Github Repo: https://github.com/iAarush/HARD-Hack-2023/tree/main

Hard Hack 2023 Ideas

Sensing the World

TO DO

- [ ] Bluetooth
- [ ] Sensor Readings
    - [ ] Flame
    - [ ] Rain
    - [x] ~~Temp~~
    - [x] ~~Brightness~~
    - [ ] Gas
        - [ ] https://learn.sparkfun.com/tutorials/hazardous-gas-monitor/all
- [ ] Chassis
    - [ ] Steering
        - [x] ~~Obstacle Detection (OD)~~
            - [x] ~~Swiveling Ultrasonic (See Github: "Hard Hack" folder)~~
                - *decrease swivel increment in code to turn slower
        - [ ] Control based on OD
    - [ ] Construction
- [ ] Final Code Compilation
- [x] ~~Power Supply~~

Final Pitch

STORY emphasis

- Fire fighting support vehicle
    - reduce risk on entry, increase situational awareness
- Environmental probe
    - exploration, record keeping for national parks/etc, land surveying

Design Improvements

- Structural, scale up and of course change materials (possibly flame retardant if fire support)
- Camera vision, GPS tracking

Components

- **Slider**
- **Ultrasonic**
- **Photoresistor**
- **Bluetooth/Wifi Module**

Car Parts (needed)

- [ ] DC motors
    - [ ] Motor stand
    - [x] ~~Wheels~~
    - [x] ~~Axels~~
    - [ ] Screws
- [x] ~~Batteries!! GRAB MORE (9V)~~

- [x] ~~Breadboard~~
- [x] ~~Servo Motors~~
- [x] ~~**Sensors**~~
- [x] ~~Temperature/humidity sensor~~
- [x] ~~Flame Sensor~~
- [x] ~~Ambient Light sensor~~
- [x] ~~Ultrasonic~~

Problem Statement Ideas

- Audiences
    - Students
    - Teaching Staff
    - General Public
    - Community Workers

Purpose Ideas

- Health
- Convenience
- Accessibility
- Environment
- Education

ChatGPT

- Smart Traffic Light system with RF
- Tracking cars with less computation (vs cameras)
- Weather/Life tracking sensor
- RF sensors for fall detection
- Some kind of chatGPT API usage

Ideas

- smart water bottle
- Scaled up greenhouse sensing system
- Bird identification

- Sound level safety checking (plus expansions)
- Modified Running shoes (tracking, form, etc)
- Mini Sensing car

- use of chatGPT within system (API usage)
- Applying a machine learning algorithm to sensor data

Workshop

[Thing files for Drogerdy - Raspberry Pi Controlled Tank Bot by timmiclark - Thingiverse]

[Rc-Rock Crawler Chassis by suisse-crawler - Thingiverse]

TO DO:

- **Code gas sensor**

```
// include libraries
#include <Servo.h>


// initialize constants, variables, and objects
const int trigPin = 9;
const int echoPin = 10;

const int ledPin = 8;

float duration, distance;
float trigger_distance = 10;

int openAngle = 90;
int angle =0;
int angleInc = 1;

//speed of sound in cm/us
float c = 0.0343;

Servo servo1, servo2;
int servo1_Pin = 11; //change later lol
```

```
int servo2_Pin = 12;
int pos1 = 0;
int pos2 = 0;

void setup() {
 servo1.attach(servo1_Pin);
 servo2.attach(servo2_Pin);
 pinMode(trigPin, OUTPUT);
 pinMode(ledPin, OUTPUT);
 pinMode(echoPin, INPUT);

 //start Serial communication
 Serial.begin(9600);
}

void loop() {
 // OUTPUT BURST //
 //starting trigPin low
 digitalWrite(trigPin, LOW);
 delayMicroseconds(2);
 // "burst" output
 digitalWrite(trigPin, HIGH);
 delayMicroseconds(10);
 digitalWrite(trigPin,LOW);
  // DETECT INPUT //
 /* pulseIn() waits to see when echoPin is targetState (aka HIGH),
 which occurs when sound waves hit the receiver, then stores
 the time value how long this is true for
 */
 duration = pulseIn(echoPin, HIGH);
 distance = duration*c/2;

 swivel();
  // CONTROL //
 if(distance <= trigger_distance){

 }

} //close loop

void swivel(){
 int pos = 0;
```

```
if (angle<openAngle){
  pos = angle+angleInc;
  servo1.write(pos);
  angle += angleInc;
}
else if (angle>= openAngle){
  pos = angle-angleInc;
  servo1.write(pos);
  angle -=angleInc;
}

}
```