NanoPack

A PACKING MACHINE FOR NANOVIEW BIOSCIENCES

Paul Stephen Hutchinson Maltaghati

George Kent-Scheller

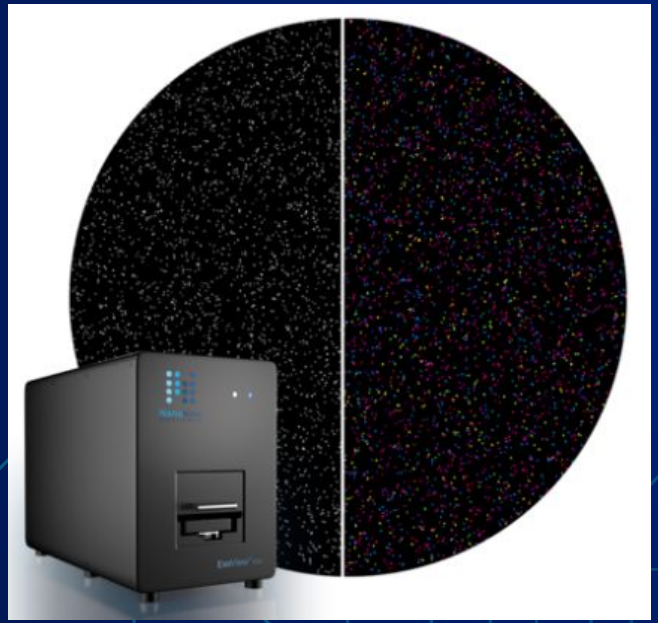
Devin Bidstrup

Justin Melville

Joseph Walsh

Critical Design Review

NanoView BIOSCIENCES



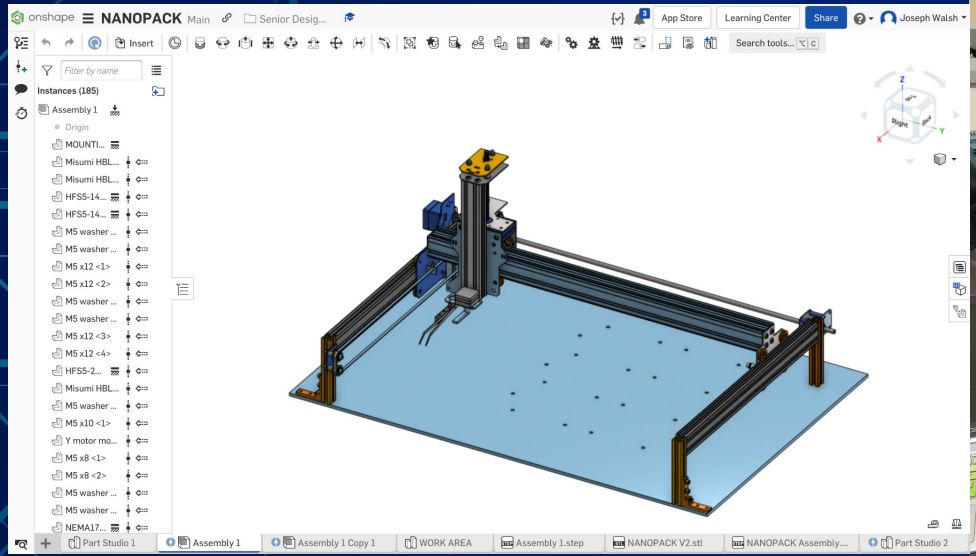
Our client, NanoView Biosciences, is a biotechnology company which makes machines capable of detecting markers on exosomes and other biological systems

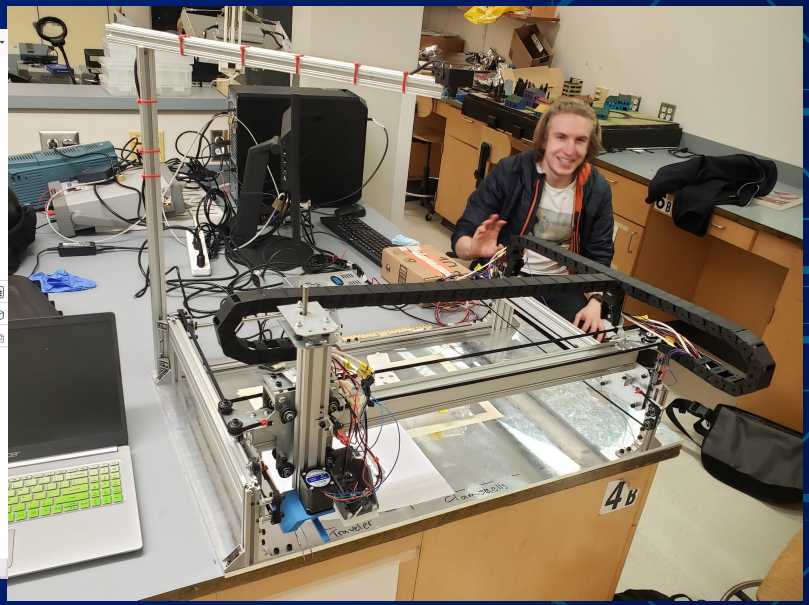
They are a startup company based in Brighton, and their President is a former BU professor, David Freedman

Problem Statement

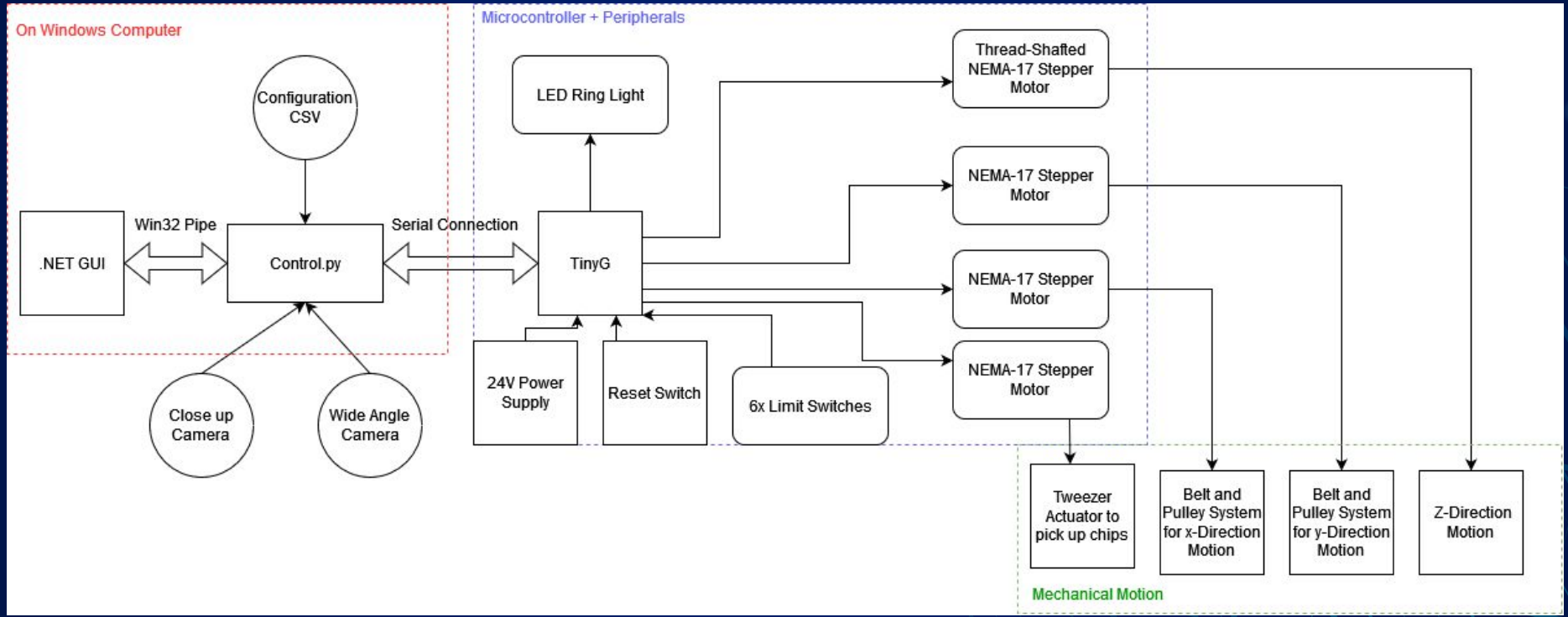
- NanoView currently uses a person with tweezers when they transfer silicon chips from their lab container, “the traveler” into user cases, “clamshells”
- Our job is to automate the process of transferring the chips from the traveler to clamshell
- This will speed up the process, prevent human error, and reduce risk of contamination

Visualization



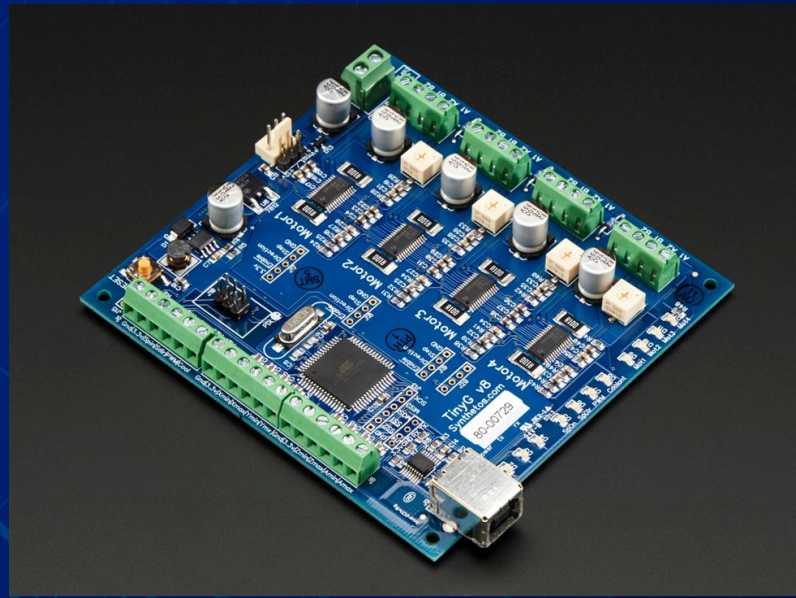


System Block Diagram

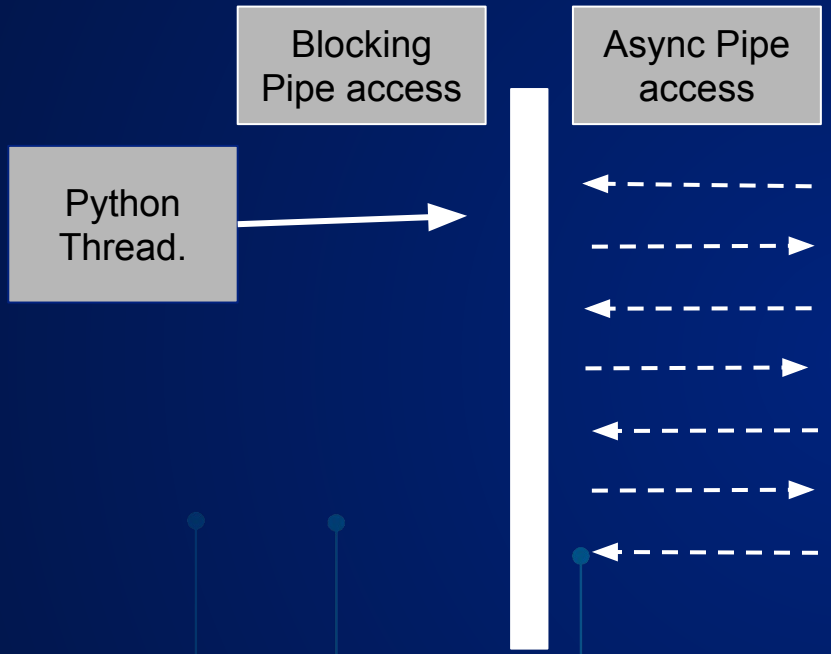


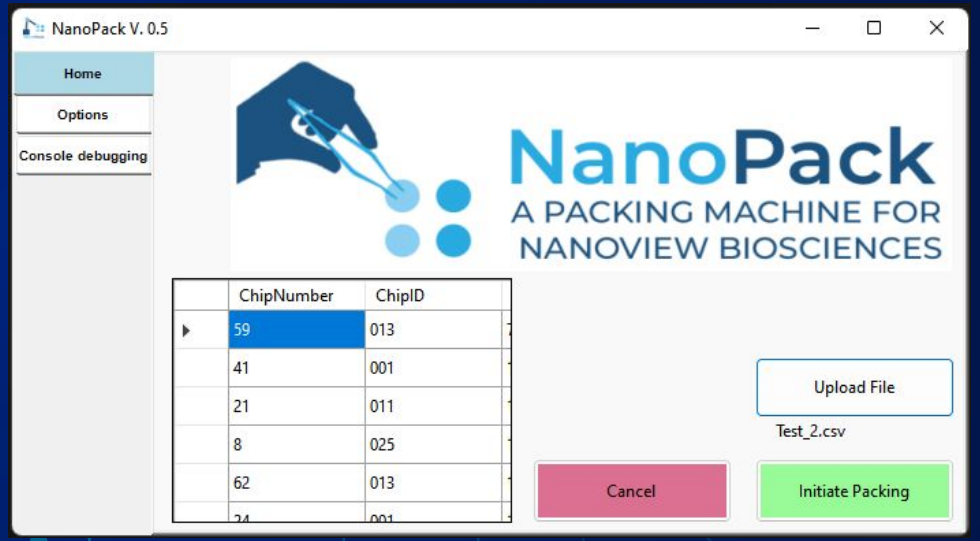
Current Status - TinyG CNC Controller

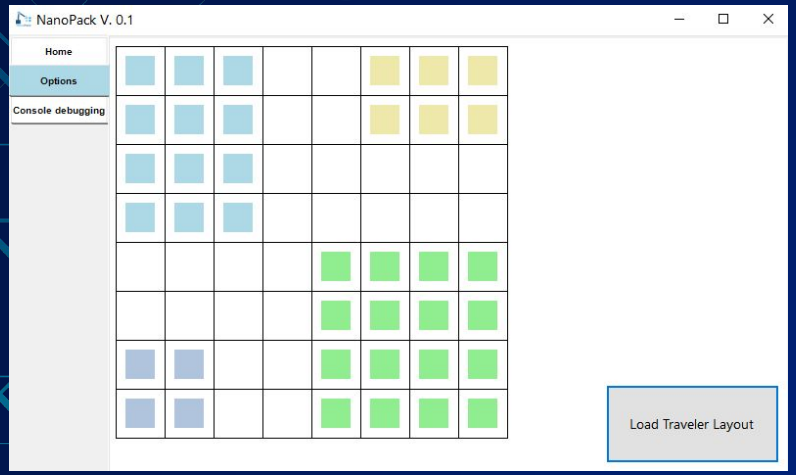
- Communicate in Python via pySerial
  - Serial port detected automatically
  - JSON-based communication
  - RTS/CTS flow control
  - Threaded reads / writes
- Configured Parameters to fit our machine
  - Parameters saved to EEPROM on TinyG
  - Updated only if needed
- Homing defines the 0 point on all three axis with limit switches
- G-code commands are sent JSON format
  - e.g. {'gc', 'G0 X 1'} to perform a G0 movement on the X-axis



Current Status - GUI

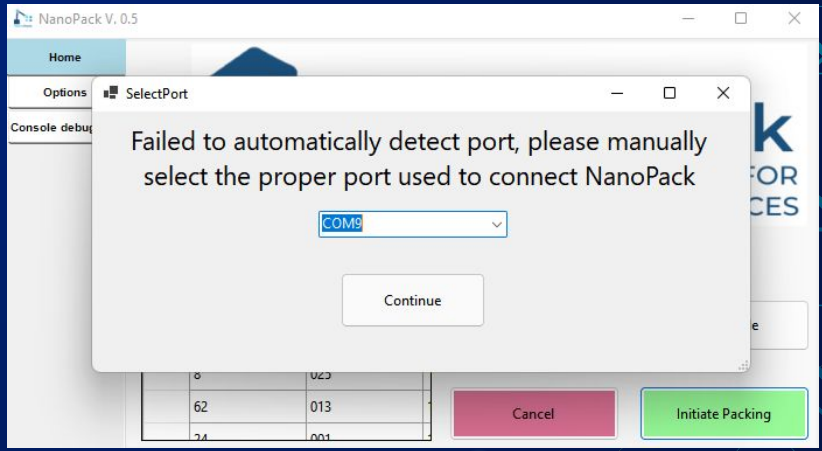




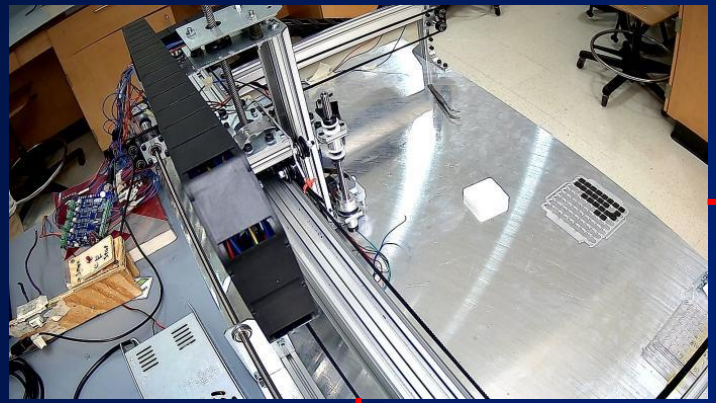


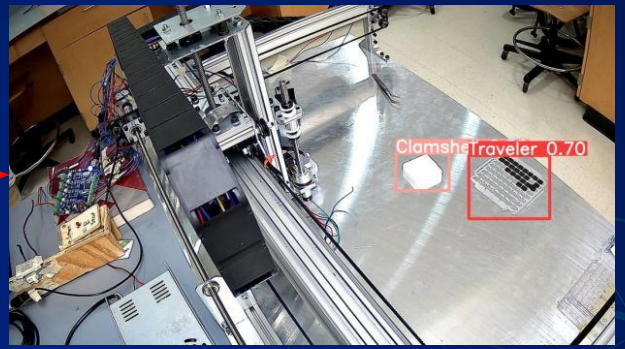
Traveler Visualization

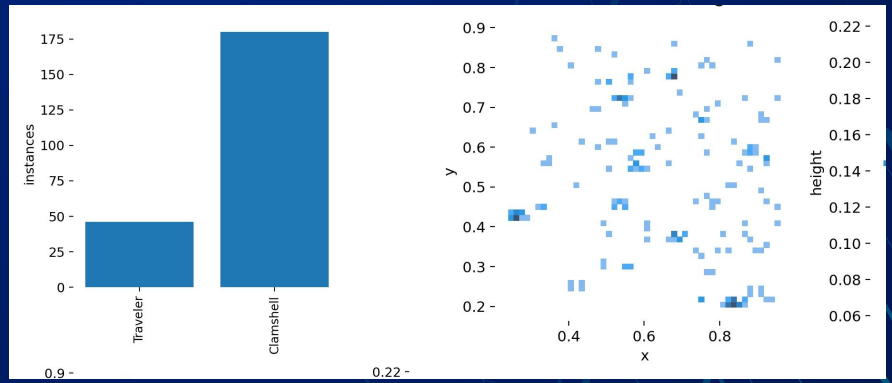
Port Selection



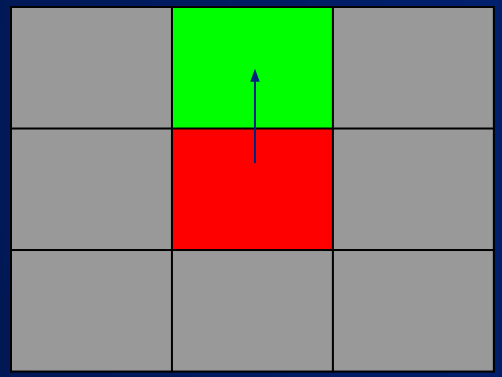
Current Status - ML - Object Detection





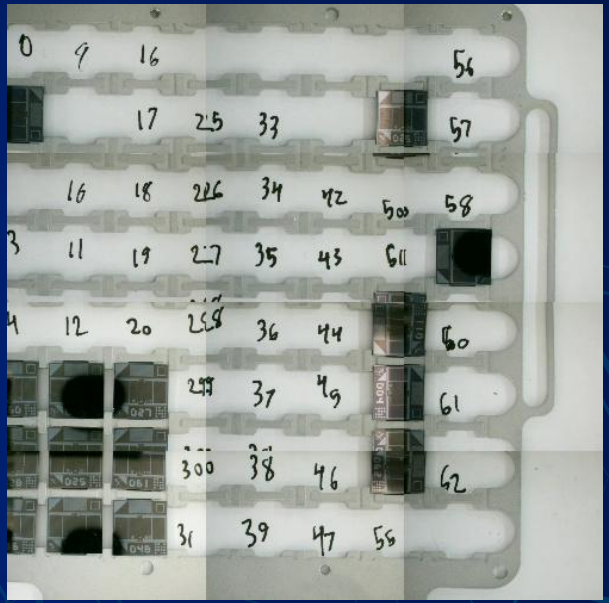


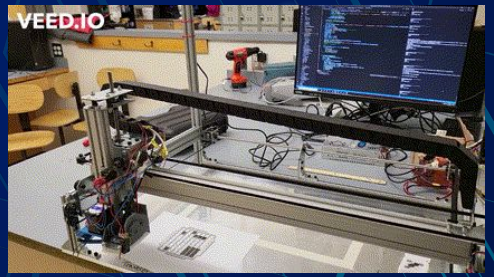
Traveler edge detection



255	255
160	160

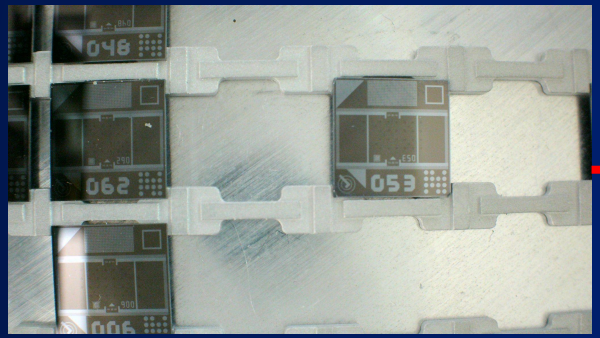
Chip Imaging

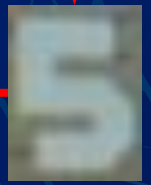




Current Status - ML - Number Recognition

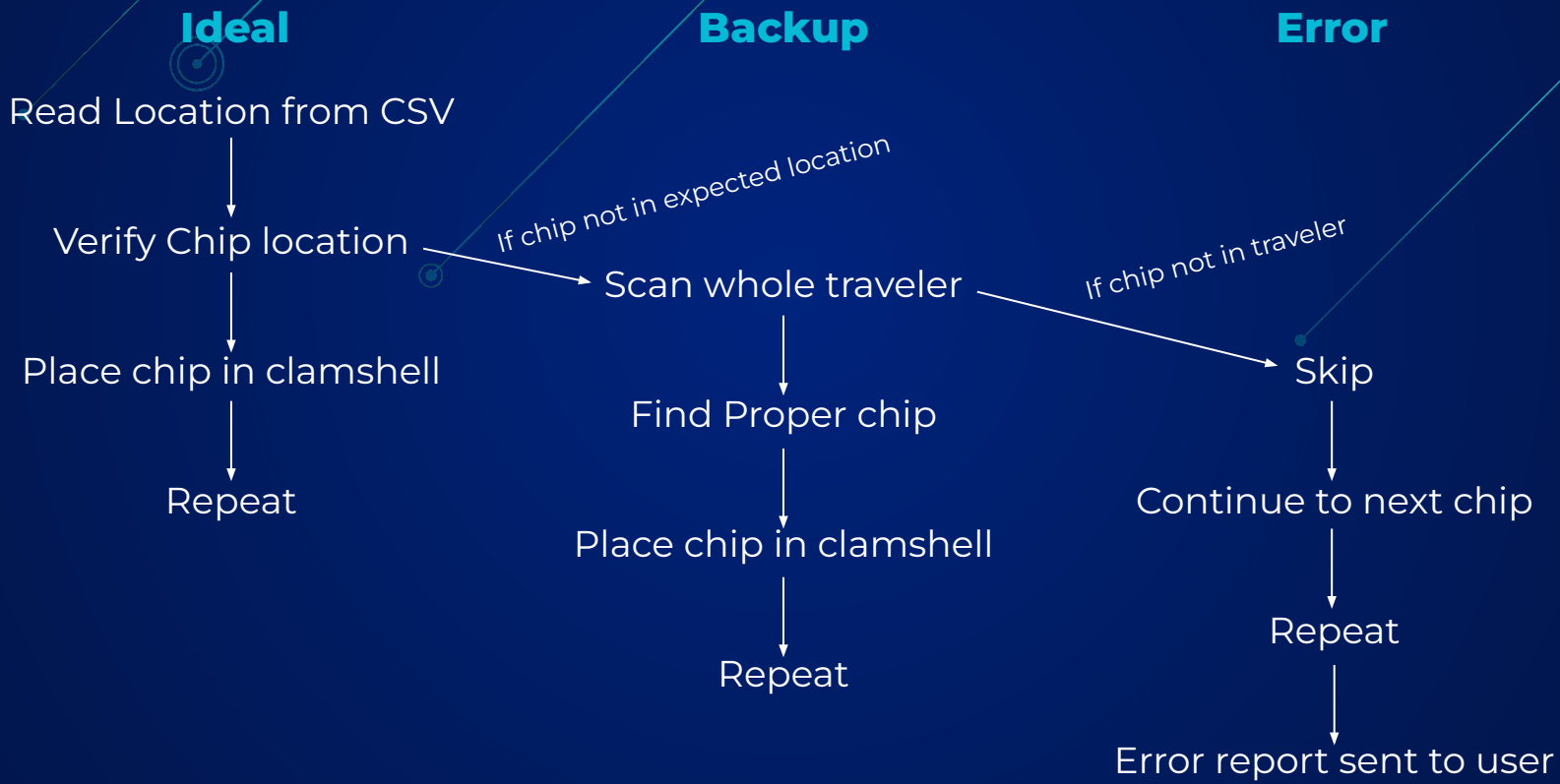
Add Number recognition, chip cropping, and number cropping model stuff here

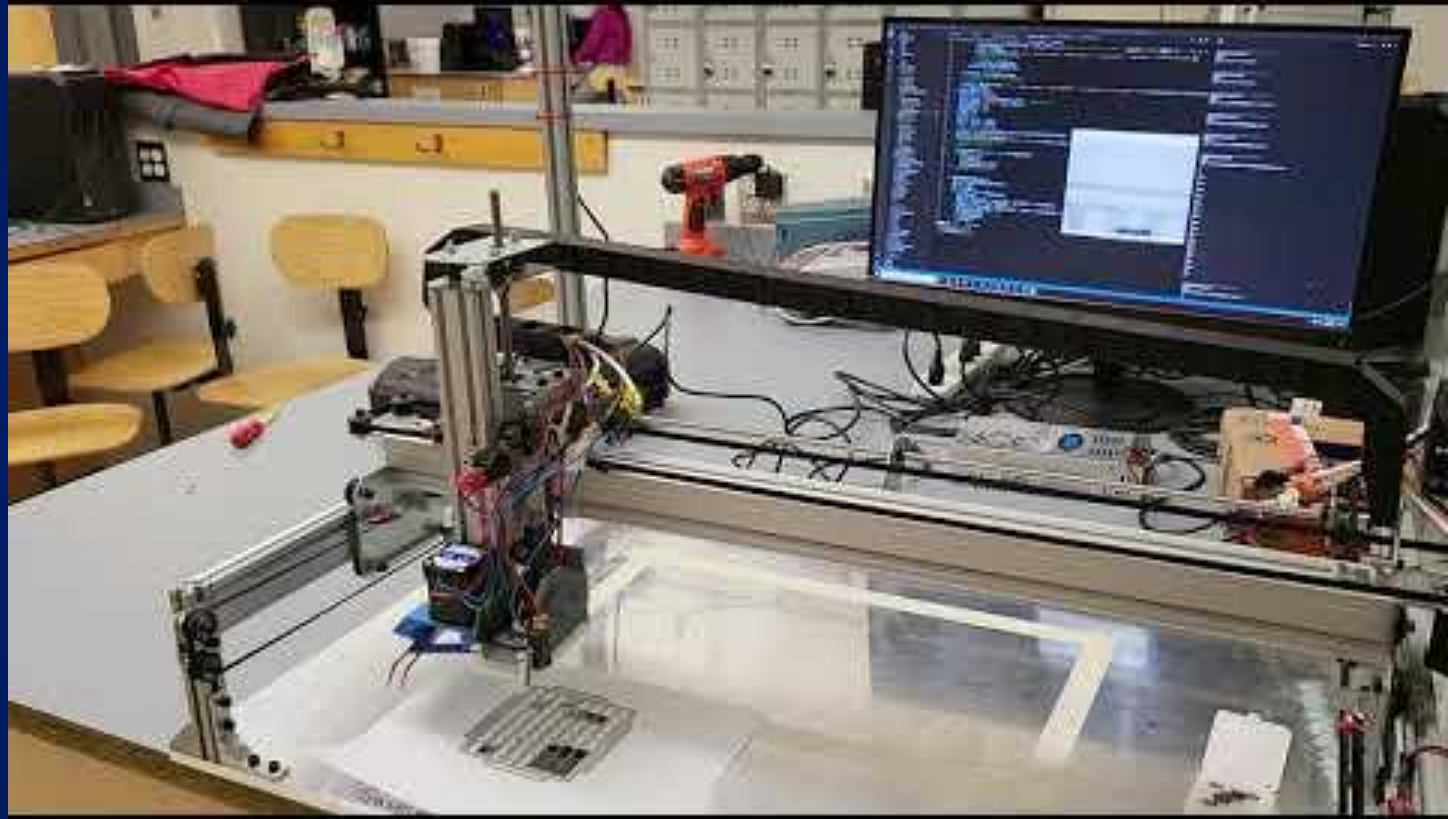


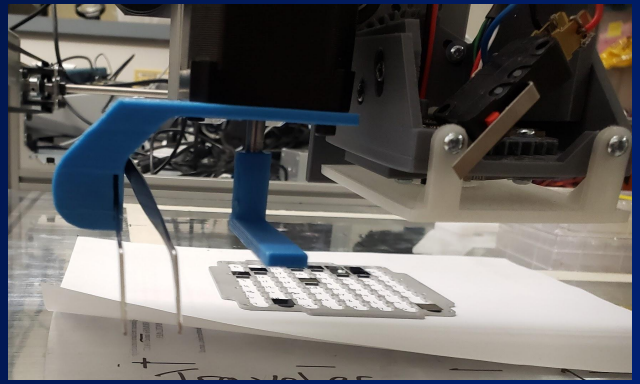


```
The model will be running on cpu device
For epoch 1 the test accuracy over the whole test set is 30 %
For epoch 2 the test accuracy over the whole test set is 53 %
For epoch 3 the test accuracy over the whole test set is 61 %
For epoch 4 the test accuracy over the whole test set is 84 %
For epoch 5 the test accuracy over the whole test set is 100 %
For epoch 6 the test accuracy over the whole test set is 100 %
For epoch 7 the test accuracy over the whole test set is 100 %
For epoch 8 the test accuracy over the whole test set is 100 %
For epoch 9 the test accuracy over the whole test set is 100 %
For epoch 10 the test accuracy over the whole test set is 100 %
Finished Training
Real labels:     zero     five     three
predicted:     zero     five     three
```

Chip Placement Pipeline

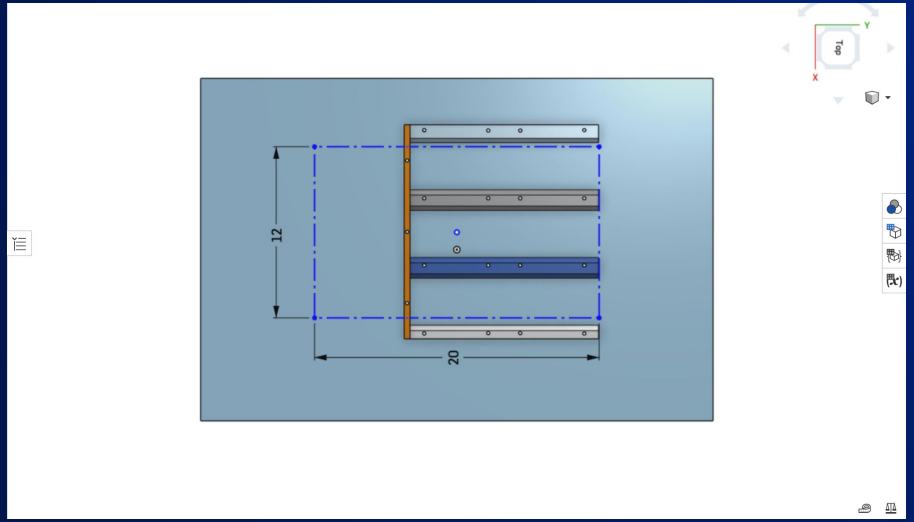


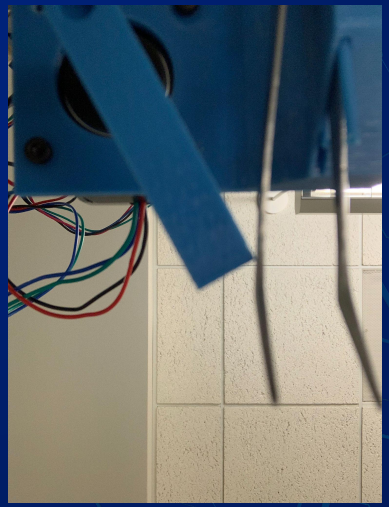




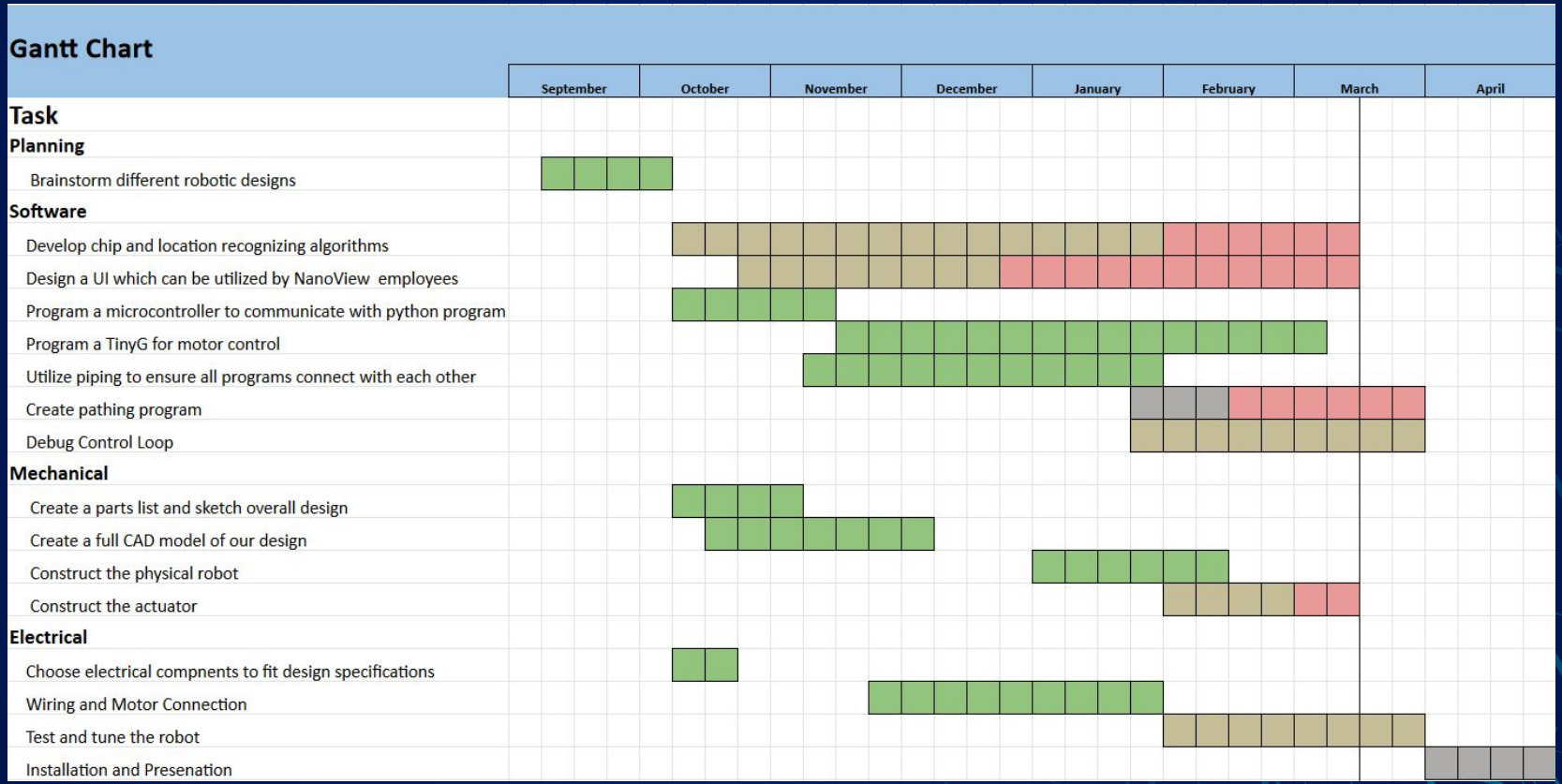
Current Status - Mechanical

- Modified design from LitePlacer robot
- Custom tweezer actuator designed to pick up chips
- Added 3D printed mount for tweezers + stepper arm to pinch tweezers





Schedule (Gantt Chart)



TODO

TinyG:

- Code accurate offsets for tweezers relative to camera
- Tune movement axis parameters
- Consider driving actuator with servo motor, solenoid, or using vacuum pump

GUI:

- Prompt user for E-Stop Button press
- Software reset the board from GUI
- Make more robust against different possible edge cases

Control:

- Implement a control loop that does pathing of chip placement

TODO(cont'd)

ML:

- Add better preprocessing on images
- Increase size labeled data set for object detection, increase labeling accuracy by realabeling data
- Allow ML to run in the background while machine moves
- Integrate the digit recognition with rest of the code

Mech:

- Tune actuator to pick up chips
- Add enclosure for tinyG and E-stop
- Create physical constraints for clamshells and traveler
- Add surface to bed of the machine to allow for better image recognition