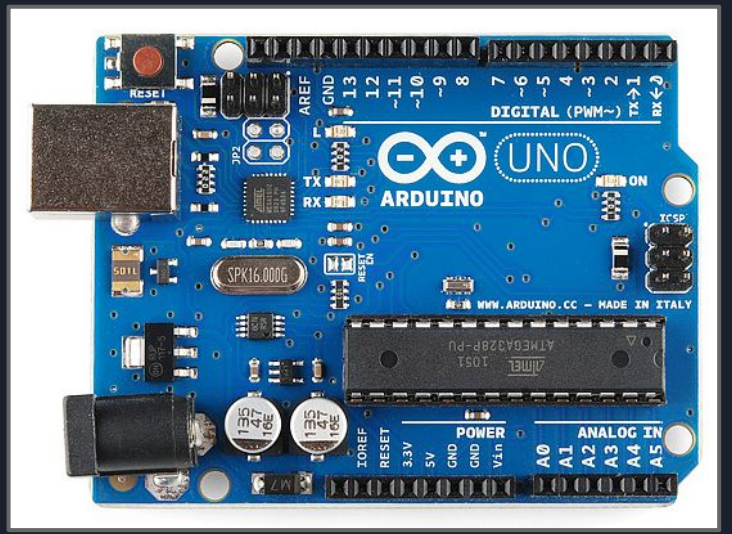
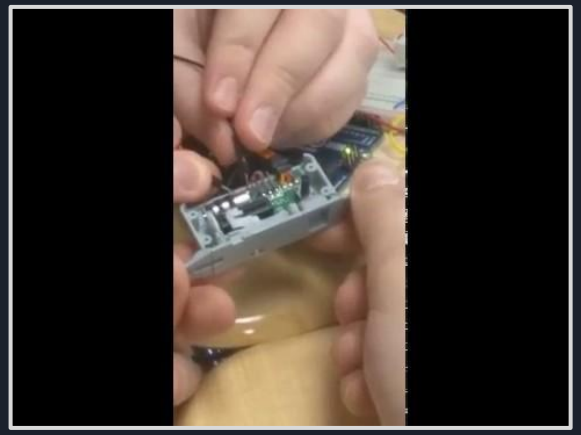
Executive Summary

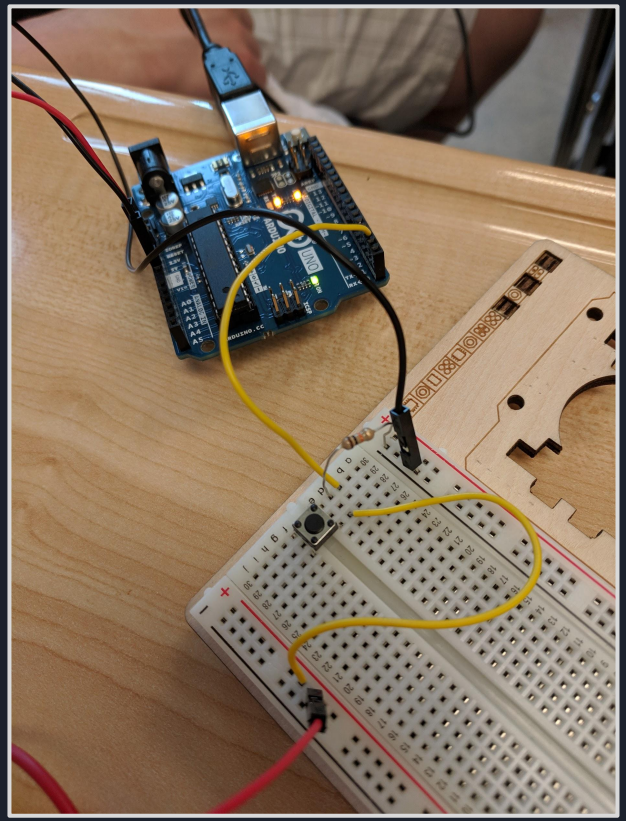
Our original module design was to simply plug the Servoless Payload Release (SPL) into the arduino using its stock cable and a stock port on the arduino. After some research, we discovered that the arduino didn't have a compatible port with the SPL, so we tried splitting the SPL's stock cable and attaching the ends to specific ports on the arduino individually. We tried sending code to get the SPL to actuate, but it didn't work. We looked online, and others had similar issues in the past which they bypassed by opening up the SPL and giving power to the motor directly. We tried sending power to the motor with just a battery and alligator clips, and it successfully actuated. Next we built a circuit and created code to get the SPL to actuate with the push of a button on the breadboard, and were successful. Next we created a circuit with a bluetooth module based off the button circuit, and we merged some bluetooth module code with the button code. We tried different variations of this hybrid circuit, but It never ended up working. We then found a different style of bluetooth circuit online and modified it and our code to get it to suit our needs. Finally, after ironing out a few bugs, we were successful! We were able to have a device with a bluetooth terminal connect to the module and actuate the SPL remotely, meaning someone connected can remotely drop a payload while a drone with the module is in flight.

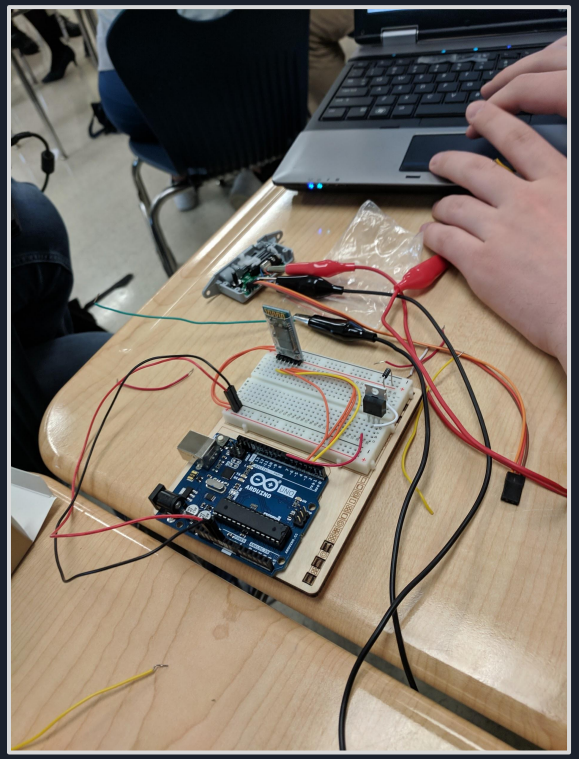
Dropping Mechanism Components



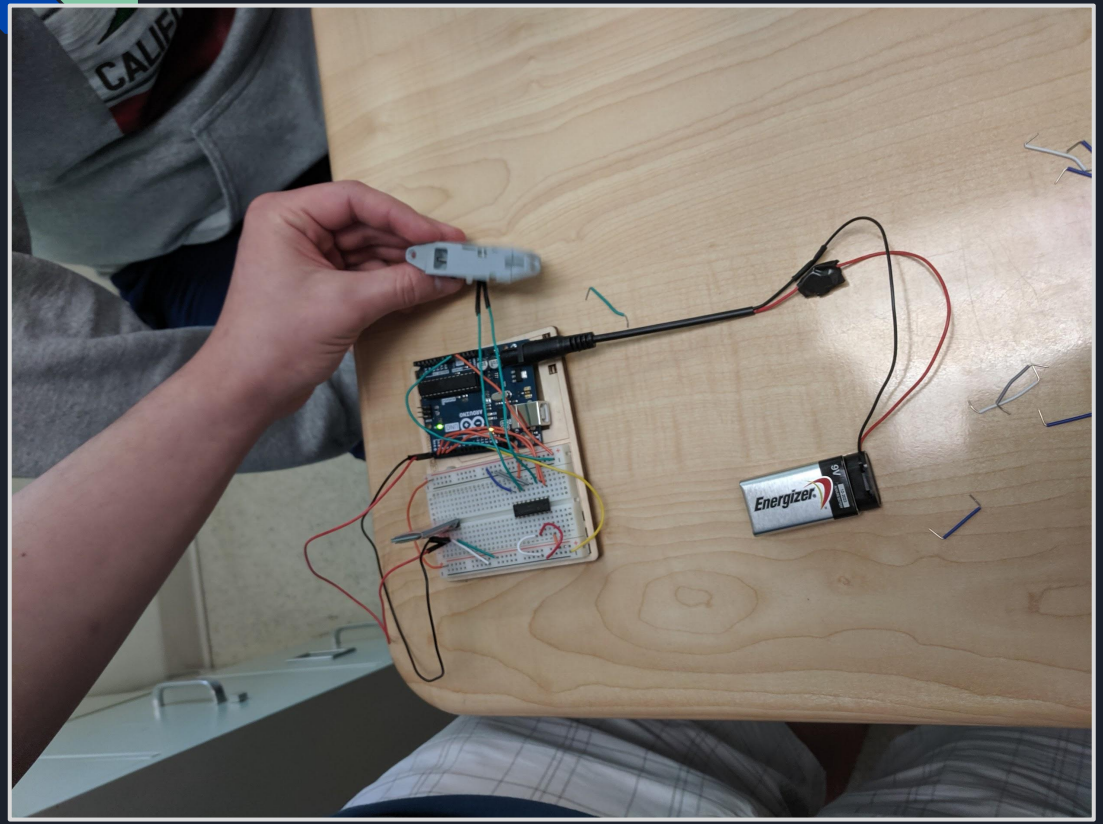
Prototypes





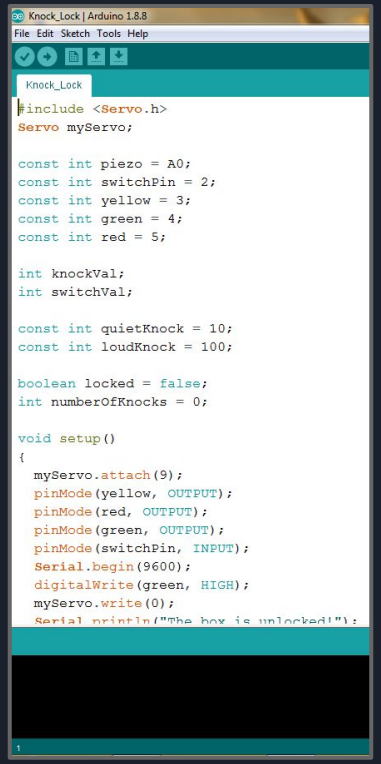


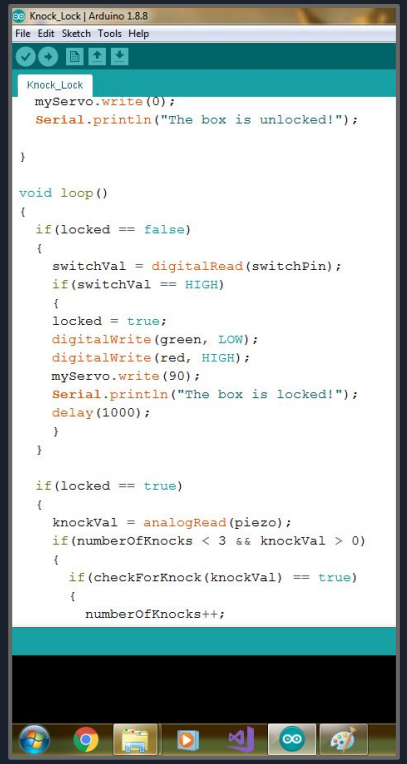
Working Mechanism Design

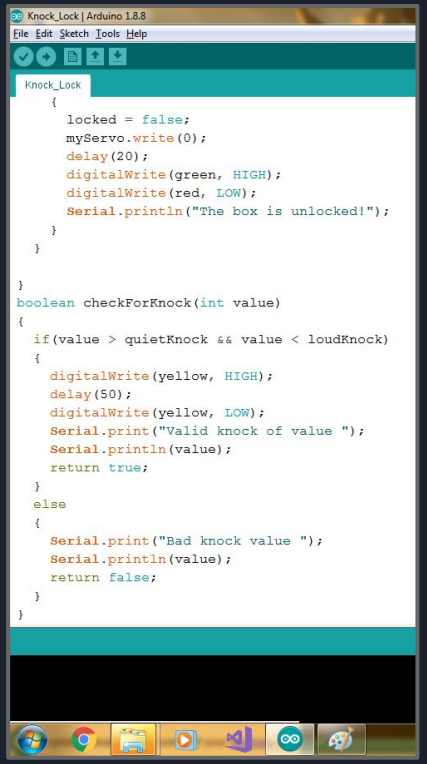


https://twitter.com/i/broadcasts/1djxXpDZLadxZ

Safe-D Code First Test







Safe-D Code Current Iteration

```
/*
 * Control DC motor with Smartphone via bluetooth
 * created by Rui Santos, https://randomnerdtutorials.com
*/

int motorPin1 = 3; // pin 2 on L293D IC
int motorPin2 = 4; // pin 7 on L293D IC
int enablePin = 5; // pin 1 on L293D IC
int state;
int flag=0;        //makes sure that the serial only prints once the state

void setup()
{
    // sets the pins as outputs:
    pinMode(motorPin1, OUTPUT);
    pinMode(motorPin2, OUTPUT);
    pinMode(enablePin, OUTPUT);
    // sets enablePin high so that motor can turn on:
    digitalWrite(enablePin, HIGH);
    // initialize serial communication at 9600 bits per second:
    Serial.begin(9600);
}

void loop()
{
    //if some date is sent, reads it and saves in state
    if(Serial.available() > 0)
    {
      state = Serial.read();
      flag=0;
    }
    // if the state is '0' the DC motor will turn off
    if (state == '0')
    {
        digitalWrite(motorPin1, LOW); // set pin 2 on L293D low
        digitalWrite(motorPin2, LOW); // set pin 7 on L293D low
        if(flag == 0)
        {
          Serial.println("Motor: off");
          flag=1;
        }
    }
    // if the state is '1' the motor will turn right
    else if (state == '1')
    {
        digitalWrite(motorPin1, LOW); // set pin 2 on L293D low
        digitalWrite(motorPin2, HIGH); // set pin 7 on L293D high
        if(flag == 0)
        {
          Serial.println("Motor: right");
          flag=1;
        }
    }
    // if the state is '2' the motor will turn left
    else if (state == '2')
    {
        digitalWrite(motorPin1, HIGH); // set pin 2 on L293D high
        digitalWrite(motorPin2, LOW); // set pin 7 on L293D low
        if(flag == 0)
        {
          Serial.println("Motor: left");
          flag=1;
        }
    }
}
```