CSU Bakersfield

School of Natural Sciences, Mathematics, and Engineering

Joana Francisco
Carlos Herrera
Nick Diaz
Cristal Ronquillo
Renzo Navarette

Outline

Abstract & Introduction

Joana Francisco

Abstract

This presentation will present a voice-operated wheelchair project using Arduino. With the voice-operated wheelchair, users will use their voice to communicate with the machine and will be able to command the wheelchair to move forward, reverse, turn right, turn left and stop using a microphone attached to the voice recognition module. The voice-operated wheelchair project will consist of a voice recognition module, Arduino Uno, motor drive module, batteries, and 2 DC motors.

The project aim to design a hands-free wheelchair control.

Introduction

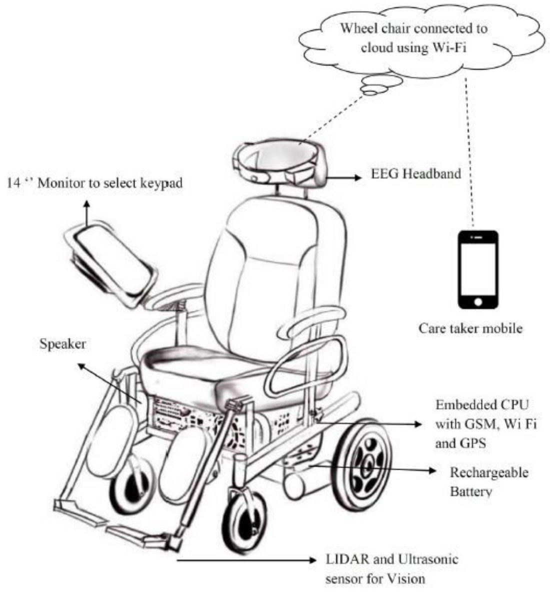
Electric powered wheelchairs provide ease and convenience to wheelchair users. Many engineers have conducted studies regarding hands-free wheelchair controls. Understanding a variety of hands-free wheelchair control is an important basis for the project.

Standing Wheelchair

In this project, the wheelchair gradually changes the seat's surface angle from horizontal to vertical while maintaining the verticality of the leg rest and back. Standing wheelchair project not only benefits users with spinal cord issues, but also all wheelchair users by not having the need to transfer or lift body weight to change from sitting position to standing position. Also helps in maintaining user's posture.

EEG Signal controlled Wheelchair

Wheelchair control system is based on a brain-computer interface. 14 electrodes and two gyroscope sensors were used in the brain-computer interface. The gyroscope played the biggest role in detecting muscle control. **Gyroscope** sensors are commonly used in robotic arm projects where a robotic arm imitates the motion of the user's gloves. The brain-computer interface headset is mounted to the user's head and controls depend on the user's head movement (to turn right, turn left or not to stop). Rise of eyebrows signals forward control.



Chin Controlled Wheelchair

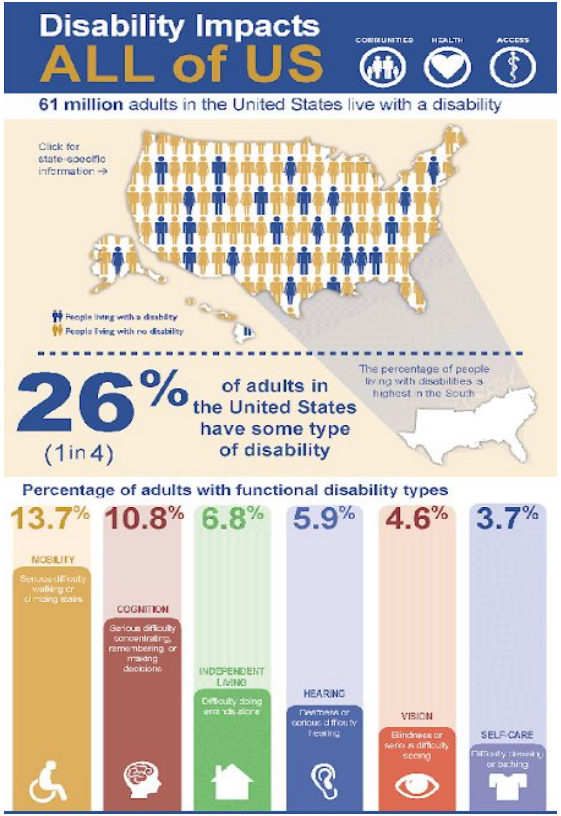
The design is similar to a standard wheelchair joystick, but instead of the joystick placed on the wheelchair armrest, chin controlled modified joystick is mounted under the user's chin, monitoring the chin movement of the user. Moving the chin up will indicate a forward control and forward speed. Same instances apply in moving the chin right or left and down for reverse.

Sip-n-puff Motorized Wheelchair

A research paper from the National Institute of Technology Rourkela in India introduced this non-bio signal approach on hands-free wheelchair control. The sip-n-puff controls used four control signals: hard sip, hard puff, sift sip, and soft puff.

Problem Statement

- **Mobility** is the most common disability in America (2018)
- Wheelchair is the most commonly used mobility device.
- According to the United States of America Census in 2018, there are 6.8Million Americans over the age of 15 use wheelchairs, both powered and manual.
- In addition, an increase of 2 Million wheelchair users is projected every year. America has the largest demographic and fastest growing population of powered wheelchair users.
- There are several types of powered wheelchairs. Price ranges from $1000 to $4000.
- Not many people have full use of their hands or arms.



Problem Statement

In order to accommodate physically ill or handicapped persons, we develop a **voice-operated wheelchair** using Arduino. With voice operated wheelchair, user will use their voice to communicate with the machine and would be able to command the wheelchair to move forward, reverse, turn right, turn left or stop using a microphone attached to either the user or the wheelchair.

Voice-Operated Wheelchair

The main objective of voice operated wheelchair project is recommended to control a wheelchair by using speech or voice recognition module. The project aims to design hands-free wheelchair control because not all wheelchair users have full control of their arms or hands. This project targets mainly the population of wheelchair users who lost their hands or arms in an accident or battles. It also designed patients who have nervous system disorders.

Voice Recognition

- IBM introduced first speech recognition machine in 1962
- Examples:
    - Apple's Siri (2011),
    - Google Now (2012),
    - Microsoft's Cortana (2013)
    - Alexa and Amazon Echo (2014) ,
    - Google Home (2016),
    - Apple HomePod (2017)

CSU Bakersfield

School of Natural Sciences, Mathematics, and Engineering

Renzo Navarrete

Voice Operated Wheelchair

- The main objective of voice operated wheelchair project is recommended to control a wheelchair by using speech or voice recognition module.

Target population:

- Physically impaired patients.
- Wheelchair users who cannot or have hard time steering manual wheelchair or joystick controlled wheelchair.

Components


DIGITAL (PWM~)
UNO
ARDUINO
POWER
ANALOG IN


EasyVR3+


UPS
12V 10Ah

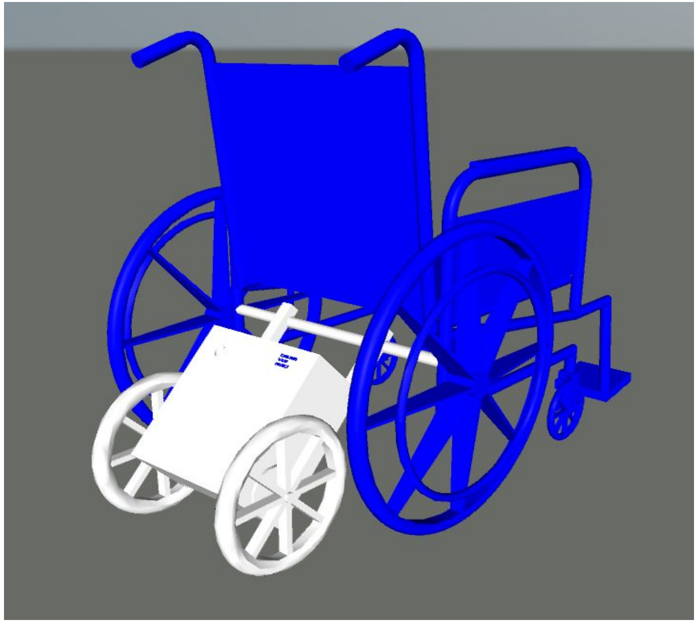
Technical Constraints

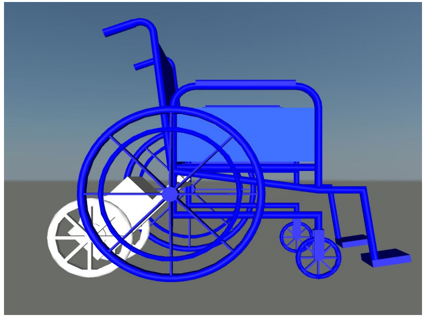
- **Weight Capacity:** 250-300lbs
- **Range of Speed:** 1.5 - 4.0mph
- **Range of Distance:** 20miles
- **Voice Recognition Accuracy:** 80-90%

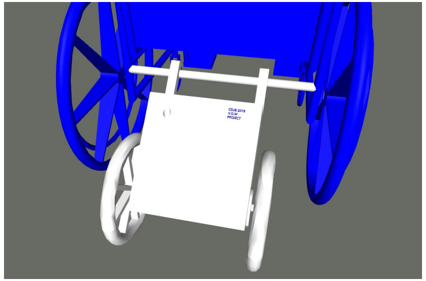
Voice Operated Wheelchair

Forward	Both motors in forward direction
Reverse	Both motors in reverse direction
Left	Right motor forward direction, Left motor off
Right	Left motor forward direction, right motor off
Stop	Both motors off

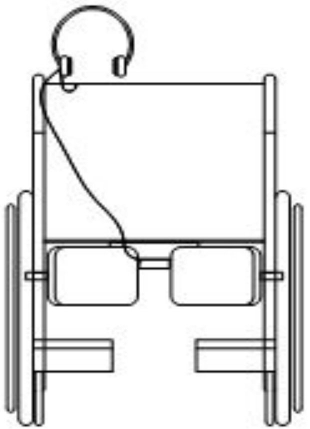
Design 1

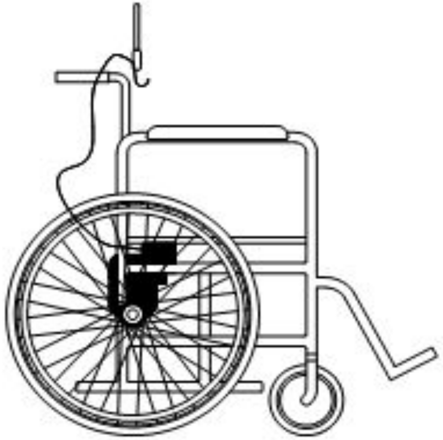






Design 2



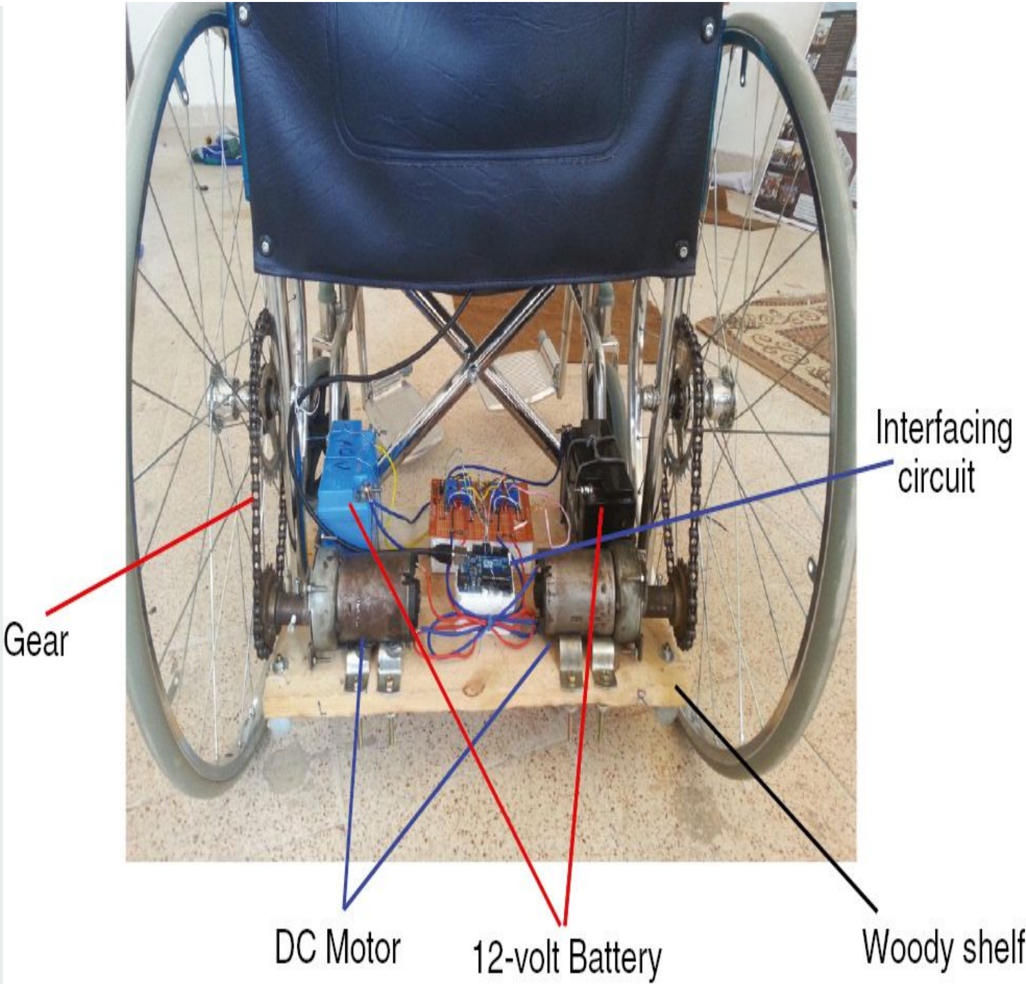


Practical design constraints of voice operated wheelchair

- Economic
- Environmental
- Social
- Political
- Ethical
- Health & Safety
- Manufacturability
- Sustainability

Economical

The cost of a voice operated wheelchair is more than twice the cost of a standard wheelchair. This is much more of a financial burden for those who required a hands-free wheelchair.



Cost Analysis - Big Scale Voice Operated Wheelchair

Constraint	Name	Quantity	Price
Hardware	AA Batteries	1	$5.00
Hardware	Voice Recognition V3	1	$50.50
Hardware	Arduino Uno	2	$18.00
Hardware	Motor Drive Module	1	$74.99
Hardware	12V Battery	1	$35.99
Hardware	2DC 12V 1/8 HP Motors	2	$240.00
Hardware	Wheelchair	1	$150.00
		Total	$814.48

Environmental

- Not designed for uneven terrain.
- Cannot travel over steep terrain.
- Cannot use a staircase for traveling, requires ramp.
- Not waterproof.
- Batteries are sensitive to cold weather.

Social

There are many similar constraints between the standard wheelchair and V.O.W. However, one key difference is that V.O.W. does not require the assistance of someone else in order to move or steer your trajectory.

You will still be constrained by accessibility to certain activities. On the bright side, you get to go to the front of line in amusement parks, assuming you can get on the ride.

Political

- Special vehicle required for transportation.
- Handicapped parking required.
- Unable to go to bathroom without assistance.
- Unable to bathe without assistance.

Ethical

- Not as effective as walking on two legs.
- Has limited speed and direction mobility.
- Unable to relieve pressure sores.
- Not much storage space.

Health & Safety

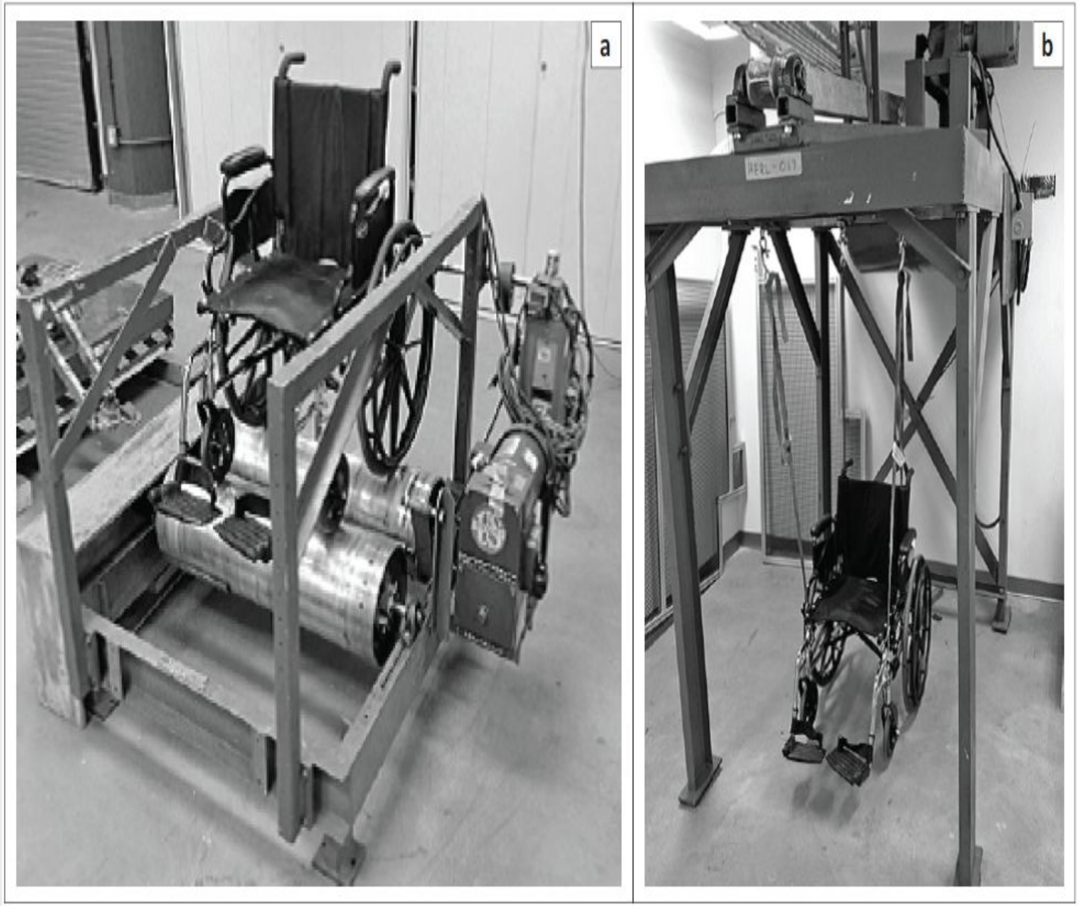
- Unable to prevent the development of sores due to long use.
- Nothing prevents the user from falling out if knocked off balance.
- Vulnerable to attacks from anyone with hostile intent.
- Motion lag can lead to collisions.

Manufacturability

- Not being mass produced.
- Requires self assembly.
- Assembly requires builder to be able to move freely and weld.

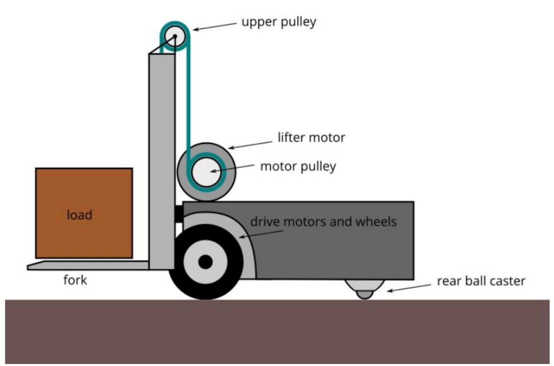
Sustainability

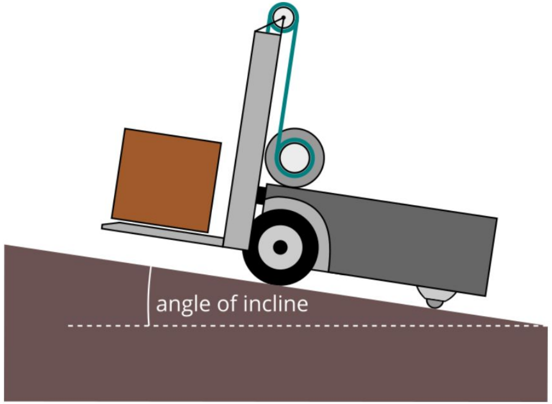
- Just as sustainable as a standard wheelchair.
- Durable. Can carry hundreds of pounds.
- Treat gently to extend the life of V.O.W.

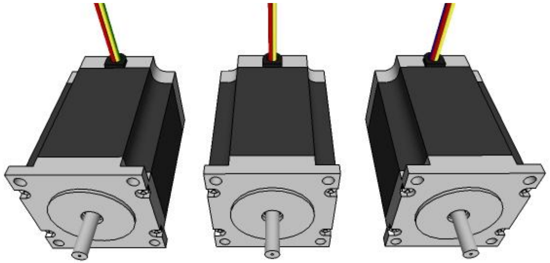


Source: Testing machines in a certified wheelchair testing laboratory (Gebrosky et al. 2013)

FIGURE 3: Multidrum test (left) and curb-drop test (right) without test dummies.

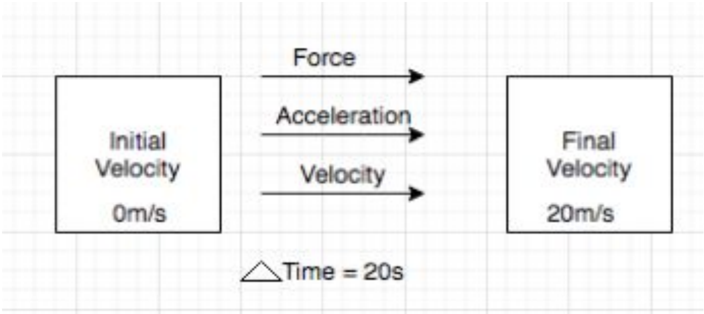






SOLUTION FORMULATION AND CONCEPTUALIZATION

MOTOR
Calculations



$$Acceleration = \frac{Final\ velocity - initial\ velocity}{Time} = \frac{20-0}{20} = 1m/s^2$$

$$Force = ma = (136kg)\left(\frac{1m}{s^2}\right) = 136Newton$$

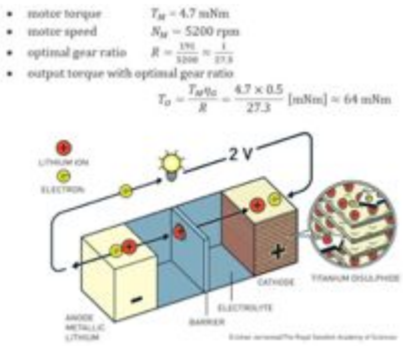
Choosing the right BATTERIES for the Voice Operated Wheelchair

Sensor Voltage	5	volts
Sensor Current Draw	.05	amps
Microcontroller Voltage	5	volts
Microcontroller Current Draw	.03	amps
Voltage Regulator Dropout	1	volts
Drive Motor Voltage	115	volts
Expected Current Draw	5.3	amps
# of Motors	2	motors
Time % Motors Used	75	%
Voltage per Pack	12	volts
mAh per Pack	10000	mAh
Packs in Parallel	1	packs
Total Continuous Power Draw	96.4	watts
Min Required Battery Voltage	115	volts
Idle Motors Battery Life	125	hours
Minimum Battery Life	0.936	hours
Typical Battery Life	1.25	hours

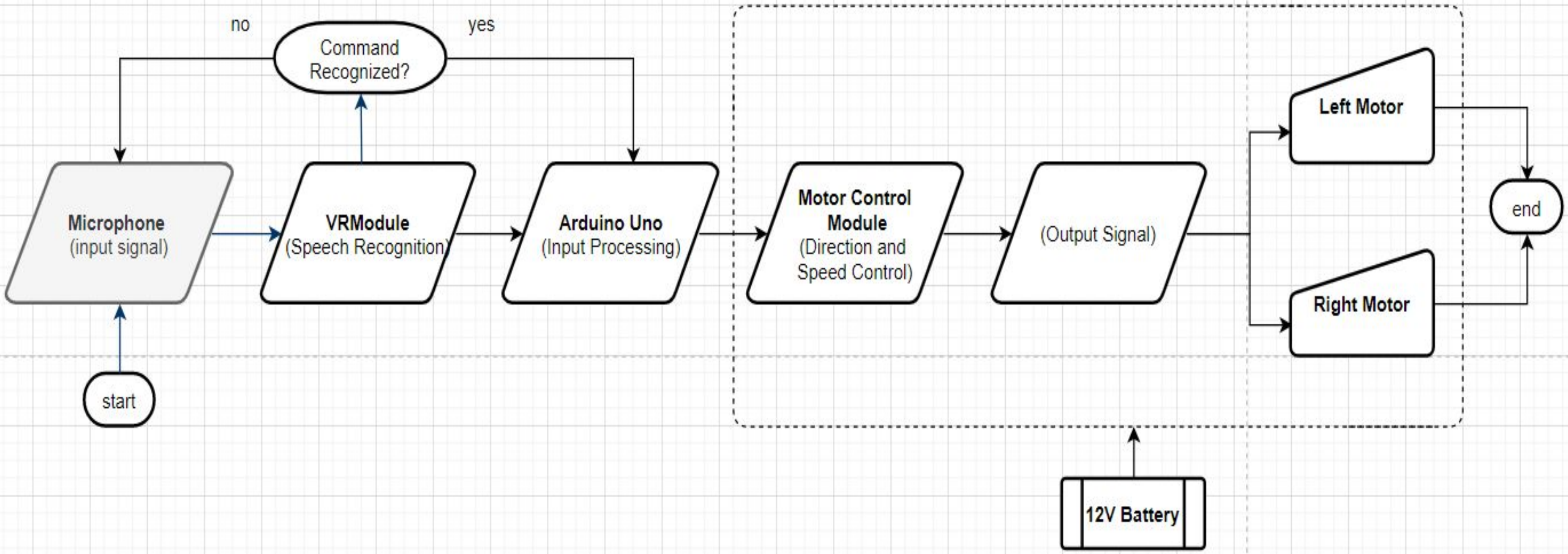
Step 1: Determine motor requirements per motor

Step 2: Determine electronics requirements

Step 3: Battery setup and specs



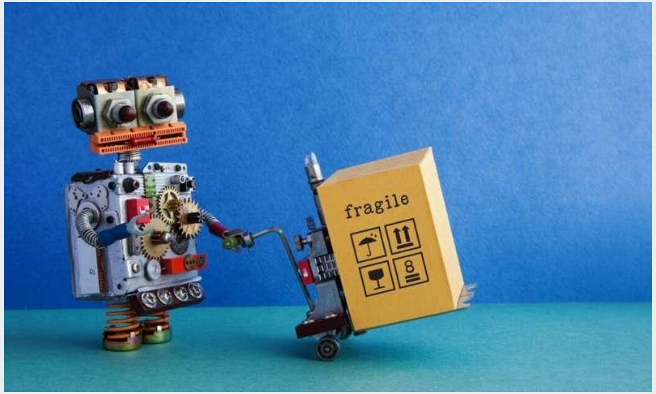
Flow Chart



no
Command Recognized?
yes
Microphone
(input signal)
VRModule
(Speech Recognition)
Arduino Uno
(Input Processing)
Motor Control Module
(Direction and Speed Control)
(Output Signal)
Left Motor
Right Motor
end
start
12V Battery

SMALL SCALE

VOICE OPERATED WHEELCHAIR

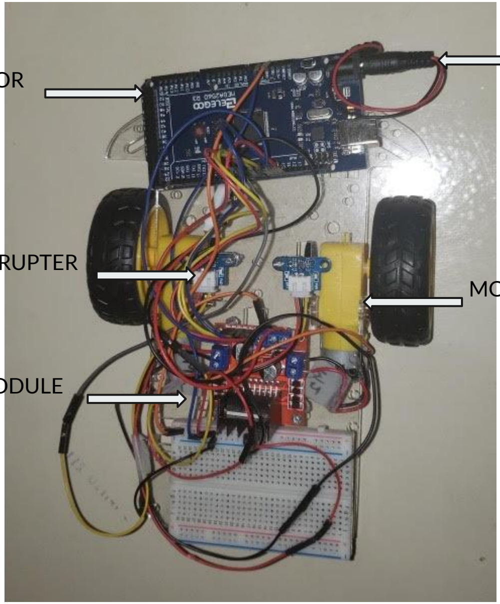


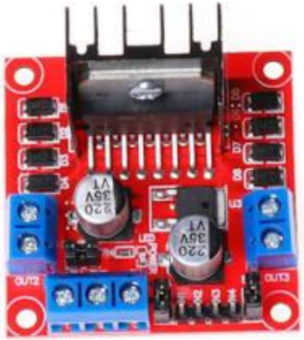
BY: CRISTAL RONQUILLO

COMPONENTS

MATERIALS

Arduino UNO
Photoelectric Encoders
Ultrasonic Sensors
Opto Interrupters
Voice Recognition Module V3
Battery Case
Robot Kit
Mega 2560 R3 Board
Motor Drive Module
Miscellaneous





This module will allow you to easily and independently control two motors of up to 2A each in both directions.

Voice recognition module and mic

The **Arduino Uno** is a microcontroller board based on the ATmega328. It has 20 digital input/output pins (of which 6 can be **used** as PWM outputs and 6 can be **used** as analog inputs), a 16 MHz resonator, a USB connection, a power jack, an in-circuit system programming (ICSP) header, and a reset button.

Battery case

Motor

A rotary encoder is a type of position sensor which is used for determining the angular position of a rotating shaft. It generates an electrical signal, either analog or digital, according to the rotational movement.

COST ANALYSIS

Cost Analysis Small Scale
For Federally Funded Contracts

VENDOR/CONTRACTOR NAME	*VOICE OPERATED WHEELCHAIR*
PROJECT / BID	*SENIOR SEM 2020*
DATE	*4/1/2020*
CONTACT NAME & PHONE #	*CRISTAL RONQUILLO*

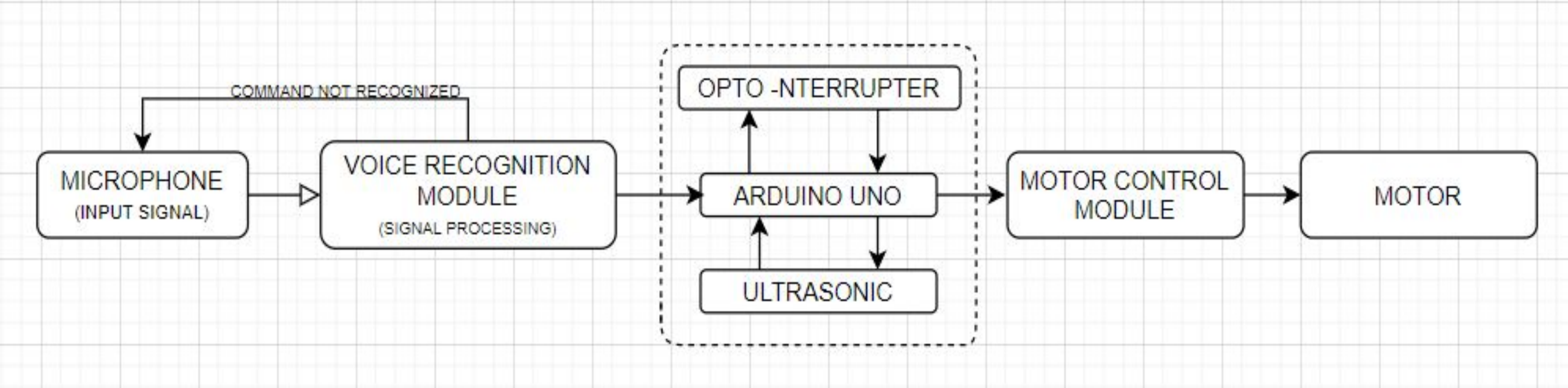
Components of Pricing to be Evaluated:

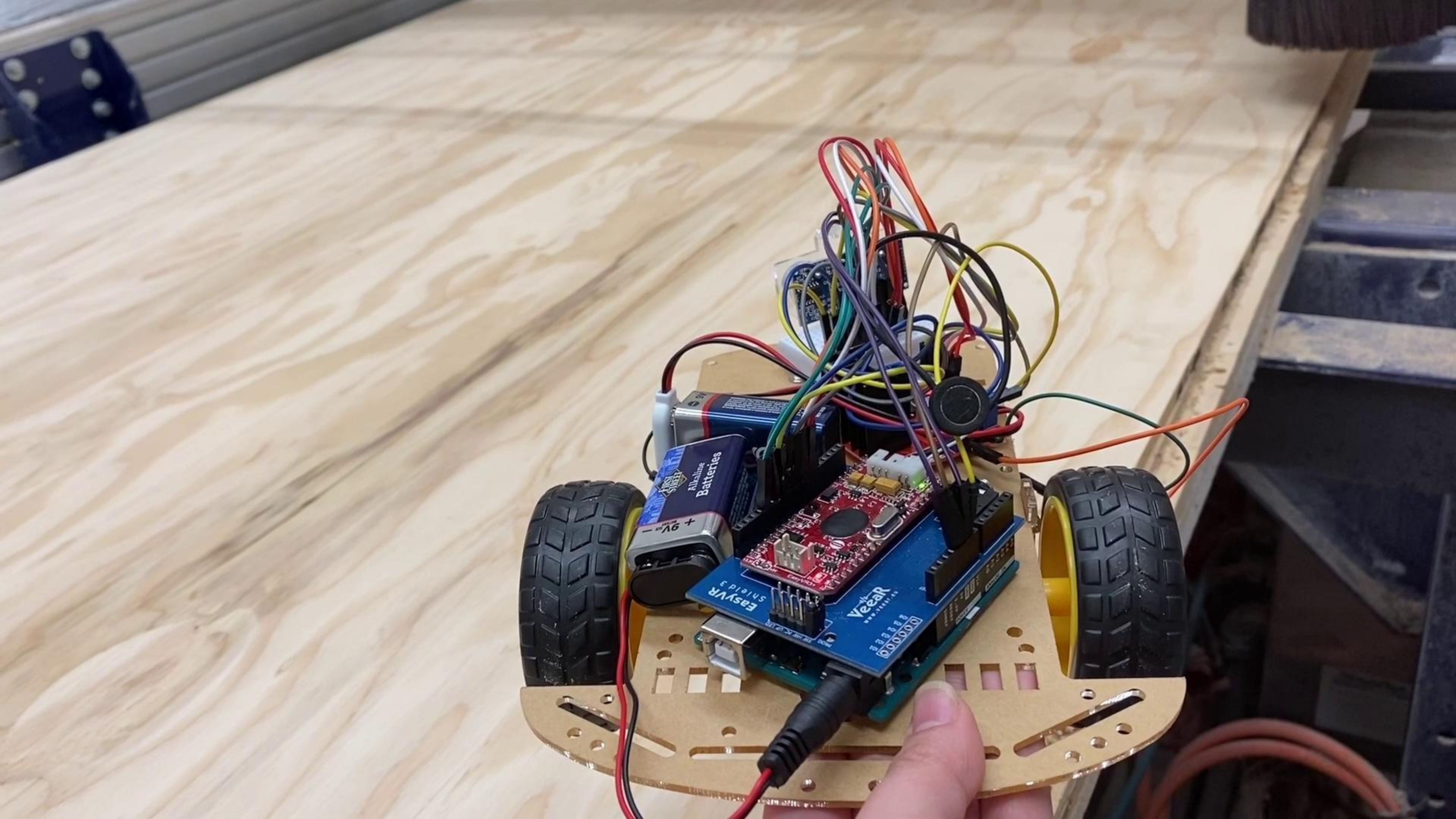
MATERIALS

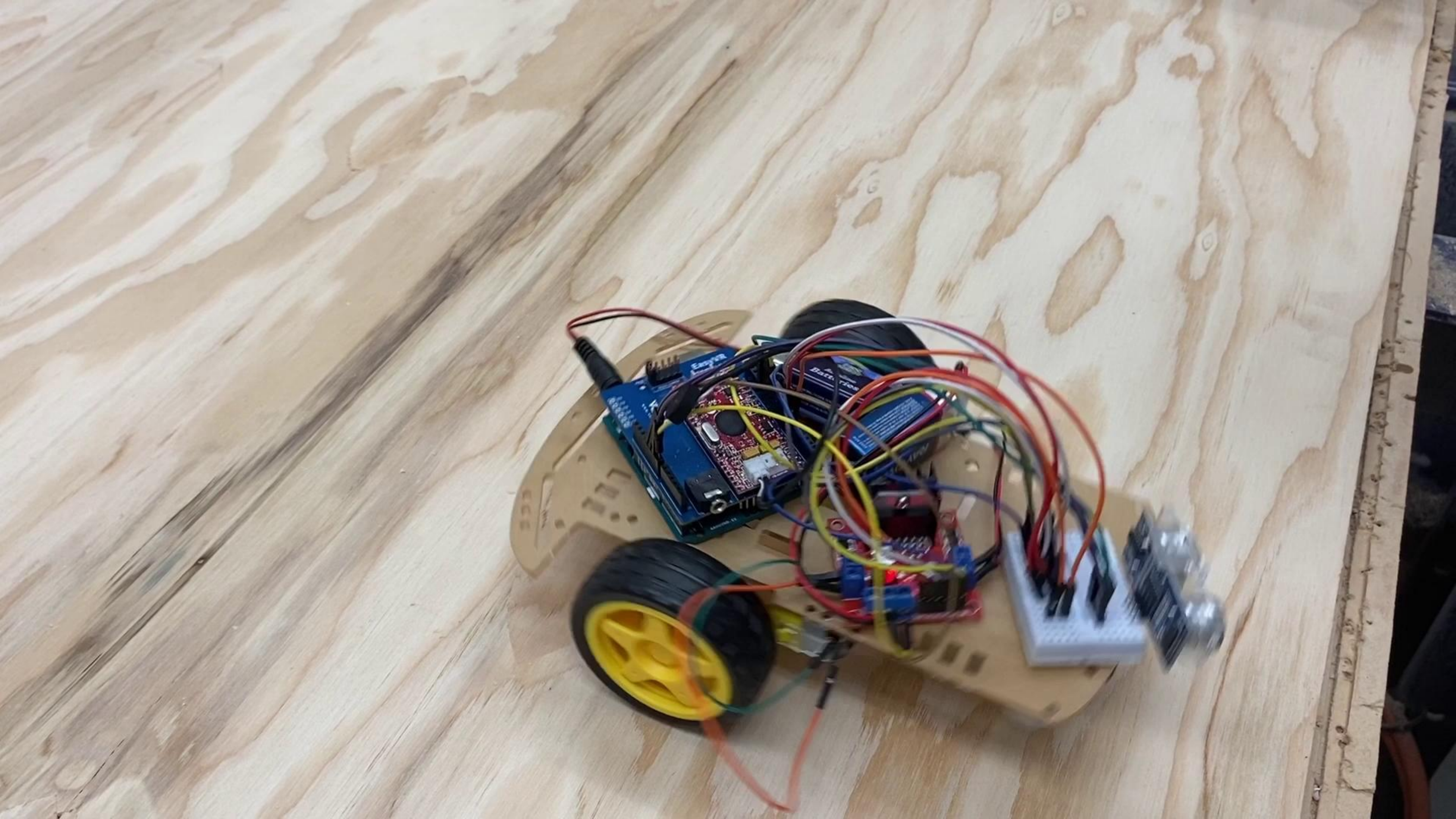
	#Units	X	Price/Unit	=	Total Cost	COMMENTS / DETAIL
Arduino UNO	1.00	X	$18.00	=	$18.00	
Photoelectric Encoders	1.00	X	$9.99	=	$9.99	
Ultrasonic Sensors	2.00	X	$10.98	=	$21.96	
Opto Interrupters	2.00	X	$10.98	=	$21.96	
Voice Recognition Module V3	1.00	X	$50.50	=	$50.50	
Battery Case	1.00	X	$6.99	=	$6.99	
Robot Kit	1.00	X	$17.99	=	$17.99	
Mega 2560 R3 Board	1.00	X	$16.98	=	$16.98	
Motor Drive Module	1.00	X	$8.99	=	$8.99	
Miscellaneous		X	$30.00	=	$0.00	*Ex: wire, wire cutter, sodering iron and etc.*

FLOW CHARTS

CODE LOGIC OVERVIEW




Alkaline
Batteries
+ 9V –
EasyVR
Shield 3
VeeaR

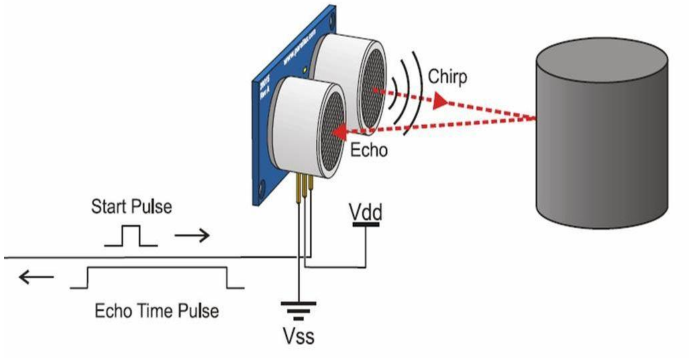

EasyVR

ADDITIONAL FEATURES

OBSTACLE AVOIDANCE AND PREPROGRAMMED COMMANDS

OBJECTIVE

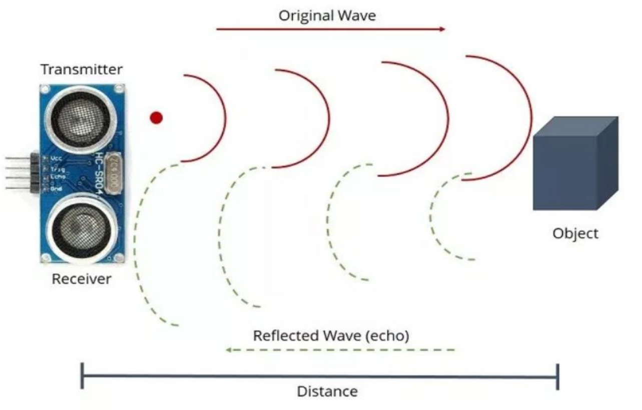
- Obstacle avoidance
- Emergency stop
- Increase safety features
- Allows autonomous functions for multiple execution
- Independence mode vs with assistant mode

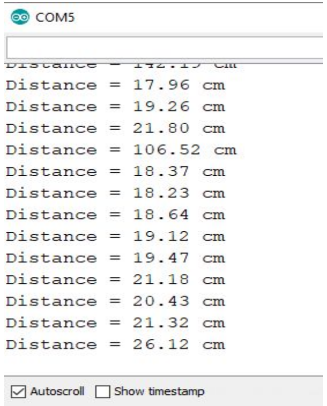


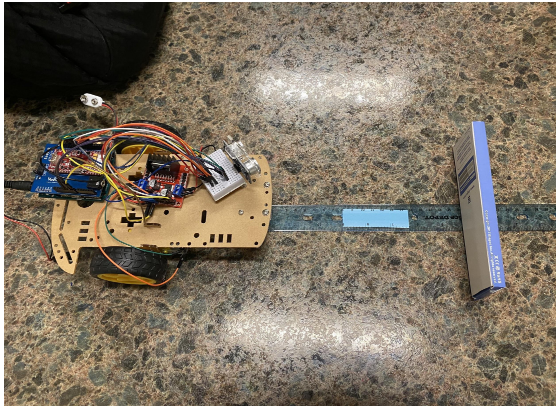
ULTRASONIC SENSOR

HC-SR04 Sensor Features

- Operating voltage: +5V
- Theoretical Measuring Distance: 2cm to 450cm
- Practical Measuring Distance: 2cm to 80cm
- Accuracy: 3mm
- Measuring angle covered: <15°
- Operating Current: <15mA
- Operating Frequency: 40Hz

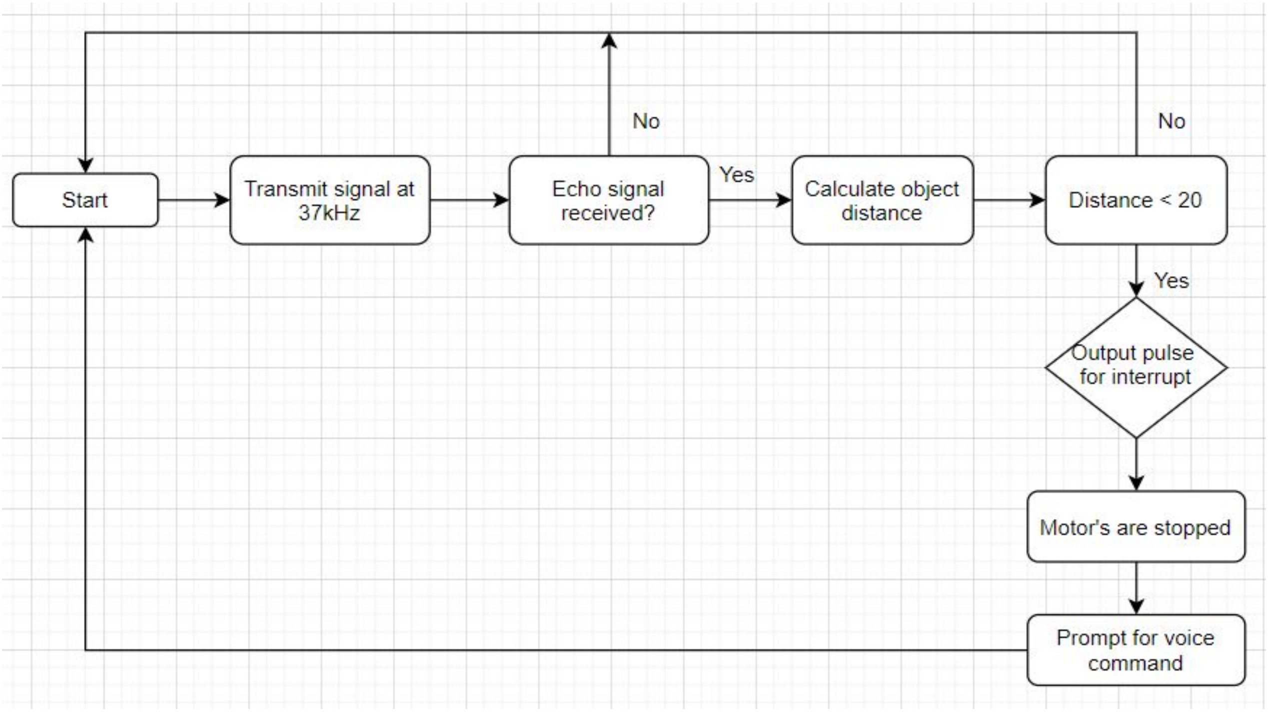






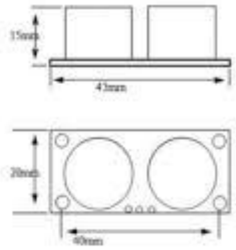
DESIGN

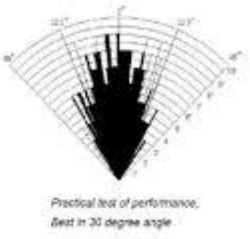
(BLOCK DIAGRAM)

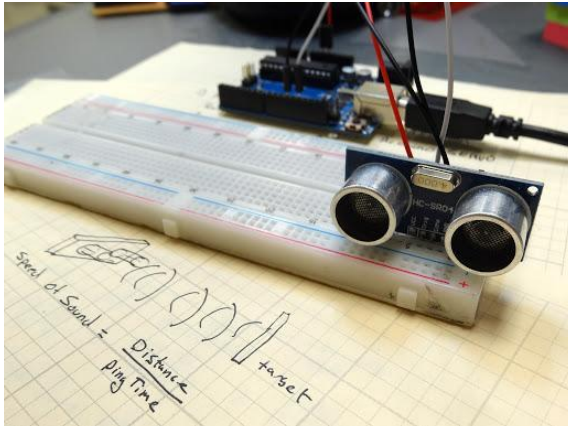


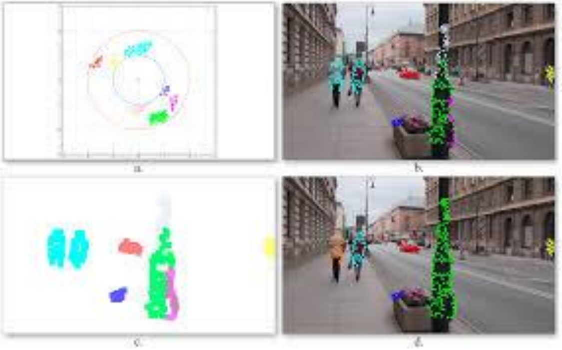
TRIAL TEST

- Speed test
- Dead zone testing
- Backing up test





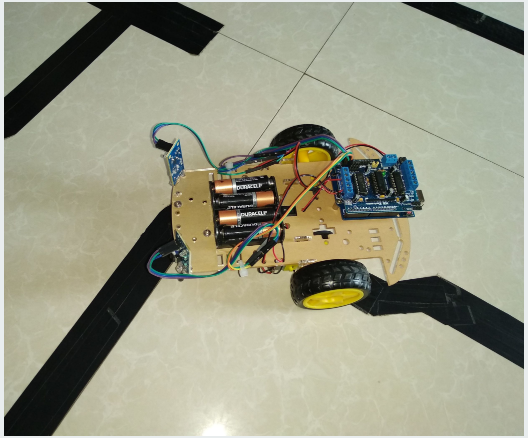


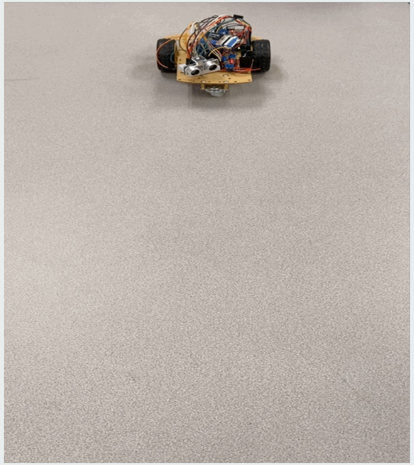


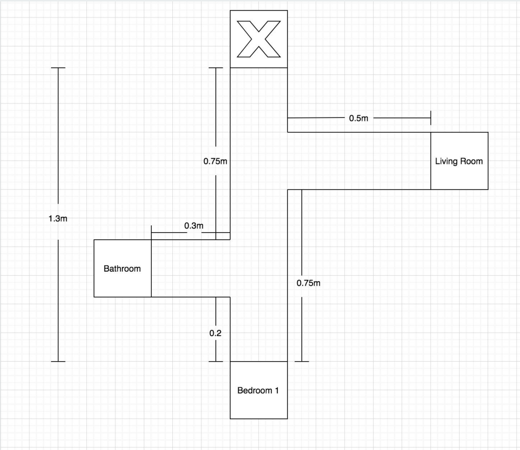
Preprogrammed Commands

- **The wheelchair must be capable to execute directions going from one point to another using one voice command.**
  - **Go to Bedroom,**
  - **Go to Bathroom,**
- <u>Basic idea of the code:</u> MoveForward((CMtoSteps(50), 255); // Forward half a meter at 255 speed. Delay(1000); SpinRight(20,255) // Spin right 20 steps at 255 speed.

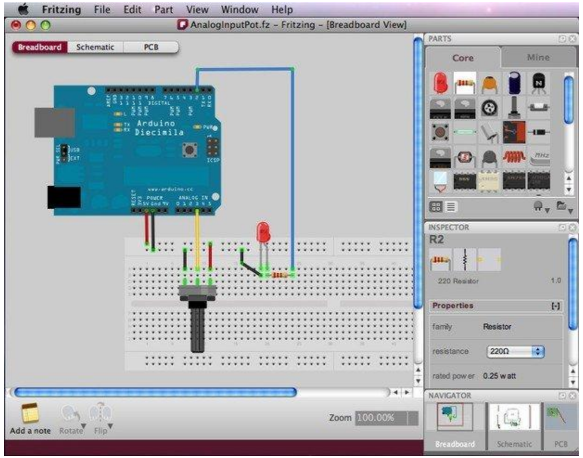
- A flatform platform will be used to demonstrate how multiple execution work



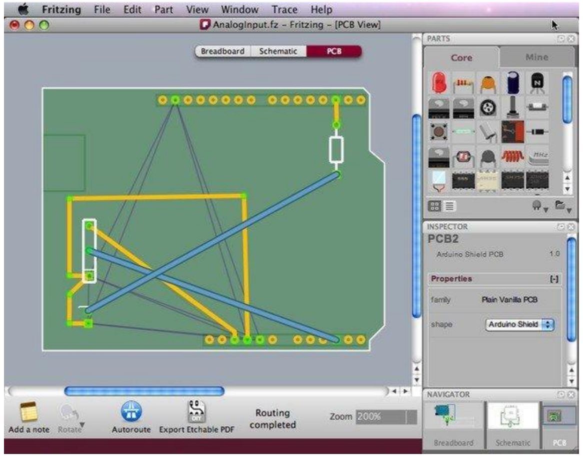




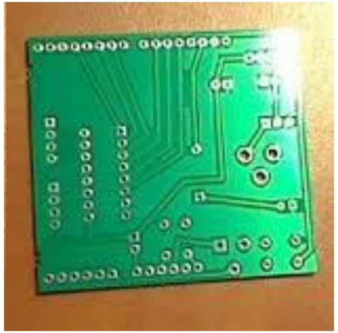
Finalizing Hardware connections



Component Routing



Board Designing



Final Product

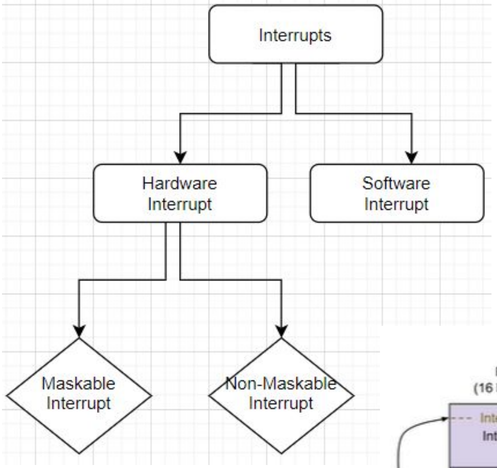
DESIGN CONSTRAINTS

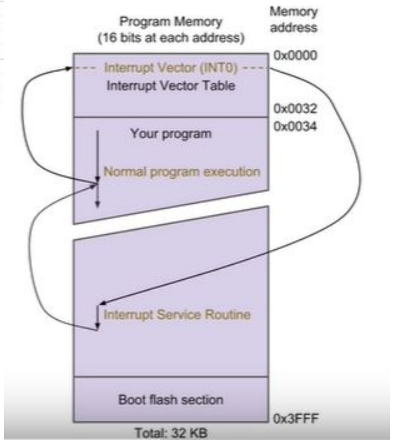
- Wheel rotation and speed
- Battery distribution
- Missing interrupts
- Microphone changed
- Switch for manual stop
- Ultrasonic sensor range increase with servo

What are Interrupts

What you need to know about Software Interrupt

1. Software interrupt is the interrupt that is generated by any internal system of the computer (instruction in the program).
2. Software interrupts are synchronized events.
3. Software interrupts increase the program counter.
4. Software interrupt is a type of interrupt that is caused either by a special instruction in the instruction set or by an exceptional condition in the processor itself.
5. Software interrupt can be categorized into two types, they include; Normal interrupt and Exception
6. Software interrupt is triggered by software (program instructions) and considered one of the ways to communicate with kernel or to trigger system calls, especially during error or exception handling.
7. Software interrupt has the highest priority than hardware interrupt.





Deliverables and Progress Report

Nicholas Diaz

Proposed Deliverables

November 2019	• Ordered Materials
December - January 2019	• Set up Easy VR3 Module with Arduino • Compiled commands in Arduino • Connected Arduino to motor drive • Ran small scale test with LEDs and Motors

February 2020	• Worked on speed and direction control • Worked on the wheelchair design
March 2020	• Finalized Wheelchair design • Worked with mobility and user comfort • Test Ultrasonic sensor and Rotary encoder
April 2020	• Troubleshoot code • Final tests • Add camera to follow lines to preset locations

Actual Deliverables

September - Meeting for project ideas
October - Research on our chosen idea
November - Finalized proposal and ordered materials
December - Tested VR module and Arduino

January: 2 week break,

Week 3 - Connected arduino to motor drive with small motors.

Week 4 - Research on upsizing

February: Week 1 - Switch to small scale. Ordered small car for small scale motor. Include new features.

Week 2 - received ordered chassis, uploaded code direction (video), problem with VR accuracy

Feb 13 - Carlos troubleshoot code turning (R/L) in place

Feb 17 - Easy VR - troubleshooting, - quiet place. (noise)

Feb 18 - gif - direction and lagging troubleshooting

Week 4 - Test and compare two VR. (show data)

- Ultrasonic Sensors (distance data)
- Encoder data (proof that both motors aren't running the same RPM) - that why we need the opto encoder

March: Mar 5 - Encoder and the new VR.

Week 1 - Ultrasonic with Servo.

Changes in Deliverables

Voice Recognition change to improve accuracy of voice commands

Use of Ultrasonic sensor to provide a stop to avoid crashing

Encoder to keep the wheels moving at the same speed to avoid drifting

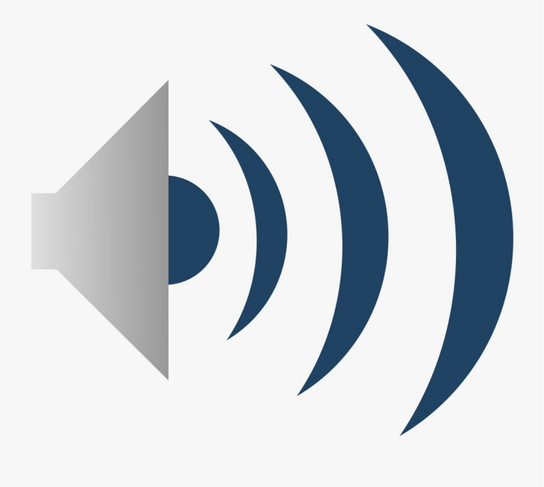
Didn’t use camera to follow guidelines since it was an extra goal if we were ahead

Changes to VR Module

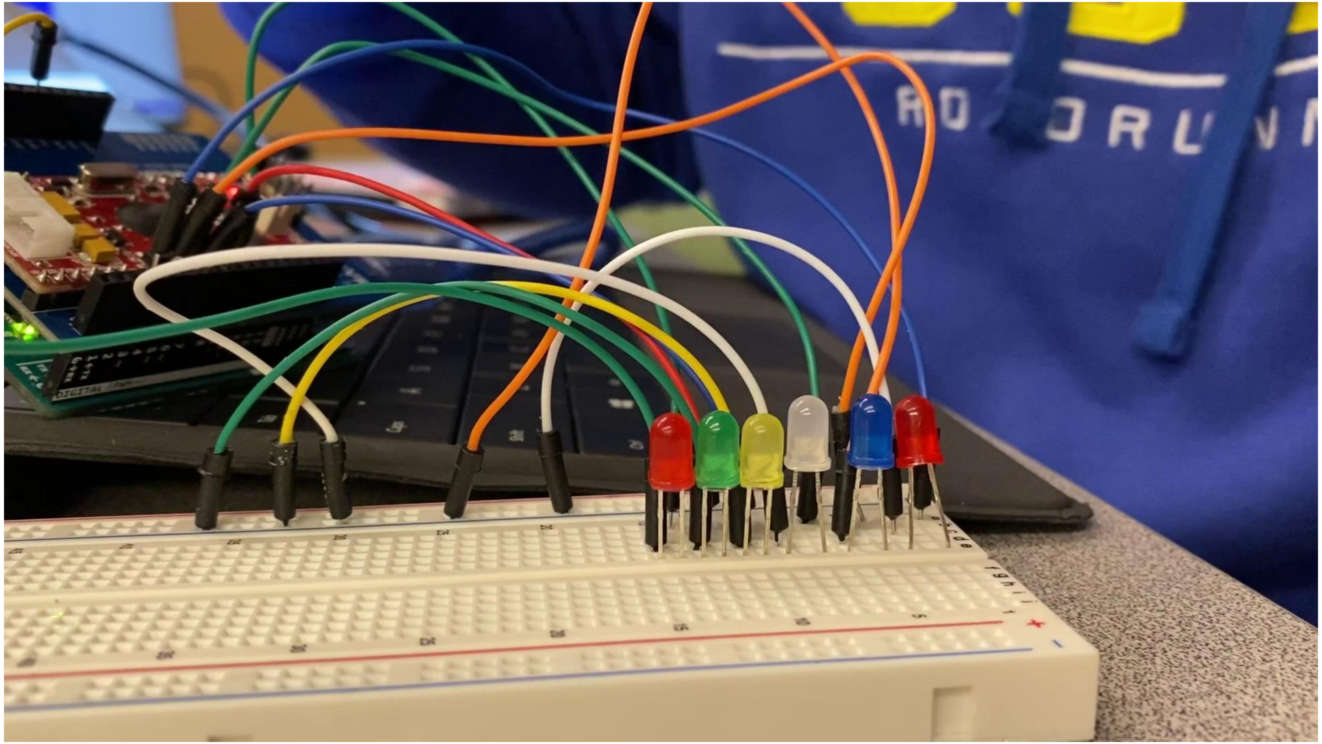
As stated before, we changed our VR module to achieve better accuracy

While the old VR worked fine and was user friendly, the accuracy of commands was low especially in louder environments

Our new VR claimed to be more accurate



Video



Distribution of Work

MY TASKS	START DATE	NOTES
Ordering the parts	November	Renzo
Testing the EasyVR with LED	December	Nick
Arduino, Motor Drive Module and Motors	January	Carlos and Cristal
Easy VR with Arduino, L298N and motors.	January	Carlos, Nick, and Renzo
Ordered Car Chasis and ultrasonic sensors (Small-Scale)	January	Joana
Car Chasis	January	Joana
Autonomous Car	January	All
Troubleshooting direction, Turning in place	January	Carlos
Troubleshooting VR	Januray	Nick
Ultrasonic Sensor measuring the distance	February	Carlos and Cristal
Ordered Interrupter	February	Renzo
Opto-Interrupter	February	Carlos and Joana
Comparison of 2 VR accuracy	February	Nick, Carlos, and Renzo
PCB board design	February	Cristal
Troubleshooting encoder	March	Carlos
Organization of circuit board	March	Cristal
Ultrasonic Sensor with Servo	March	Joana and Nick

Progress with Deadlines

Even with the changes, we were on track towards completing the wheelchair

Due to recent events, meeting to work on implementation has stopped

Worked on the code to simulate how it will run

Issues and Solutions

As stated before, we had issues with the VR module and solved it by replacing it

The issue with drifting was solved by implementing a rotary encoder to calculate and determine the rotation of each wheel

An issue with doing a large scale wheelchair was major cost for the motors and chair which we solved by switching to a smaller scale and calculating for actual implementation

Conclusion

Carlos Herrera

Finalized Project

- Project was only partially completed due to COVID-19 pandemic.
  - Only the voice recognition and the movement were completed

  - Sonic sensor obstacle avoidance, and preprogrammed motion halted due to social distancing orders.

    no more in person meetings

    would need to meet in person to exchange parts

Testing Accuracy of Voice Recognition V3

Final results of Voice Recognition

- Voice Recognition Mode V3 much better than Easy VR Shield 3.0.

- Easy VR Shield 3.0 abysmal performance with minimal noise.
    -

Accuracy V3	ideal	med noise
success:	96	77
fail:	4	28
accuracy:	0.96	0.73

Easy VR 3.0	ideal	med noise
success:	78	N/A
fail:	22	N/A
accuracy:	0.78	N/A

Movement Commands

- A single motor is designated the master and a constant power is applied to the master motor.
- The second motor, the slave, must continuously check deviation from the master. A correction is applied to fix error.
- Every iteration for a check continuously updates the power to the slave.

Results Master Slave

Using Master Slave approach, the RPM synchronization has a mean deviation of 2.68% between the two motors.

```
0000 Motor Speed1: 132.00 RPM - Motor Speed2: 135.00 RPM
000 Motor Speed1: 129.00 RPM - Motor Speed2: 135.00 RPM
000 Motor Speed1: 132.00 RPM - Motor Speed2: 135.00 RPM
0000 Motor Speed1: 132.00 RPM - Motor Speed2: 132.00 RPM
000 Motor Speed1: 129.00 RPM - Motor Speed2: 135.00 RPM
000 Motor Speed1: 132.00 RPM - Motor Speed2: 135.00 RPM
0000 Motor Speed1: 129.00 RPM - Motor Speed2: 135.00 RPM
000 Motor Speed1: 132.00 RPM - Motor Speed2: 135.00 RPM
000 Motor Speed1: 132.00 RPM - Motor Speed2: 135.00 RPM
0000 Motor Speed1: 129.00 RPM - Motor Speed2: 132.00 RPM
```

Problems

- Sensor for slave was not reading correctly.
    - Right sensor does not work properly.
    - Does not count a lot of the breakpoints on the indicator wheel.
    - A new sensor would need to be ordered.

Reflection-Issues

Where did it go well, where did you run in problems.

- Underestimated cost to make VOW
- ordered unnecessary parts.
- Took too long to troubleshoot
- Pandemic inhibited ability to meet in person and exchange parts.

Suggestions

- Any suggestions if these and how these problems might have been avoided? If you would have to start all over again, what would you have done differently?
  - Spent more time planning the project and all the things it ought to be able to do and sticking with this plan to the end.
  - Done more research BEFORE ordering our parts.
  - ALWAYS check the most obvious hardware issue before assuming it's something else (e.g. check power source first, then check wiring next, last check software).

Desired College Resources

- It would have been helpful to take the Robotics course offered at CSUB, but some of us didn't have the chance to.

- More access to measuring instruments would have helped. Sometimes we would have been able to resolve a hardware issue in minutes if we had a DMM or an Oscilloscope available to use outside of regular lab hours.
- Funding

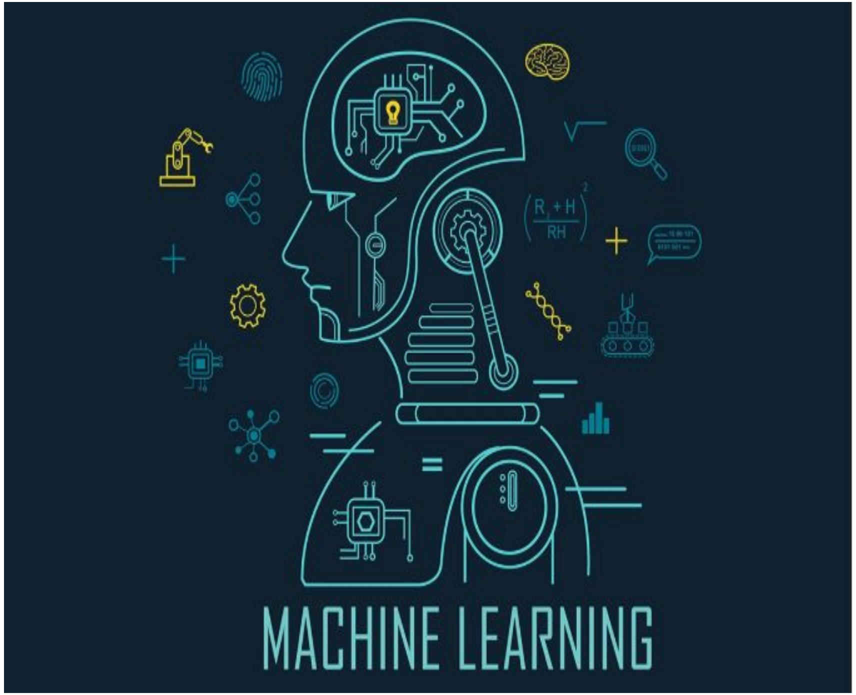
Reflection As A Team

- All Roles Equally Important
- Be willing to challenge ideas you don't agree with.
- Cooperation and ability to Compromise
- Project Management
- Troubleshooting
- Improved Coding Abilities

Contemporary Issues

- Limits of Current Technology- As research into AI improves, so will the performance of voice recognition technology.

- Safety- Given that with even the best technology available, a voice controlled wheelchair has safety issues since commands would need to understood near 100% of the time to avoid serious injury.



Use of Acquired Academic Knowledge

- Electric Circuits (Analog and Digital)
- Digital Communication
- C++
- Operating Systems
- Arduino Forums/Workshops

Future Improvements

Would like to add more features:

- Line Following Algorithms
- Pre-programmed movement
- More User Friendly
- Portability

Acknowledgements

- Dr Wei Li, assisting in robot movement issues.
- Fab Lab, for donating small DC motors and 9V batteries.
- Donation of a wheelchair by friend of group member

References

DroneBot Workshop. “Using Rotary Encoders with Arduino.” Youtube, 1 Mar. 2020, https://www.youtube.com/watch?v=V1txmR8GXzE.

RobotC. (2012). “Using Encoders to Drive Straight”. 15 Mar. 2020 from https://www.robotc.net/wikiarchive/Tutorials/Arduino_Projects/Mobile_Robotics/VEX/Using_encoders_to_drive_straight

Veear Projects Inc. 2019. Easy VR 3.
http://www.veear.eu/files/EasyVR%203%20User%20Manual%201.0.16.pdf

References

L. Xin, S. Gao, J. Tang and X. Xu, "Design of a Brain Controlled Wheelchair," *2018 IEEE 4th International Conference on Control Science and Systems Engineering (ICCSSE)*, Wuhan, China, 2018, pp. 112-116.

K. Matsuzawa and C. Ishii, "Control of an electric wheelchair with a brain-computer interface headset," *2016 International Conference on Advanced Mechatronic Systems (ICAMechS)*, Melbourne, VIC, 2016, pp. 504-509.

W. Seamone and G. Schmeisser, “New Control Techniques for Wheelchair Mobility,” APL Technical Digest, Vol. 2, No. 3, 1981.