2022

3-D Printed Arduino Powered Drone

Michael Floccare
John Carroll University, mfloccare22@jcu.edu

3-D Printed Arduino Powered Drone

Michael Floccare

John Carroll University, 1 John Carroll Blvd. University Heights, OH 44118

Abstract

With the 3D technology available today creating something in a lab has never been easier. Using 3D printers, the body of a drone is created and printed using a combination of programs. With a body created using PLA filament the drone is durable and light weight. The drone's flight system and controller are programmed with the built-in software from Arduino. Adding the Arduino nano to the drone body then gives the drone capabilities to fly.

I. Introduction

Technological advances over the past decades have given way to a multitude of unmanned arial vehicles (UAV). UAVs are in use in militaries across the world, as planes, rovers, drones, etc. We also see UAVs in civilian life as well. Store bought drones, cars, planes, hovercrafts, etc. are all examples of UAVs. Other options are also available, and for the do-it-yourselfer one can build it themselves. Projects by Electronoobs and RClife on YouTube were used as inspiration and guidance for this project.

The most important components of the drone are the electronics. This includes the Arduino and the microprocessor. The Arduino is an open-source electronics platform that can be used as a microcontroller board. Arduino has its own programming language and software (IDE). This can be used alongside a microprocessor unit with sensors to create a program for flight. The sensors that are needed alongside the Arduino are a gyroscope, accelerometer, magnetometer, barometer, and even a GPS. With the use of a radio controller transceiver the drone can be operated remotely through the commands given by the controller.

Another essential component of the drone is the chassis. This can be done by using 3D printers, a relatively new technology that has taken off in recent years. 3D printers have the capabilities of printing plastics, metals, woods, and even concrete. The possibilities for 3D printed objects are limitless. Models can be created in Fusion360 or similar programs which then can be exported to the printer's own modeling program. The two printers that were used are the MakerBot Replicator and Ultimaker S5 Pro. The MakerBot Replicator uses MakerBot print to slice the model and bring the file to life, similarly, Ultimaker Cura is used to bring models to life for the Ultimaker S5 Pro. These programs take your 3D model and slice it into layers that the printer can then print. Additionally, it also calculates the optimal route the extruder will take for the fastest print time. Other options such as infill percentage, supports, diameter of extruding filament can be altered for each print. Changing the setting for prints generates unique characteristics for all prints.

Here we will describe the construction and programming of the Arduino drone. This project costs less than $300 in material and parts. The major cost of this project is from the electronics that were unable to be 3D printed. All parts of the chassis, including the arms and propellers were created and printed using a 3D printer. The experimental portion was done by comparing the quality of a print alongside the strength and mass of each part. This was instrumental in optimizing the mass of the drone.

II. Theory

In order to sustain flight in a drone the propellers must spin in different directions. Two propellers must spin clockwise while the other two spin counterclockwise. If the rotors were aligned like a square, the two rotors on the sides would spin counterclockwise, and the rotors on the top and bottom would spin clockwise. A visual representation of this can be seen in figure one. If the propellers rotated all in the same direction the drone would spin like a helicopter without the tail rotor.[1] This creates stability for flight and the ability to adapt during flight. The drone should continuously take measurements from the onboard sensors. These measurements are recorded automatically by a micro processing unit.

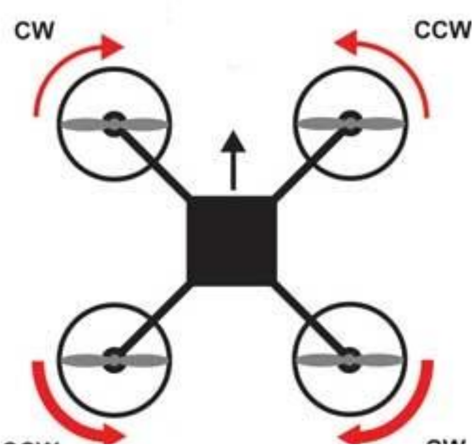

Figure 1: Visual Representation of the direction of spin for each rotor. Top left and bottom right are clockwise. Top right and bottom left are counterclockwise.[1]

The measurements recorded by the sensors adjust the four degrees of freedom. The four degrees of freedom are yaw, roll, pitch, and altitude. Yaw is the rotation of the drone on a plane perpendicular to that of the drone.

Roll is the tilt of the drone left to right, and pitch is the same except it refers to tilting forward or backward.[1] Roll and pitch are accomplished by decreasing the thrust of one rotor and increasing the thrust of the opposite rotor. This causes the drone to tilt, the force vector is split into two components, horizontal and vertical. The split of the force vector causes the drone to move in the direction of rotor with less thrust. If the front rotors had less thrust than the back the drone would tilt forward and then move forward. The tilt caused by this can result in an unstable drone and it to lose altitude. To compensated for this all rotors must increase in thrust to ensure the drone stays in the air[1]. The rotors are controlled by electron speed controller (ESC). The ESC relies on the inputs from the controller to output power to the rotors.[2] The thrust of each rotor is controlled by its own ESC.

The thrust that each rotor receives during flight controls the direction in which the drone moves. The propeller's dimensions directly correlate to the movement of the drone. If a propeller has a 4" pitch the drone will move 4" forward every full rotation. The larger the propeller the more the drone would move every rotation. The downside of this is that the larger the propeller, whether diameter or pitch, the more energy it takes to spin it.[1]

III. Experimental

During this project we looked at multiple variables during printing the infill, the mass, and the strength. All three characteristics were related to one another. The infill percentage of a print is the amount of material that is inside the walls of an object. While 3D printing the extruder creates a grid like pattern inside the object. The lower the percentage the less material is used and the quicker a print is completed. If the infill was 40%, the part was heavier and stronger, if the infill was 20% the part was lighter and more flexible. The infill needs to high enough were it gives enough support, but is not too heavy. By weighing the flexibility factor for each piece, determining infill became easier. This relationship needed to be thought of when printing different aspects of the drone. Another choice that needed to be made was what printer was more beneficial for each part. The MakerBot Replicator printed smaller pieces accurately and faster but a smaller printing area. The Ultimaker S5 Pro printed at a slower rate, had the ability to print two different materials at once, and a large printing area. Ultimately, it was decided that the pieces that were small and could be printed in larger batches would be printed on the MakerBot and the larger pieces would be printed on the Ultimaker. We chose the MakerBot for the smaller pieces as the accuracy of the printing tip resulted in better quality pieces. The ability to print small pieces in a batch was far easier with this printer as the supports were easy to remove and did not alter the individual pieces structural integrity. As for the larger pieces, the Ultimaker had a stronger plastic called tough PLA that it printed with. This gave pieces such as the top and bottom of the chassis a stronger structure to support the battery, arms, and Arduino.

Throughout the printing process the mass of the drone was carefully examined. Each motor has 0.860 kg of thrust, giving the entire drone a maximum weight of 3.440 kg. We also took the average consumer drone weight into consideration, 0.250 to 1.300 kg. The goal mass for the drone was 1 Kg. This was determined by taking the average consumer drone and subtracting weight that comes with components not being utilized in this build such as cameras and an outer shell. With the mass of all the parts in consideration, as seen in table 1, finding areas to take material and mass away was important. To reduce the mass, the first version of the spacers had connecting pieces, the second version we removed all material connecting the spacers. This redesign saved 10g from the overall mass. To be able to have sustained flight time it was important to examine the mass of each part. The parts that were purchased, such as the

Table 1: Infill Percentages and Mass of Each Part of the Drone

Part	Infill	Mass (g)
Top Plate	25%	22
Bottom Plate	40%	24
Arms - 4	40%	30
Battery - 11.1v LiPo 5200 mAh	-	300
Motor - 4	-	220
Arduino Nano	-	13.61
MPU6050	-	18.144
Propellers		
Spacers V2 - 12	10%	6
Total Mass		633.754

battery and motors, had a strict mass. There is nothing that could be altered to change the mass of these items. The 3D printed parts could differ in mass depending on the infill of the print. The higher the infill is the heavier the piece will be. Certain pieces can be printed at a lower infill, but others need the support and rigidness that a higher infill provides. The bottom plate for example needs to be printed with 40% infill. This results in a strong base that will not break due to the mass of the parts it holds. Conversely, the top plate could be printed with 25% infill. This produced enough structural strength and flexibility with adding little weight to the overall mass. If the part was printed with a higher infill the weight would have increased substantially. To find the ideal print infill was difficult and took printing the same item two to three times to find what was best for overall build.

With a drone body completed, the flight controller must now be built. The flight controller utilizes the Arduino nano and an MPU6050 to relay inputs from the remote control to the ESCs. The MPU6050 is a micro electro-mechanical system. It acts as a three-axis accelerometer and gyroscope. The MPU6050 measures velocity, orientation, acceleration, and displacement.[3] The combination of the programmed Arduino nano, MPU6050, and the NRF24 receiver let a user input a command using the remote control. Since the MPU6050 and Arduino nano are two separate boards a printed circuit (PCB) was used to solder and connect the proper inputs. This was a tedious task, as every connection needed to be checked with an ohm meter. This ensured that there were no shorts in the circuit.

The remote control utilizes an Arduino mini and NRF24 transmitter to communicate with the receiver attached to the drone. Similar to the program that is in each of the Arduino nanos, the Arduino mini is uploaded with a controller program. This program is easily customizable to the specifications of the controller. In this case, changes were made to the input in the program to accommodate for the inputs used in the Arduino. The code was then uploaded to the Arduino using a USB cable. The code was written using Arduino's IDE software. This allows users to write their own code and makes the project making process entirely user created. Similar to the flight controller, a two-layer PCB was used to solder all of the components of the controller. This included two potentiometers, four buttons, two switches, Arduino Pro mini, an MPU6050, two two-axis joysticks, and a NRF24 transmitter. Implementing the previous components with a few capacitors and a 3.3v surface mounted voltage regulator the code can be uploaded with a unique transmitter ID. This gives the transmitter a unique code that only the receiver with the same ID will comprehend the commands that it is giving.

The Arduino programs along with the sensors of the MPU6050 calculate and adjust the thrust of each rotor. This creates stable flight and allows the drone to move in a smooth manner. The drone would be very rigid if it could only go up, down, and side to side. With the MPU6050 communicating with the ESC it is possible for the drone to tilt and go around corners easier.

IV. Results & Discussion

Learning how 3D printers work and their capabilities give creating objects infinite possibilities. Creating a drone using a 3D printer was seamless when using .stl files. The files provided by ElectroNoobs aligned with the goal of making a lightweight drone. While printing questions regarding what infill percentage should be used, if supports were needed, and even what printer were asked constantly. These questions proved to be necessary for each piece.

Each piece used its own settings for printing because each piece served a unique function. Some of the challenges during printing was keeping the material adhered to the plate. This was difficult on the MakerBot printer as the plate was not heated. If the bottom surface of the print did not stay to the build plate it could cause warping and even complete print failure. This occurred multiple times while printing smaller pieces. The warping rendered certain pieces to be useless by being stuck to the build base. Similar struggles occurred with the Ultimaker, the support feature needed to be used with the second extruder and dissolvable filament. The filaments extrude similar to PLA, but when cooled and introduced to water the filament rubs off quite easily. This process was used to print propellers. However, this was unsuccessful as the printer had trouble using both filaments. This was overcome by using propellers bought from a drone store. These worked much better as they were smooth and uniform.

Soldering small circuits was also difficult and caused many problems. To start, the first Adruino nano board was over heated and was unusable. This was caused by soldering the board onto the pins prior to soldering the circuit on the underside of the board. After finding this issue, the Arduino board was replaced, and each connection was checked with an ohm meter to ensure the circuit was not shorted. Once this issue was resolved, the Arduino code could be uploaded onto the board. This took some tinkering as some of the inputs needed to be assigned. This was something that the youtube videos and tutorial page laid out step by step. The issue beyond this was unforeseen. A set of pins that the ESCs plug into was forgotten in the tutorial, and while plugging the ESCs it shorted the circuit, frying yet another Arduino nano board. This was caused by the positive end of the ESC going directly to the digital PWM inputs.

With all the bad, there was still some good. The controller and receiver worked well with the flight controller. Using a Multiwii program the electronics were able to be tested. Before the second board was shorted and fried there was rotation in each of the motors. This was a sign of progress towards flight.

V. Conclusion

This project was not a total failure. It taught valuable lessons in research, design, and development. Failing multiple times during this project created experiences in troubleshooting. Whether with circuits or 3D printing finding solutions is important to creating a working product. Designing products take multiple attempts. We did not finish this project in one semester, but the steps needed to finish the project are small. First, a new Arduino nano board and MPU6050 wired together to create a new flight controller unit. This will then connect to the receiver which will communicate with the controller. Additionally replacing the 11.1 V – 5200mAh battery with a 1300-1500mAh will reduce even more weight and create longer flight time. With these few adjustments this project will be a complete success. This project taught valuable lessons that will be taken into the field upon graduation. To create a successful product, one must fail first. There can never be perfection after one try, something can always be improved.

References

[1]C. Johnson, Drone (2020). https://drone-zoom.com/how-do-drones-work-and-fly/

[2]CTIA Research Team, CTIA (2019). https://www.ctia.org/news/up-up-and-away-how-do-droneswork#:~:text=Drones%20use%20their%20rotors%E2%80%94which,descend%2C%20when%20pilots%20perform%20the

[3]CiferTech, Arduino Project Hub (2019). https://create.arduino.cc/projecthub/CiferTech/what-is-mpu6050-b3b178?ref=user&ref_id=1197733&offset=10

[4]Electronoobs, How to build your own drone (2017). http://electronoobs.com/eng_robotica_tut5.php