GROUP 23 Military Surveillance Robotic Vehicle

Ryan Hromada - EE

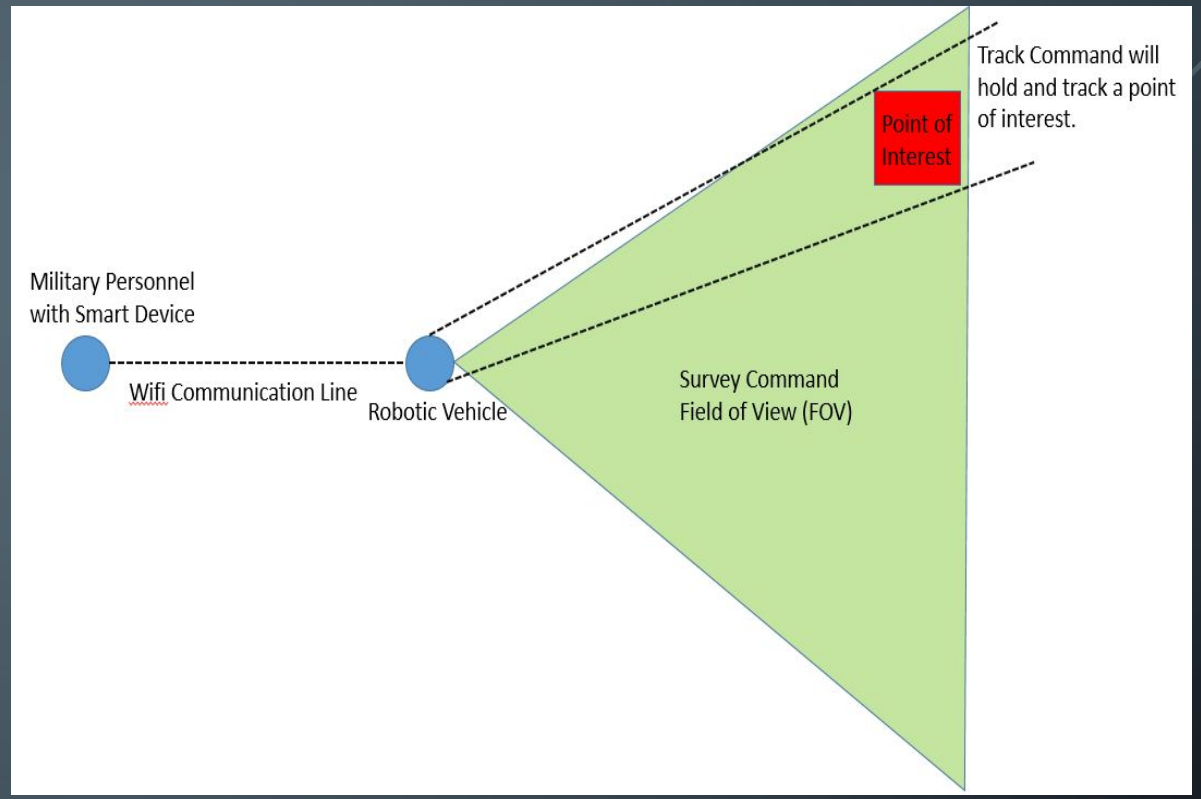
John Baumgartner - EE

Austin King - CpE

Kevin Plaza - CpE

INTRODUCTION

- Autonomous tracking vehicle
- Commands:
  - Basic Movement Commands
  - Wander (Obstacle Avoidance)
  - Survey/Track (Image Detection and follow)
- Commanded through android app



GOALS & MOTIVATION

- Create a new design that addresses problems in existing products
  - Portability
  - Simplicity
  - Usability
- Make ground surveillance safer and more effective

SPECIFICATION SUMMARY

Size	No Larger then 2’/2’/2’ (L/W/H)
Control	Accept functional voice commands via Wi-Fi
Range	25 feet from control device
Vision	Minimum of 120-degree scanning in front of vehicle
Cost	Less then $450 per vehicle
Device	Useable on an Android devices
Function 1	Obstacle avoidance through automatic collision detection.
Function 2	Track POI for at least 30 seconds
Function 3	Wander mode to find a POI

EXISTING SYSTEMS

- Throwbot XT vehicle made by ReconRobotics

- California Institute of Technology vehicle
  - Similar function to our design
  - Large size and weight, portability issues
  - Analog Controls
  - More rugged

TEAM MEMBER ROLES

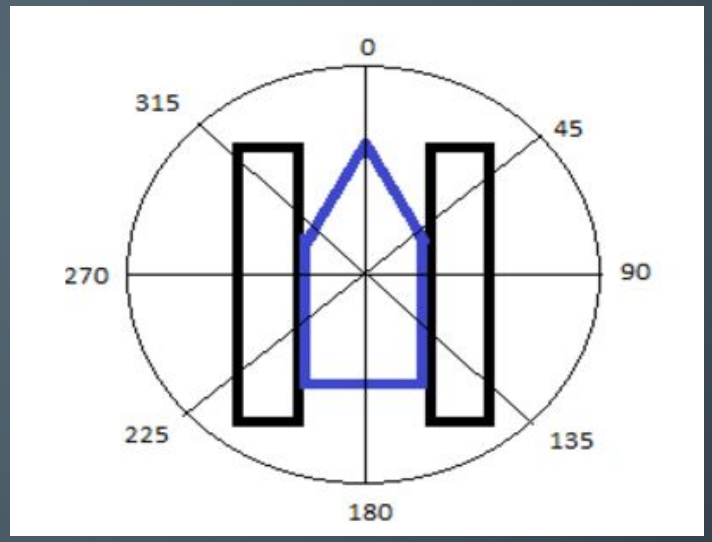
- Ryan Hromada: EE role- PCB design, component interfacing, electronics testing
- Adam Baumgartner: EE role- PCB design, servo control and interface, vehicle assembly
- Austin King: CpE role- Software design and testing
- Kevin Plaza: CpE role- Software design and testing

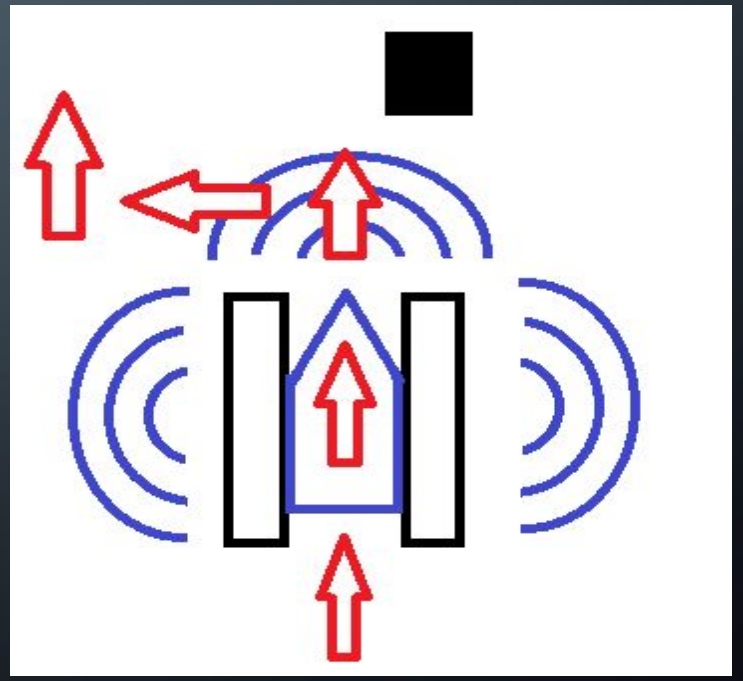
HARDWARE OVERVIEW

- Sensor Array
- Accelerometer
- Camera & Bluetooth
- Attiny & I2C
- Encoders
- Movement

SENSOR ARRAY

- How will the vehicle movement work?
- How can the device know what's in front of it?
- How does the device know where it's been?
- How does the device know what a left turn is?



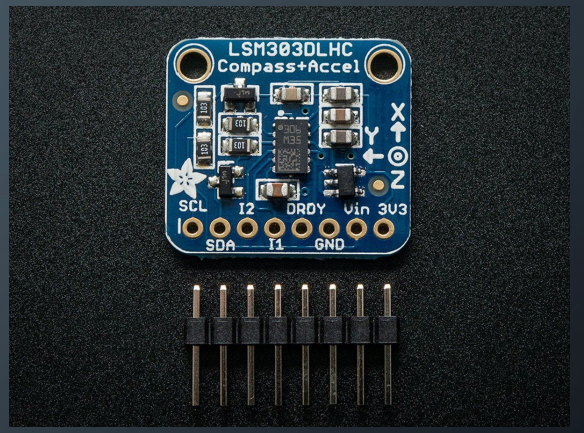


ULTRASONIC SENSOR ARRAY

- Inexpensive components
- Low voltage tolerance
  - Difficult to interface with Raspberry Pi
- Range of up to 3 meters
  - Effective range of 1 meter
  - Not clean data
- Can combine trigger and echo pins to make an array

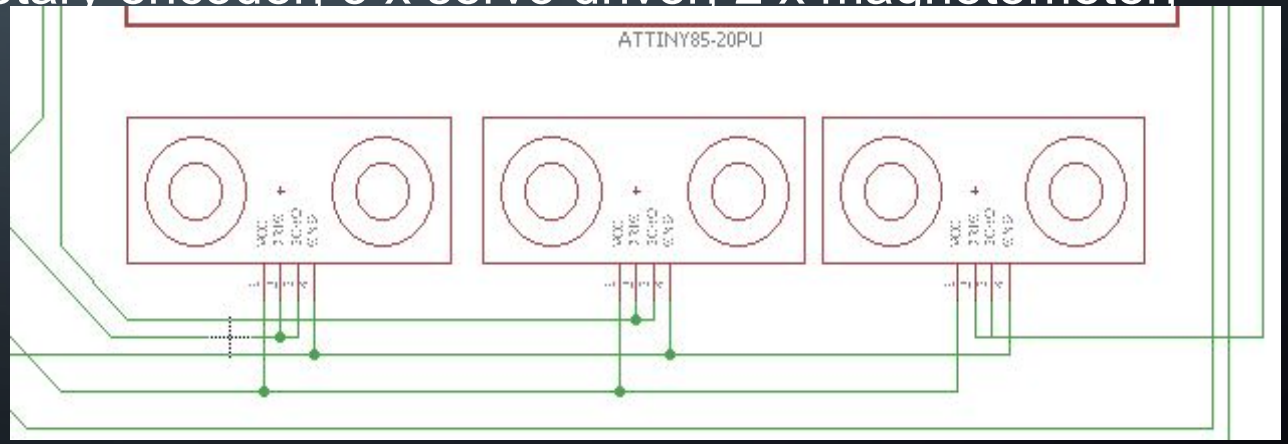
ACCELEROMETER, CAMERA, BLUETOOTH

- Utilizes built in Bluetooth and Wi-Fi functionality
  - allows for easy programming and communications
- High enough image resolution for image detection
  - Plug and play; 1080p; 8 megapixel
- Small form factor accelerometer
  - i2c, magnetometer
  - flexible voltage input



COMMUNICATIONS TO THE MICROCONTROLLER

- How will the sensor array communicate with the micro controller?
- What voltage limitations are there?
- Reduce the number of GPIO ports used
  - 2 x ultrasonic sensor, 1 x rotary encoder, 5 x servo driver, 2 x magnetometer, Power and Ground,
  - 14 minimum GPIO pins



INTER-INTEGRATED CIRCUIT (I2C)

- Unique address for each component
- Serial based communications
- Industry standard in many microelectronics
- Reduces the communication lines
  - 2 minimum GPIO Pins

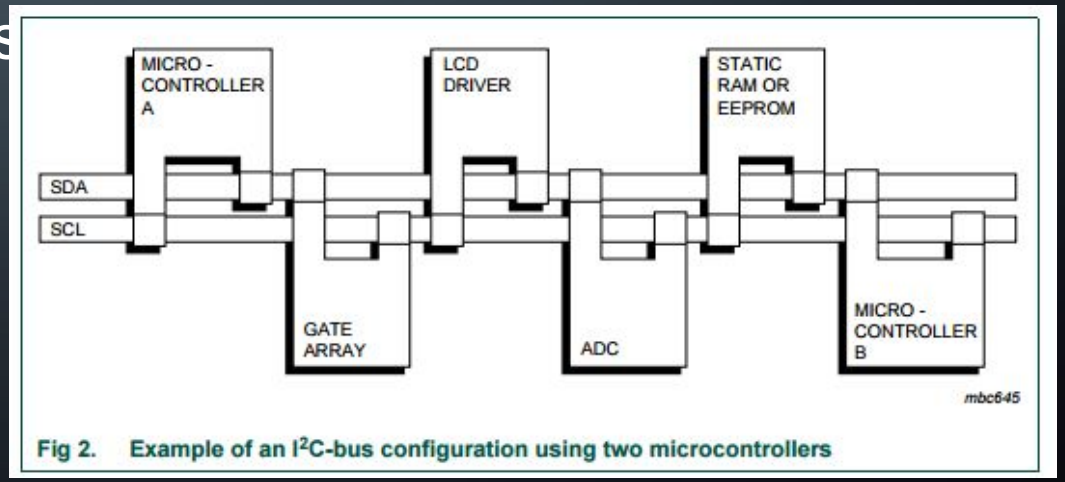
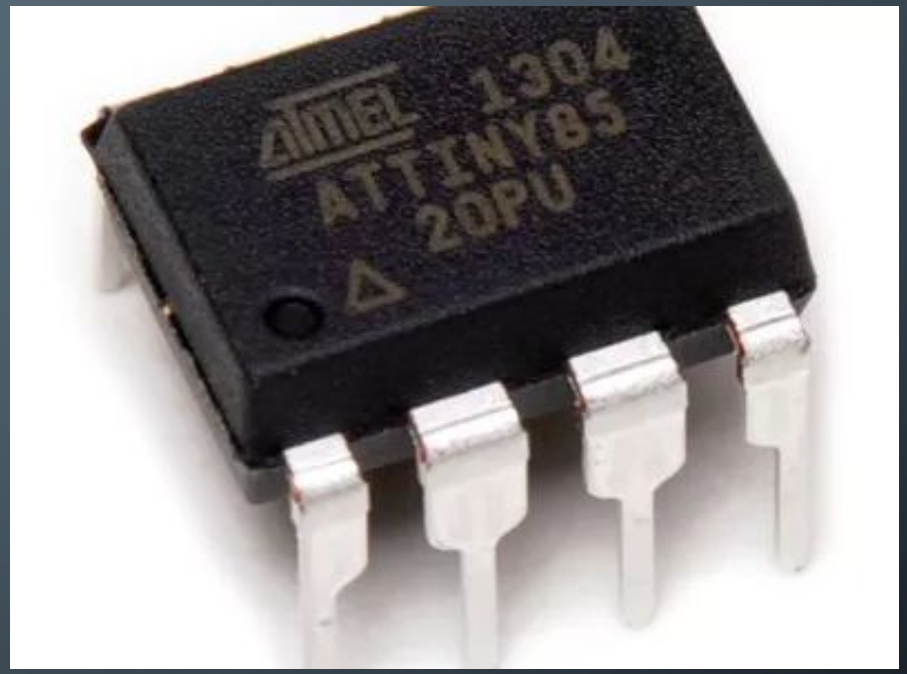


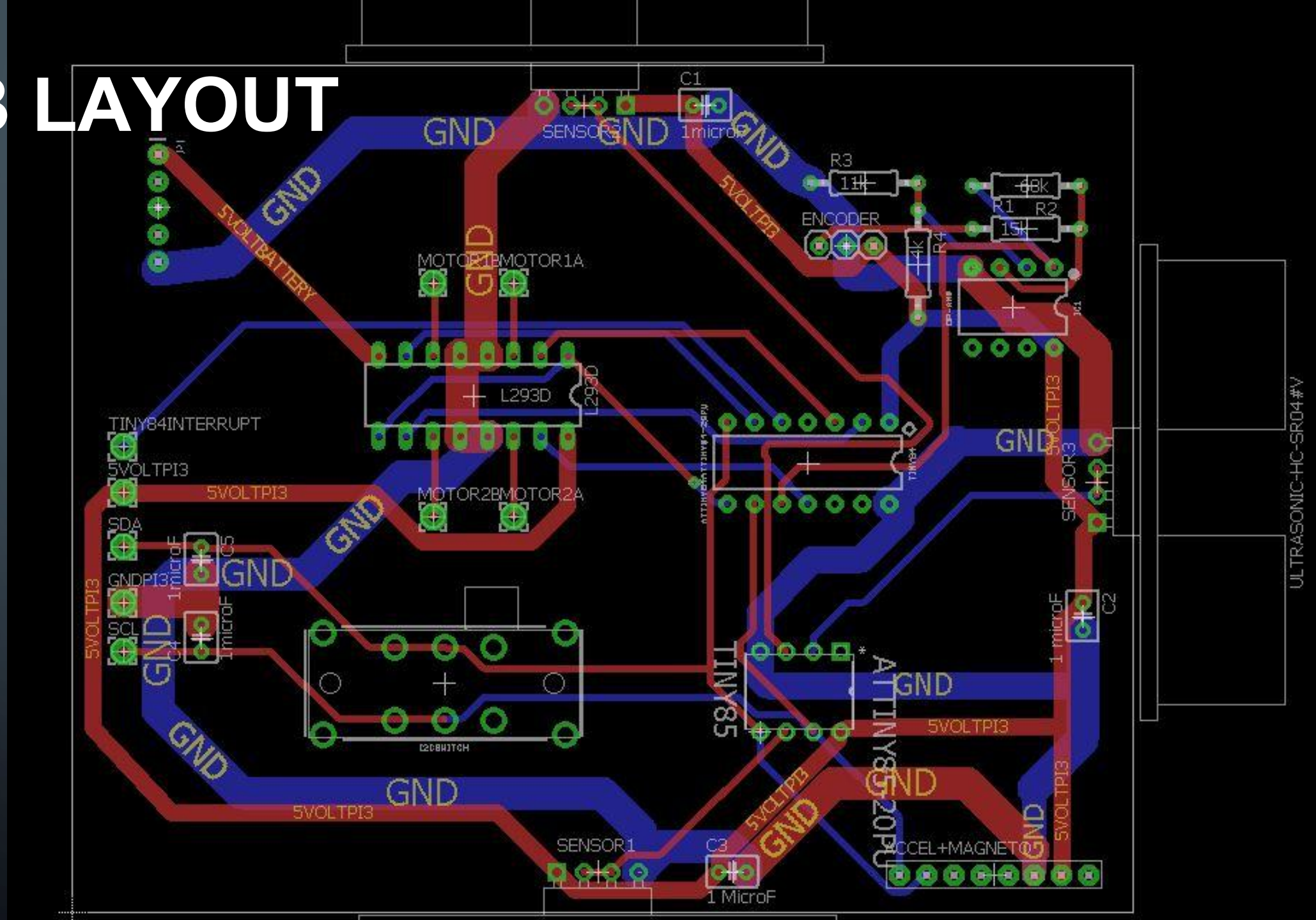
Fig 2. Example of an I²C-bus configuration using two microcontrollers

ATTINY85 & I2C

- Low Power Consumption
  - 300 Microamps
  - Large operation voltage
- 6 programmable pins
  - 2 x I2C Lines
  - 3 ADC lines
  - 1 Reset line multi use line
- Slave-Master Ability for offloading code

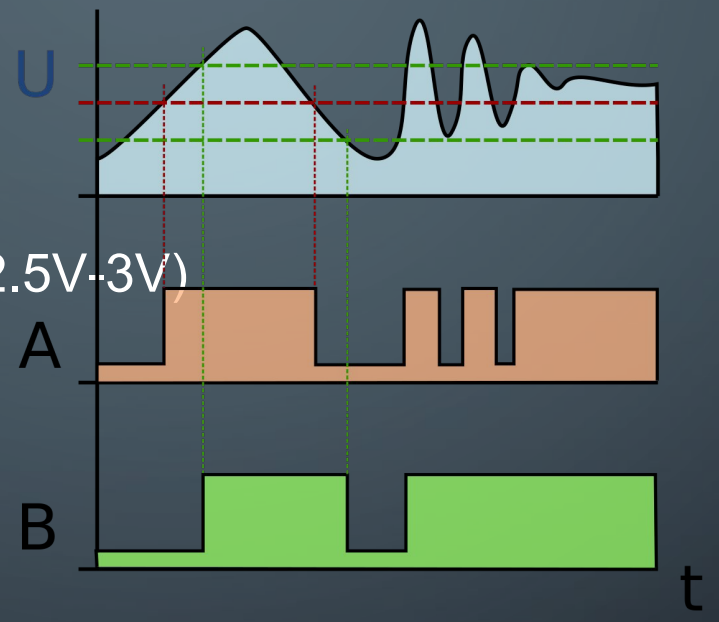


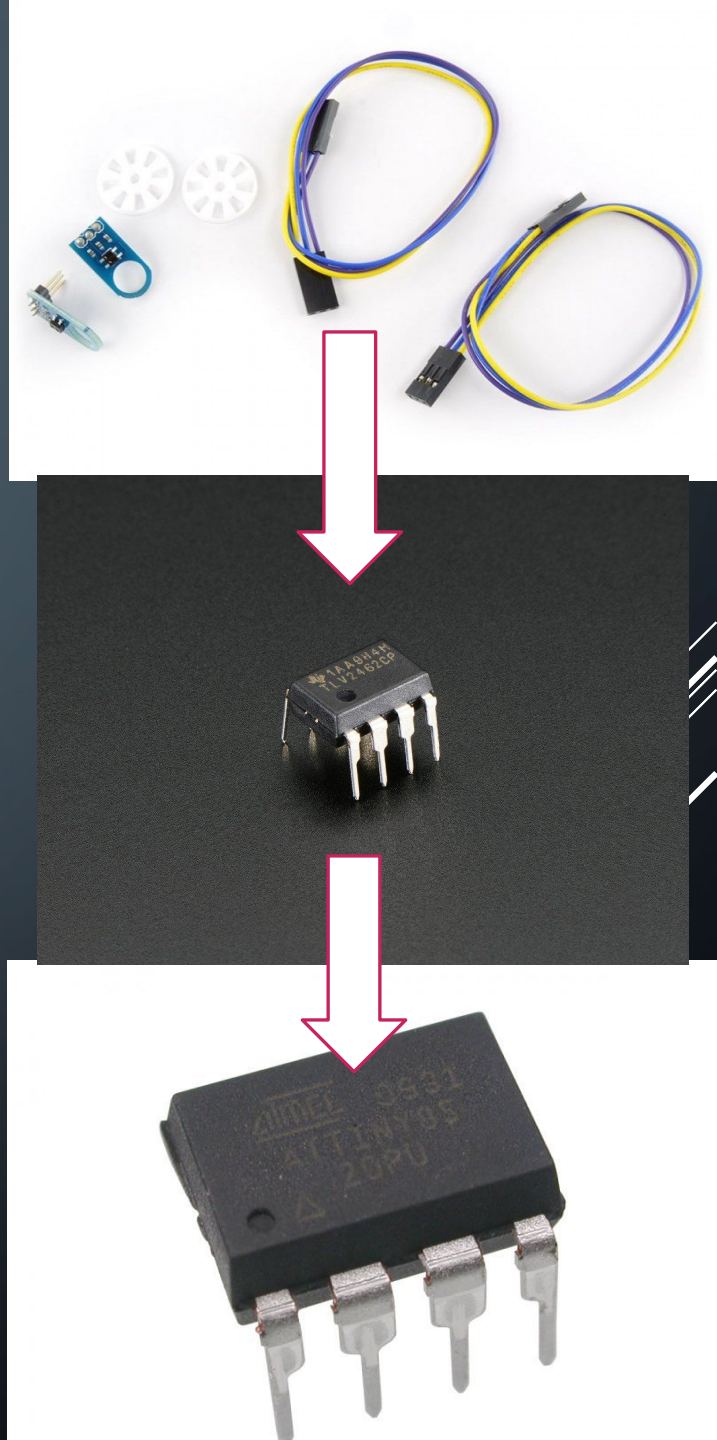
PCB LAYOUT


GND
SENSOR2
C1
1microF
5VOLTBATTERY
R3
11K
ENCODER
R1
R2
68k
15K
R4
4K
MOTOR1BMOTOR1A
L293D
TINY84INTERRUPT
5VOLTPI3
SDA
GNDPI3
SCL
MOTOR2BMOTOR2A
C5
C4
C2
SENSOR3
TINY85
ATTINY85-20PU
ACCEL+MAGNETO
SENSOR1
C3
1 MicroF
ULTRASONIC-HC-SR04#V

ENCODERS

- Analog Rotary Encoders
  - Output High (3.5V-4V); Low (2.5V-3V)
- Schmitt Trigger
- ATtiny85
  - Analog To Digital Conversion
  - Communication

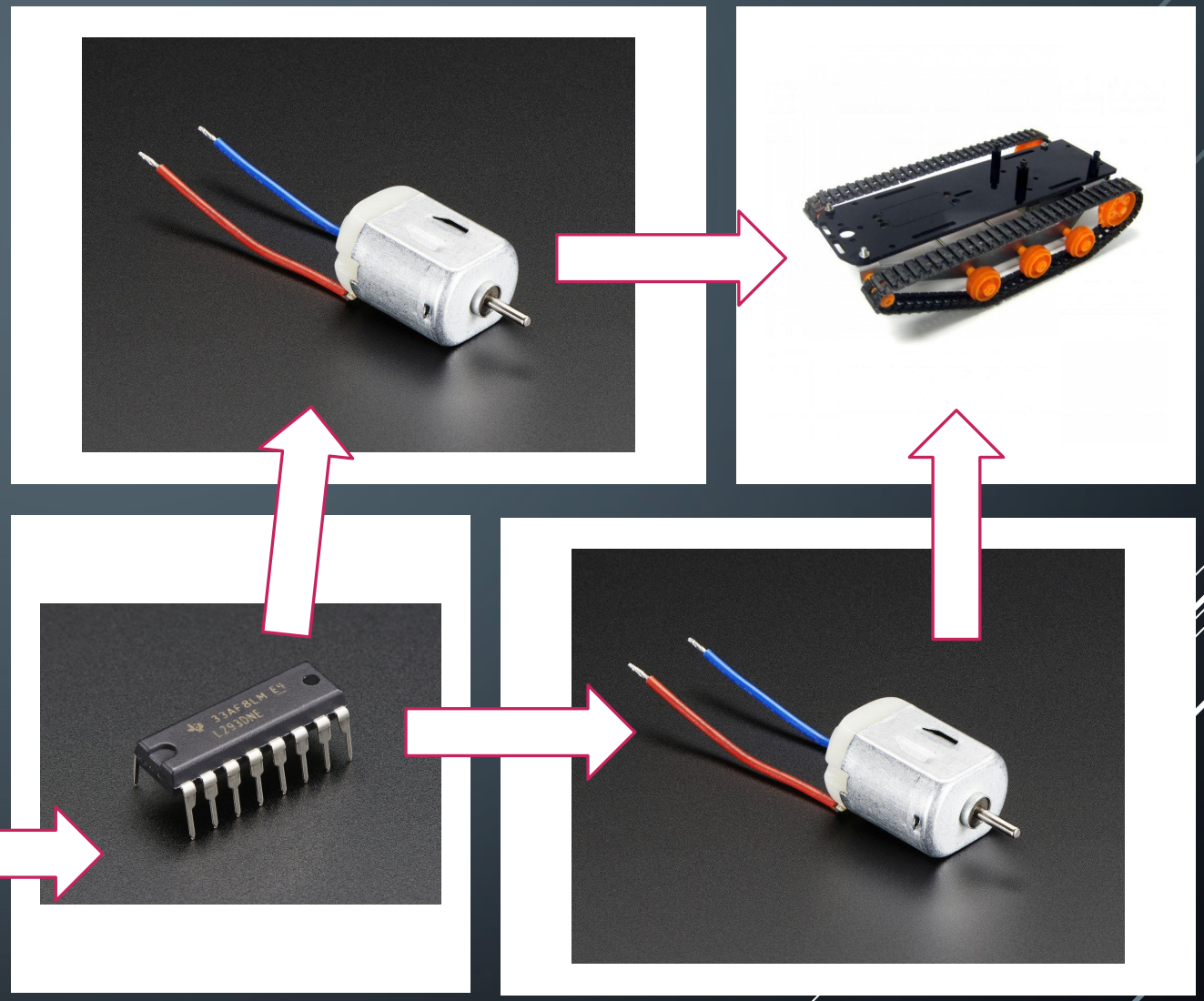




MOVEMENT

- L293D Motor Driver
- Brushed DC Motors- 130 Size
  - Noisy; Current Requirements
- Tracked Robot Chassis

Microcontroller

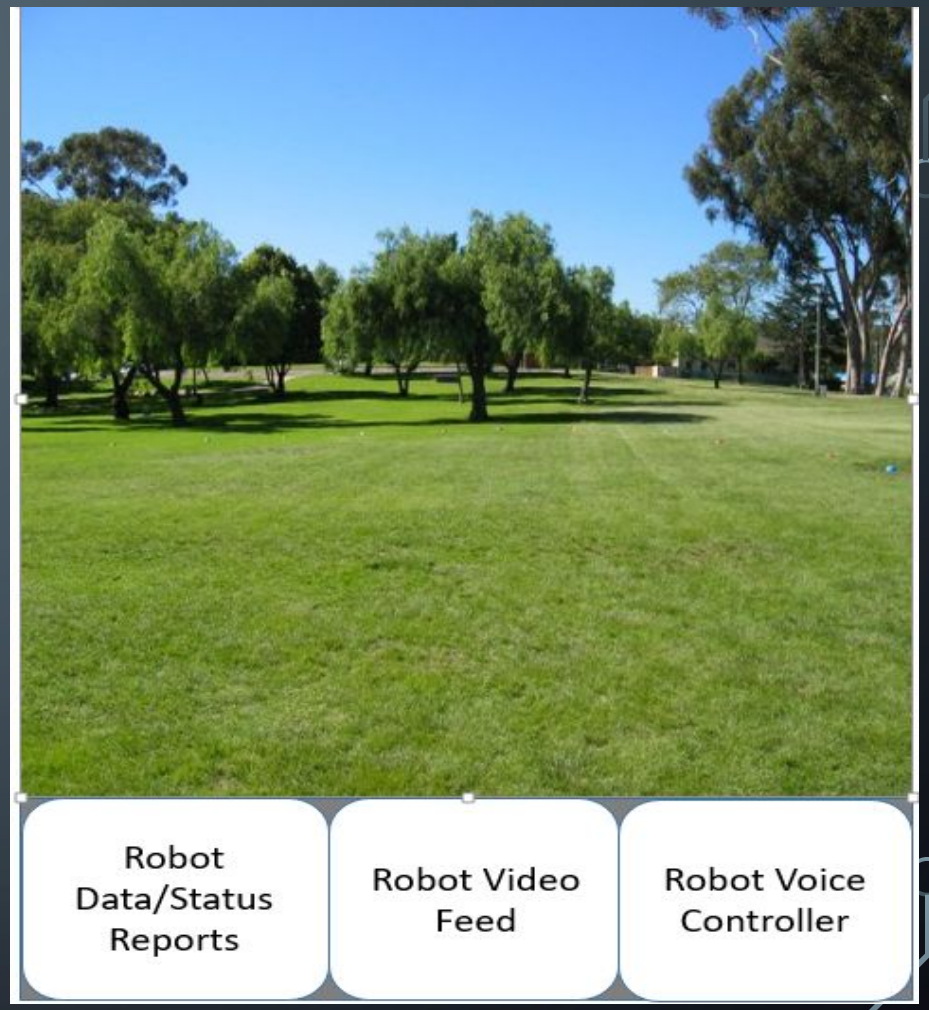


PROGRAMMING OVERVIEW

- App Design
- Voice Commands
- Computer Vision
- Collision Detection

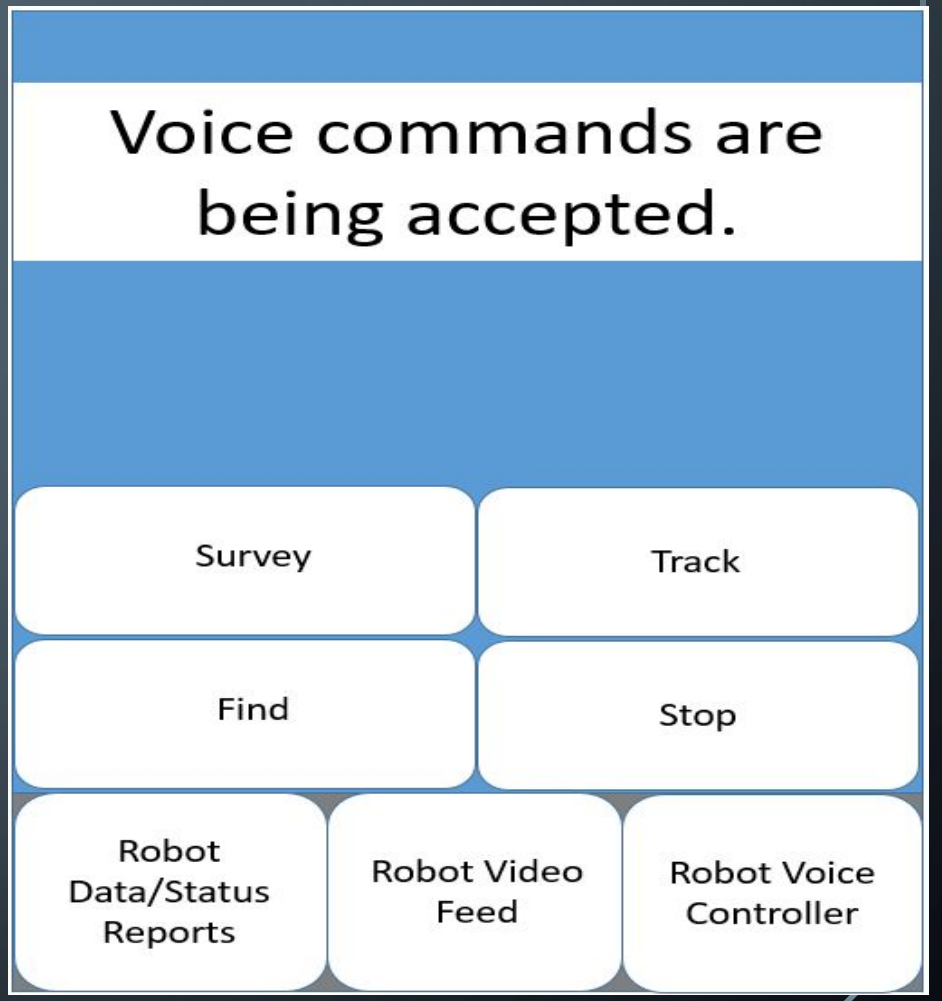
ANDROID APPLICATION

- Phone app using PhoneGap Platform
  - HTML, CSS, Javascript
  - Was Android Studios
- Connecting to Raspberry Pi via WiFi
- Video is sent to phone via the app



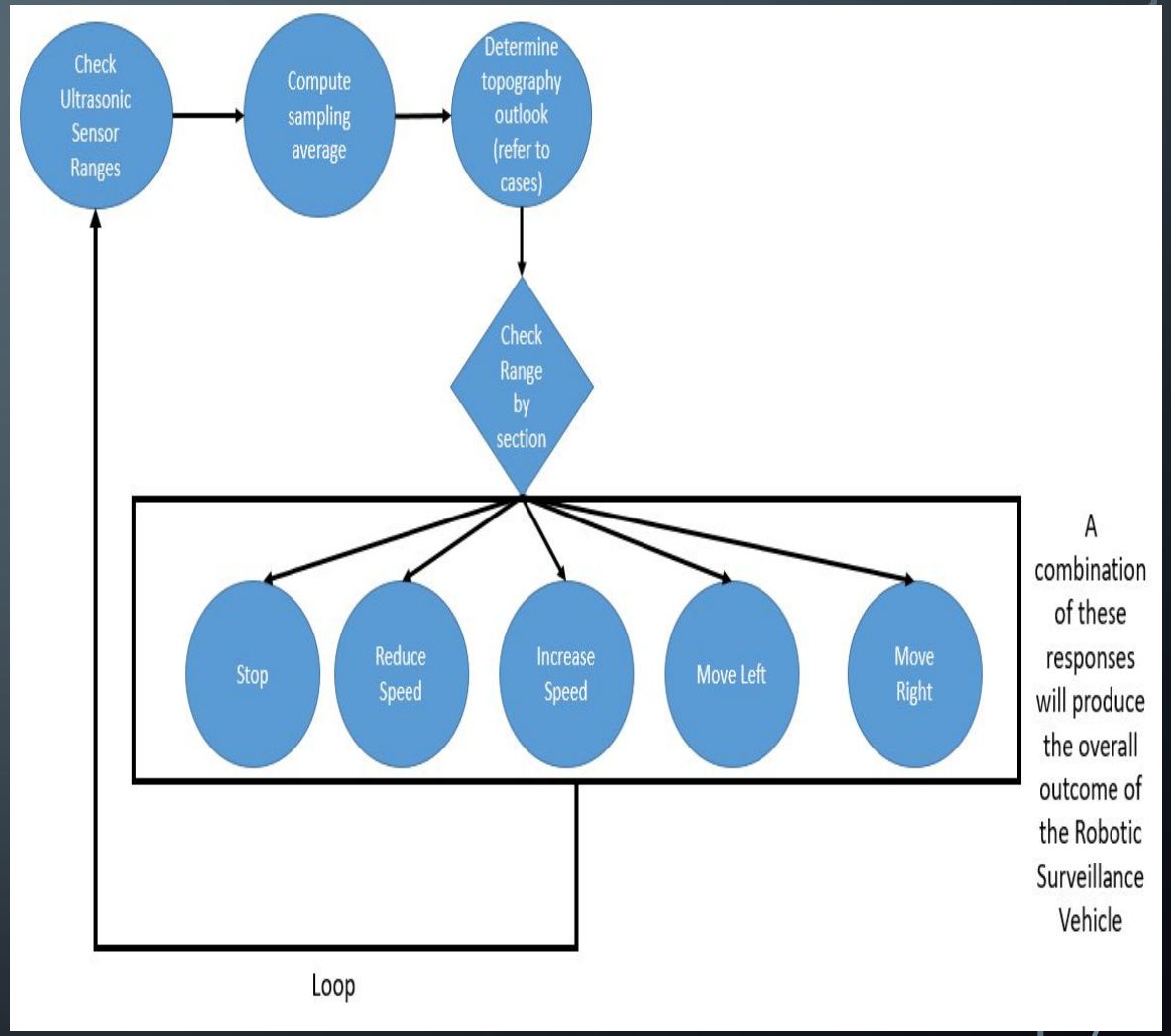
VOICE COMMANDS

- Commands accepted through device
  - Cordova Plugins will be used be voice and video
- App will select mode based on voice command
- Command will be sent to vehicle from app



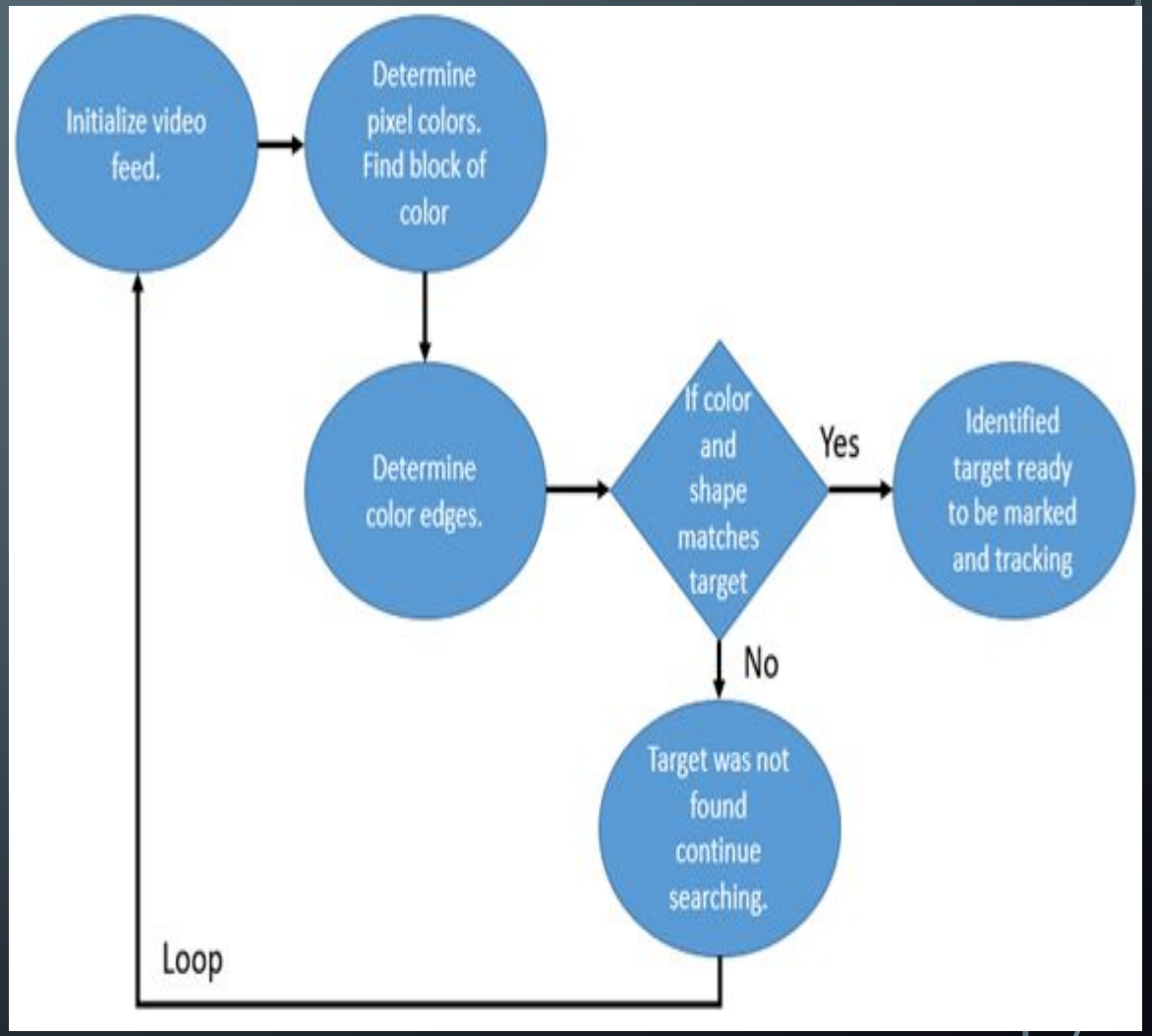
COLLISION DETECTION

- Uses sonar sensor array
- Sensors feed distance values to ATtinys, Raspberry Pi pings for distance within threshold (~ 3-100cm)
- Current distance and rate of change of the distance will be used to determine logic controls
- Motor controls based on determinations will be done using servo controller.



COMPUTER VISION

- Camera on vehicle records images in front of vehicle
- Computer vision software will use camera images to find or track a specified target
- Once target is detected, software will alert the user and track
- SimpleCV library used with Python language to program the computer vision



CONCLUSION

BUDGET

HOUSE OF QUALITY

REVIEW OF ISSUES

BUDGET VS ACTUALS

Component	Budget	Component Used	Actual
PCB	$50.00	Bayarea Circuits	$26.00
MCU	$40.00	Raspberry Pi3	$35.00
GPS/WIFI	$24.00	Included in Raspberry Pi	-
RF/Bluetooth	$5.00	Not used	-
Robot Kit	$20.00	DFRobotShop Rover Chassis Kit w/Expander and motors	$68.00
Battery	$20.00	Poweradd 10000mAh	$16.00
Sensor Array	$11.00	HC-SR04 Ultrasonic Module	$6.00
Compass	$3.00	Adafruit LSM 303 Accel/Magneto	$17.50
Camera	$25.00	Raspberry Pi Camera module V2	$29.90
MCU2	$10.00	Attiny85 & Attiny84 & L293D	$7.90
Transformer	$10.00	Dip Sockets, Hardware,	$5.70
Total	$218.00		$212.00

HOUSE OF QUALITY

			Marketing				Targets
			Cost	*Accuate*	*Intuitive*	*Portable*	
			-	+	+	+	
Engineering	*Dimensions*	-	↓	-	-	↑↑	24''/24''/24'' (L/W/H)
	Wifi Enabled	+	↓	↑	↑	↑	≤ 1Kg
	Voice Control	+	↓↓	↑	↑↑	↑	Accurate voice commands
	Battery Operated	+	↓	-	-	↑	>2 Hours
	Range	+	↓	-	-	-	150 ft
	Response Time	-	↓	↑	-	-	≤ 3s
	Collision Detection	+	↓	↑↑	-	-	Avoid collisions causing loss of track or damage to the vehicle
	Cost	-	↓	-	↓	↓↓	≤ $350

REVIEW OF ISSUES

- I2c Breakdown
- ATtiny code learning curve
- Operational Noise
- Phone app communication with Raspberry Pi
    - App development learning curve.
- Processing power unsuitable for goals.
- Cross discipline makes teamwork necessary.