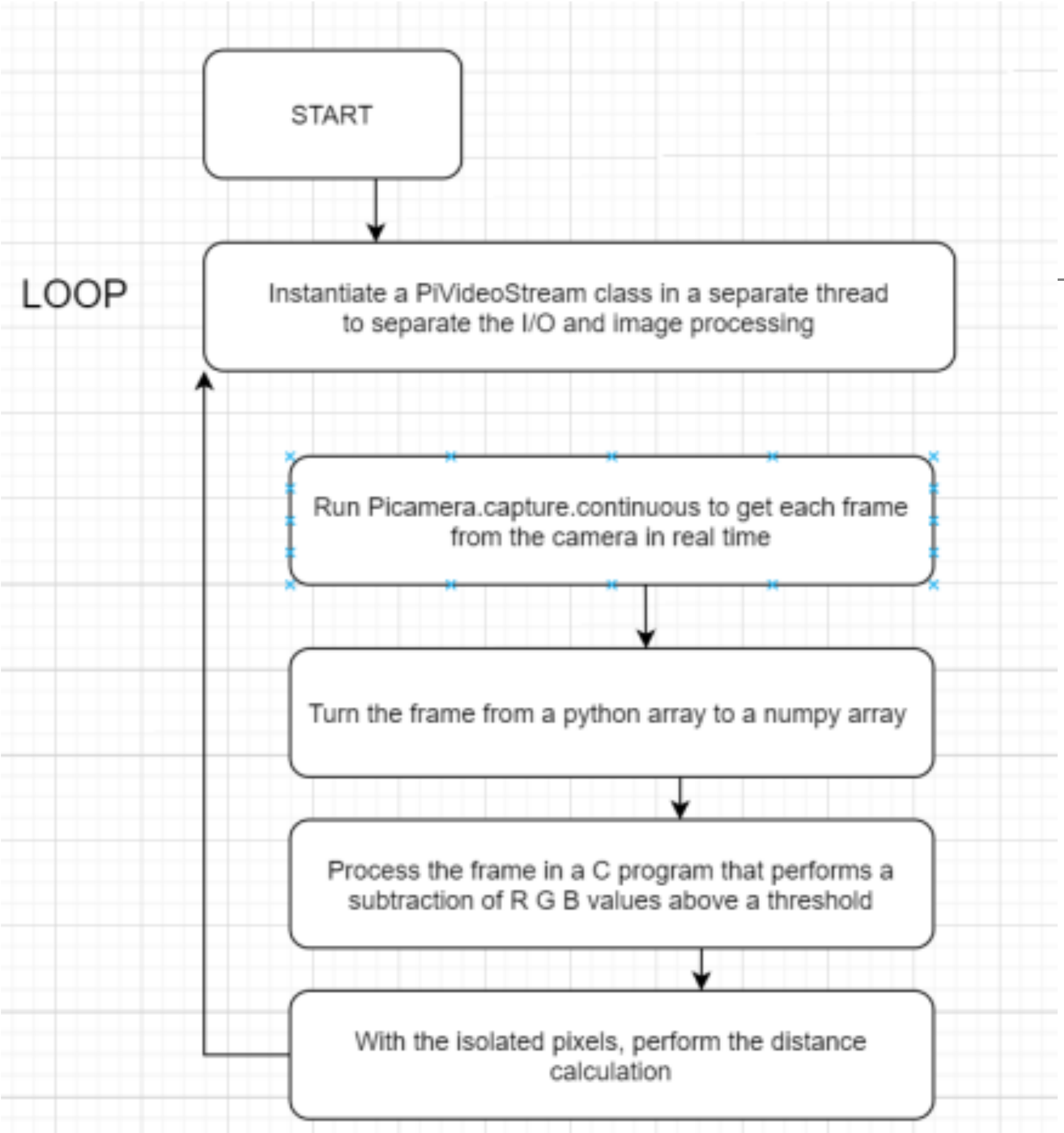
} All of these are constants!

Image Processing - Goal

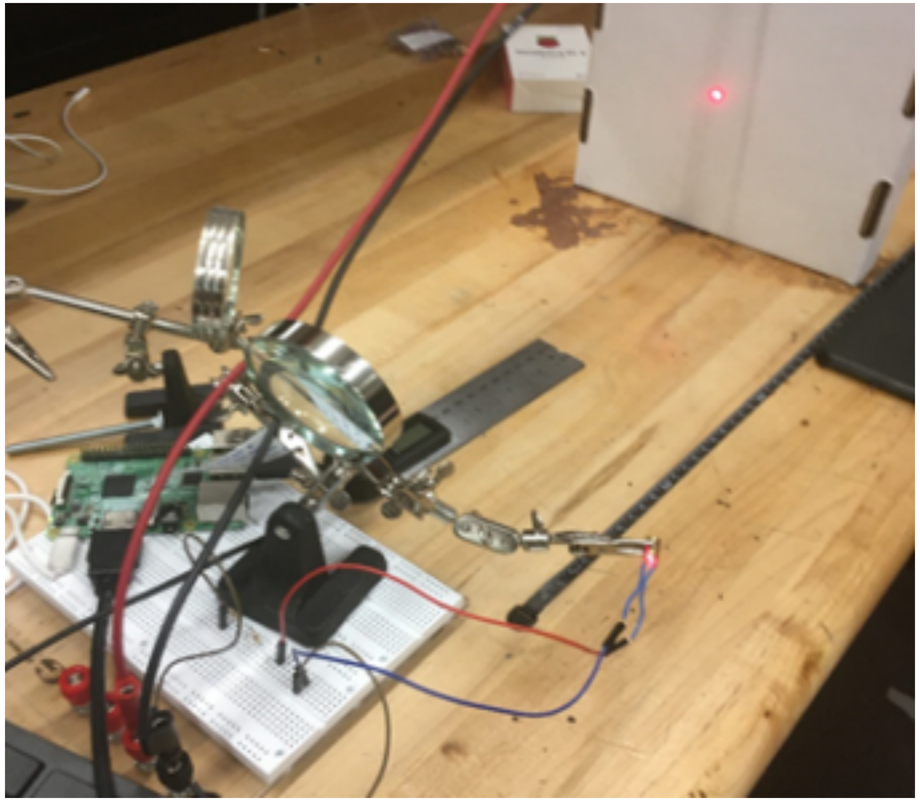


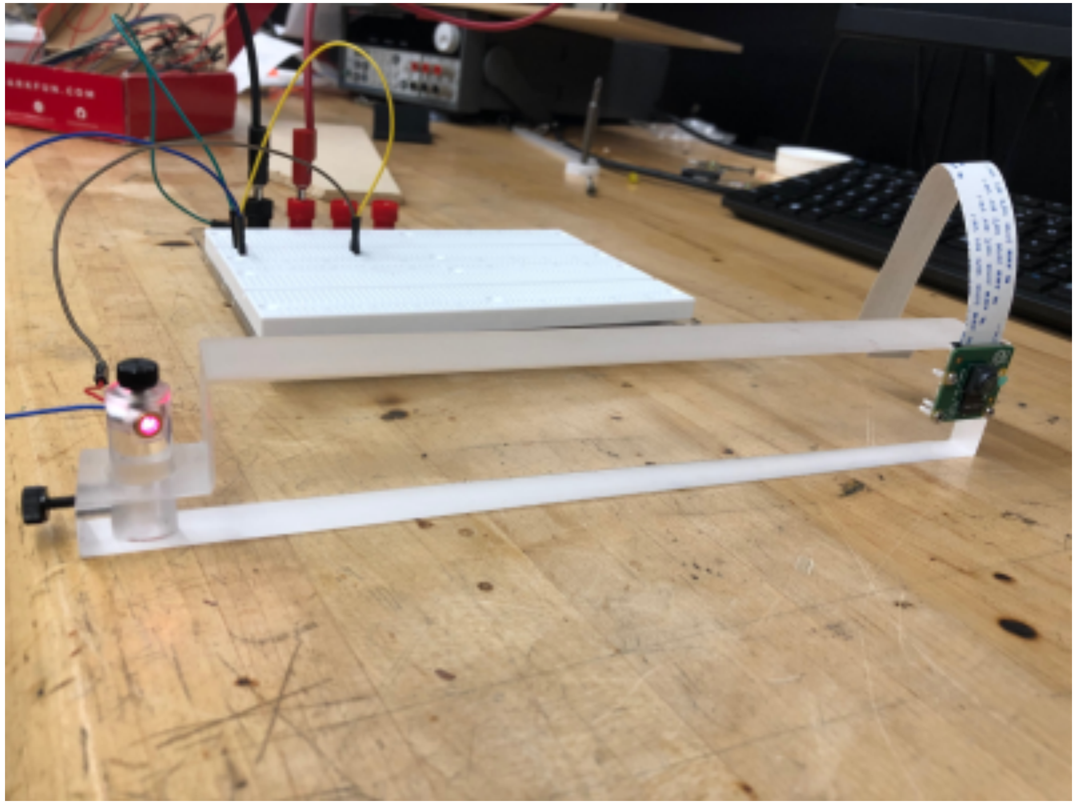
The goal of the image processing is to run an algorithm to detect the laser dot (find the pixels where it is found on the image) on a moving object and then calculate the distance

Image Processing - Basic Algorithm



Distance Sensor Testing



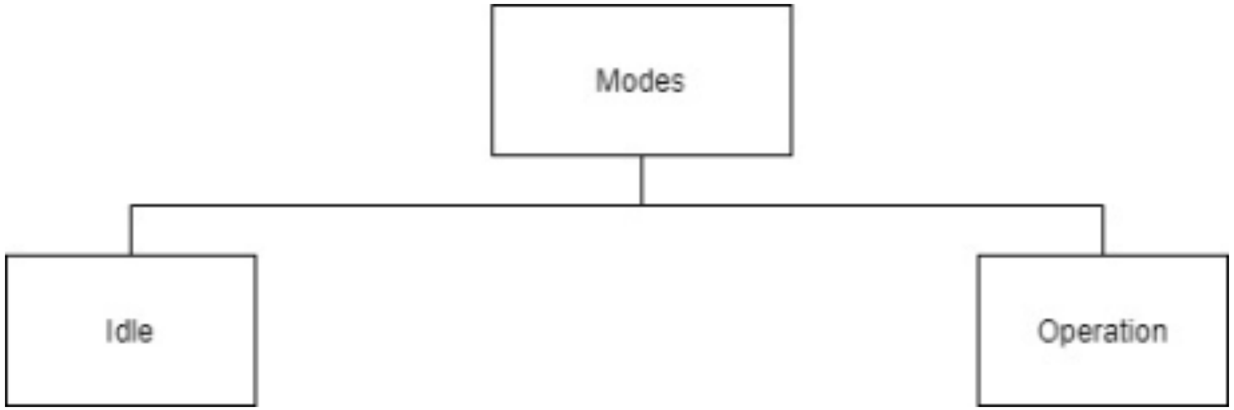


Graphical User Interface (GUI)

Framework Choice

- We have chosen the PyQt framework, with its GUI building tool
- The GUI is the main interface that the user can interact with to modify settings for the entire system

GUI Functionality



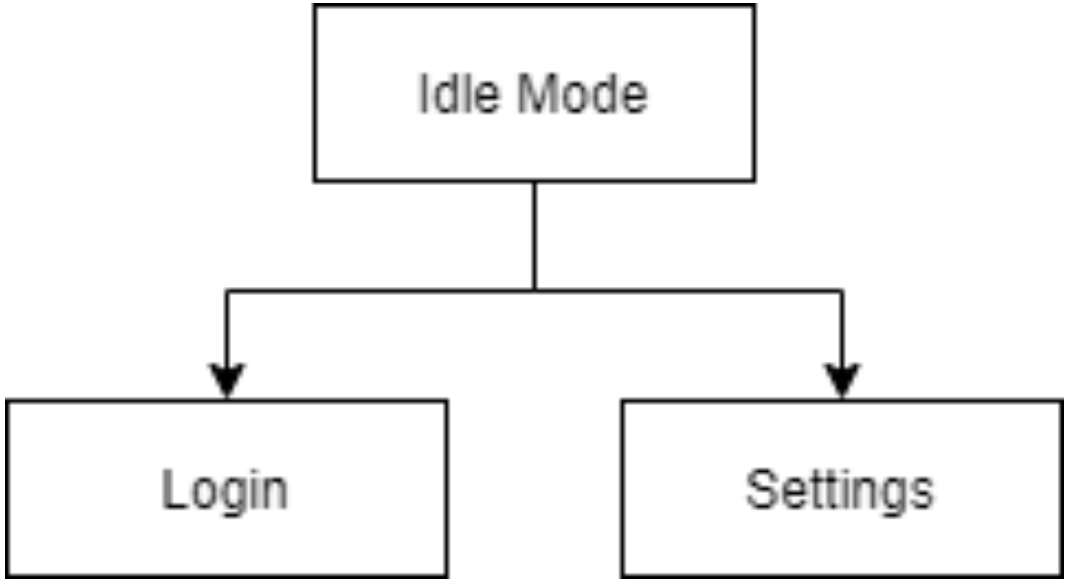
- The GUI has two basic modes of operation:
- An Operation mode that shows the distance between the vehicle and the LED lights as they are attempting to park
- An Idle mode when the user is trying to access the settings

GUI Continued - Mode Switching

- The Raspberry Pi runs two processes simultaneously in idle mode
- The first one runs the camera continuously, polling to check for a distance under 10 feet, while the other performs idle mode operations
- When new distance measurements are detected within a range set by the user, the GUI switches from Idle Mode to Operation Mode
- When distance measurement is either not found or stays about the same for 10 seconds straight, GUI switches from Operation Mode to Idle Mode

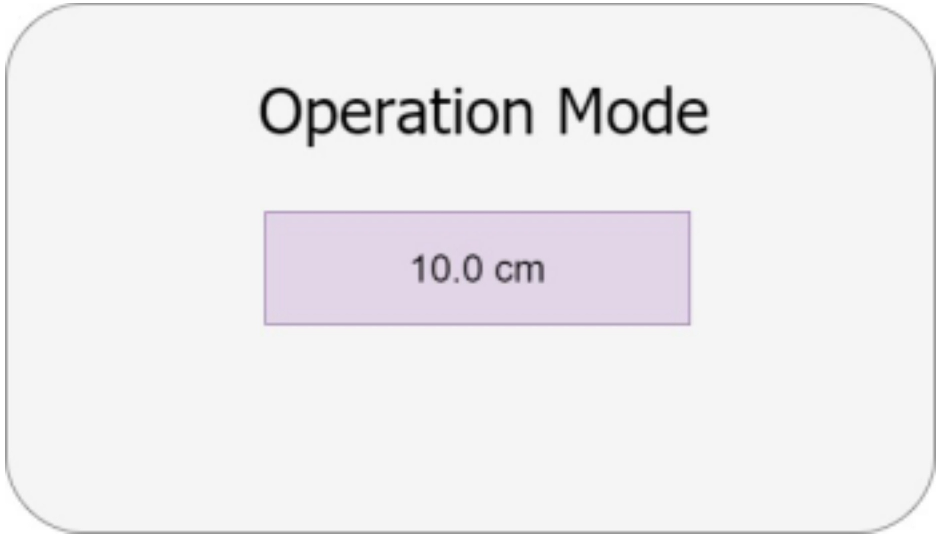
GUI Continued - Idle Mode

- Uses the PyQT framework
- The GUI displays two options to select from in Idle Mode
- It has the option to go to settings or to login
- Contains account and general settings
- Login control functionality such as accessing settings. Can be disabled
- GUI also displays the date and time



GUI Continued - Operation Mode

- Operation Mode is the GUI state when the user is attempting to park
- The GUI displays a numerical value to the user in large text. This value dynamically changes as the user adjusts the position of the vehicle and represents the parallel distance between the user and the LED lights.
- Operation Mode works alongside the lights to give user the feedback visually through the lights or numerically if they prefer. The text is large enough to see and occupies at least 50% of the total GUI screen space. It is read-only.



Light Display System

- Light display will be used to alert users when they are close to proximity to the garage wall
- Green light will be displayed when distance sensor senses the car approximately 10 feet away
- Yellow light will be displayed afterwards when distance sensor senses car around 3-5 feet
- Red light will signal the driver to stop when their car is 2-3 feet away
- Light system will have default setting, but can be changed based on user configuration in settings from the GUI system
- The closer the car gets to the garage wall, the faster the light display will blink

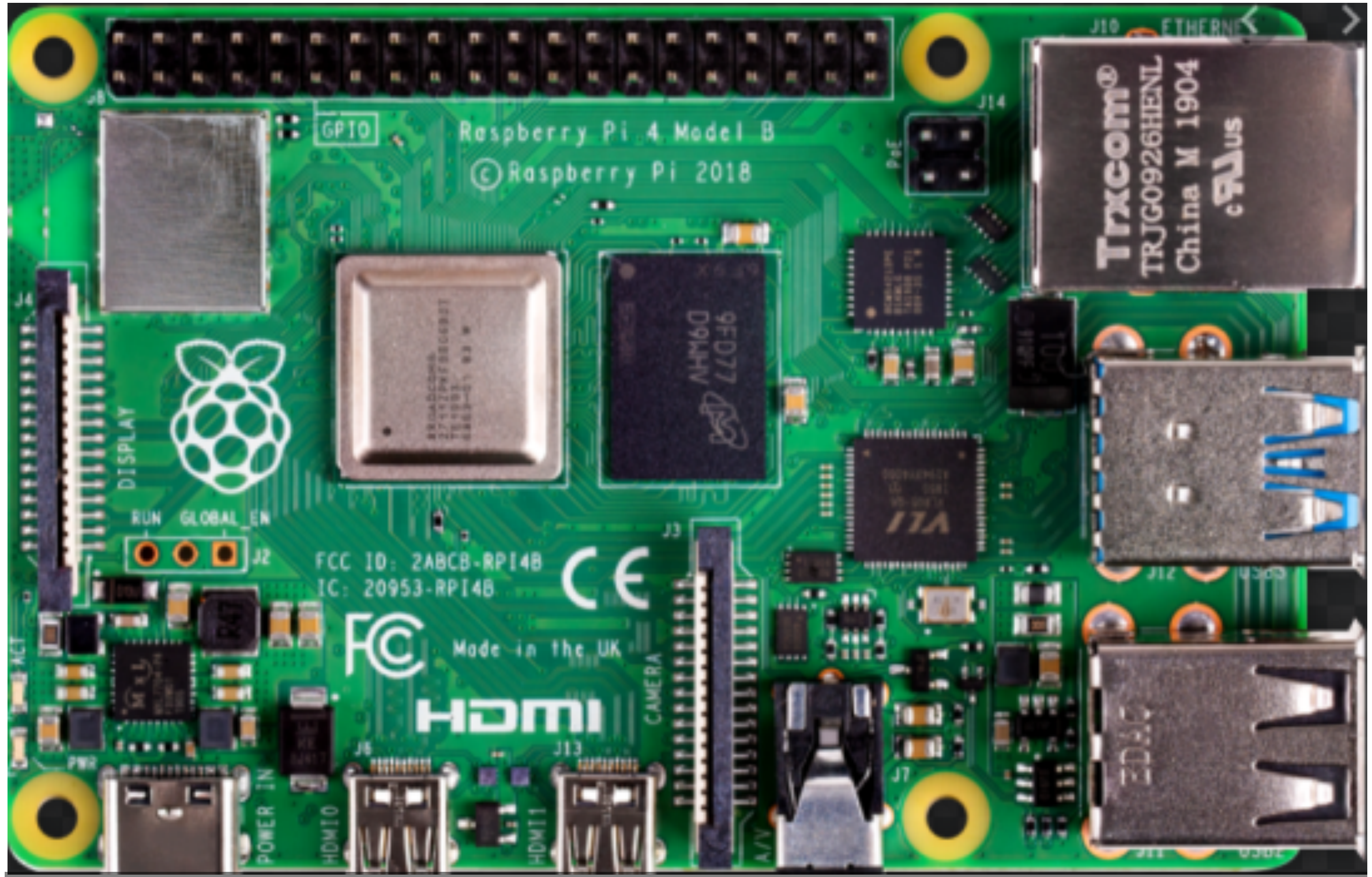
Subsystem Performance

- The distance sensor is able to perform measurements from 1 to 10 feet within an error of +/- 5 cm ✓
- The distance sensor fails to detect the laser or gives inaccurate measurements for highly reflective surfaces or very dark surfaces ✗
- LED lights are able to accurately and quickly respond to all distance measurements ✓
- Original enclosure plan was cancelled due to coronavirus closing CREOL machine shop ✗
- GUI performs switching seamlessly and supports most features ✓
- Account does not save settings ✗

Components

Microcontroller Comparison

	Arduino UNO R3 Clone	MSP430FG4618	Raspberry Pi Model 3 B
Operating Voltage	5V	1.8-3.6V	5V
Max Clock Frequency	16 MHz	8 Mhz	4-core @ 1.2 GHz
Memory	2 KB SRAM	8KB RAM	1 GB RAM
Communication Protocols	UART	UART, SPI	USB, WiFi, BlueTooth
GPIO Pin Count	14	80	40
Average Power Consumption	~.5 W	880 uW	2.5 W
Supported Devices	None	None	PiCamera, PiDisplay
Primary Programming Environment	Arduino IDE in C Language	Code Composer Studio in C Language	Python IDLE
Price	$10	$117	$~60 + accessories



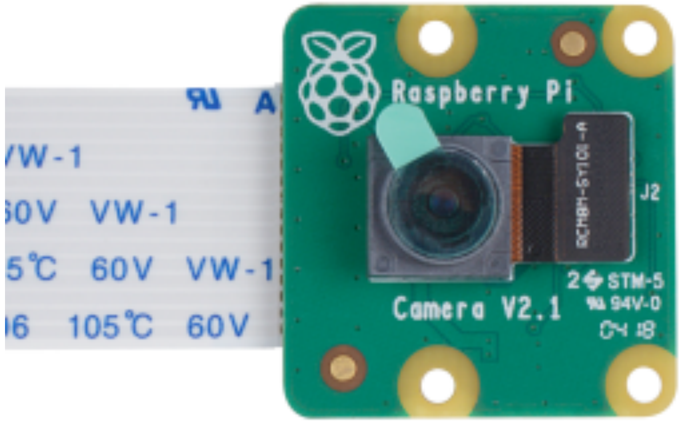
Raspberry Pi 3 Model B

Microcontroller

- 4-core 1.2 GHz CPU
- 1 GB RAM
- USB communication between devices
- 40 GPIO pins
- Native compatibility with external cameras and LCD displays
- Power Requirements: 5VDC, 3A

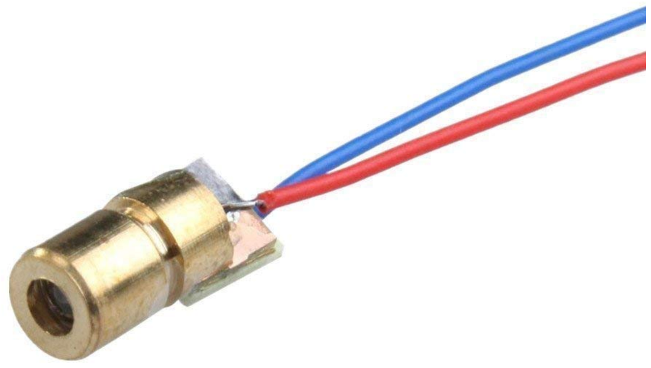
Camera Choice

- Final camera selection was **Raspberry Pi Camera Module V2** due to its easy integration with our Raspberry Pi microcontroller
- Supports many resolutions and video capture as well as individual frame capture
- Python package allows for extracting stream of frames for image processing
- Power is fed from Raspberry Pi



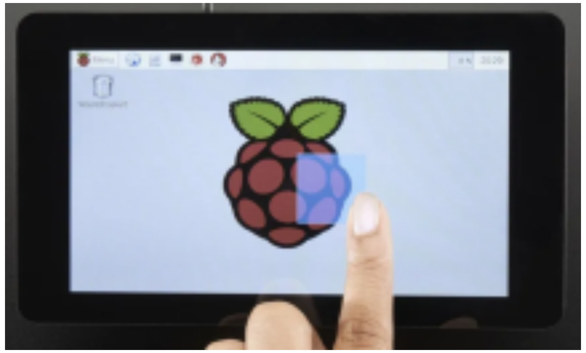
Laser Diode Choice

- **HiLetGo Red (650 nm) Dot Laser Diode**
- Working Voltage: 5 V
- Operating Current: < 20 mA
- Output Power: 5 mW
- Class IIIa laser, safe for skin exposure and momentary/accidental eye exposure
- Chosen because it was cheap and contained a package of 10 laser diodes which could be used for testing



Display

Raspberry Pi 7” Touchscreen Display

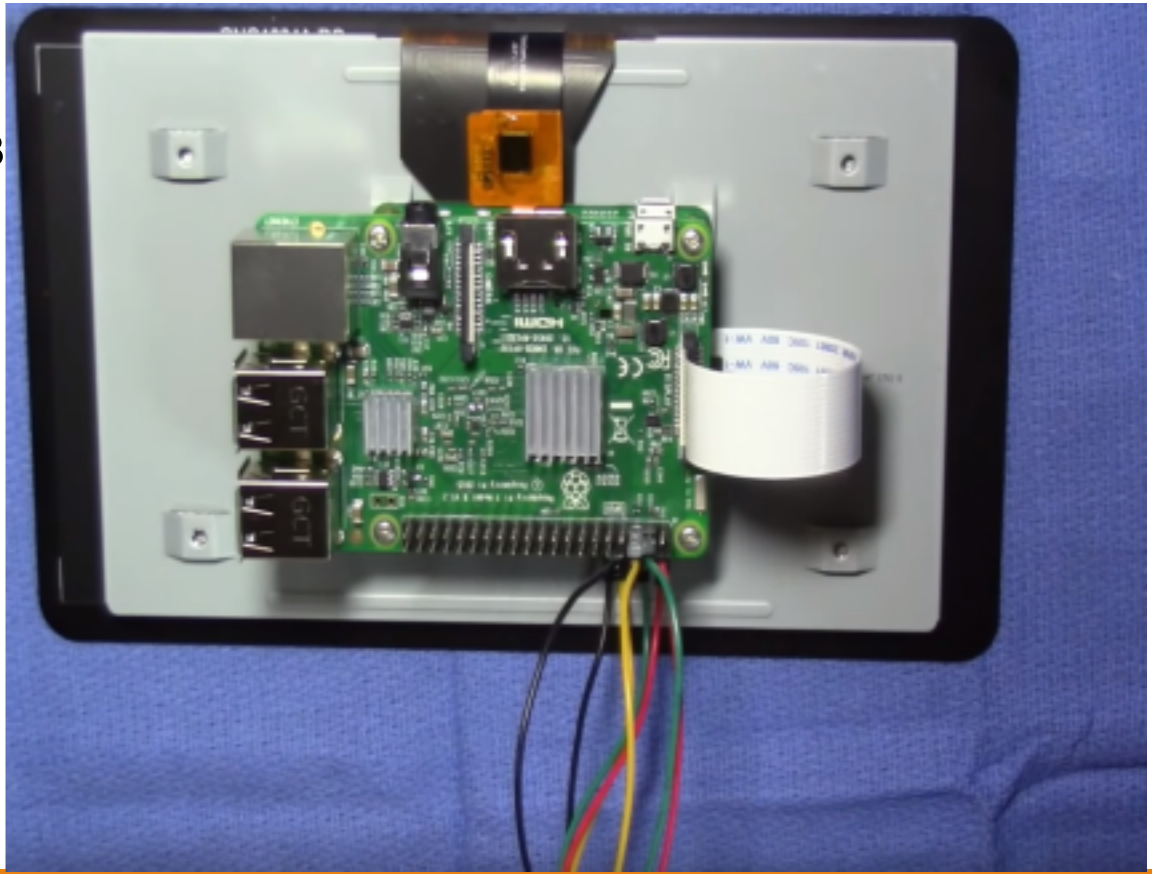


touch screen capability

native resolution of 800x48 pixels

viewable screen size is 1.55mm x 86mm

Mounted Raspberry Pi to Display



Example GUI

The purpose of the touch display screen is to give the user an interface to control the system via a Graphical User Interface

LED Lighting Comparison

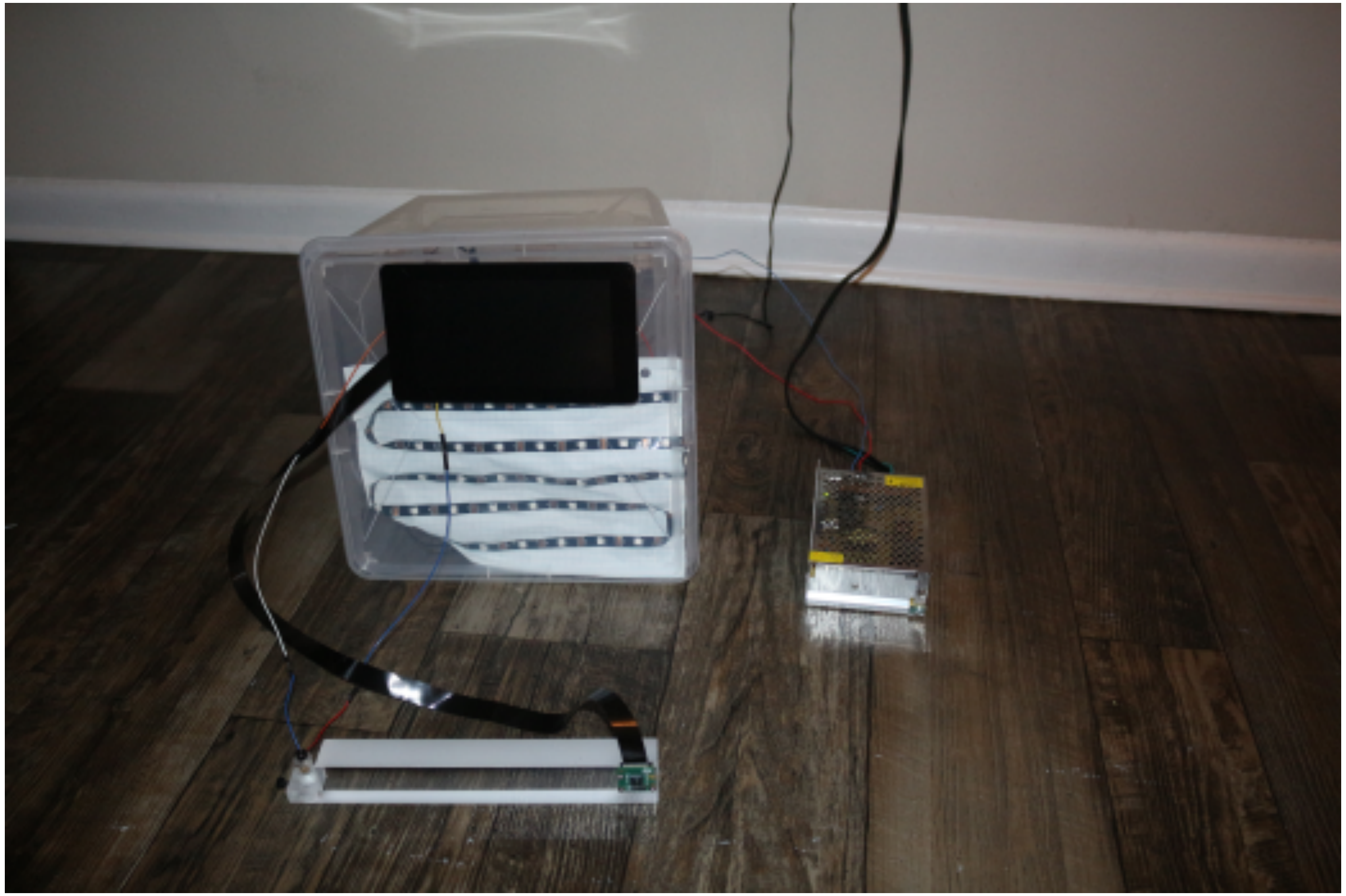
Type	Cost	Input Power	Controller
Fiber Optic Lighting	>$80	12 Volts	N/A
Non Addressable LED lights	>$30	12 Volts	Cannot be controlled with MCU
Addressable LED WS2801	>$20	3.3 - 5.5 Volts	14 PIN configurations
Addressable LED WS2811	>$20	12 Volts	8 PIN configurations
Addressable LED WS2812B	>$20	5 Volts	4 PIN configurations

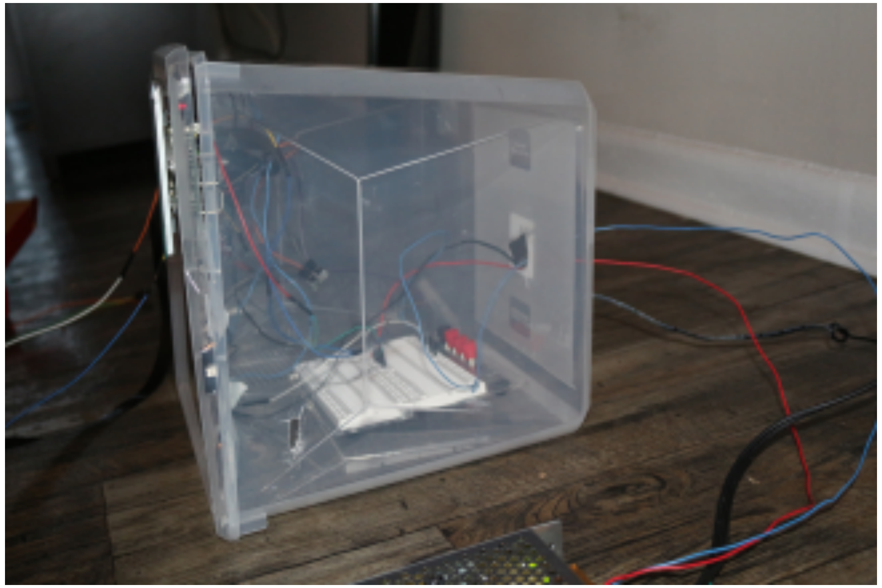
LED Lighting Choice

- We are using WS2812B individually addressable LED lights (Neopixel)
- LEDs will display specific colors/turn lights on or off depending on the distance measurements
- Each LED will be able to be individually configured and addressed in programming
- Power requirements: 5.5V and 1.67A/m
- Will need a Logic Level Converter
  - Converts the 3.3V to 5.5V

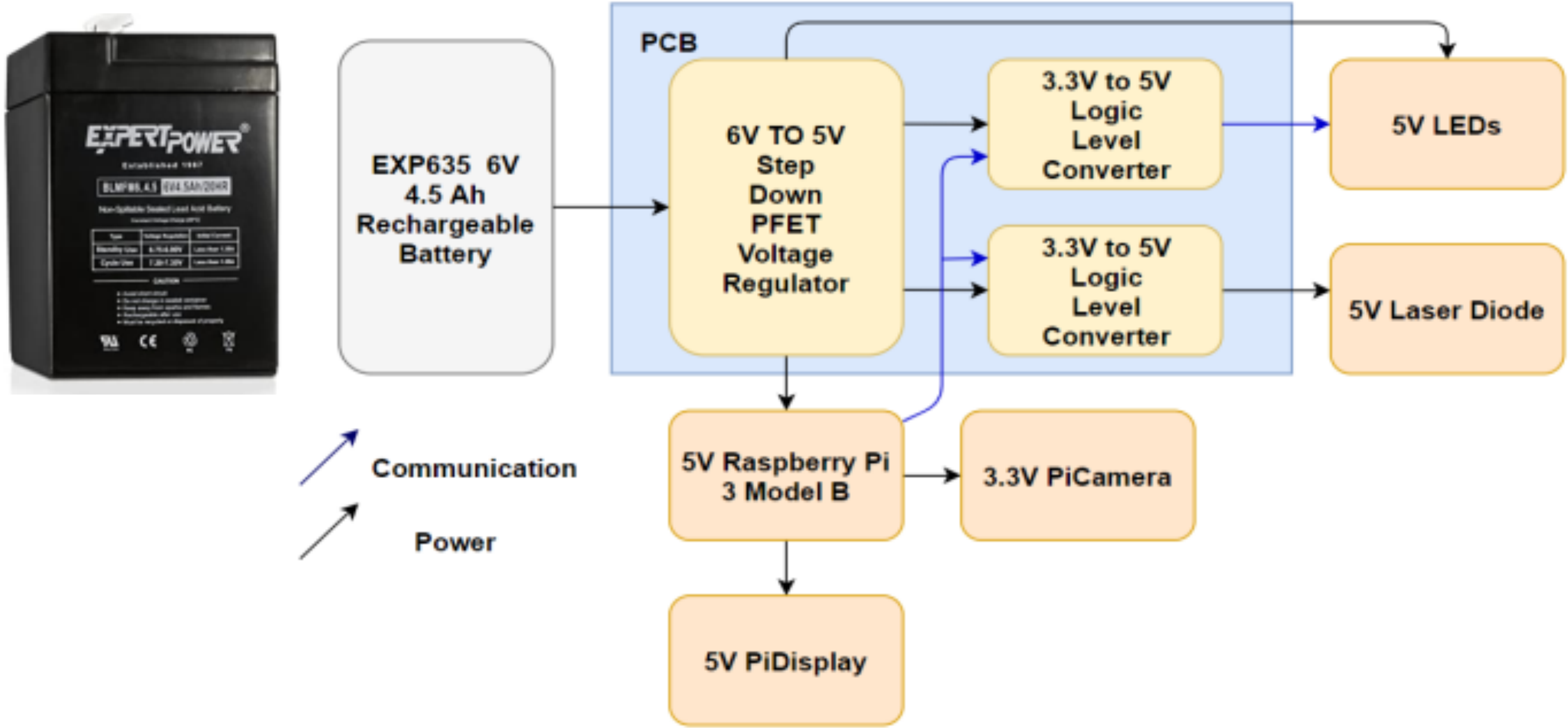
Enclosure

- Intended to be designed and manufactured from acrylic at CREOL machine shop, and include all components including the distance sensor
- Instead, we made a cheap enclosure from a transparent plastic storage container
- Components mounted inside and on the surface of the storage container
- Distance sensor separate from enclosure to ensure accurate results
- During operation the distance sensor will be some distance below the rest of the system, lined up with the front bumper of the car

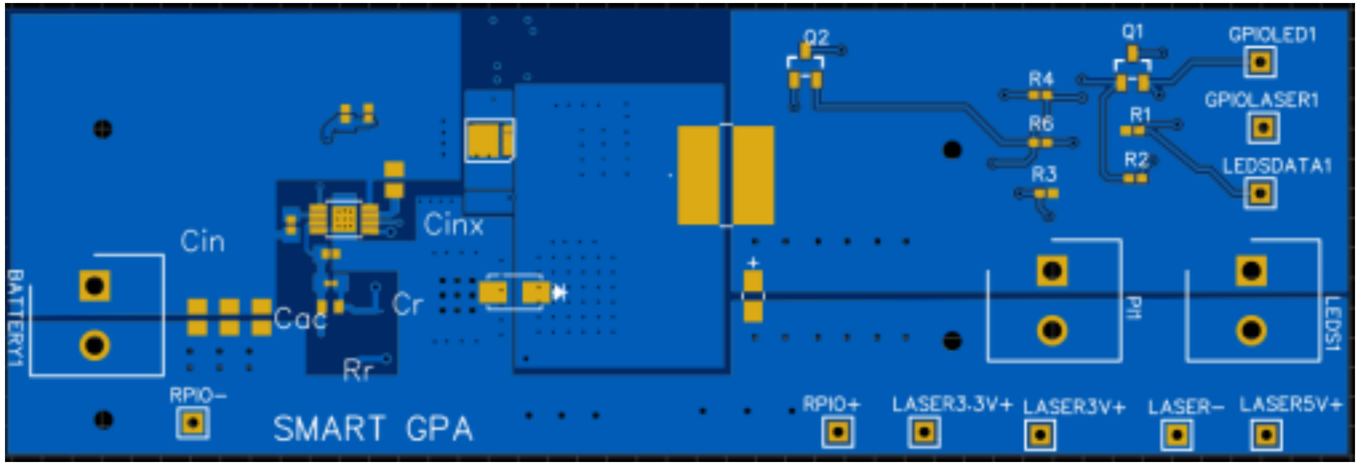


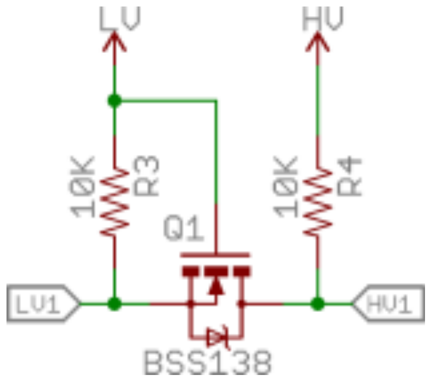


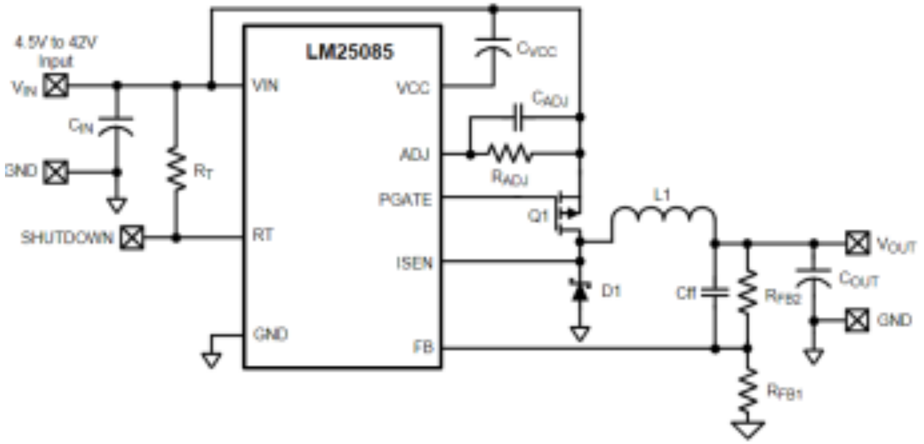
Overall PCB Block Diagram (Intended)



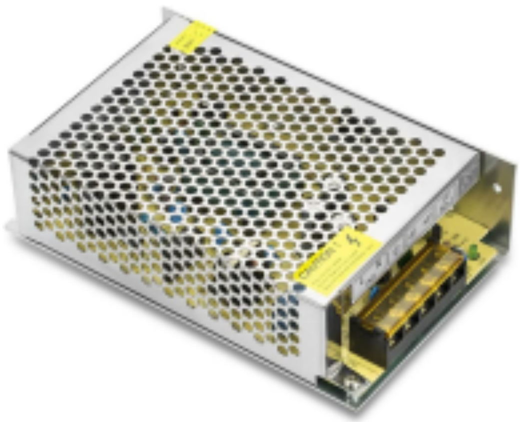
PCB Final Design

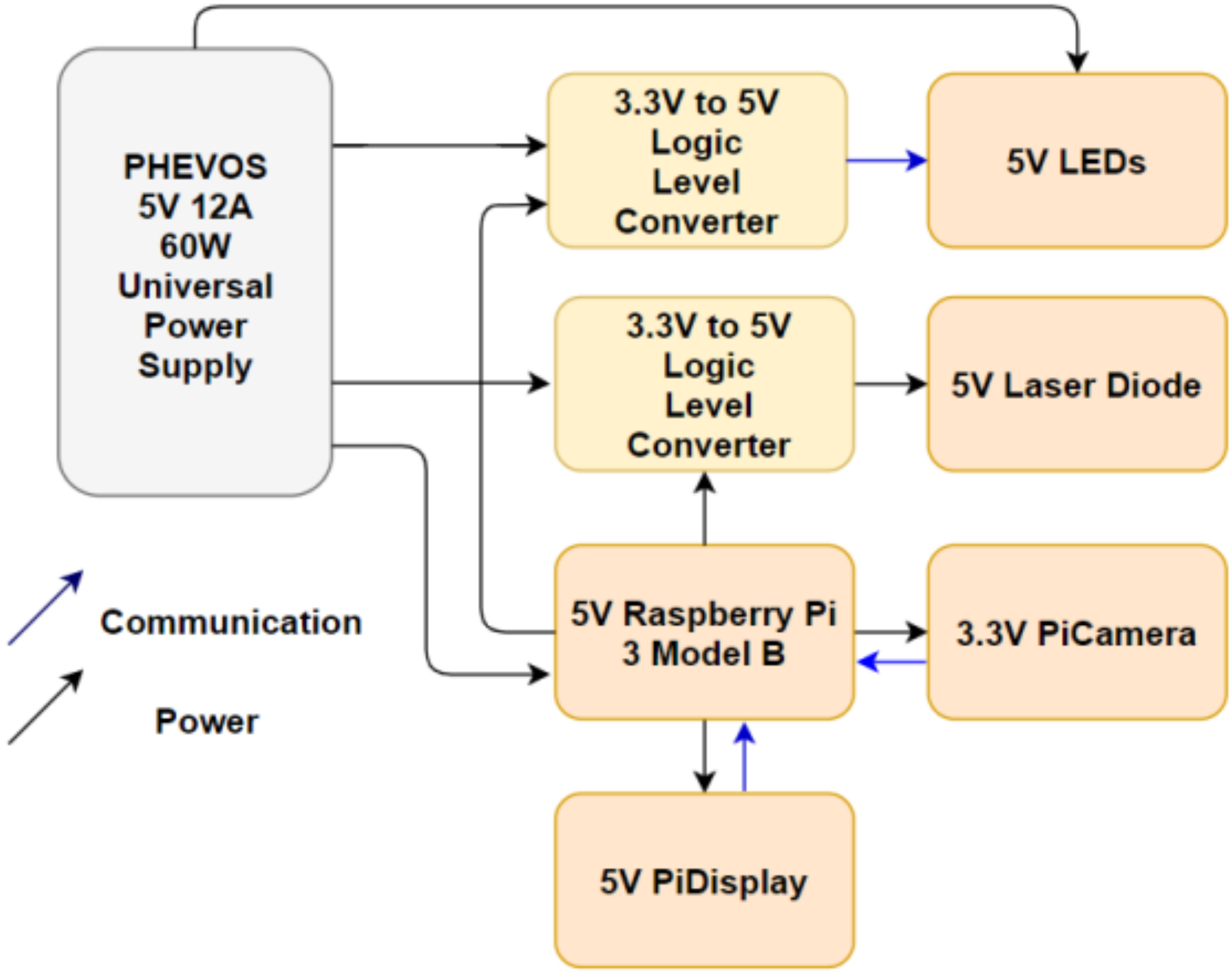






Current Block Diagram





Administrative Content

Cost Estimation

Item	Quantity	Cost Estimate
Enclosure	1	$5
LED lights	1	$20
PCBs & Components	1	$100
Raspberry Pi	1	$50
Raspberry Pi Camera	1	$25
Red Laser Diodes	10	$10
Battery	1	$20
Touch Screen Display	1	$65
Total:		$285

Budget : $500

Project Difficulties

- Lack of PCB and soldering skills and resources
- Controlling 5V components with 3.3V logic
- Python array manipulation is slow, so measurement rate is hard to improve without using code in a different language
- Raspberry Pi Camera Module has slow frame capture at the resolution we needed, limiting our measurement rate
- Laser triangulation method is difficult to implement at the range we desired, on the surfaces we needed to measure on
- Our display was broken two days before being able to film and due to COVID-19, we are unable to replace it. We resorted to an HDMI display for the demo.

Work Distribution

	Primary	Secondary
PCB/Power	Jennifer Castillo	Nicholas Zollo
Light Display	Daniela Otero	Jennifer Castillo
Laser Distance Sensor	Nicholas Zollo	Jorge Dardon
GUI Display	Jorge Dardon	Daniela Otero

Cancelled Features

- Due to the coronavirus separating our team members, we were not able to implement many of the features we planned for the Smart GPA
- We planned a light detection system using dual photodiodes, to alert the user if their car lights were left on after parking
- We also planned additional tuning features to change the internal variables of the SMART GPA from the touch display
- Complete PCB and enclosure also had to be cancelled due to needed facilities closing down
- Toggleable camera view on the touch display was cancelled early on due to incompatibility with Python package needed for the Neopixel LED lights

Thank you for watching!