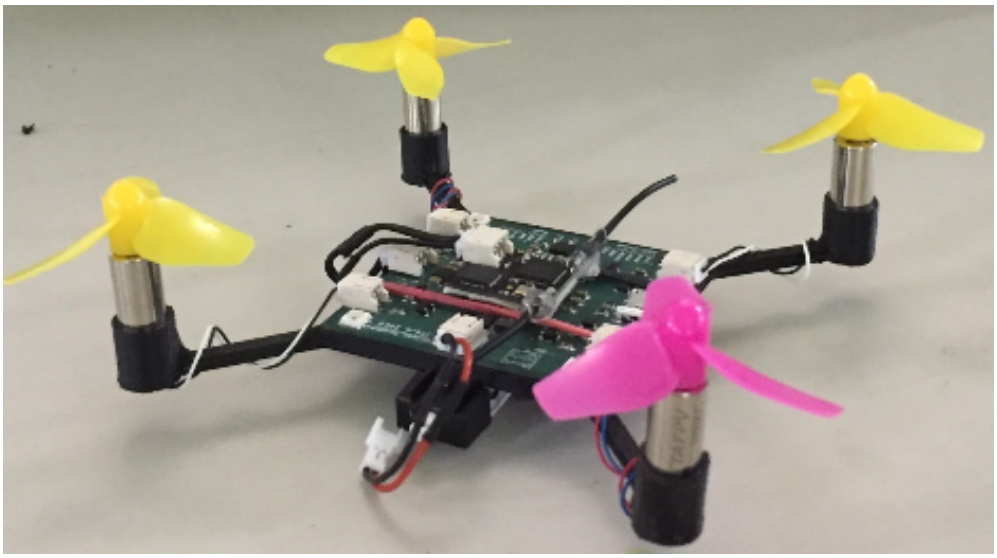
SAAGAR ARYA
DRONE PROJECT

AUGUST 2017 – APRIL 2019

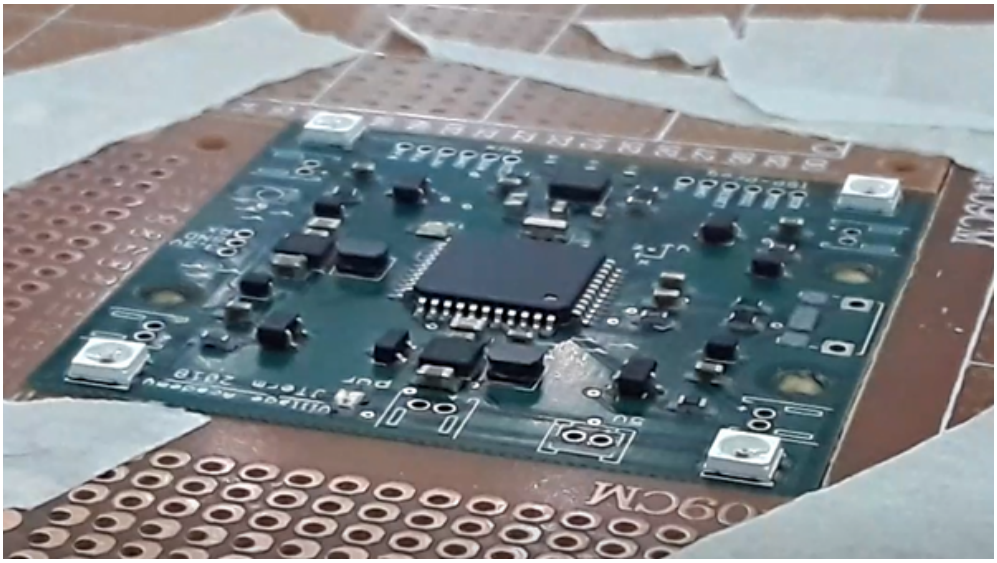


Completed Drone

Drone in Flight During Testing

SUMMARY

Building my drone was one of the first projects I worked on. As only a 7th grader with almost no experience in anything engineering-related, I approached Mr. McDonley, the engineering and computer science teacher. After a candid discussion about how much work it would take due to my inexperience, he agreed to provide guidance where he could, and for nearly the entire year, I worked 2-3 hours on most days after school. After a semester of this constant work, I learned how to 3D model, about electric components and circuits, how to comb data sheets, and finalized the schematic. This finalized schematic required me to learn how inductors and microchips work, calculate resistance and impedance values, and how the crystalline structure of Quartz allows it to be an accurate on-board clock. After converting the schematic to a Printed Circuit Board, I had to learn to place the parts to minimize distances and vias, balance the board by spacing heavy components, and create onboard heatsinks for the Voltage Regulators. After receiving the PCB from fabrication and hand-placing the minuscule surface mount parts, I ran a debugger to ensure all the wires and components were connected before running a bootloader of the Arduino Firmware (Based on C++) which made iteasier to program.



Printed Circuit Board's Components Being Placed

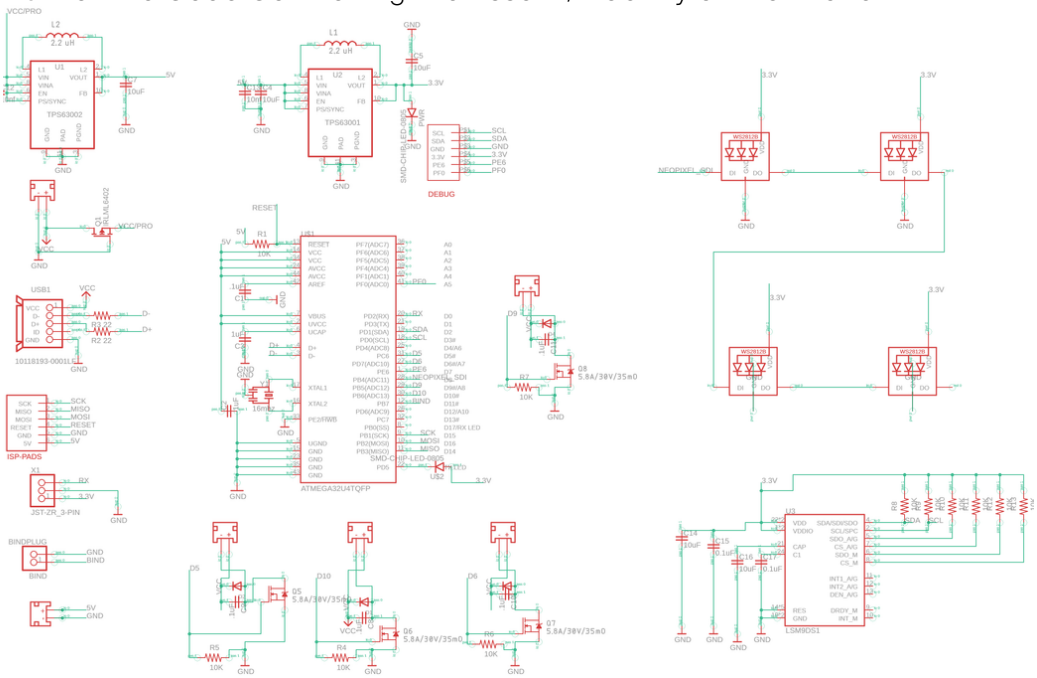
Further, using my knowledge of 3D modeling, I created a light 3D printed frame that allowed for the PCB to sit compactly between 4 motors, a battery, and a receiver. As I began programming the drone, I learned to code in the Arduino environment and tune P.I.D. values, tie in libraries for the various onboard sensors and components, and use GitHub and Reddit Forums to create a working set of code that let the drone fly.

FlightController.cpp | Adafruit_NeoPixel.cpp | Adafruit_NeoPixel.h | FlightController.h | Imu.h | LSM9ds1_Registers.h | Motors.cpp

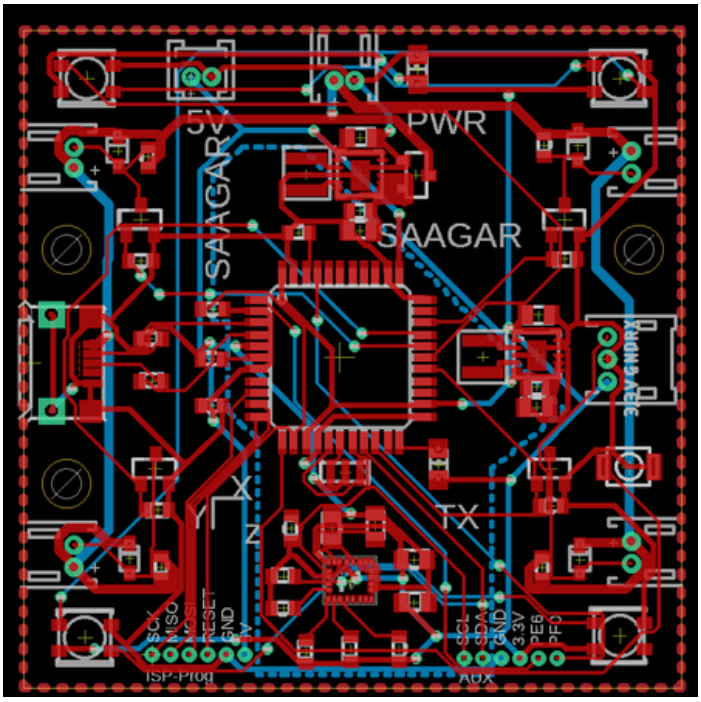
```
if(!failSafe)
{
  //Calculate PID
  prevPitchError = pitchError;
  prevRollError = rollError;
  prevYawError = yawError;

  pitchError  = deadband(spPitch - pitch, 2)*2;
  rollError   = deadband(spRoll - roll, 5);
  yawError    = deadband(spYaw - yaw, 10);

  pitchErrorACC = pitchErrorACC + pitchError*(cycleTime>>9);
  pitchErrorACC = constrain(pitchErrorACC, -MOTORLIMIT/2, MOTORLIMIT/2);
  rollErrorACC  = rollErrorACC + rollError*(cycleTime>>9);
  rollErrorACC  = constrain(rollErrorACC, -MOTORLIMIT, MOTORLIMIT);
  yawErrorACC   = yawErrorACC + yawError*(cycleTime>>9);
  yawErrorACC   = constrain(yawErrorACC, -MOTORLIMIT/2, MOTORLIMIT/2);

  pitchErrorDer = (pitchError - prevPitchError) / (cycleTime>>9);
  rollErrorDer  = (rollError - prevRollError) / (cycleTime>>9);
  yawErrorDer   = (yawError - prevYawError) / (cycleTime>>9);

  pidPitch = constrain(5*pitchError + 0.25*pitchErrorACC + 100*pitchErrorDer, -4*MOTORLIMIT, 4*MOTORLIMIT);
  pidRoll  = constrain(0*rollError + 0*rollErrorACC + 0*rollErrorDer, -MOTORLIMIT, MOTORLIMIT);
  pidYaw   = constrain(1*yawError + 0.25*yawErrorACC + 4*yawErrorDer, -MOTORLIMIT, MOTORLIMIT);
  pidPitch = 0;
```

Code based on The MultiWii GPL Project

Part of the Code Controlling the Assent/Mobility of the Drone



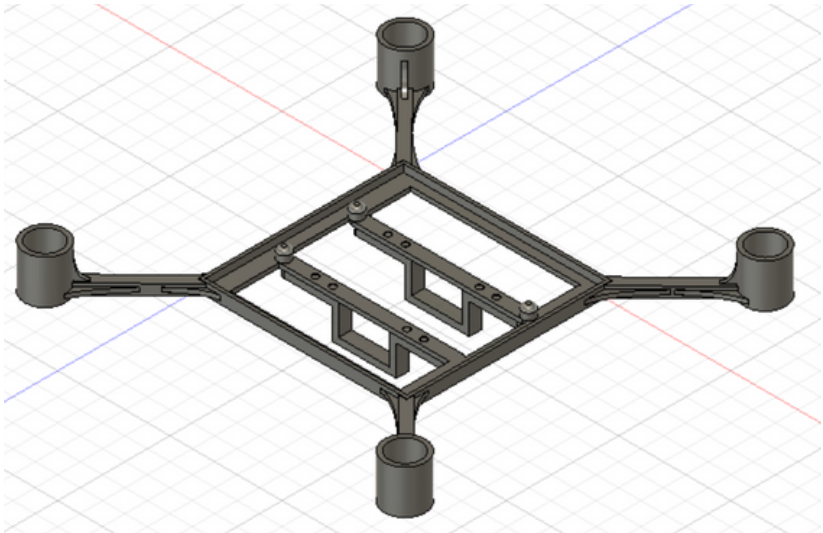
Schematic of the Drone's Printed Circuit Board on Autodesk Eagle



Printed Circuit Board Modeled on Autodesk Eagle

SKILLS LEARNED/USED

Autodesk Fusion 360
Ultimaker Cura
Autodesk Eagle
Arduino (Based on C++)
Aerospace concepts
Electrical Engineering



3D Model of the Drone's Chassis on Autodesk Fusion 360