Version 1

I made the frame out of wood, screws and glue
After making the frame, I decided to have four flowers across the bottom because it looked really big.

Pollinator will run across the top rail

Flowers will go here

This is the first motor I designed. It is just the stepper motor to test how it moves. It had rubber wheels inside to drive it along the rail. Here is what I discovered

<u>Issues:</u>

- Speed – It is too slow. It went 0.595cm/s.
- Balance - It was leaning to the side
- Traction – The wheels were not gripping very well
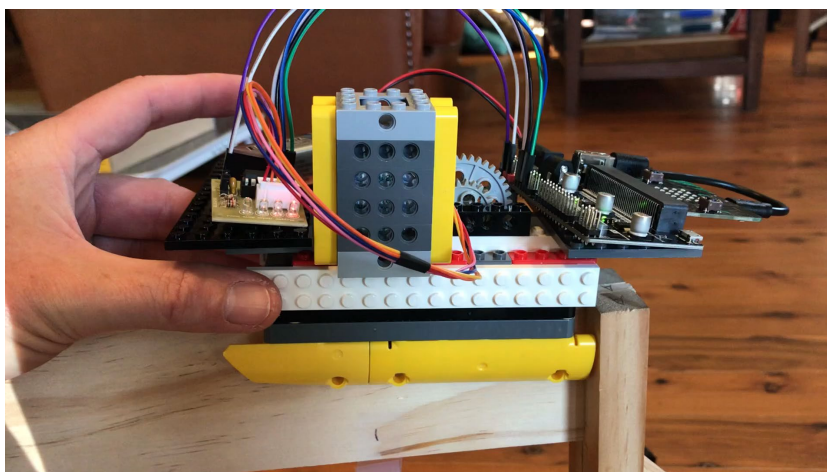
In the next version I need to fix these problems.

Version 2

In this version I needed to make it faster and better balanced on the rail. I changed from a rubber wheel to a gear and put teeth on the rail for the gear to lock into, improving traction. This increased my speed. However, I noticed the gears inside were not always locking. To correct the slipping gears, I changed the drive mechanism from just gears driving each other to a chain drive to allow for the slight misalignment between the stepper motor and the Lego gears.
To fix the balance problem, I made the side supports longer which seemed to fix it.

The next step is adding the servo motor for the pollinating arm.
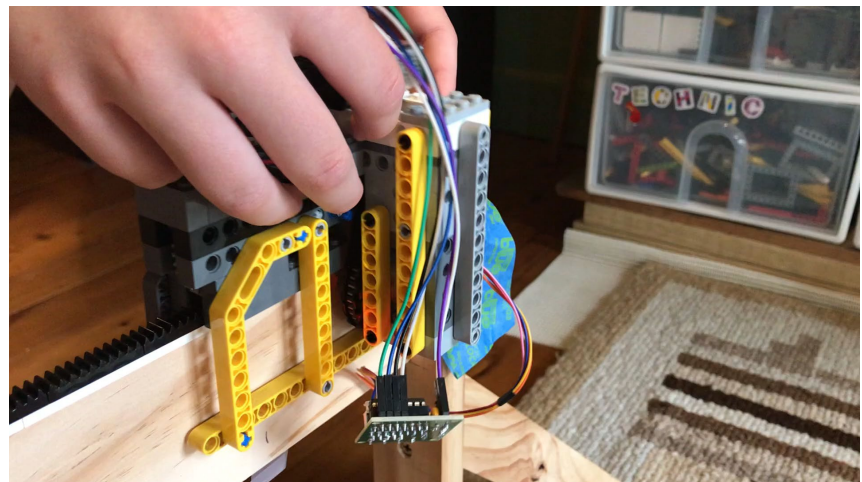
VERSION 3

In this version I added the servo motor.
First I wrote the code to drive the servo then move the stepper motor.

I downloaded the code onto the microbit. I tried it and nothing happened. I was using a 9V battery so I tested the voltage with my dad's multimeter. It was only 7.3V so I got a new battery.
I tried again with the new battery but I still couldn't see anything working.

To try and find the problem I checked the physical connections, tried a simpler code with and without the servo motor, and tried changing the order of the code. While doing this I noticed a gear twitch which told me it was moving, just not enough. I changed the code to increase the stepper motor's movement and this worked, showing me that this was the problem all along.
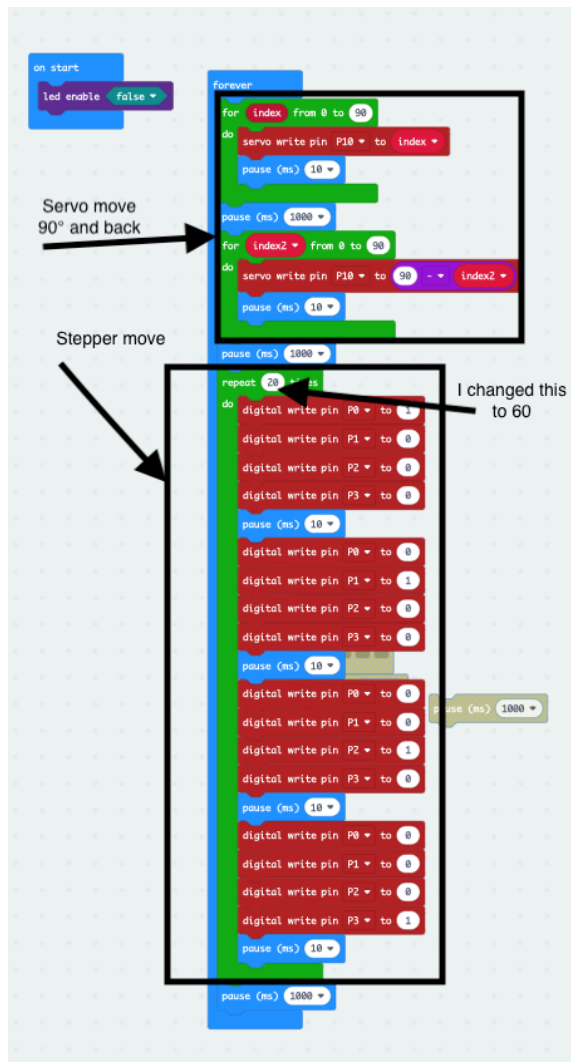
VERSION 4

Added the ultrasonic sensor

When I tried the ultrasonic sensor, it didn't work

The first plan was to use the LCD display to test if the ultrasonic sensor was working.

I wrote the code to do this but for some reason it didn't work. To solve this problem I rechecked the code, connection and pins, googled for alternative solutions but none of these worked. I then checked the ultrasonic sensor and LCD display separately to see if one of these were at fault. It turns out the LCD was the issue.

I rewrote the code so the Ultrasonic sensor reading displays on the LEDs on the microbit instead of the LCD display.

The Ultrasonic sensor will detect the presence of a flower, signalling the stepper motor to stop and the servo motor to begin pollination.

Ultrasonic sensor bit

```
on start
led enable  false ▾

forever
  set distance ▾ to   ping trig P4 ▾
                      echo P10 ▾
                      unit cm ▾
  if  distance ▾ < ▾ 0  and ▾  distance ▾ > ▾ 0  then
    for index from 0 to 90
    do
      servo write pin P5 (write only) ▾ to index ▾
      pause (ms) 10 ▾
    pause (ms) 1000 ▾
    for index2 ▾ from 0 to 90
    do
      servo write pin P5 (write only) ▾ to  90 - ▾ index2 ▾
      pause (ms) 10 ▾
    pause (ms) 1000 ▾
  else if  distance ▾ < ▾ 0  and ▾  distance ▾ > ▾ 0  then ⊖
    start melody  nyan ▾  repeating forever ▾
  else ⊖
    repeat 60 times
    do
      digital write pin P0 ▾ to 1
      digital write pin P1 ▾ to 0
      digital write pin P2 ▾ to 0
      digital write pin P3 ▾ to 0
      pause (ms) 10 ▾
      digital write pin P0 ▾ to 0
      digital write pin P1 ▾ to 1
      digital write pin P2 ▾ to 0
      digital write pin P3 ▾ to 0
      pause (ms) 10 ▾
      digital write pin P0 ▾ to 0
      digital write pin P1 ▾ to 0
      digital write pin P2 ▾ to 1
      digital write pin P3 ▾ to 0
      pause (ms) 10 ▾
      digital write pin P0 ▾ to 0
      digital write pin P1 ▾ to 0
      digital write pin P2 ▾ to 0
      digital write pin P3 ▾ to 1
      pause (ms) 10 ▾
    pause (ms) 1000 ▾
  ⊕
```
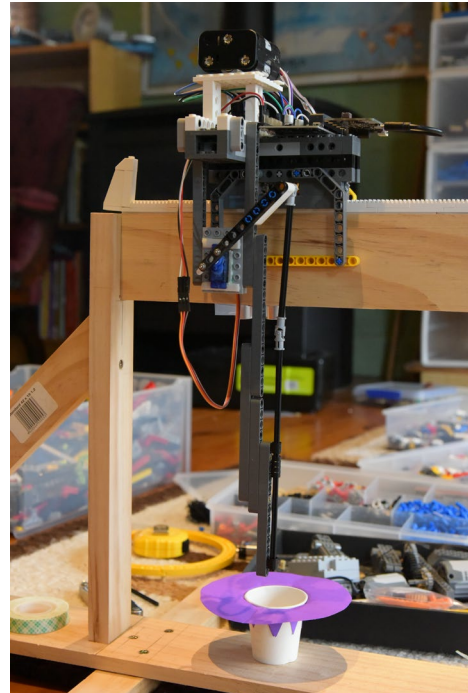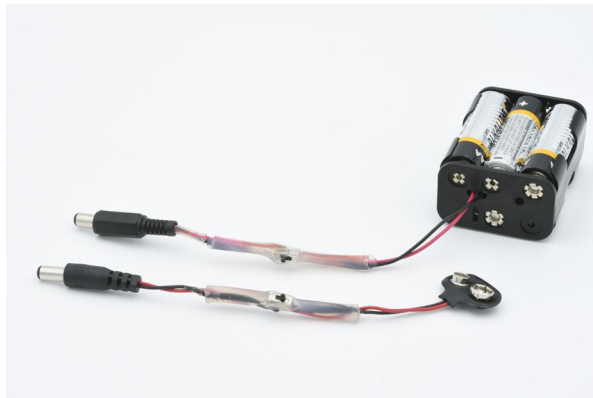
VERSION 5

I changed the power source to six AA batteries in a pack instead of one 9V battery to make it last longer
I also built the flowers and made the pollinator arm. I tested its length and movement. Now it fits perfectly into the flower.
It is mounted on the trolley.
 I have tested ultrasonic sensor distances and adjusted the code so that all components work together.
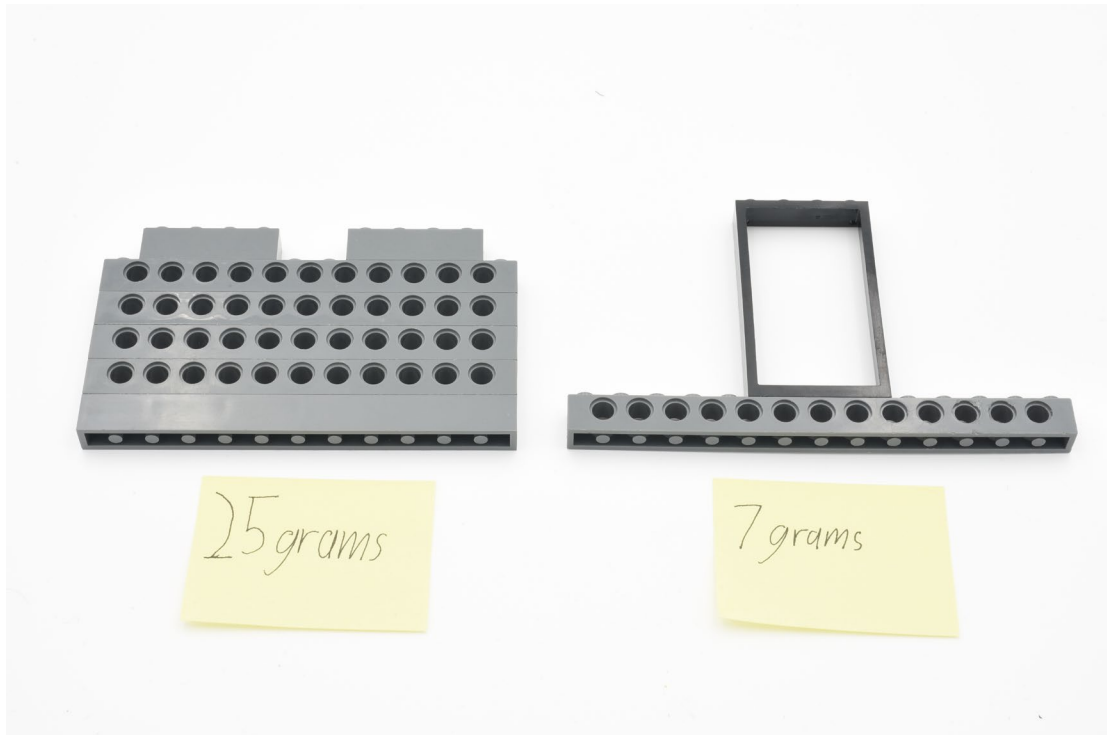

VERSION 6

I tested all three components but there was no travelling movement on the rack, but off the rack the stepper motor was working. This meant that either the battery was flat or the device was too heavy. First I tested the batteries but they were fine which meant this was not the problem.
I changed the gear ratio from 40:16 to 40:24 to make it easier for the stepper motor to drive hoping that this would allow it to move. It did not work but I kept the gear ratio anyway.
Then I tried making it lighter, testing the movement after each change. I changed the side supports with lighter pieces (door frames (7g) instead of solid blocks (25g)). I simplified the pollinator arm to make it lighter, lightened the ultrasonic sensor mount, then moved the battery tray to rebalance the device.
After these changes, the device can now travel along the rail, allowing me to begin testing with the flowers.

VERSION 7

With lighter everything it was still only working about ½ time and the batteries were going flat fast. Instead of batteries I changed to using a power pack attached to the module with a cord. It is not on the module however. I also made some minor adjustments to the ultrasonic sensor mounting.
I tried this version and it worked.
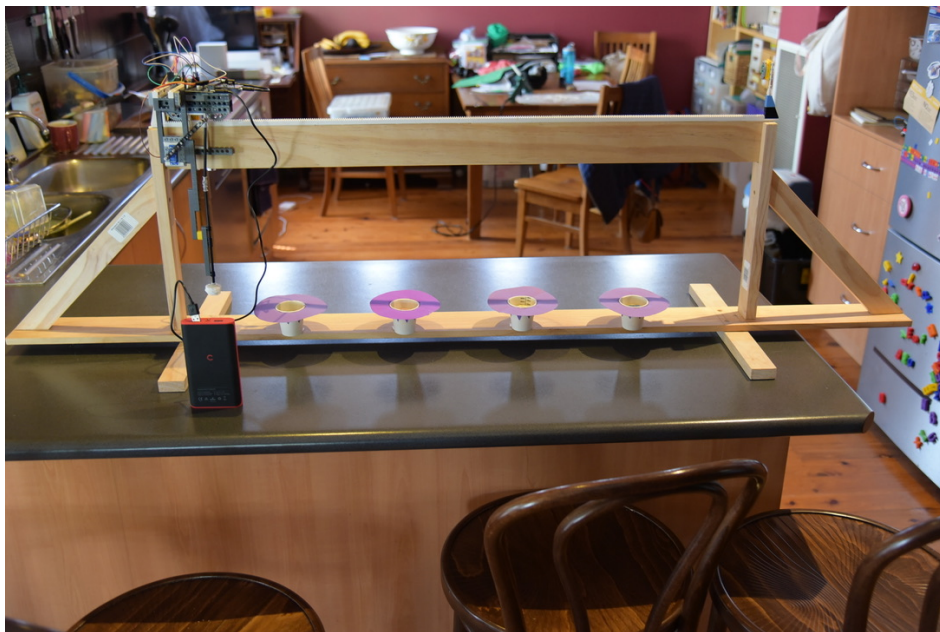
Summary

Even though I have 7 versions, version 5 is the first complete iteration of my mechanical pollinator. I ran into problems in earlier versions so I didn't bother adding all the components until those problems were resolved.

My design works well about 85% of the time. The main issue throughout the process was the strength of the stepper in moving the pollinator along the rail.