Arduino

First Person View (FPV) Quadcopter

Drexel Lagare

Senior Design I

CMPE 4372.01

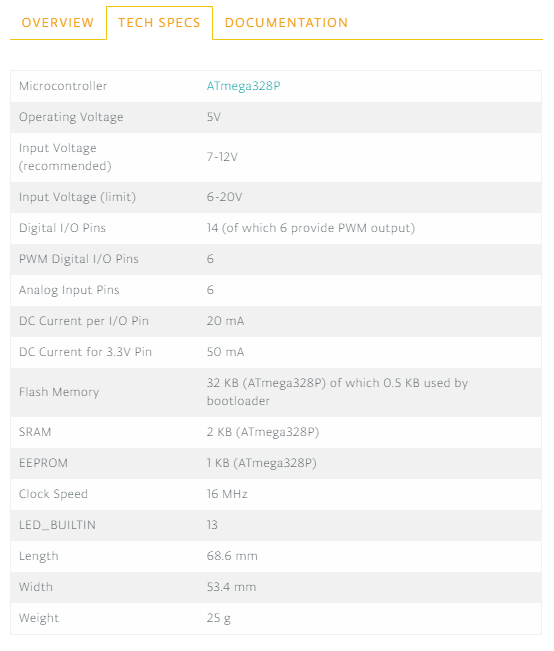
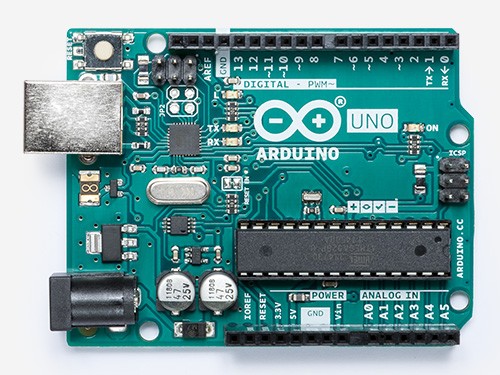
**What is Arduino?**

Arduino is an open-source electronics platform based on easy-to-use hardware and software. [Arduino boards](https://www.arduino.cc/en/Main/Products) are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the [Arduino programming language](https://www.arduino.cc/en/Reference/HomePage) (based on [Wiring](http://wiring.org.co/)), and [the Arduino Software (IDE)](https://www.arduino.cc/en/Main/Software), based on [Processing](https://processing.org/).

**Arduino**

**Arduino UNO**

(<https://store.arduino.cc/usa/arduino-uno-rev3>)

****

* Most Used and documented of all

the Arduino Family

* Recommended as the best board to get started

With electronics and coding

* Most popular to use when building quadcopter
* It has 14 digital input/output pins (of which 6 can be

used as PWM outputs), 6 analog inputs, a 16 MHz

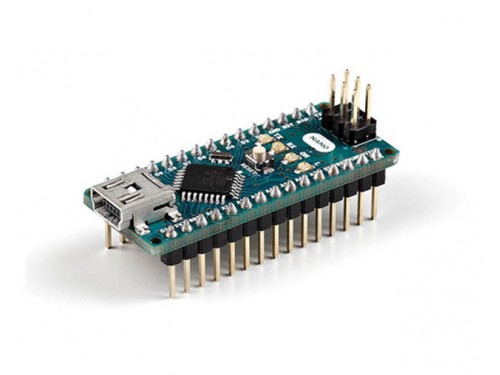
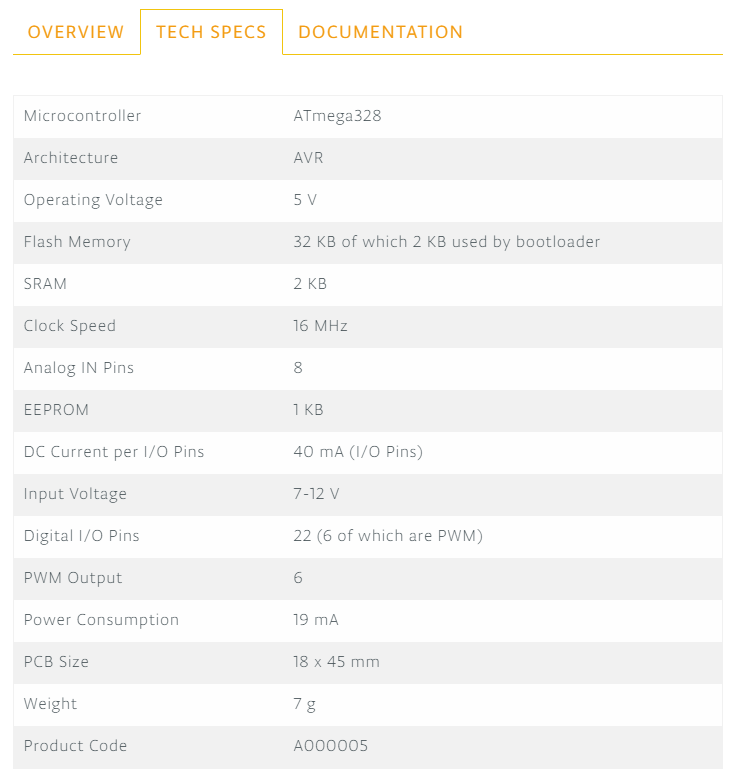
quartz crystal, a USB connection, a power jack,

an ICSP header and a reset button

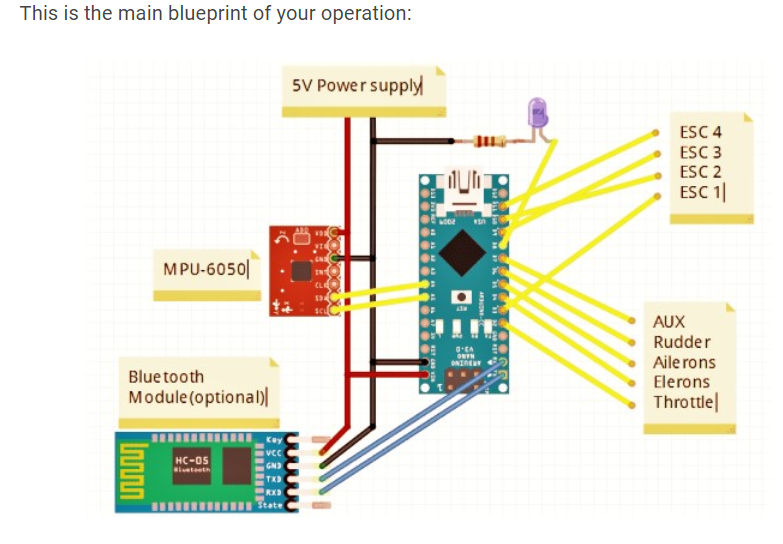
* Arduino Uno Quadcopter Guide: <http://www.droneybee.com/arduino-quadcopter-guide/>
* Project YMFC-AL Guide - The Arduino auto-level quadcopter: <http://www.brokking.net/ymfc-al_main.html>
* DIY ARDUINO FLIGHT CONTROLLER Guide: <https://www.instructables.com/id/DIY-ARDUINO-FLIGHT-CONTROLLER/>
* Autopilot Drone Guide: <https://www.hackster.io/suhaskd/autopilot-drone-c07cc1>

**Arduino Nano**

(<https://store.arduino.cc/usa/arduino-nano>)



* One of the most popular to use for quadcopter DIY builds
* Each of the 14 digital pins on the Nano can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions.
* Arduino Nano Quadcopter Guide: <https://www.mydronelab.com/blog/arduino-quadcopter.html>

****

**Arduino Software** (<https://www.arduino.cc/en/main/software>)

* Code online and save your sketches on the cloud
* Always have the most up-to-date version of the IDE
* Includes all contributed libraries and support for new Arduino Boards

**Arduino Programming Language** (<https://www.arduino.cc/reference/en/>)

* Functions
* Variables
* Structure
* Libraries

**Arduino with Prices:**

|  |  |
| --- | --- |
| **Arduino** | **Price** |
| [**Arduino Uno Rev3**](https://store.arduino.cc/usa/arduino-uno-rev3) | **$22.00** |
| [**Arduino Ethernet Shield 2**](https://store.arduino.cc/usa/arduino-ethernet-shield-2) | **$23.65** |
| [**Arduino Mega 2560 Rev3**](https://store.arduino.cc/usa/mega-2560-r3) | **$38.50** |
| [**Arduino Due without Headers**](https://store.arduino.cc/usa/arduino-due-without-headers) | **$37.40** |
| [**Arduino Due**](https://store.arduino.cc/usa/due) | **$38.50** |
| [**Arduino Micro**](https://store.arduino.cc/usa/arduino-micro) | **$19.80** |
| [**Arduino Micro without headers**](https://store.arduino.cc/usa/arduino-micro-without-headers) | **$17.60** |
| [**Arduino Leonardo without Headers**](https://store.arduino.cc/usa/arduino-leonardo-without-headers) | **$17.60** |
| [**Arduino Leonardo with Headers**](https://store.arduino.cc/usa/leonardo) | **$19.80** |
| [**Arduino MKR WiFi 1010**](https://store.arduino.cc/usa/mkr-wifi-1010) | **$33.90** |
| [**Arduino MKR1000 WIFI**](https://store.arduino.cc/usa/arduino-mkr1000) | **$34.99** |

|  |  |
| --- | --- |
| **Arduino** | **Price** |
| [**Arduino MKR1000 WIFI with Headers Mounted**](https://store.arduino.cc/usa/arduino-mkr1000-with-headers-mounted) | **$35.99** |
| [**ARDUINO UNO WiFi REV2**](https://store.arduino.cc/usa/arduino-uno-wifi-rev2) | **$44.90** |
| [**Arduino Motor Shield Rev3**](https://store.arduino.cc/usa/arduino-motor-shield-rev3) | **$22.00** |
| [**Arduino MKR GSM 1400**](https://store.arduino.cc/usa/mkr-gsm-1400) | **$69.90** |
| [**Arduino MKR WAN 1300 (LoRa connectivity)**](https://store.arduino.cc/usa/mkr-wan-1300) | **$39.90** |
| [**Arduino MKR Vidor 4000**](https://store.arduino.cc/usa/mkr-vidor-4000) | **$74.90** |
| [**Arduino MKR ZERO (I2S bus & SD for sound, music & digital audio data)**](https://store.arduino.cc/usa/arduino-mkrzero) | **$21.90** |
| [**Arduino Yún Rev 2**](https://store.arduino.cc/usa/arduino-yun-rev-2) | **$59.00** |
| [**Arduino MKR NB 1500**](https://store.arduino.cc/usa/arduino-mkr-nb-1500) | **$83.90** |
| [Arduino MKR RGB Shield](https://store.arduino.cc/usa/mkr-rgb-shield) | **$44.85** |
| [**Arduino Nano**](https://store.arduino.cc/usa/arduino-nano) | **$22.00** |
| [**Arduino Zero**](https://store.arduino.cc/usa/arduino-zero) | **$42.90** |

**Comparisons of each reference.**

1. **Why this component over the other component.**
2. **Comparison of builds from guides**

**Guides to compare:**

**Android Uno**

1. Arduino Uno Quadcopter Guide: <http://www.droneybee.com/arduino-quadcopter-guide/>
2. Project YMFC-AL Guide - The Arduino auto-level quadcopter: <http://www.brokking.net/ymfc-al_main.html>
3. DIY ARDUINO FLIGHT CONTROLLER Guide: <https://www.instructables.com/id/DIY-ARDUINO-FLIGHT-CONTROLLER/>
4. Autopilot Drone Guide: <https://www.hackster.io/suhaskd/autopilot-drone-c07cc1>

**Android Nano**

1. Arduino Nano Quadcopter Guide: <https://www.mydronelab.com/blog/arduino-quadcopter.html>

(1) Arduino Uno Quadcopter Guide: <http://www.droneybee.com/arduino-quadcopter-guide/>

1. The 1st Approach is to use one of the drone kits on [Best quadcopter kits](http://www.droneybee.com/best-quadcopter-kits-beginners/) website to have the frame, motors, ESC(speed controller) and battery picked for the quadcopter. The flight controller will be the Arduino Uno and all that is left is to find is a transmitter and receiver.
2. The 2nd approach is to start from scratch and pick the frame, motors, ESC(speed controller), battery, transmitter and receiver for the quadcopter which is mentioned in [chapter 2](http://www.droneybee.com/arduino-quadcopter-guide/#tab-con-2) of the dronebee website. The flight controller will be the Arduino Uno.
3. Gives insight on Choosing motors and propellers on this [website](http://www.droneybee.com/choosing-motors-and-propellers-for-multirotors/). Recommendation to for aim for at least a Thrust to Weight ratio of 2:1 meaning that at 50% thrust the drone should be able to hover. The other 50% should be for speeding and gaining altitude.
4. How to program flight controller on this [website](https://blog.owenson.me/build-your-own-quadcopter-flight-controller/).
5. **Autonomous Flight :** Use GPS Navigation and Altitude Hold module for autonomous flight implementation
   1. **GPS Navigation:** The ArduPilot provides libraries for parsing GPS data into latitude and longitude, you’ll just need a PID to convert desired speed into pitch/roll, and another PID for converting distance to waypoint into desired speed. You can use the compass to work out direction to your waypoint, and then just translate that into the right amount of pitch and yaw.
   2. **Altitude Hold**: You can sense altitude with the barometer that is build onto the ArduPilot board. You’ll need two PIDs, one to calculated throttle alterations from desired ascent/descent rate and a second to calculate desired ascent/descent from distance to desired altitude.

**(2) Project YMFC-AL Guide -** The Arduino auto-level quadcopter: <http://www.brokking.net/ymfc-al_main.html>

1. YMFC-AL quadcopter is an auto-leveling Arduino based quadcopter. Auto leveling means that when the sticks on the remote is released the quadcopter will level itself.
2. Uses MPU-6050 gyro/accelerometer for the auto level feature on the drone instead of having to program the auto level on the Arduino Uno.
3. Complete YMFC-AL software package with code already configured. Open using the Arduino IDE.
4. Code already written from YMFC-AL software package, just need to follow videos on Joop Brokking [youtube](https://www.youtube.com/watch?v=XFxqFQwRumc&list=PL0K4VDicBzsibZqfa42DVxC8CGCMB7G2G).
5. Quadcopter PID controller and PID tuning explained and shown on [video](https://www.youtube.com/watch?v=JBvnB0279-Q&list=PL0K4VDicBzsibZqfa42DVxC8CGCMB7G2G&index=6).
6. Autonomous flight GPS Implementation for Quadcopter [video](https://www.youtube.com/watch?v=kX4udwHve6w&list=PL0K4VDicBzshwCpUHzIB6hOLQVkDFHbxC&index=9).
7. Waypoint App in Visual Studio for the YMFC-32 Arduino Drone [video](https://www.youtube.com/watch?v=EeJeSE_5rns&list=PL0K4VDicBzshwCpUHzIB6hOLQVkDFHbxC).

**(3) DIY ARDUINO FLIGHT CONTROLLER Guide:** <https://www.instructables.com/id/DIY-ARDUINO-FLIGHT-CONTROLLER/>

**1.** Arduino based flight controller that can do wireless **PID tuning, Orientation lock and altitude hold and position lock with the help of GPS.** PID is an acronym for Proportional, Integral and Derivative in which the name comes from how the controller deals with disturbances in the system.

**2.** This build focuses on the flight controller. Assumes you already have built the drone with the necessary parts and just need to program the Flight controller.

**3.** Arduino flight controller will have features like the NAZA and APM flight controllers. Bluetooth module used to help tuning step without the use of computer

**4.** Gives tutorial on designing and building PCB fabrication.

**5.** **Multiwii Arduino program and Multiwii GUI** for PID tuning, orientation lock, altitude hold and position lock with GPS.

**6.** Program drone with your phone: [**EZ-GUI Ground Station**](https://play.google.com/store/apps/details?id=com.ezio.multiwii)

**(4) Autopilot Drone Guide:** <https://www.hackster.io/suhaskd/autopilot-drone-c07cc1>

**1.** Arduino Autopilot drone uses multiple sensors and a wireless camera controlled by two microcontrollers. **DOES NOT** use a RC remote to control the drone.

**2.** Drone will be handled by itself, flight controller will control the actions of the drone using the sensors to balance the drone.

**3.** Arduino Uno will be the brain of the system on the drone which will give the proper signal in the form of PWM (Pulse Width Modulation) to flight controller.

**4.** Flight controller needs PWM signals for input to control the individual motors. These signals are usually generated by the RC remote but since we are not using one the Arduino Uno will provide PWM the signal.

**5.** OpenPilot CC3D microcontroller with a camera to capture live data with weather monitoring sensors. Has build in gyro and accelerator controller with auto leveling feature.

**6.** Uses Bluetooth module to turn drone on/off and display the live data using an Android mobile device.

**(5) Arduino Nano Quadcopter Guide:** <https://www.mydronelab.com/blog/arduino-quadcopter.html>

**1.** Approach of this guide is to let user pick the frame, motors, propellers, electronic speed controller, battery, inertial measurement unit(IMU), and RC transmitter from multiple suggestions to create drone.

**2.** You can either get a regular flight controller or use the Arduino UNO to build your own flight controller.

**3.** Connects Arduino Uno with [**MPU-6050 Module**](https://www.amazon.com/HiLetgo-MPU-6050-Accelerometer-Gyroscope-Converter/dp/B01DK83ZYQ?tag=dronelabtrack-20) gyroscope/Accelerometer sensor module to get values for accelerometer and gyroscope data.

**4**. Guide has tutorial on how to use Multiwii to program Arduino Uno Flight Controller and Multiwii GUI as the interface

**5.** Multiwii GUI allows you to change the PID values and fine tune your quad to desired preferences. PID is an acronym for Proportional, Integral and Derivative in which the name comes from how the controller deals with disturbances in the system.