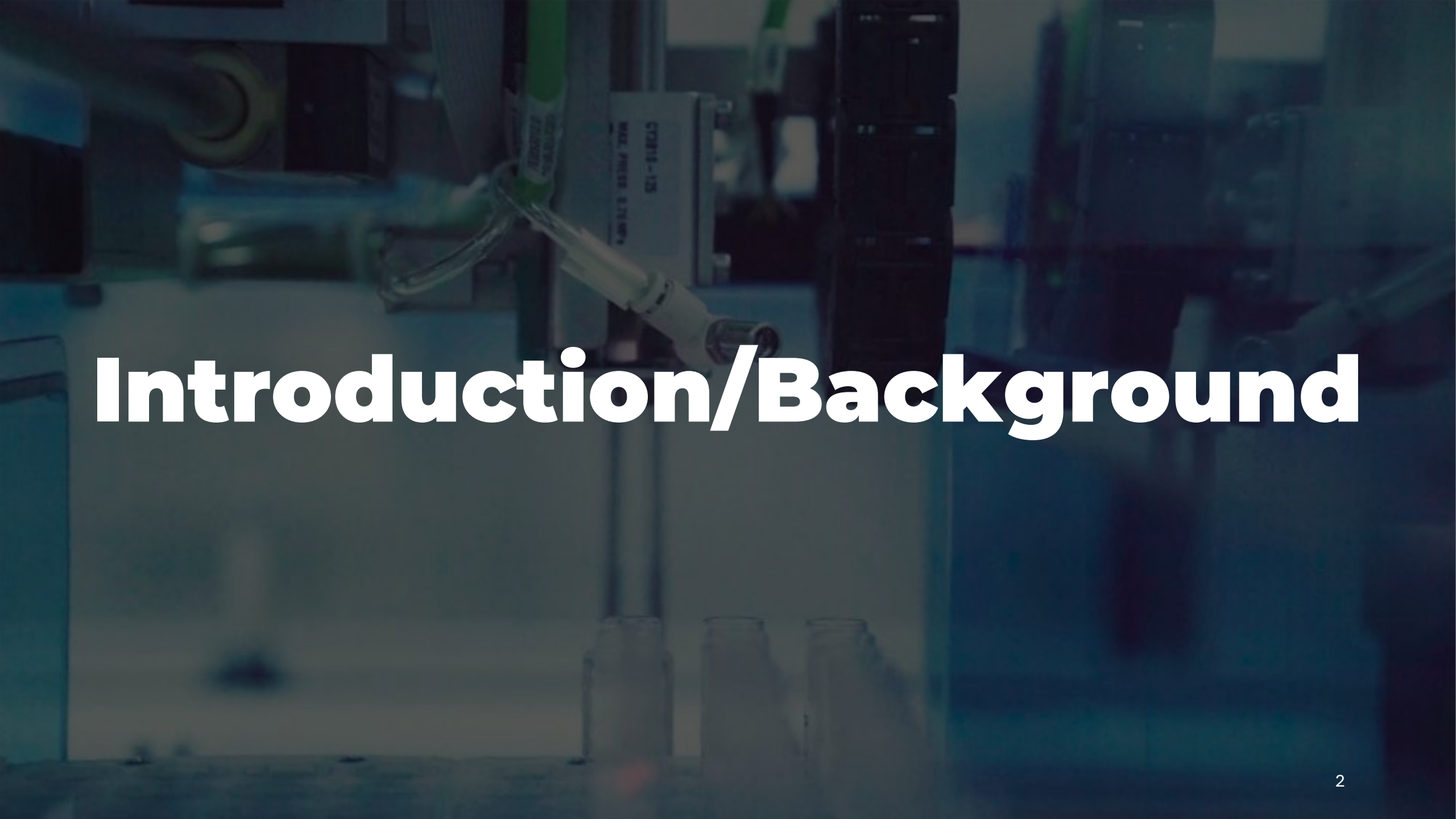


Automated High-Throughput Pipetting Machine





Introduction/Background

High Throughput Screening



Rapidly Identifies Hits in Large Numbers of Samples



Takes Advantage of Automated Equipment



Addresses the Shortcomings of Traditional Drug Discovery

Drug Discovery

Drug Discovery Entails: Identification, Preclinical Tests, Clinical Trials



High Costs and Labour-Intensive Process



Large Quantities of Data Can Be a Bottleneck

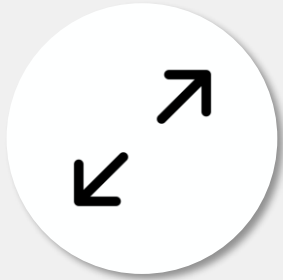


One of the shortcomings of HTS in **drug discovery is the frequent generation of false positives/negatives**

Needs Statement

We aim to create an automated multichannel pipetting device for high throughput screening that is accurate for drug discovery usage.

Our Requirements



**Max pipetting
volume of 100
 μL**



**Minimum
pipetting
volume of 10 μL**



**Pipetting
speed must
be $>0.1 \mu\text{L/s}$**



**Pipetting
speed must
be $<14 \mu\text{L/s}$**

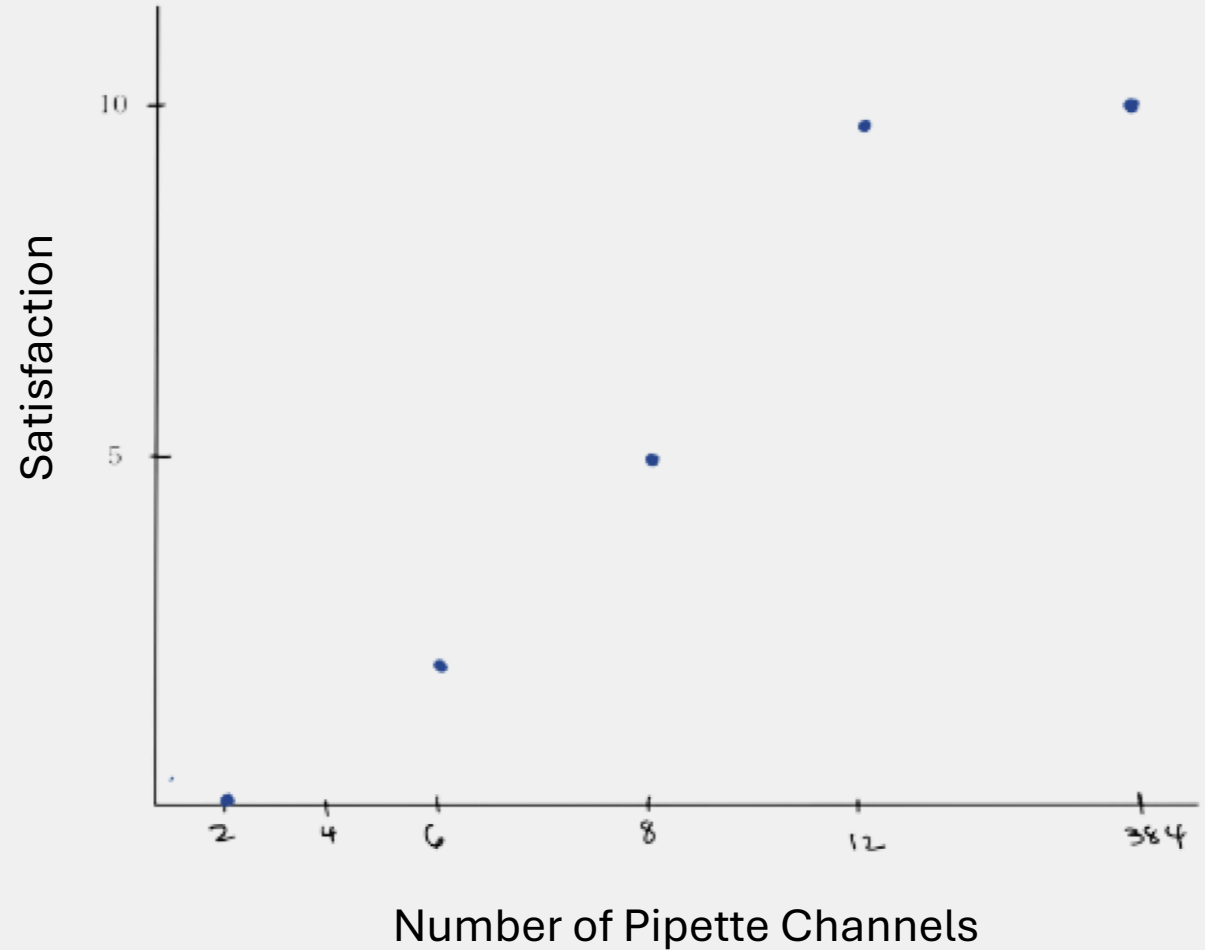


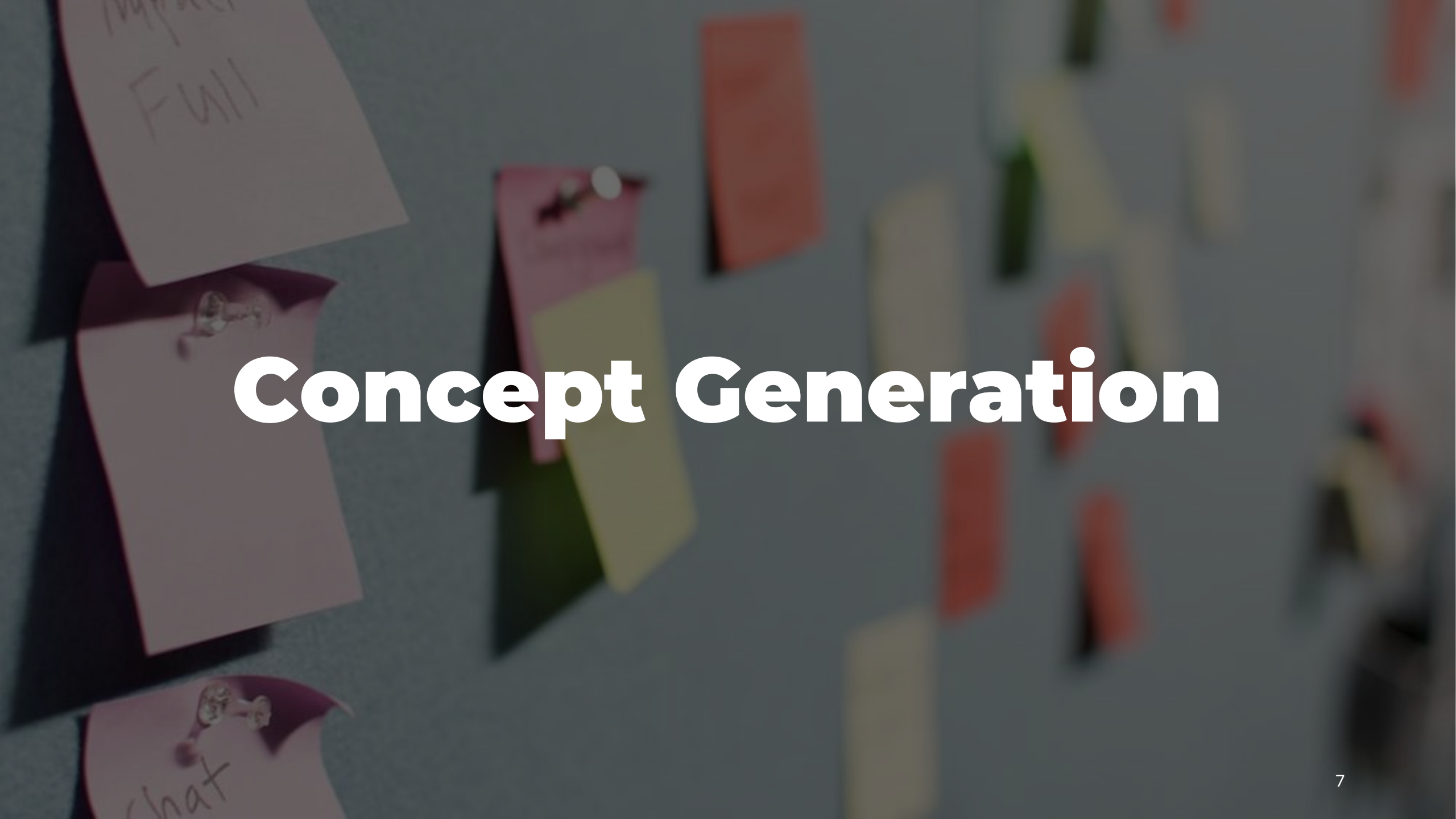
**Pipette tip distance
to bottom of 96 well
plate must not
exceed 1.35mm**

Evaluation Criteria

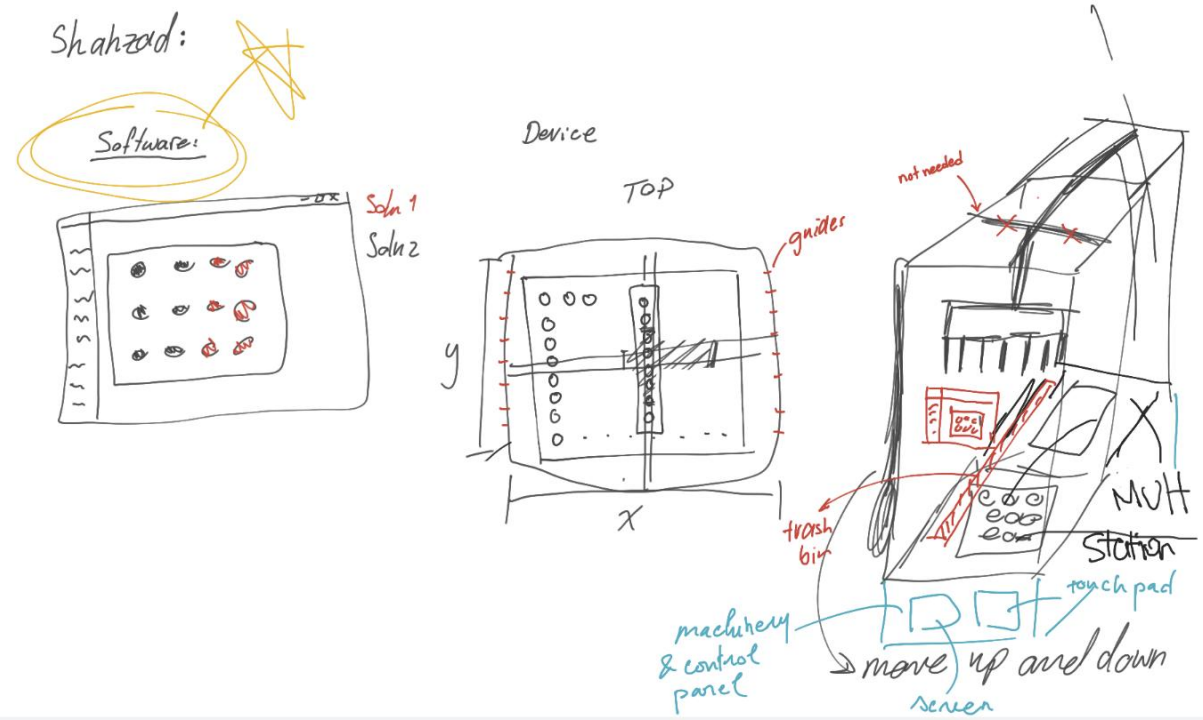
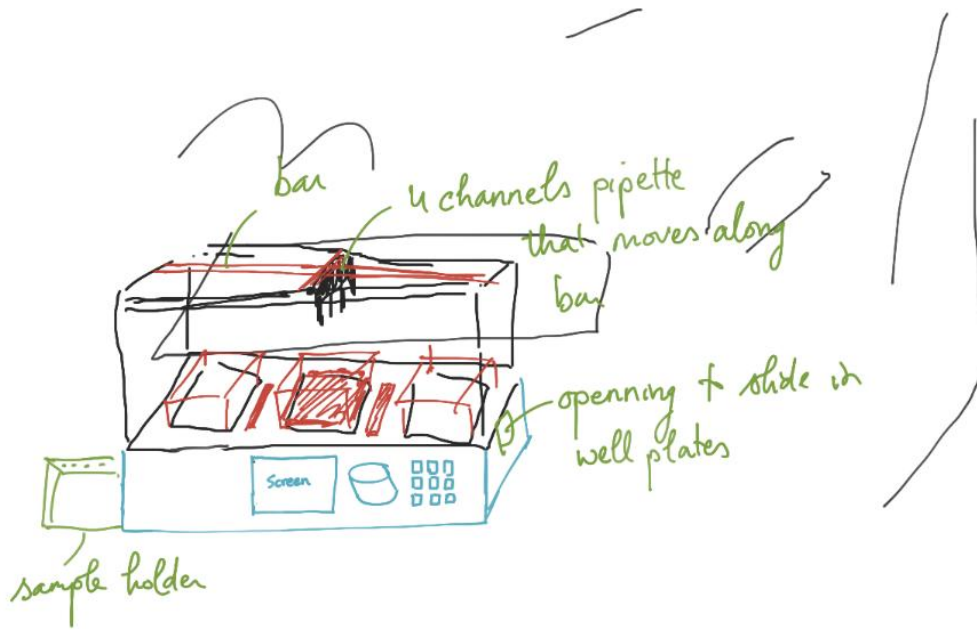
A greater number of **pipetting channels** would allow for greater throughput

Satisfaction Vs. Number of Pipette Channels





Concept Generation



Concept #1 – Papa's Pipetteria



Smaller form factor, feasible design



Only 4 pipetting channels

Concept #2 – The Regurgitator



Large number of pipetting channels



Large and heavy, expensive to make commercially

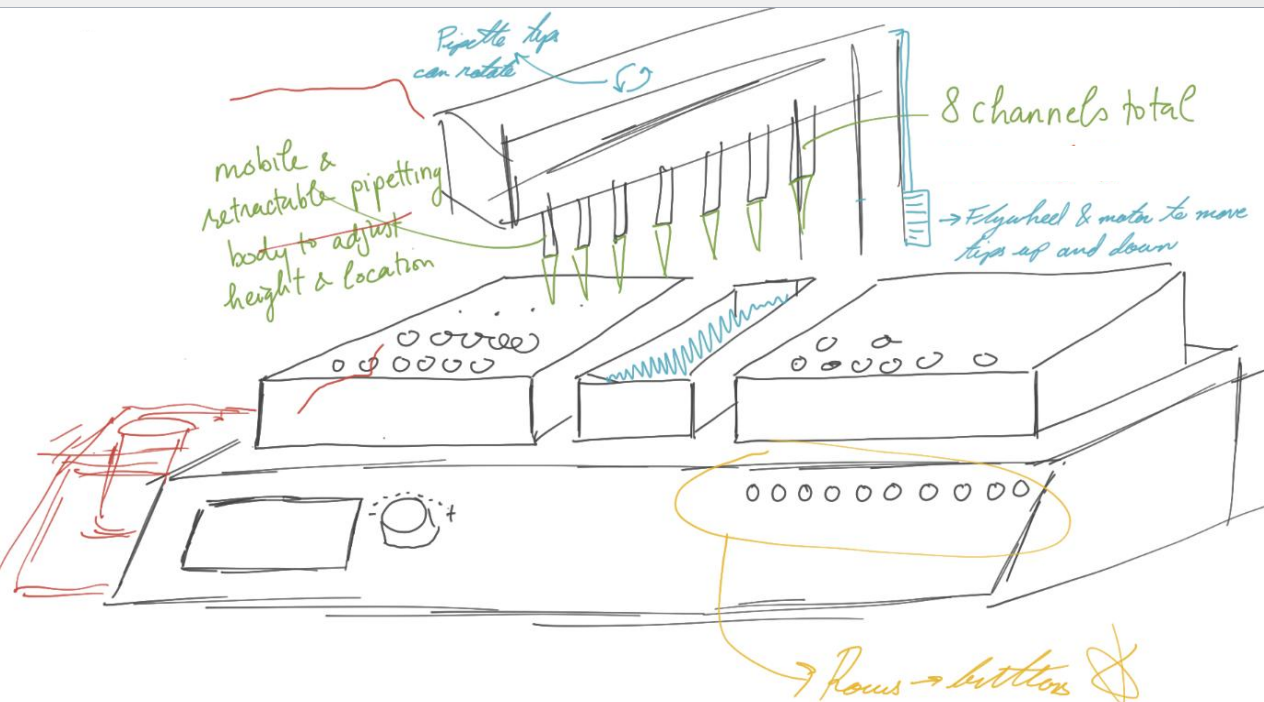
Concept #3 – The Ejectinator



Feasible design, good number of pipetting channels



Requires a large number of operations from the user



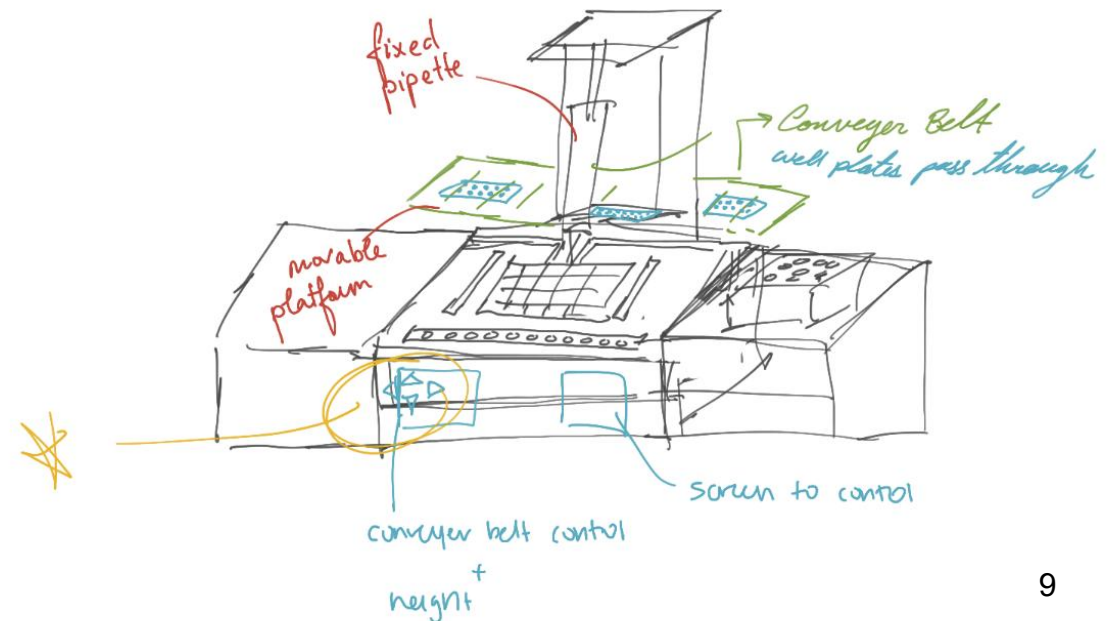
Concept #4 – Radio Box Pipette



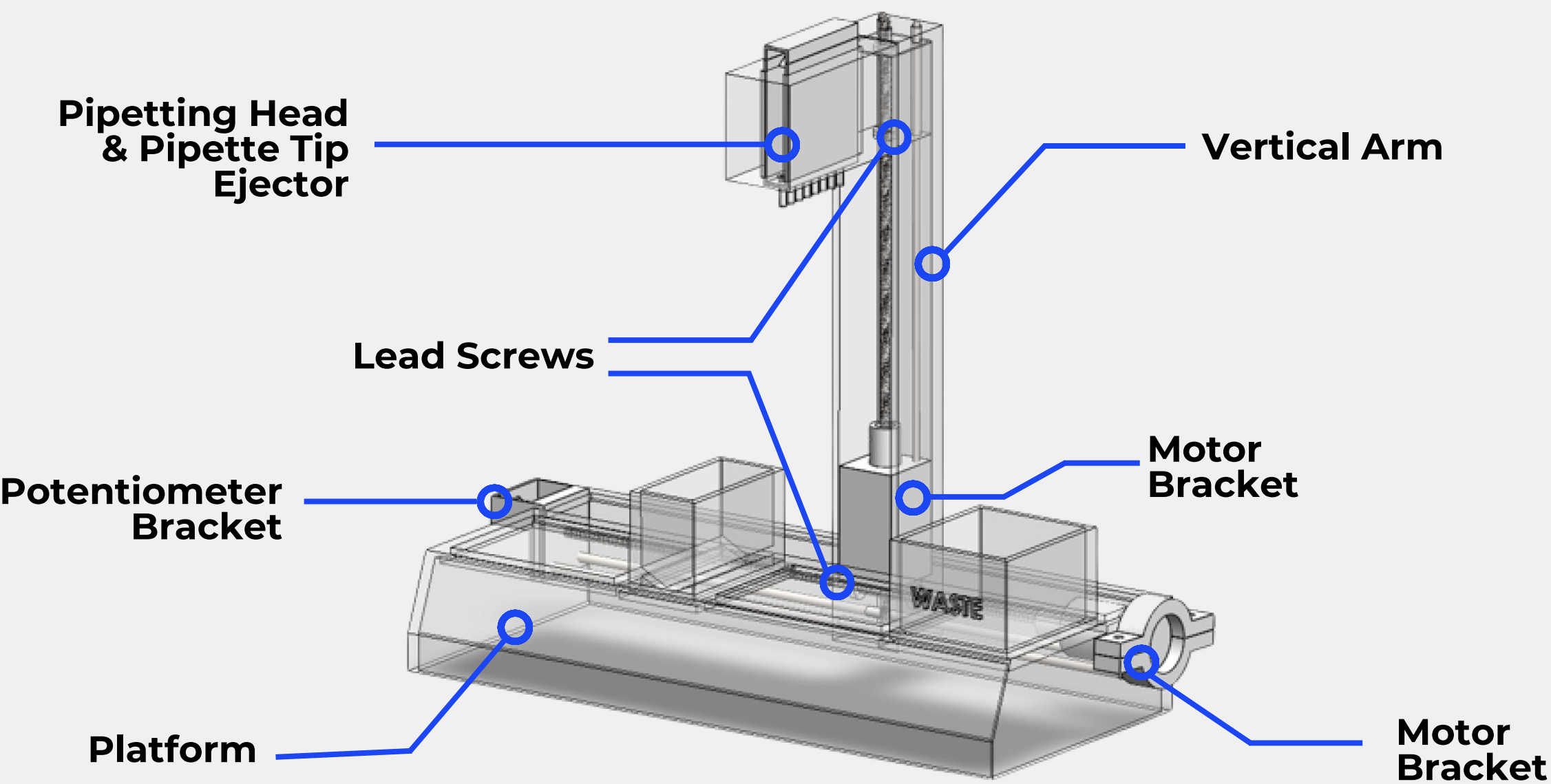
Relatively cheaper to construct due to fixed pipette



Only one pipetting channel



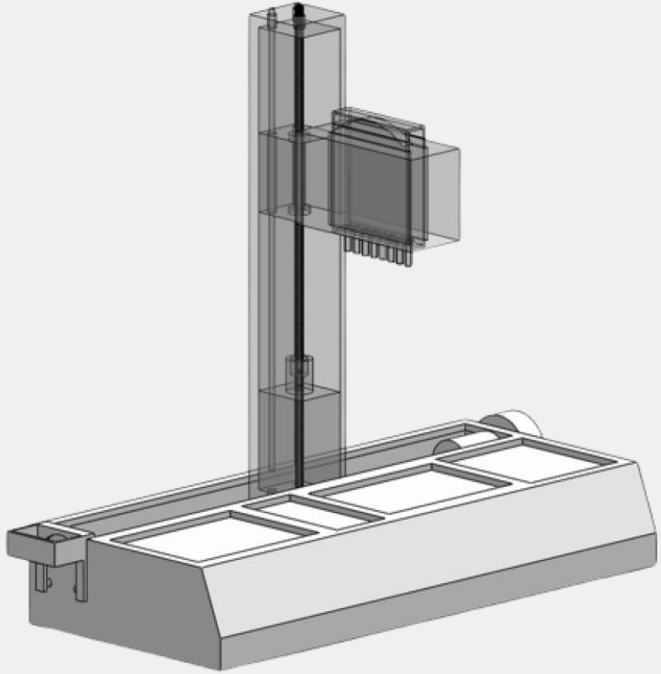
Final Concept



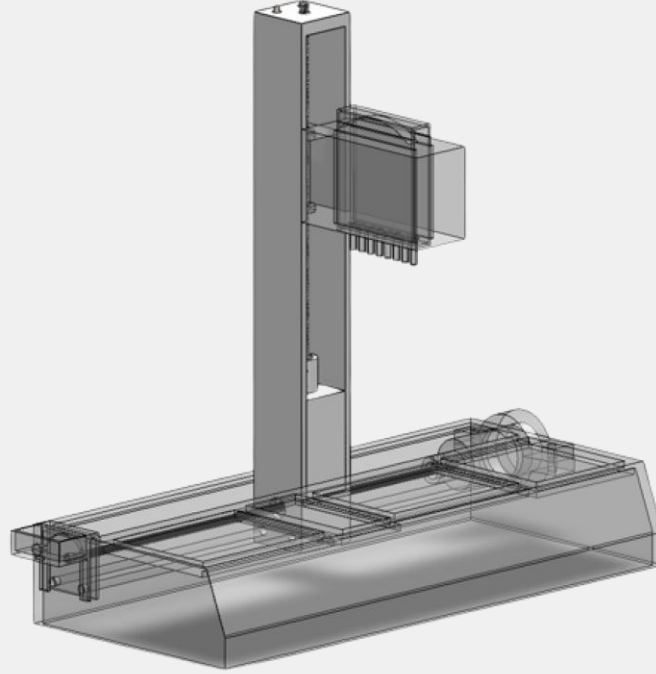
Prototyping

A collage of prototyping tools and materials. In the background, a 3D printer with a transparent enclosure is visible. To the left, a laptop displays a 3D model of a mechanical part. In the foreground, there are several 3D printed objects: a red gear assembly, a white cylindrical part, a colorful topographical model, and two green trays containing small, colorful printed parts. To the right, a laser cutter is partially visible, and a wooden board with a laser-cut design lies on the desk. Safety glasses and a small digital display are also present.

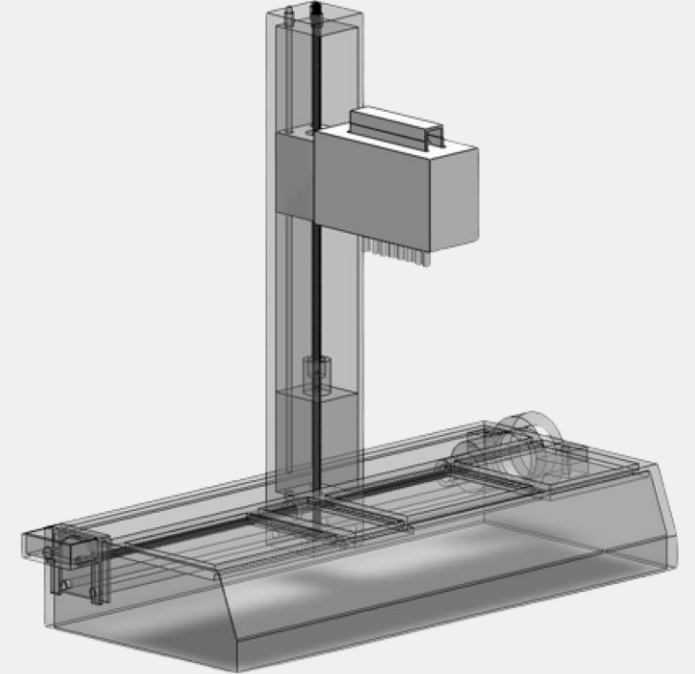
Prototype Details



Main Device **Platform/Body**

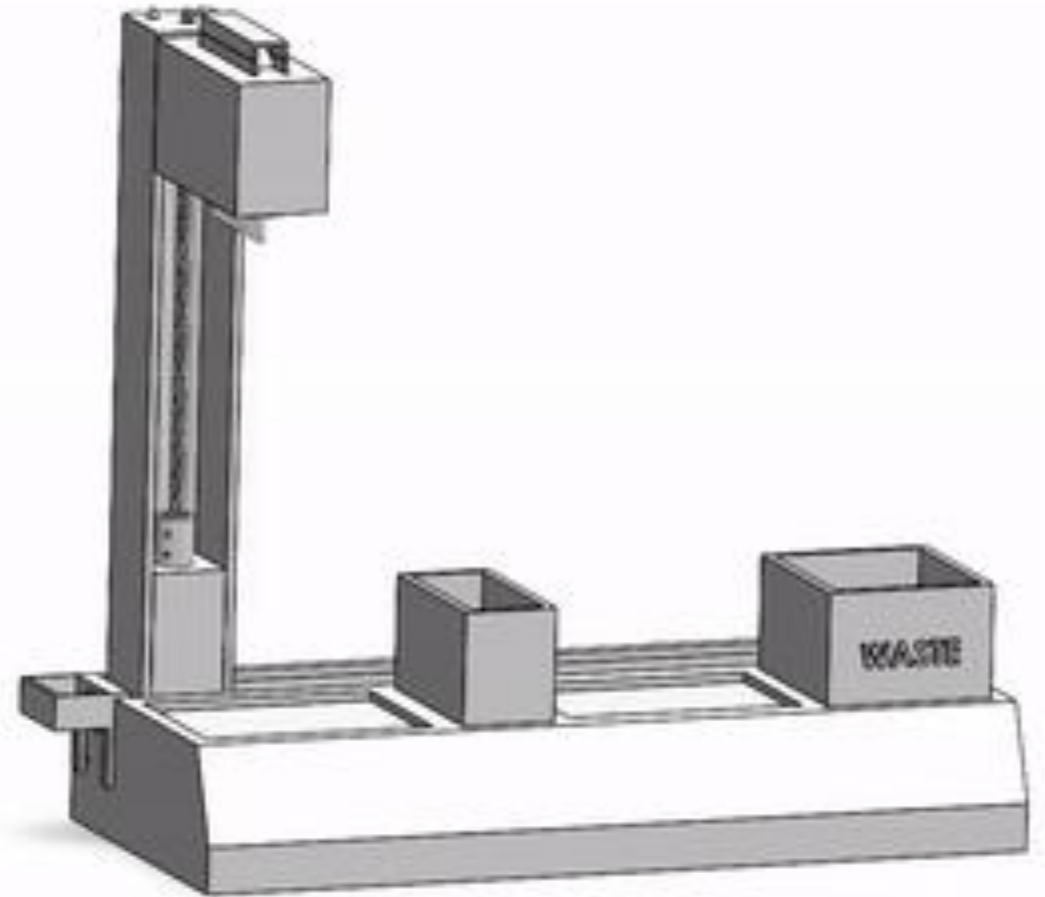
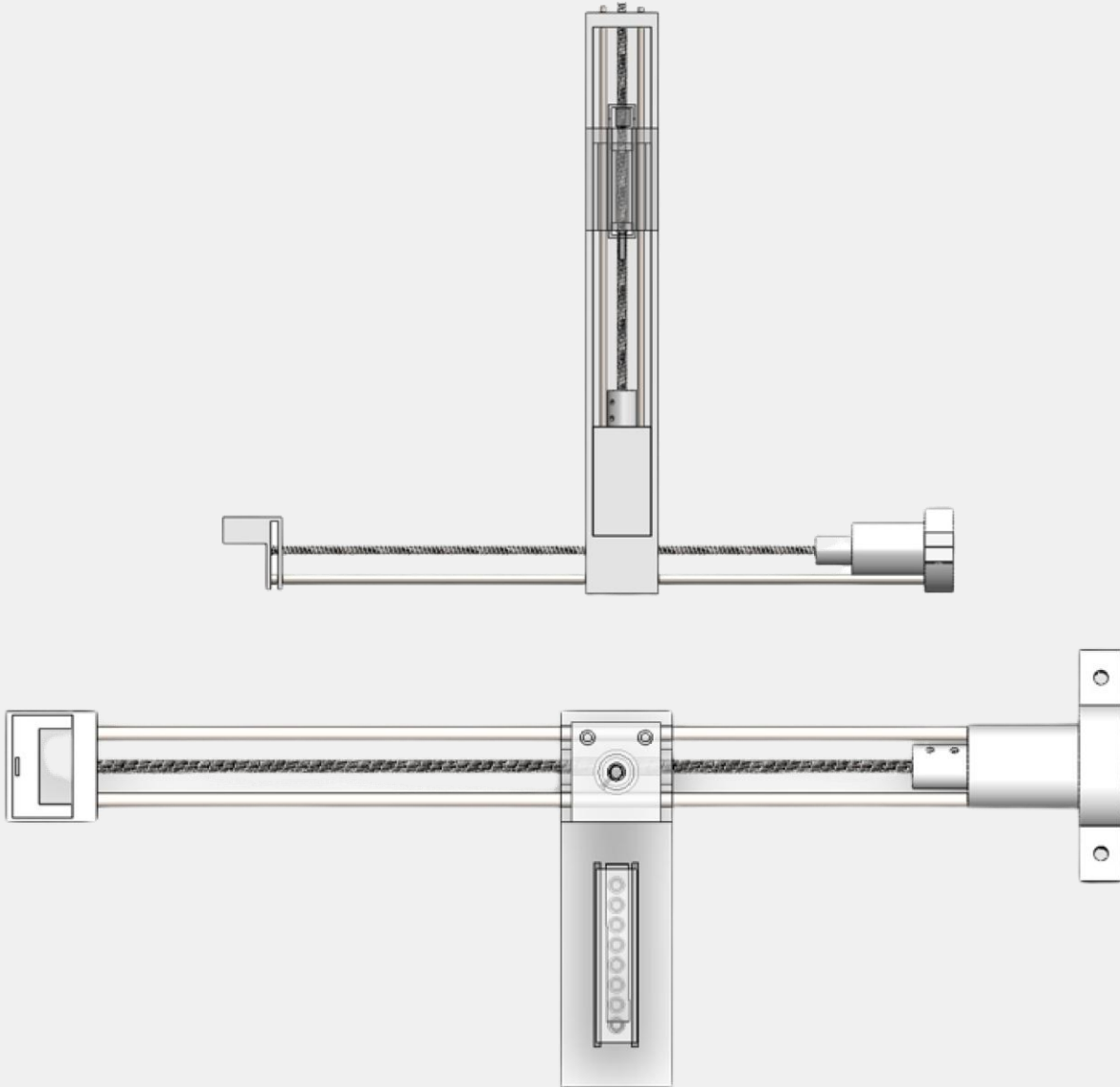


Device **Vertical Arm**



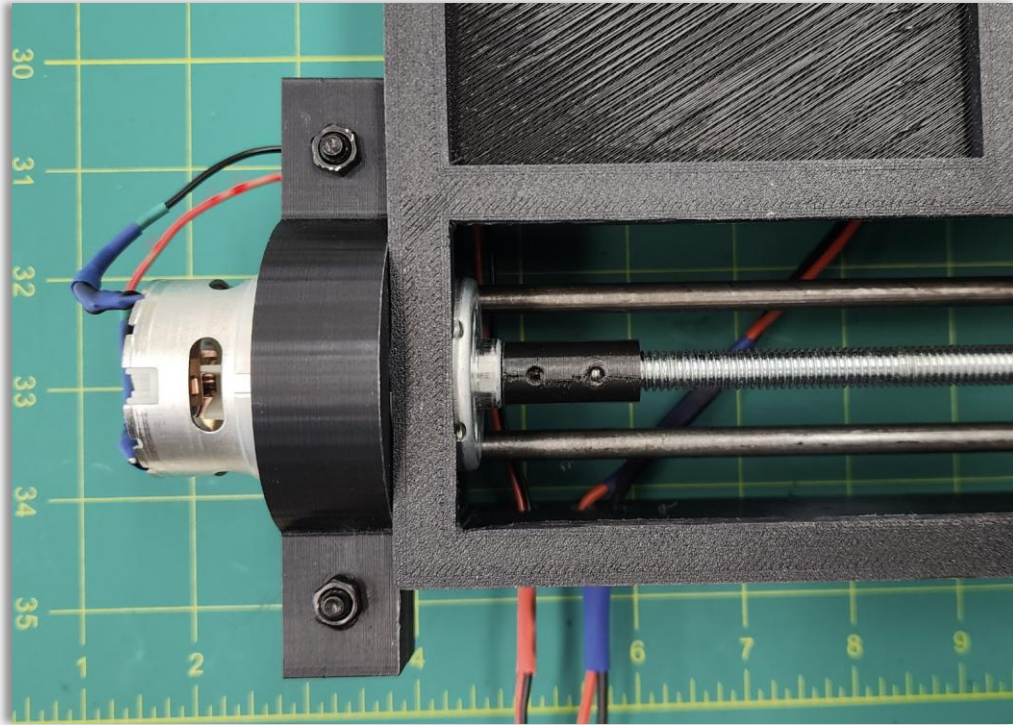
Device Horizontal **Arm**

Prototype Motion

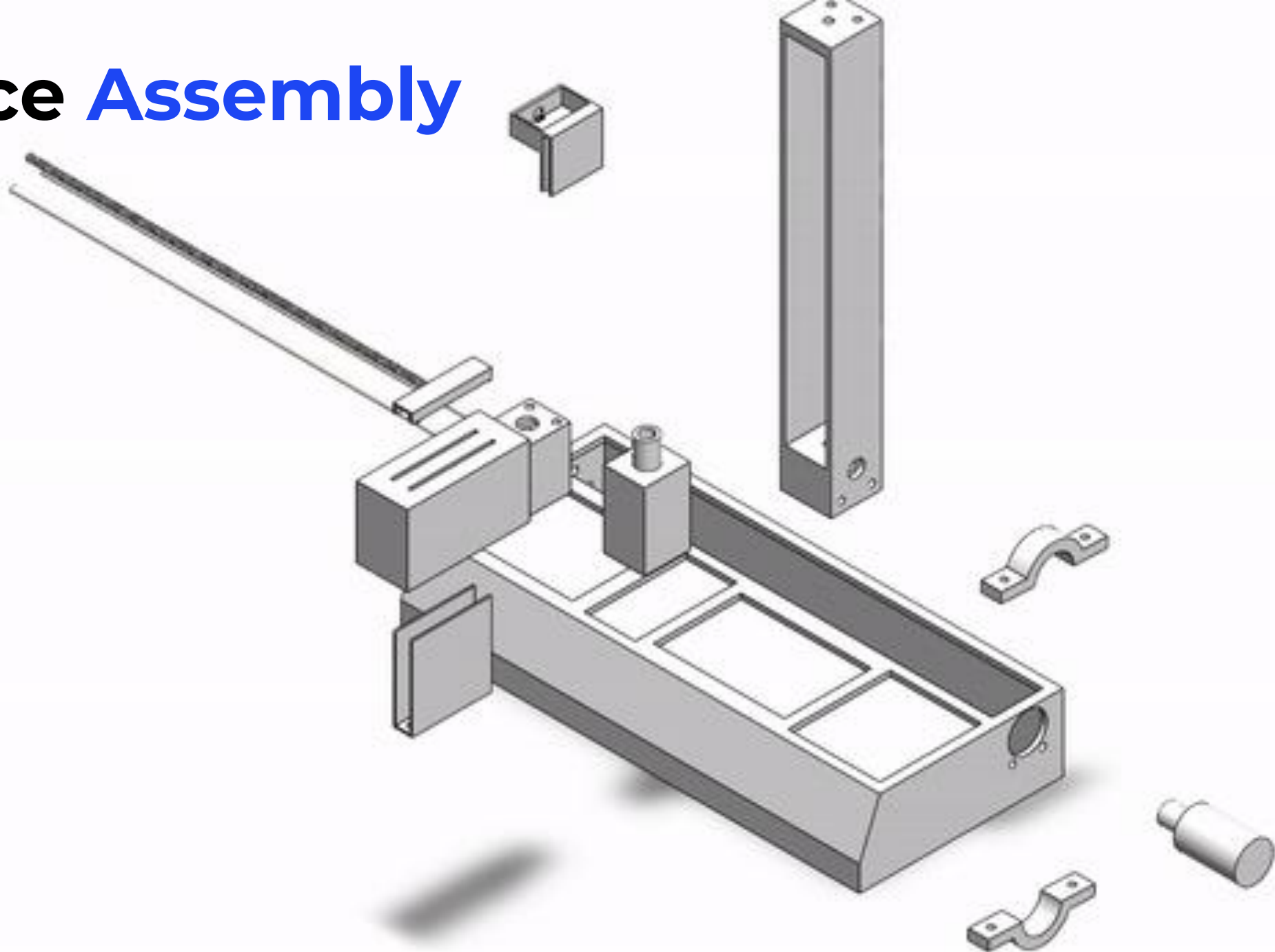


How it Works

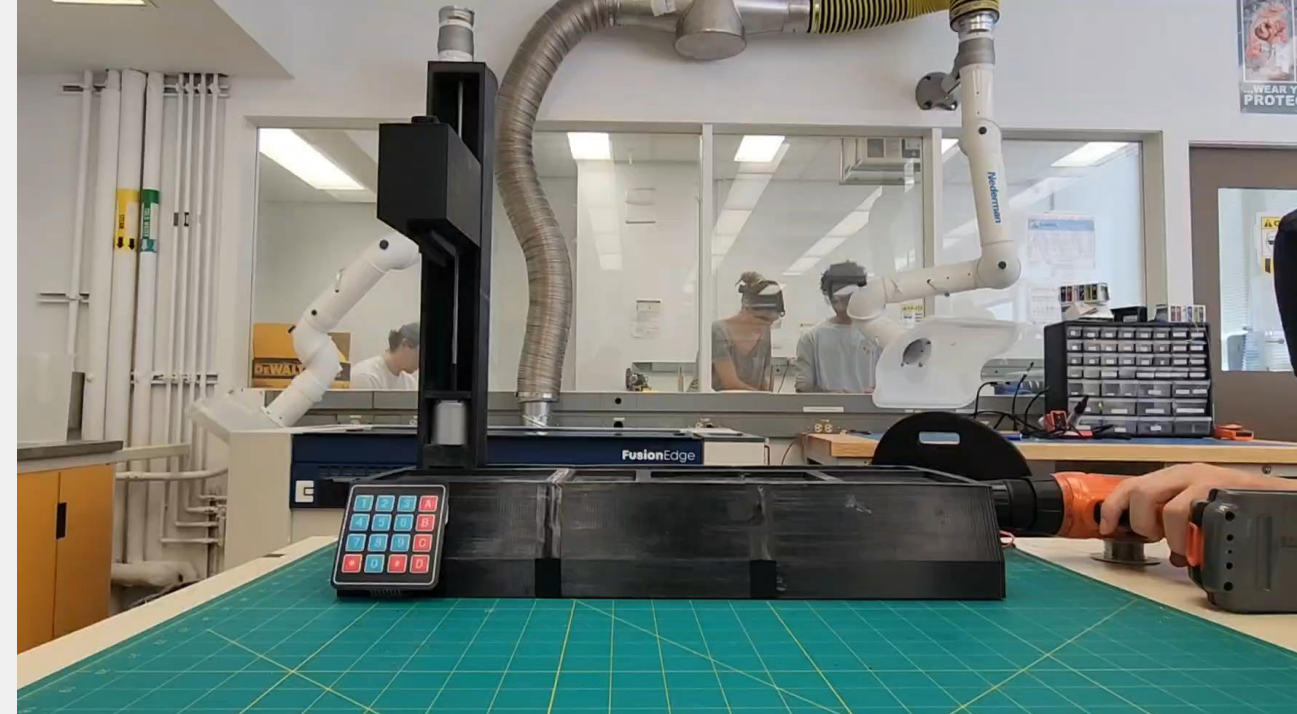
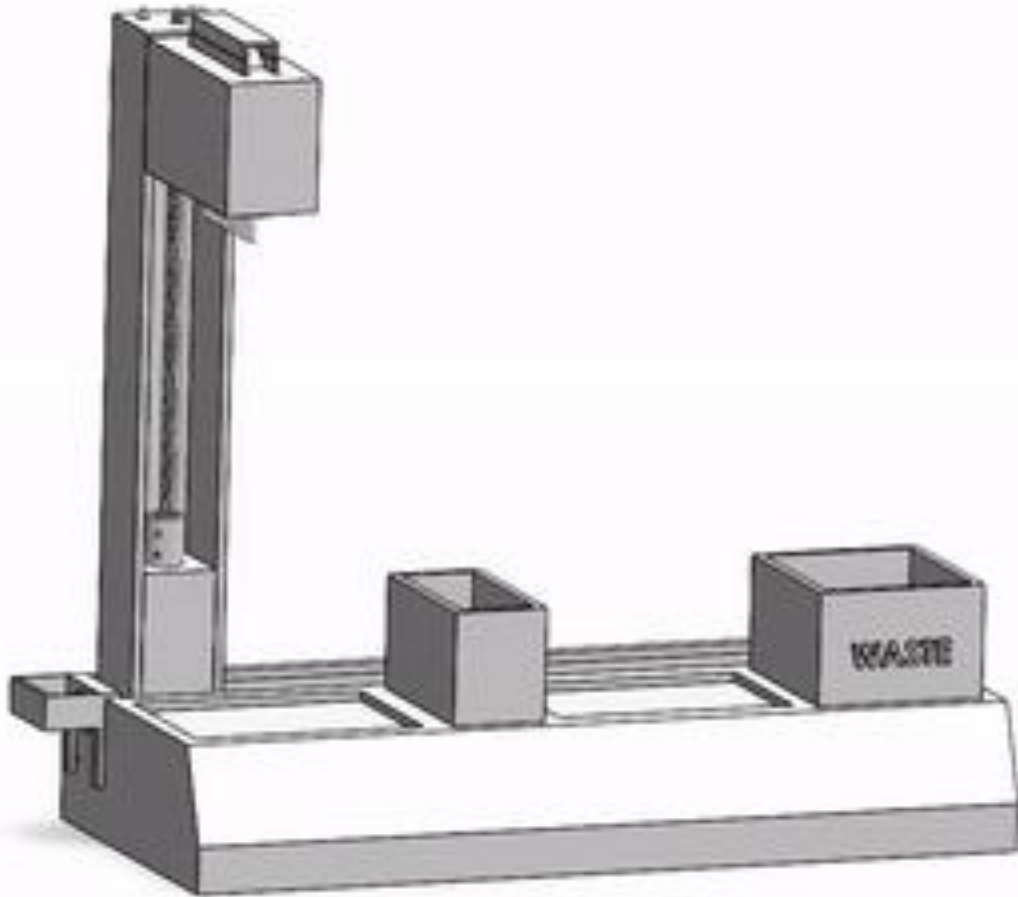
Linear motion is achieved using **lead screws** made with a $\frac{1}{4}$ inch threaded rod that pushes and pulls the vertical arm as the rod rotates



Device Assembly



Testing



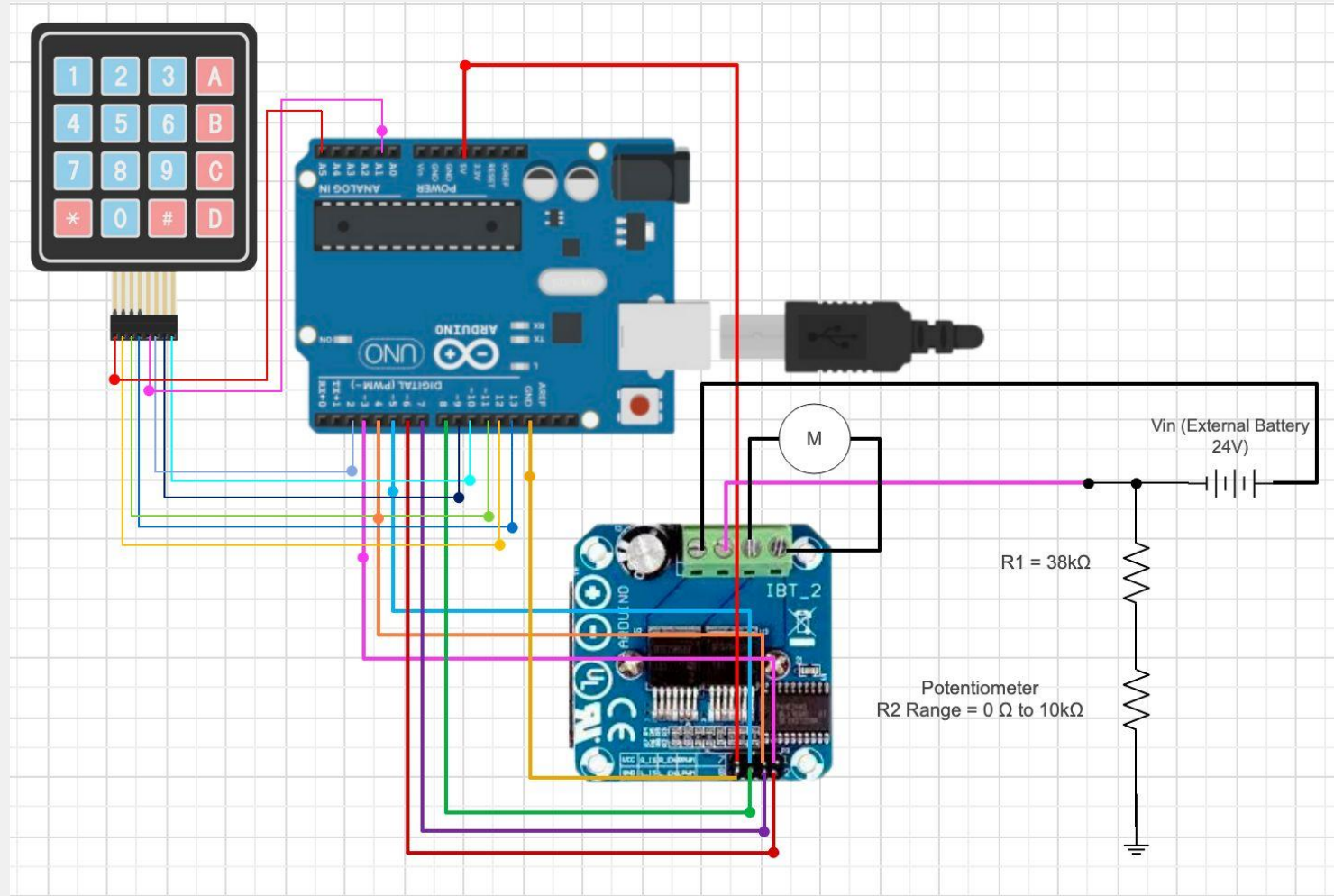
Circuitry/Programming

A photograph of an electronics workspace. In the foreground, a black digital multimeter with a red probe is on the left. To its right is a breadboard populated with various electronic components like resistors, capacitors, and integrated circuits. In the background, there are other electronic components and a power supply unit. The entire image is overlaid with a semi-transparent dark grey filter, and the text 'Circuitry/Programming' is centered in a large, white, bold font.

Circuit Design

- **Main components:** DC motors and BTS7690 motor drivers, Keypad, Arduino UNO, Voltage divider, Rotary Potentiometer
- **Sensing circuit** used to obtain horizontal positional information of the vertical arm
- Voltage divider **Vs**. Wheatstone Bridge

*Circuit diagram for horizontal movement of vertical arm



Programming

```
178 int object_turn = object_distance / cm; //Conversion of distance into number of turn
179
180 //returns the calculated supposed number of turns for the object
181 return object_turn;
182 }
183
184
185 //Counter function that counts the number of 360 degrees turns made by the shaft of the linear potentiometer
186 //Checks if the measured angle is 0
187 //Calls the functions above to check the angle
188 bool ShaftTurn_Counter(int count, int calibratedturn){
189     //Checks if the measured angle is 0
190     //Calls the functions above to check the angle
191     while(count<=calibratedturn){
192         //Serial.println(Angle_Conversion(Measure_Analog(potentiometerPointer))); //for debugging purposes
193         if (Angle_Conversion(Measure_Analog(potentiometerPointer)) == 0) {
194             //Serial.println(Angle_Conversion(Measure_Analog(potentiometerPointer))); //for debugging purposes
195             Serial.println(count); //for debugging purposes
196
197             //Makes sure that the count only increases if the appearance of the angle happens at a 2 second interval from each other
198             if (millis() - recentTime >= 600) {
199                 count++; // Increase count
200                 recentTime = millis(); // Updating the time
201             }
202         }
203     }
204     Serial.println("finished"); //for debugging purposes
205 }
```

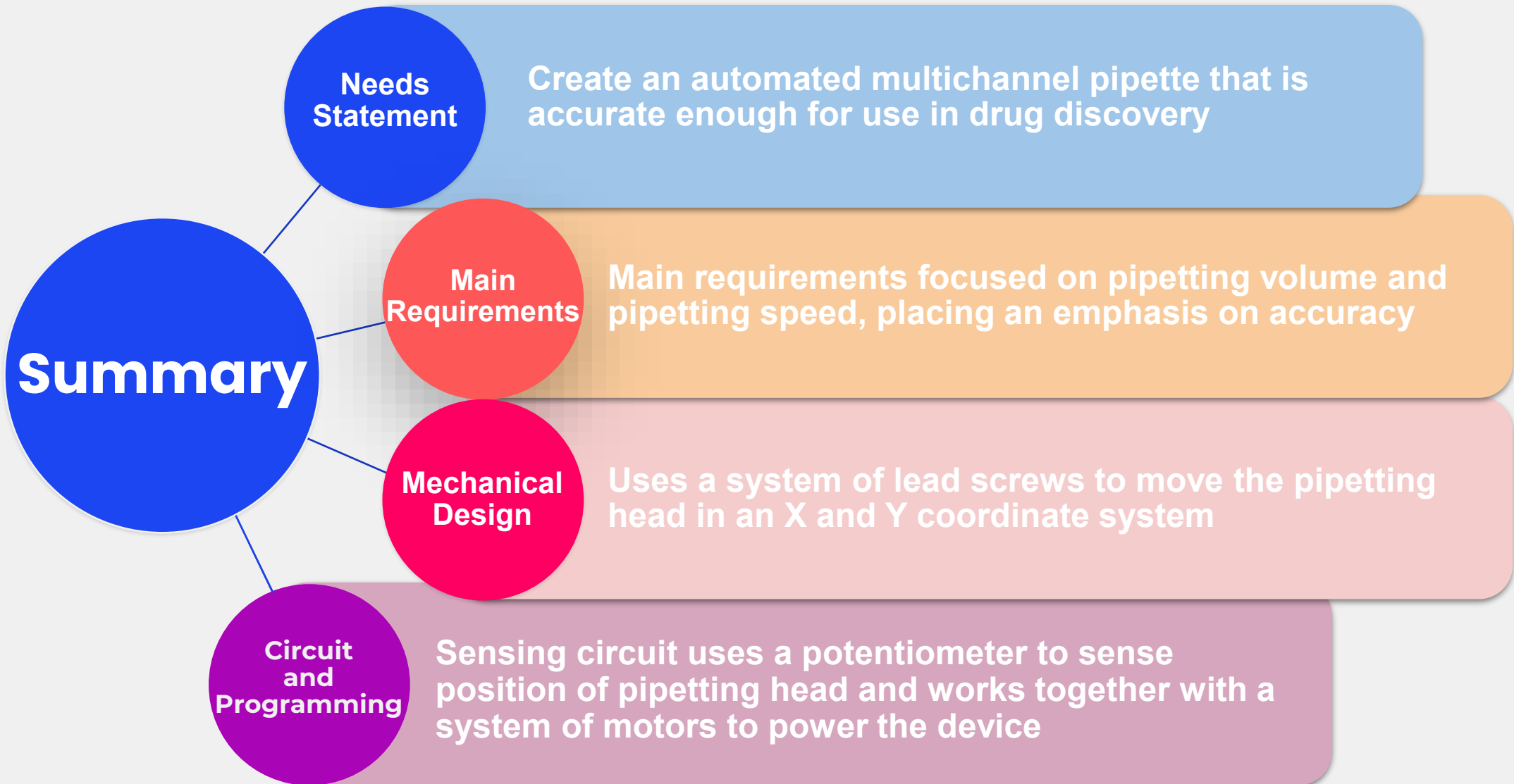
Output Serial Monitor

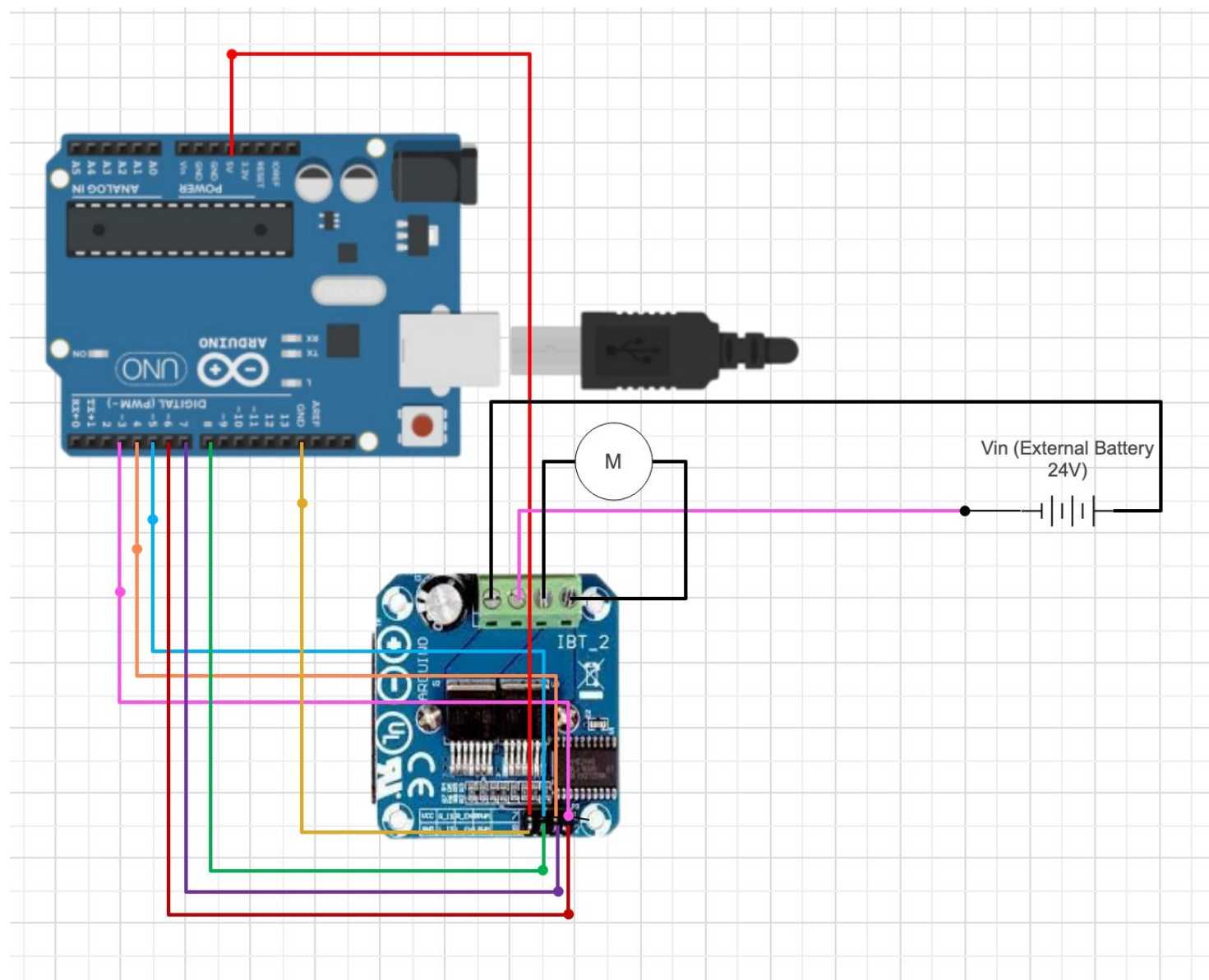
Compiling sketch...

Ln 182, Col 7 Arduino Uno on /dev/tty.usbmodem142201

MacBook Pro

***Motor is running from when the keypad is pressed up until it falls out of the hand**





Weighted decision matrix													
Evaluation Criteria	Weight	Radiobox		The Ejectinator		Papa's Pipetteria		The Regurgitator		Clip On Rotary		Final Concept	
		raw	weighted	raw	weighted	raw	weighted	raw	weighted	raw	weighted	raw	weighted
Number of Pipetting Channels	0.25	5	1.25	5	1.25	1	0.25	9	2.25	5	1.25	5	1.25
Number of Operations	0.2	0	0	2	0.4	4	0.8	2	0.4	4	0.8	3	0.6
Cost	0.25	6	1.5	8	2	0	0	0	0	7	1.75	8.5	2.125
Speed of Aspiration	0.15	4.5	0.675	4.5	0.675	7	1.05	2.3	0.345	4.5	0.675	4.5	0.675
Weight	0.15	6	0.9	8	1.2	0	0	0	0	4	0.6	9	1.35
TOTAL Weighted SCORES	1		4.325		5.525		2.1		2.995		5.075		6