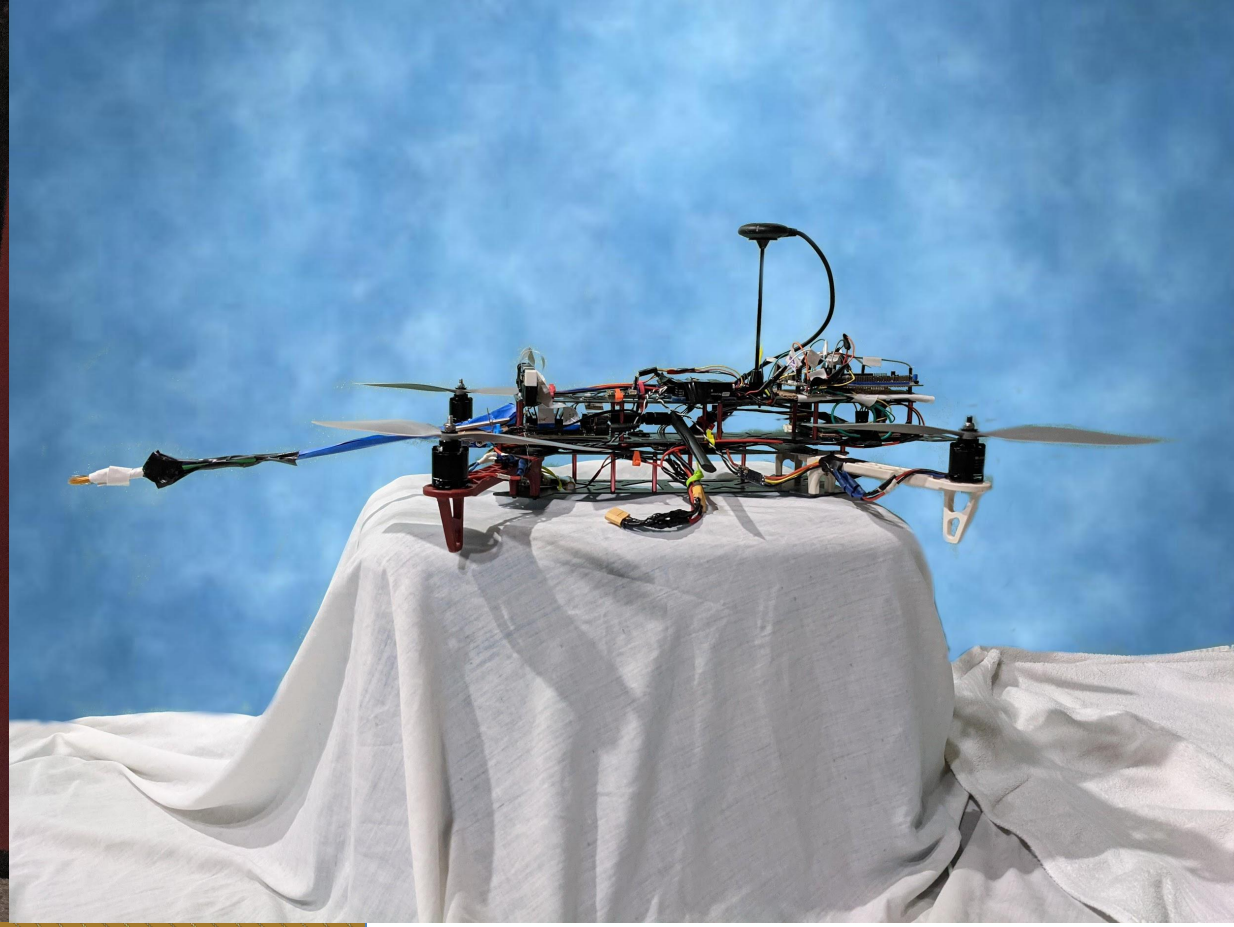



# ECE 477 FINAL REVIEW:

# TEAM 4

PURDUE
UNIVERSITY

# OUTLINE

- Project Overview
- Block Diagram
- Design Challenges
- Individual Contributions
- Project Demonstration
- Questions

PURDUE
UNIVERSITY

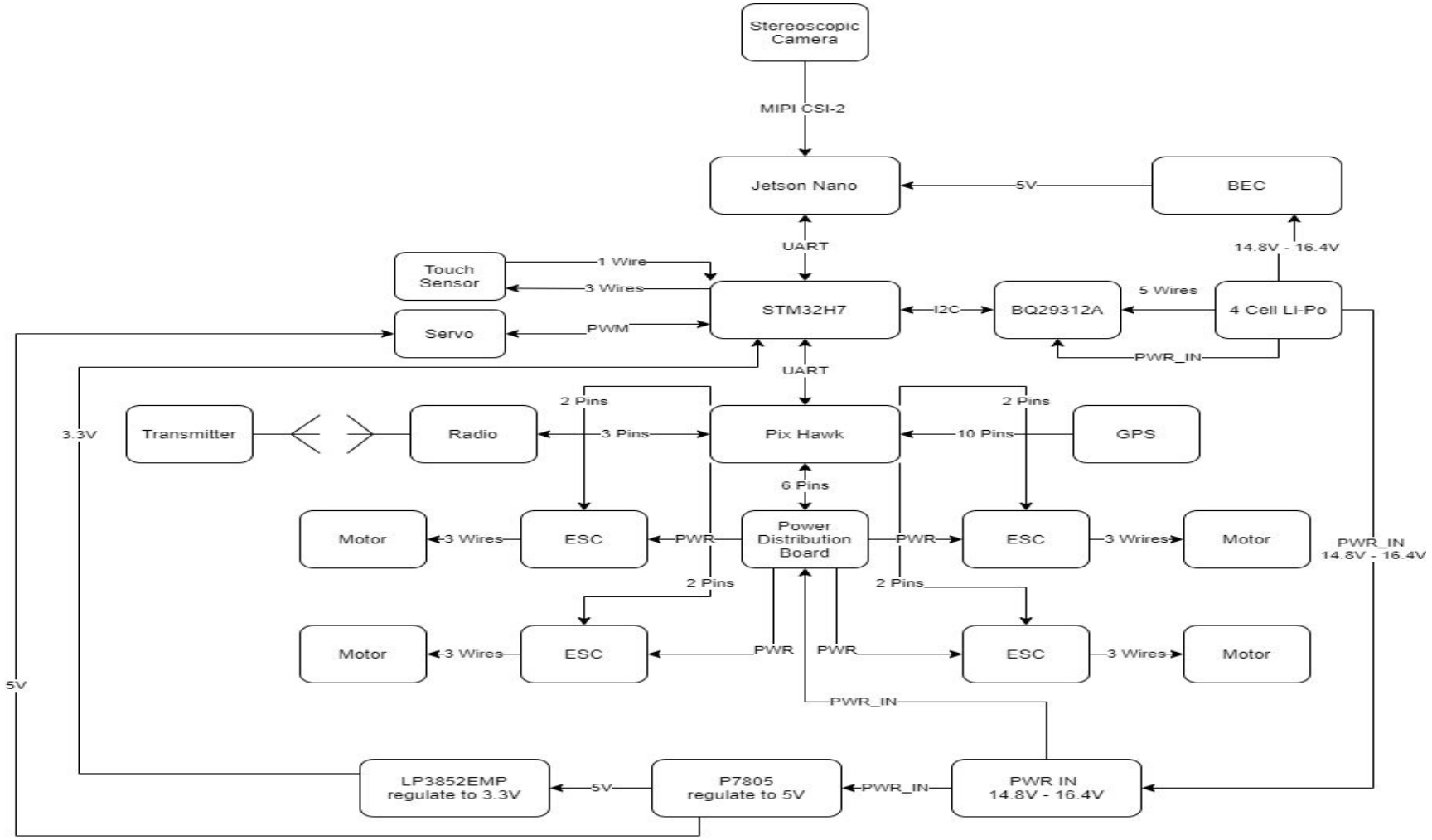
# PROJECT OVERVIEW

- “Pollinating Bee Drone”
- Autonomous aircraft pollinates apple blossoms
- Image Processing
- Flower contact activated actuator
- Battery Monitoring
- Safety switches
- On board WiFi

Pictured: The pollinating bee drone

PURDUE UNIVERSITY

# BLOCK DIAGRAM




PURDUE UNIVERSITY

# DESIGN CHALLENGES

Battery Monitor

- Couldn’t use coulomb counter due to amperage on single PCB
  - Solution: Fourth order approximation of battery life using voltage
- How to monitor in-flight?
  - Solution: Send data to Nano, ssh in


Retrieved from: battery site

PURDUE UNIVERSITY

# DESIGN CHALLENGES

Size of mounting components on drone:

- Need to be able to mount Jetson Nano, PixHawk Mini, STM32, and other sensors on drone frame.
- Components must be safe from damage upon landing or normal drone operation.
- Solution:
  - We decided upon using a large drone frame to ensure proper mounting locations available for necessary components.
  - Utilized a durable frame material to be able to absorb vibrations and forces upon landing.
  - Centrally mounted all components to ensure they will not contact rotors or branches during operation.

PURDUE
UNIVERSITY

# DESIGN CHALLENGES

| Video Stream Optimization Path | |
|---|---|
| Python(OpenCv) | 7s @ 5 fps |
| Rewrite in C++ | 2s @ 10 fps |
| New Bug - Synchronization of image frames! | Approximately 1s lag between image streams |
| Switch to lower level framework(Gstreamer -> libargus(Cuda)) | 0s @ 2 fps |
| Implement Multithreading | 2 producer, 2 consumer, 1 main thread |
| Synchronization of frames got worse | |
| Implement synchronization primitives across threads | 0s @ 30 fps |
| Add image processing | 2s @ 5fps |
| Reduce resolution, Cuda Processing, Blur Optimization | 0.1 s @ 20 fps |

PURDUE
UNIVERSITY

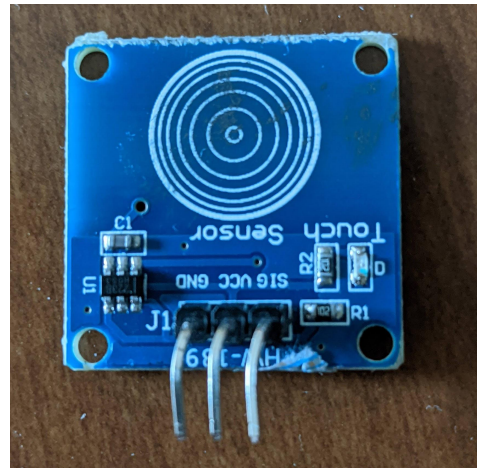
# DESIGN CHALLENGES

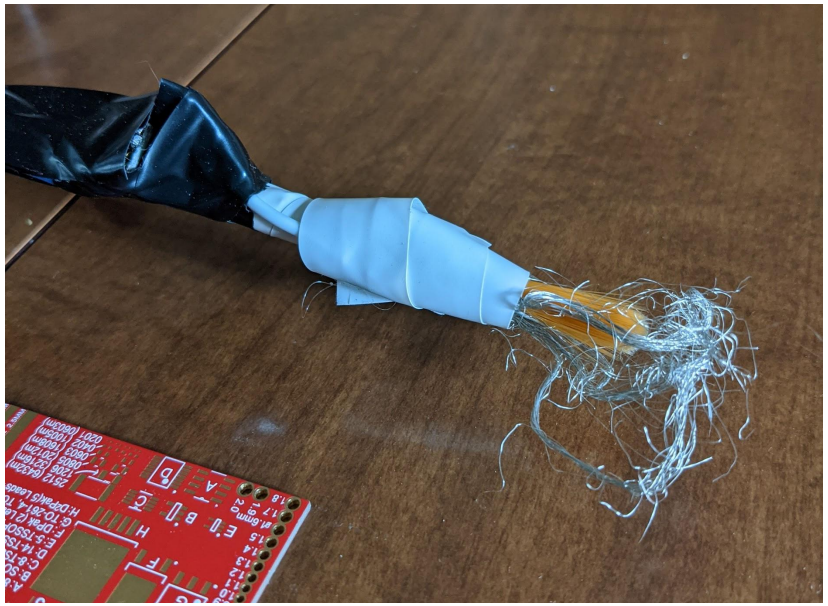
Dual Core UART Transmission

- method to queue up > 200 byte UART messages for MAVLink protocol
- messages could be from either core
- must be handled in background (DMA) given the size of the packets.
- Solution:
  - adjusted linker file for core 4 to free up SRAM3 for both cores.
  - made a linked list queue in there
  - a timer empties it out every 50ms

Capacitive Touch Sensor

- sensors came attached to a sensor pad on a PCB.
- Soldered on some wires to the sensor pin and placed them near the brush.



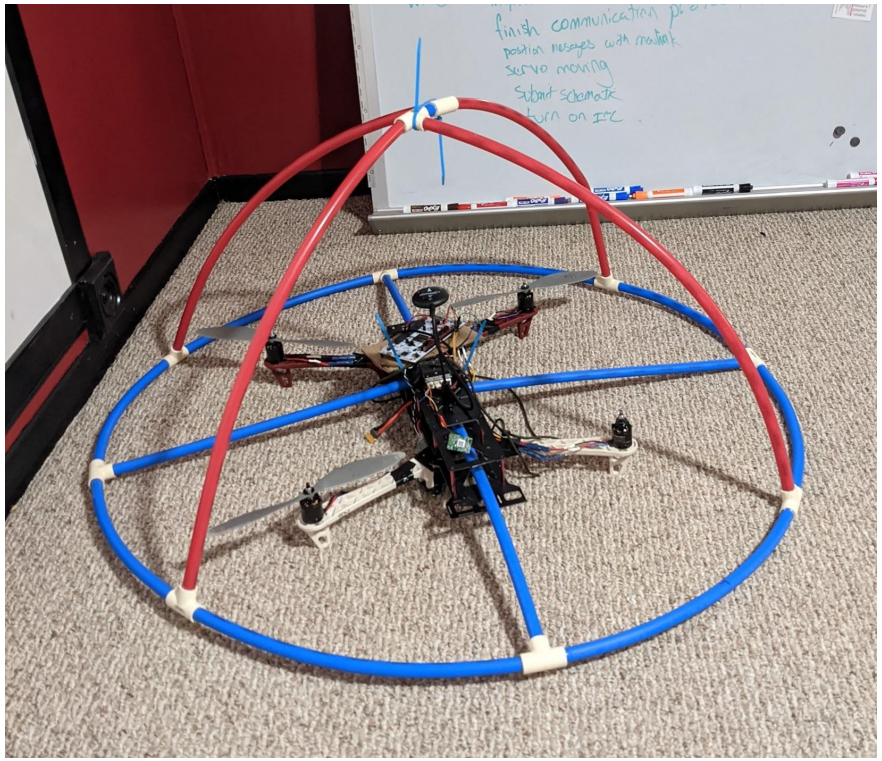



PURDUE UNIVERSITY

# INDIVIDUAL CONTRIBUTIONS

## Molz Rasheed

- Mavlink protocol implementation and dual core communication
- Drone flight routines
- finalization and assembly of pollinator touch sensing and actuation

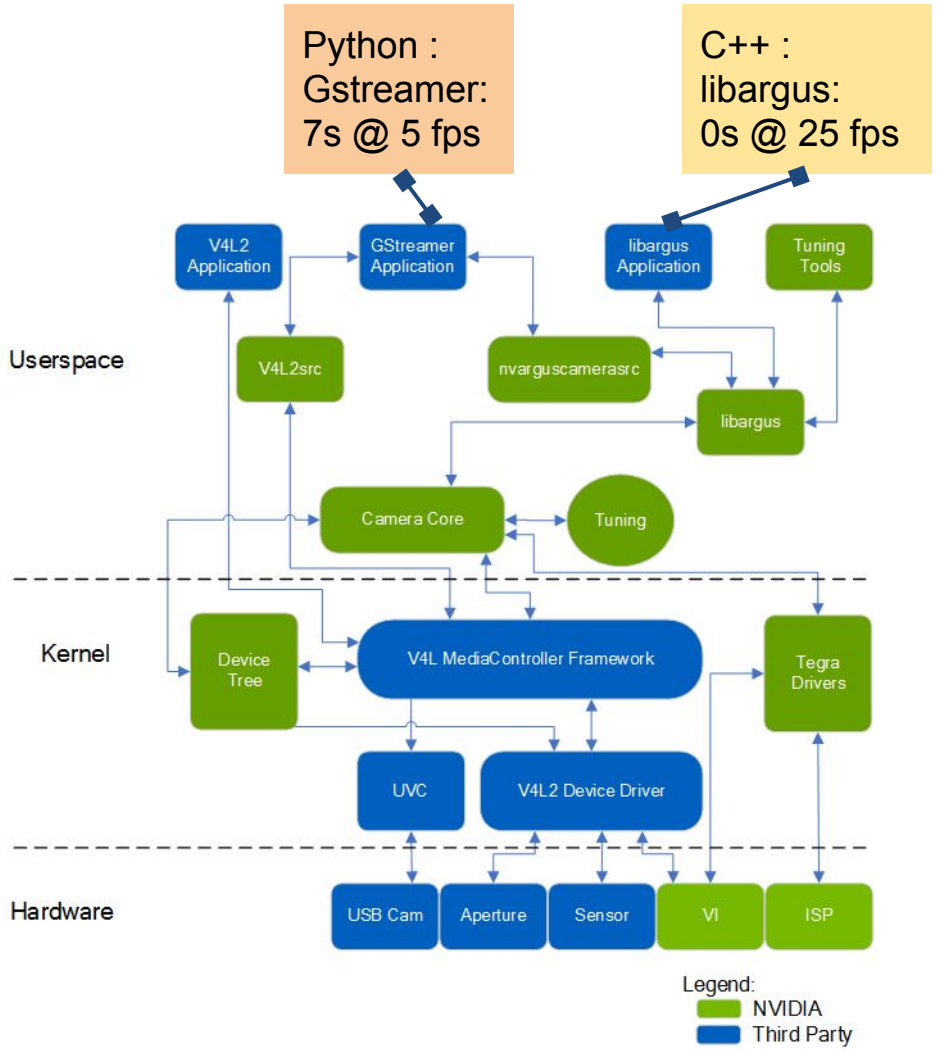



PURDUE UNIVERSITY

# INDIVIDUAL CONTRIBUTIONS

## Ethan Campbell

- Jetson Nano:
  - Inter-Hardware Communications: Uart
    - PySerial
  - Stereo Vision Processing
    - Two CSI Raspberry Pi Cameras
    - Python -> CPP optimization
      - low level integration Gstreamer
      - True multi-threading
      - libargus API
    - OpenCV
      - Image processing
      - Disparity Mapping

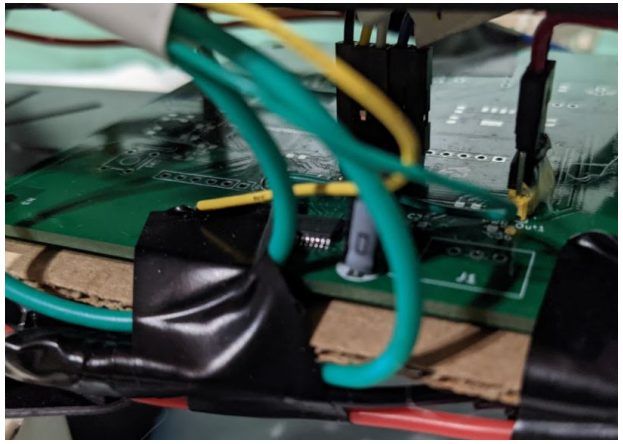


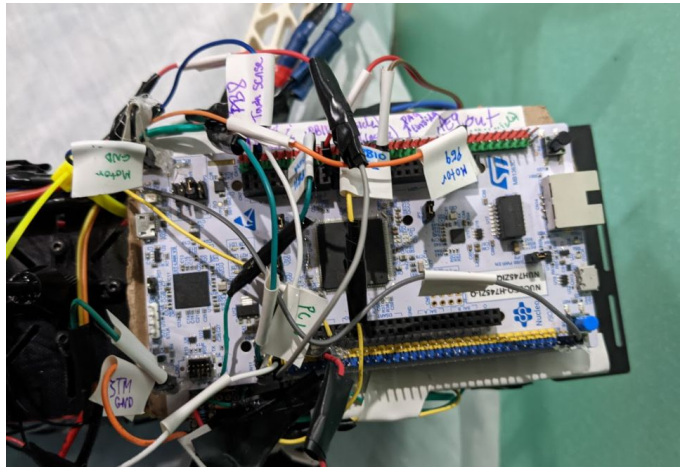

*docs.nvidia.com*

PURDUE UNIVERSITY

# INDIVIDUAL CONTRIBUTIONS

## Jackie Malayter





- Initial implementation of color masking and blob detection
  - Python OpenCV
- Battery Monitor
  - I2C communication
  - Modeling voltage versus discharge capacity
- STM UART peripheral with Nano
  - Moved to interrupt based receive
  - Wrote transmit
- Final mounting and test
  - Soldered components
  - Tested subsystems
  - Secured/replaced components

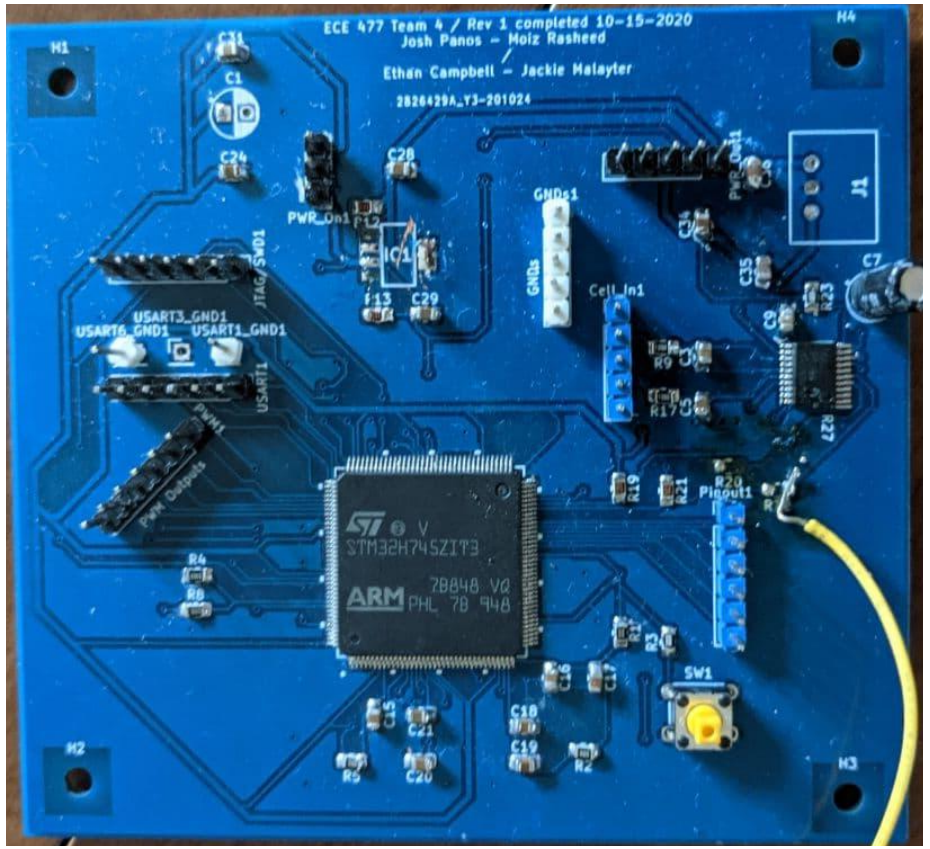
PURDUE UNIVERSITY

Pictured: (Top) Battery monitor on the PCB
(Bottom) “Amazing” wiring job on dev board

# INDIVIDUAL CONTRIBUTIONS

## Joshua Panos

- Drone Construction:
  - Motor sizing for expected drone weight and flight durations
- PCB design and assembly:
  - Providing a safe and effective circuit to use the battery monitor on our drone.
  - Develop a board that is able manage multiple connections and ICs.
- Prototyping of capacitive sensor, servo and pollinator.




PURDUE UNIVERSITY

# PROJECT DEMONSTRATION

View our project demo video here!

PSSCs:

1) An ability to detect the location of a flower in an image(s) frame.
2) An ability to communicate flower positional data between the STM32 and the Jetson Nano.
3) An ability to actuate and detect contact with a capacitive switch on the pollen collecting appendage.
4) An ability to monitor battery life with the STM32 during drone flight.
5) An ability to communicate with the flight controller over the mavlink protocol.

PURDUE UNIVERSITY

# Questions?