1. **Abstract**

The objective of this project is to build a quad-copter that can be controlled by gesture wirelessly. User is able to control motions of the quad-copter by wearing the controller glove and performing predefined gestures.

This report covers the design, analysis, manufacturing, and testing of an autonomous quad-copter. A control system was designed and implemented through the use of onboard microprocessor and inertial measurement system. The goal of the quad-copter was to maintain a hover at a used-defined altitude while minimizing lateral drift. In addition to achieve self-stability, the quad-copter attained a few weight reduction from other designs and which led to an increased flight time.

# Acknowledgments

# We want to express our gratitude to all the people who have given their heart whelming full support in making this project a magnificent experience.

# We also wanted to thank our family who inspired, encouraged and fully supported us for every trial that comes our way and for supported us to doing this project and for giving us not just financial, but morally and spiritually support.

# Although any expression of acknowledgment will fail to fully capture the importance of their role, we would still like to thank each of them, and also to say how much we appreciated their contributions and enjoyed our interactions.

# To the staff of College of Engineering and Computer Engineering Department, in particular, for their support of science, that made us to get this place of knowledge and for their effort to getting us the graduation year.

# A lot thanks go to Head of Computer Department Dr. Luai Malhis , who approved the project , and encouraged us to fly in this project.

# We would also like to thank all those that helped the workshops become a reality, and all others who, in various capacities, helped this project come to a successful conclusion.

# We whole heartedly thank you for the kindness and patience that you have given us.

1. **Introduction**

The quad-copter is one of the most complex flying machines due to its versatility and maneuverability to perform many types of tasks. Classical quad-copters are usually equipped with a four rotors. Our specific project is concerned with the design and control of a miniature rotorcraft, known as a quad-copter.

Quad-copters are symmetrical vehicles with four equally sized rotors at the end of four equal length rods. Early designs of quad-copters were completed in the 1920's by Etienne Omichen, Dr. George de Bothezat and Ivan Jerome. These designs, however, never truly grasped the attention of the public or the in case of Dr Bothezat and Jerome the military. Therefore, neither Omichen's or Bothezat and Jerome's were mass produced. This fact, however, does not discredit the advantages of quad-copters. Unlike their counter parts, quad-copters make use of multiple rotors allowing for a greater amount of thrust and consequently a greater amount of maneuverability. Also, the quad rotors symmetrical design allows for easier control of the overall stability of the aircraft.

Each of the rotors on the quad-rotor helicopter produces both thrust and torque. Given that the front and rear motors both rotate counter-clockwise and the other two rotate clockwise, the net aerodynamic torque will be zero, as seen in Figure 1.

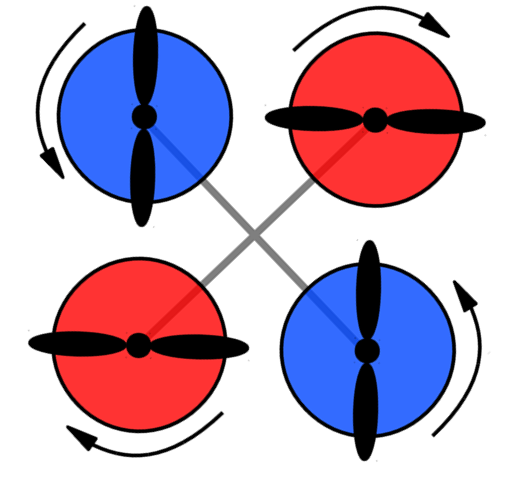
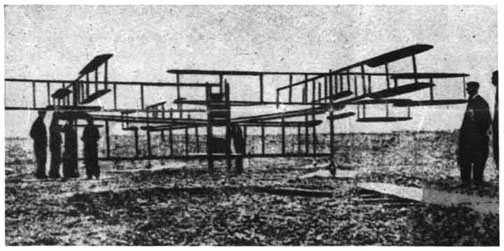


Figure ghhhjj

**Figure 1- Torque Patterns and Related Motion**

Our project involves the design of an autonomous quad-copter. Along with optimizing frame design and weight reduction, we have designed controls for the quad rotor with the use of the ATmega328 and a custom C code. With the controls by hand motion which make controlling quad-copter very smooth and easy .

1. **Background**

Although unconventional in appearance by contemporary standards, four rotor helicopters distinguished themselves early in the development of rotary wing aircraft beginning in 1907. On August 24th of that year, the first flight of a manned helicopter took place when the Bre´guet-Richet Gyroplane No. 1 lifted off the ground in France. Depicted in Figure 2.1, the Gyroplane No. 1 featured four rotors mounted at the tips of a cross- shaped fuselage. The brief flight attained an altitude of only two feet and, because stability and control were lacking, the quad-copter motion was limited by four tethers.

Hiller Aviation Museum (used with permission)

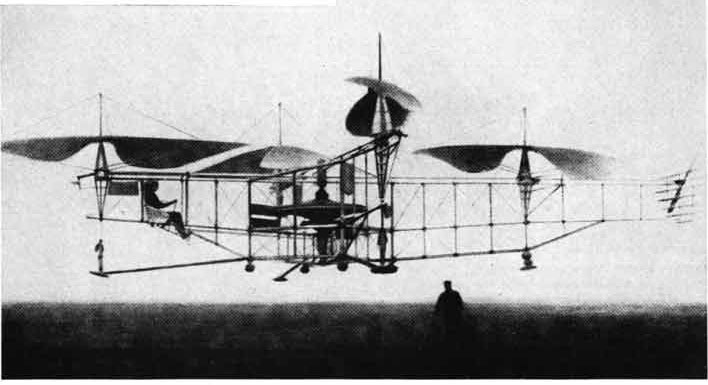
Figure 2.1: Bre´guet-Richet Gyroplane No. 1 (1907)

In 1921 the United States Army Air Corps funded a more successful program led by Dr. George de Bothezat. De Bothezat’s quad-copter, pictured in Figure 2.2, was developed in secrecy at McCook Field, predecessor to Wright-Patterson Air Force Base, Ohio, until its first flight on December 18th, 1922. In addition to the four main rotors, two additional propellers were used for directional control and two more situated above the engine provided both cooling and some additional lift. Over the next year De Bothezat conducted over 100 test flights. One notable example highlighted the quad-copter’s stability by flying with three men hanging below three of the four engines to provide an asymmetric weight distribution. De Bothezat received additional funding to demonstrate improved performance, but when it failed to yield anticipated results and the Army Air Corps cancelled the program in 1923.

National Museum of the Air Force

Figure 2.2: De Bothezat’s quadrotor flying at McCook Field (1922)

Later, Etienne Oemichen established the viability of the quad-copter configuration when he set the Fe´de´ration Aeronautique Internationale’s helicopter distance record on April 14,1924. Oemichen went on to fly over one thousand test flights with his quad-copter, pictured in Figure 2.3, which also used additional propellers for control. By 1956, D. H. Kaplan had further refined the quad-copter and demonstrated the ability to control aircraft attitude using differential thrust between pairs of rotors. This concept was tested on the H-frame Convert a wings Model A. Although the Model A never entered into production, the use of differential thrust for control has carried forward to the contemporary four rotor model aircraft and research testbeds.



Hiller Aviation Museum (used with permission)

Figure 2.3: Oemichen’s record setting quadrotor (1924)

An early quad-copter UAV was built by the Paisecki Aircraft Corporation to meet naval requirements for a weapons delivery platform launched from naval destroyers. The PA-4 Sea Bat achieved hover and maintained a level attitude during a tethered flight in 1958 using differential tilt for control. Then, except for experimental tilt-rotors, which rotate the engine nacelles to transition between rotary wing and fixed wing modes of operation, the quad-copter design fell out of favor for the next half century.

**4. Principle of Operation**

**Quad-copter movment:**

Before we take a look on the physics, it is important to understand how a quad-copter moves and how we can control it.The figure 55 shows the different movement of a quad-copter and how we control it.

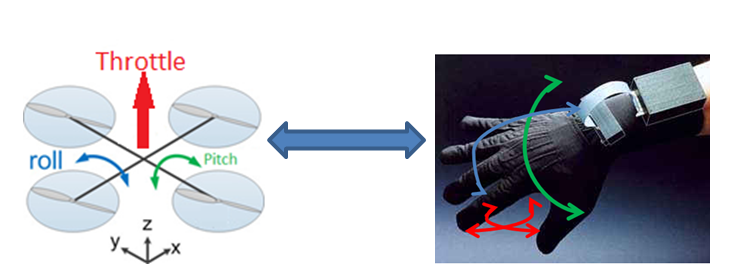
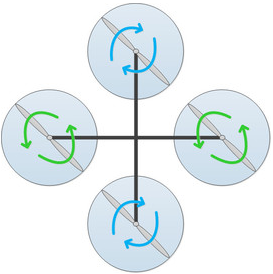


Figure 55: Movement of Quad-Copter and Hand.

Movement of Quad-Copter:

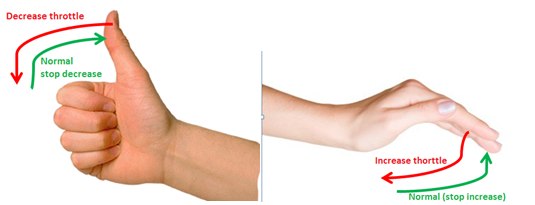
Pitch and roll angles are generated with different speed of opposed rotors. To move the quad-copter upward (throttle), the speed of every motor is increased. The tricky part is the yaw angle. If every rotor would turn in the same direction, the quad-copter would start turning around the z axis (like a helicopter without a rear rotor). Therefore rotation directions are configured as seen in figure 3.

****

**Figure 55: Rotor Direction.**

**Hand Motion:**

We control the roll of quad-copter by rotate the hand left and right, and control the pitch by rotate the hand up and down, to control the value of throttle(speed) by fingers motion as show in figure 56.

****

**Figure 56: Finger motion to control throttle.**

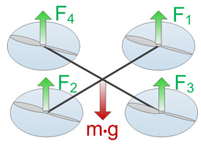
**Hovering:**

For hovering a balance of forces is needed. The picture below shows such a situation. If we want the quad-copter to hover, **SUM(Fi)** must be equal **m•g** (where **m** is the mass of the quad-copter, **g** the gravity acceleration and **F1** - **F4** the forces of the motors). For this simple example we assume all motors are equal and have the same force. So if **SUM(Fi)** is smaller then **m•g** than the quad-copter is declining, if it is greater, it is climbing.

SUM(Fi) > m•g <=> climb

SUM(Fi) = m•g <=> hover

SUM(Fi) < m•g <=> decline

****

**Figure 66: Balance of Power while hovering.**

## Tilting

Now let us take a look on what is happening when we tilt the quad-copter. Figure 5 shows such a situation. For simplification only two of the four rotors are shown. We see that the force is divided in two different parts. **FL1** and **FL2** are the part of the force used to lift the quad-copter. **FT1**and **FT2** represents the part used for the translation. It is obvious that the lift part becomes smaller with increasing **φ**.

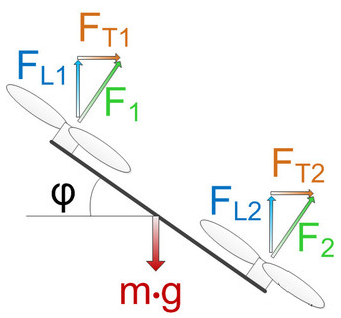


Figure 55: Force Distribution for Tilting.

1. Hardware Implementation

6.1 Overview

In this chapter we will discuss the component of our project. Which consist of two main parts. First section will illustrate the quad-copter components. Second section will illustrate wireless gesture components.

**Quad-copter components:**

1-Frame.

2-Microcontroller ( Arduino Uno).

3-Motors(A2217-9 Brushless Outrunner Motor).

4- **Electronic Speed Controller (ESC)**

5- Lithium Polymer Battery.

6- Propeller.

7- *Inertial Measurement Unit (IMU Digital Combo Board).*

8-RF receiver.

The following figure shows the **quad-copter components:**

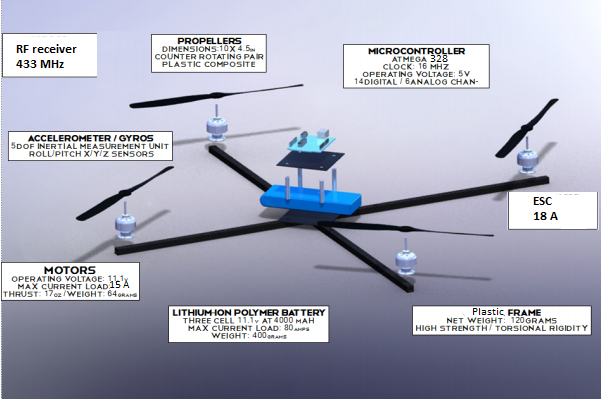


Figure 55: **Quad-copter components.**

Wireless  gesture  components:

1. Microcontroller ( Arduino Mini Pro).
2. *Accelerometer* (ADXL 335).
3. Flex sensor.
4. RF transmitter.

The following figure shows the Wireless  gesture  components**:**

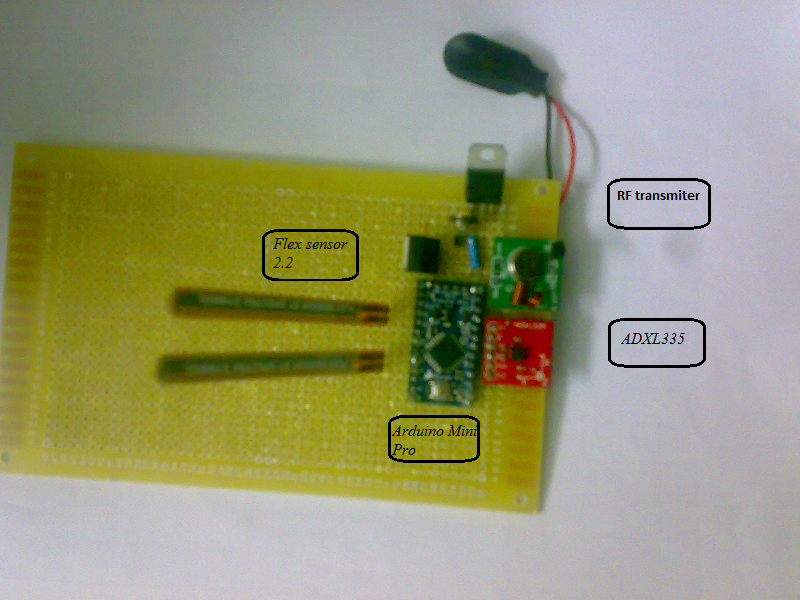
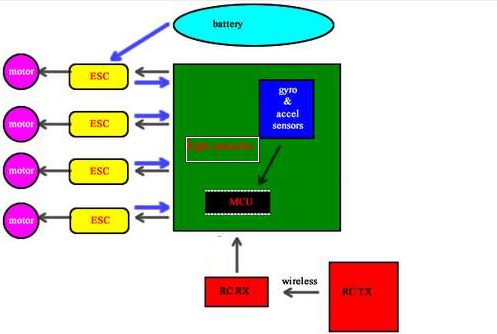


Figure 55: Wireless  gesture  components.

* 1. **Quad-copter Component Modeling**

****

6.2.1 **Frame**

In considering the frame, the first consideration is the material to be used. It must be lightweight, sturdy, and affordable. The forces which act on the aircraft primarily will be gravity and air pressure. Gravity allows for construction under the guidance of a limited mass to allow for structural stability on the ground, as well as control of the copter in the air. Air pressure, which is used to determine the airspeed, will affect the quad-copter’s stress on the screws at higher altitudes. The higher the altitude, the lighter the air, the smaller the forces against the frame, which implies the copter’s frame, is being stretched. This is what is kept in mind when considering for the base material for our aircraft. For the project, three materials are possibilities due to their popularity in the RC World: wood, aluminum and plastic.

Plastic was the best choice. With Plastic, test flights can be performed repeatedly without requiring reinforcement from another source. Furthermore, due to its increased strength to stress, plastic is less likely to bend due to take-off or stable flight; also, it carries a stronger stability to the frame. Plastic is less weight from the other material to meet the minimum quad-copter requirements.

.

The lightweight frame must be designed to support all the quad-copter subsystems. So a prototype frame was designed with a 12cm X 12cm square plastic central plate with four radiating 27cm. Plastic struts which are shown in figure 49.

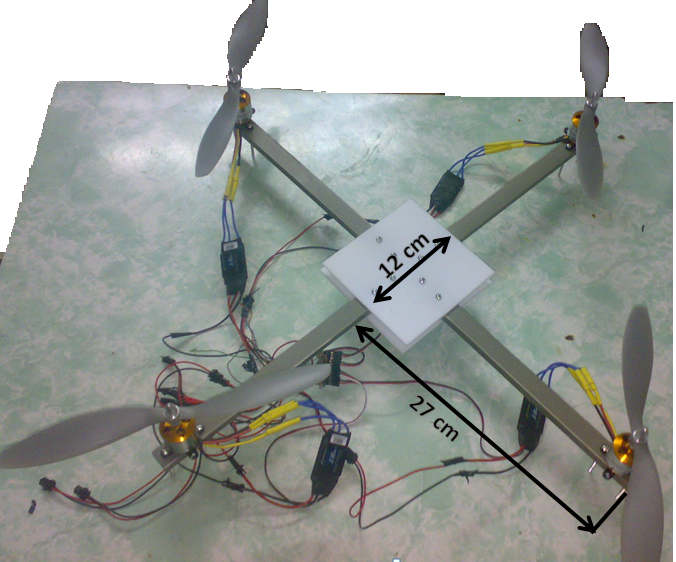


Figure 49: Plastic prototype frame with close-up of central plate

**6.2.2Microcontroller**

The Arduino Uno, Shows in Figure 31 and Figure 32, is a microcontroller board based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header. Table 31 shows the specification of Arduino Uno.

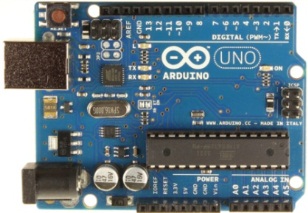


Figure 31: Arduino Uno R3 Front.

|  |  |
| --- | --- |
| Parameter | Value |
| Microcontroller | ATmega328 |
| Operating Voltage | 5V |
| Input Voltage (recommended) | 7-12V |
| Input Voltage (limits) | 6-20V |
| Digital I/O Pins | 14 (of which 6 provide PWM output) |
| Analog Input Pins | 6 |
| DC Current per I/O Pin | 40 mA |
| DC Current for 3.3V Pin | 50 mA |
| Flash Memory | 32 KB (ATmega328) of which 0.5 KB used by bootloader |
| SRAM | 2 KB (ATmega328) |
| EEPROM | 1 KB (ATmega328) |
| Clock Speed | 16 MHz |

Table 31: Arduino Uno specification.

**ATmega328**

High Performance, Low Power AVR 8-Bit Processor in 28 pin DIP package. It's like the ATmega168, with double the flash space. 32K of program space. 23 I/O lines, 6 of which are channels for the 10-bit ADC. Runs up to 20MHz with external crystal. Package can be programmed in circuit. 1.8V to 5V operating voltage. Figure 32 shows the Pin configuration.

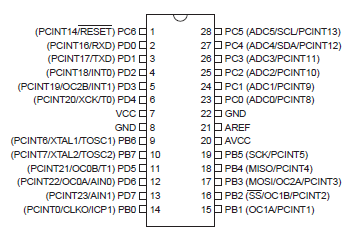


Figure 32: Pinout Configuration of ATmega328

The ATmega328 is a low-power CMOS 8-bit microcontroller based on the

AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega328 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

**Memory**

The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM.

**Input and Output**

Each of the 14 digital pins on the Uno can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

1. Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
2. External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.
3. PWM: 3, 5, 6, 9, 10, and 11.
4. SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.
5. The ATmega328 also supports I2C (TWI).

**6.2.3Motors**

**6.2.3.1 Intoduction**

Motors are the starting point when calculating flight stability and control. The motors chosen should meet the following objectives:

• Lightweight.

• High speed and torque.

• Cost effective.

• PWM speed controlled.

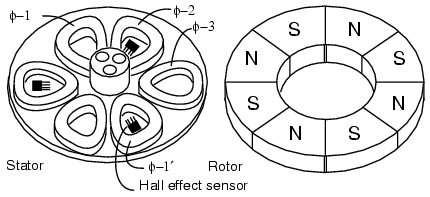
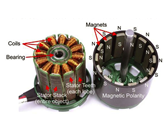
• Synchronized.

After a deep search to find the motor that achieves specifications mentioned earlier. Brushless motor is the best options available.

* + - 1. **Brushless Motor**

**6.2.3.2.1 Introduction**

The brushless motor, unlike the DC brushed motor, has the permanent magnets glued on the rotor. It has usually 4 magnets around the perimeter. The stator of the motor is composed by the electromagnets, usually 4 of them, placed in a cross pattern with 90o angle between them. The major advantage of the brushless motors is that, due to the fact that the rotor carries only the permanent magnets, it needs of NO power at all. No connection needs to be done with the rotor, thus, no brush-commutator pair needs to be made! This is how the brushless motors took their name from. This feature gives the brushless motor great increament in reliability, as the brushes wear off very fast. Moreover, brushless motors are more silent and more efficient in terms of power consumption.

The 3-φ pancake motor (Figure above) has 6-stator poles and 8-rotor poles. The rotor is a flat ferrite ring magnetized with eight axially magnetized alternating poles. We do not show that the rotor is capped by a mild steel plate for mounting to the bearing in the middle of the stator. The steel plate also helps complete the magnetic circuit. The stator poles are also mounted atop a steel plate, helping to close the magnetic circuit. The flat stator coils are trapezoidal to more closely fit the coils, and approximate the rotor poles. The 6-stator coils comprise three winding phases.

If the three stator phases were successively energized, a rotating magnetic field would be generated. The permanent magnet rotor would follow as in the case of a synchronous motor. A two pole rotor would follow this field at the same rotation rate as the rotating field. However, our 8-pole rotor will rotate at a sub multiple of this rate due the the extra poles in the rotor.

* + - 1. **A2217-9 Brushless Outrunner Motor**

This is an overall excellent brushless motor for average sized quad-copter (< 2kg). Recommended props are the APC 10x4.7. Selection an ESC must be rated at 18A. We love these motors because they come with pre soldered bullet connectors and they do not have a long shaft sticking out at the motor mount area, making it easier to mount our motors to the quad-copter. Figure 34 shows our Brushless Outrunner Motor.



Figure 34: A2217-9 Brushless Outrunner Motor.

The motor features a 4mm hardened steel shaft, dual ball bearings for the shaft and has 3.5mm gold spring male connectors already attached with 3 female connectors included for our speed control. Also includes aluminum radial cross mount set with 6mm prop adapter. Mounting holes are spaced 19mm and 16mm on center and are tapped for 3mm (M3) and 2.6mm (M2.6) screws. Figure 35 shows the motor specification.

|  |  |
| --- | --- |
| Parameter | Value |
| Kv | 950 RPM/V |
| Max Efficiency | 80% |
| Max Efficiency Current | 5 - 15A (>75%) |
| No Load Current | 0.9A @10V |
| Resistance | 0.095 ohms |
| Max Current | 18A for 60S |
| Max Watts | 200W |
| Weight | 73.4 g / 2.59 oz |
| Size | 27.8 mm x 34 mm |
| Shaft Diameter | 4mm |
| Poles | 14 |

Figure 35: A2217-9 Brushless Outrunner Motor Specification.

* 1. **Electronic Speed Controller (ESC)**
     1. **Introduction**

A complicating factor with the use of brushless motors is the introduction of an electronic speed controller (ESC), a device external to the motor which electronically performs the commutation achieved mechanically in brushless motors. The ESC converts the battery pack DC voltage to a three phase alternating signal which is synchronized to the rotation of the rotor and applied to the armature windings. The motor speed is then proportional to the root-mean-square (RMS) value of the armature voltage and is set by the ESC in response to a pulse width modulated control signal. The relationship between the control signal and the voltage level is not necessarily linear and must be confirmed experimentally. For example, the ESC units used for this quad-copter could be programmed to provide either a linear power response or a linear speed response.

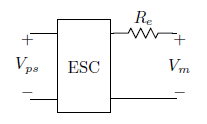


Figure 3.2: Electronic Speed Controller Model

Direct measurement of the three phase voltage or current can be made with a true RMS meter with sufficient bandwidth and current limits only if it is known whether the armature windings are connected in a wye or a delta configuration. An alternate approach is to model the ESC as shown in Figure 3.2. Accounting for the power consumed by the electronics, Pe, the power available to the motor can be determined from the power supply voltage and current as:

Pm = Vps Ips − Pe (3.8)

Power consumed by the ESC is expected to be negligible under normal operating conditions, however, this loss may be a factor when collecting measurements under low power, such as when calculating motor core loss with no load. Assumed constant, Pe can be determined from Equation (3.8) by applying power to the ESC without connecting the motor leads. The resistor, Re, shown in Figure 3.2, represents the resistance of the solid state switches and the value is provided by the manufacturer.

An issue encountered with the ESC used for the quad-copter is the presence of an automatic calibration routine that scales the output voltage based on the range of the motor control signal received. During preflight of a fixed wing remote controlled aircraft, the vehicle is restrained while raising the throttle to the maximum value ensuring that the calibration covers the full range of the motor control signal. However, a preflight run-up is not desirable with the quad-copter configuration.

It is important to understand how brushless motors and ESCs work, so that we don't destroy our ESCs.

To reverse the direction of motor spin, WE DO NOT REVERSE THE POWER INPUT TO THE ESCs.

We note that there are three wires that go between the motor and the ESC. These motors are three phase motors, meaning there are three coils inside. The coils are energized in sequence to make the motors spin. So the ESC's job is to energize the coils in sequence, but it needs to time each energization correctly so the motor can actually accelerate to the right speed. The ESC has a microcontroller inside that turns on or off the coils using FETs and also determines timing by measuring the feedback in the coils caused by the movement of the magnets.

* + 1. **Brushless ESC 18A**

This is a solid Electronic Speed Controller (ESC), shows in Figure 36, ideal for use with our quad-copter. It can support a max of 18 Amp continuous output to handle both large and small motors. Use this ESC with our Configurator to perform a throttle calibration with all motors at once with our quad-copter. Table 36 shows the ESC specification.

****

Figure 36: Brushless ESC 18A.

|  |  |
| --- | --- |
| Parameter | Value |
| Output | Continuous 18A, Burst 22A up to 10 Secs |
| Input Voltage | 2-4 cells lithium battery or 5-12 cells NiCd/NIMh battery |
| BEC | 2A / 5V (Linear mode) |
| Max Speed | 210,000rpm for 2 Poles BLM, 70,000rpm for 6 poles BLM, 35,000rpm for 12 poles Brushless Motors |
| Size | 45mm (L) \* 24mm (W) \* 11mm (H) |
| Weight | 18g |

Table 36: Brushless ESC 18A Specification.

**Signal output from MCU to ESC:**

The output from our microcontroller have a short period (1-2 ms pulse width ), as show in figure 55, (thus a higher frequency, about 250 to 300 Hz). The Turnigy Plush ESCs have been reported to be able to handle the higher frequency, and thus more adjustments in motor speed can be made per second, therefor the quad-copter will become more stable.

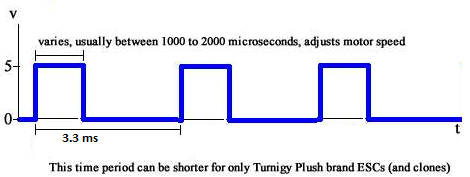


Figure 55: Signal output from microcontroller to ESC.

**Signal output from ESC to motor:**

The frequency of signal output from ESC to motors is 10-30Khz), as show in figure 55,. However, the output voltage 10.8-12 V, the motor consume 5-15A.

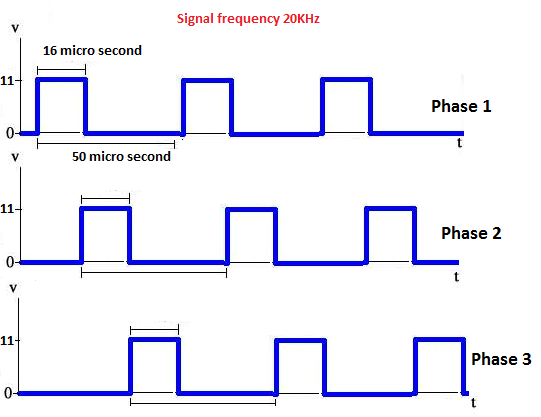


Figure 55: Signal output from ESC to motor.

We load the following settings to all of ESCs by programing a special code through Arduino:

* Brake: Off
* Battery Type: Li-xx
* Cut Off Type: Soft-Cut
* Cut Off Voltage: Low
* Start Mode: Normal
* Timing Mode: High
* Li-Po Cells: 0 (meaning auto-detect)
* Governor Mode: Off

**Programmable Items:**

1. Brake Settings: brake enabled/brake disabled (default is brake disabled).
2. Battery Type: Li-xx(Li-ion or Li-poly/Ni-xx(NiMh or Nicd) (default is Li-xx).
3. Low Voltage Protection Mode (Cutoff Mode): power reduction/ power cutoff (default is power reduction).
4. Low Voltage Protection Threshold (Cutoff Threshold): low/medium/high (default is medium cutoff voltage).
5. Startup Mode: normal/soft/super-soft (default is normal startup)
6. Timing: low/medium/high (default is medium timing). In normal cases, low timing can be used for most motors. But, for high efficiency, we recommend the Low timing for 2 poles motor and Medium timing for 6 pole motors and above. For higher speed, High timing can be used.
   1. **Lithium Polymer Batteries**
      1. **Introduction**

Lithium Polymer batteries are well suited for the quad-copter due to their high specific energy. These batteries also have a low internal resistance which can be considered negligible for some applications. For high power applications, however, the loss associated with this resistance, dissipated as heat in the battery pack, accounts for a significant portion of the power budget and must be considered. The impact of the internal resistance is evident, which plots data for increasing thrust measurements.

* + 1. **30C 11.1v 5000mAh lithium polymer battery**

Lithium Polymer batteries, shows in figure 39, are well suited for our quad-copter due to their high specific energy, large capacity, large discharging current and long working time. These batteries also have a low internal resistance which can be considered negligible for our quad-copter. In addition to this features, it is ultra-thin in thickness, ultra-light in weight, design flexibility, high energy density, leakage-proof, long cycling life and environmental-friendly. Table 39 shows the battery specification.

****

Figure 39: 30C 11.1v 5000mAh lithium polymer battery.

|  |  |
| --- | --- |
| Parameter | Value |
| Voltage | 11.1 V |
| Nominal capacity | 5000 mAh |
| Continuous discharge current | 30 C |
| Dimension | 131\*41\*21 mm |

Table 39: Lithium Polymer Battery Specification.

Our battery has flat discharge rate (30C), this allow high current from the battery with small Ampere per hours. We can calculate the Max current that given from the battery by using eq. 32:

Max current = battery current \*discharge rate (32)

Max current =5A \*30 =150A

Each motor need a current 5-15A. In average, all four motors consume 40A. We can calculate the flight period that supported by the battery, using eq. 32:

The period of flight =current /consume current\*60 minutes (32)

Period =5A/40A\*60=7.5 minutes

After determining the battery type with required specification we made power distribution to other components in quad-copter as shown in Figure 40.

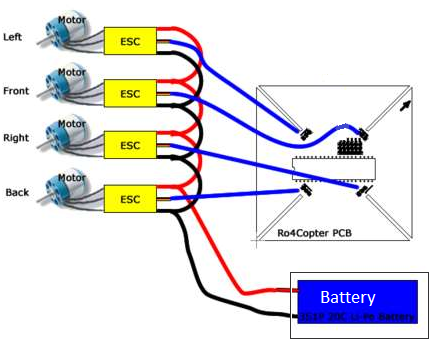
****

Figure 40: Power Distribution.

* 1. **Propeller**

The propellers must be in counter-rotating pairs (a "pusher" and a "puller"). We use 10x4.7 APC, shown in Figure 41, slow-fly propellers. 10 indicates diameter in inches and 4.7 indicates pitch. Larger diameter means more lift but requires more powerful motors. 10 inches is about right for the frame size I am using.

****

Figure 40: 10x4.7 APC Propellers.

With four independent motors, flight control can be established without the use of cyclic pitch and therefore simple fixed pitch propellers can be employed. With motors arranged in counter-rotating pairs, sets of motors that are matched in terms of twist and airfoil are desirable. Fortunately, pusher propellers, normally mounted at the back of a fixed wing aircraft, can be applied since the brushless motors are easily reversed by merely swapping two of the three motor leads. Although selection is limited, large diameter propellers are available in matching pusher and tractor configurations.

Characterization of the propulsion system required measurement of thrust data collected along with the electrical motor parameters. The thrust test stand configuration for this project consisted of a hinged lever with the motor mounted on the free end. A pusher propeller attached to the motor shaft provided a downward directed force that was measured using a digital scale. No special provisions were made to account for the airflow, which limited the maximum thrust that could be measured. At higher throttle settings with an exit velocity of 52 mph, the induced air flow circulated about the room and corrupted the data by impinging on the test stand. Data collected was sufficient to characterize the normal range of operation based on current limits within the motor. Measurements with a larger propeller, without having to rely on an extrapolated model, would require a better ability to control the airflow during the data runs.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Propeller | Volts (V) | Amps (A) | Thrust (g/oz) | Power (W) |
| APC 10 X 4.7 E | 12.31 V | 14.20 A | 29.62 oz | 172.41 W |
| APC 10 X 7 E | 10.83 V | 12.04 A | 24.67 oz | 128.37 W |
| APC 10 X 7 E | 11.06 V | 9.31 A | 16.61 oz | 99.78 W |
| APC 11 X 4.7 E | 10.78 V | 13.86 A | 27.92 oz | 145.41 W |
| APC 11 X 7 E | 12.12 V | 17.91 A | 31.41 oz | 212.55 W |

Table 42: Propeller Characterization.

* 1. **Flight stability sensors**
     1. **Introduction**

In the interest of flight stability, or achieving level controlled flight, a combination of sensors will be required to continually monitor the roll and pitch of the quad-copter such that the microcontroller can process the data and effect real time corrections. The pitch is a measurement of the nose of the quad-copter pointing either upwards, positive pitch, or downwards, negative pitch. The roll is a measurement of the rotation around the longitudinal axis of the quad-copter with the right or starboard side down being a positive roll. The yaw is a measurement of the rotation around the vertical axis. These parameters are illustrated below in Figure 2 for clarity.

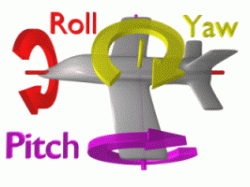
**

Figure 2: Visual representation of roll, pitch, and yaw.

The system, consisting of a combination of sensors that detect pitch and roll for the purpose of maintaining level flight, should satisfy the following goals and objectives:

* The system should be robust enough to collect roll and pitch data of the quad-copter at a frequency that facilitates real time flight correction.
* The system selected should be advanced enough so as not to constrain data utilization methods. Meaning more rather than less capacity.
* The system should utilize a simple data transfer subsystem that demands minimal MCU processing.
* The system should preferably output an analog signal.
* The system should be compact in size.

#### Inertial Measurement Unit (IMU Digital Combo Board)

#### This is a simple breakout for the ADXL345 accelerometer and the ITG-3200 gyro. With this board, shown in Figure 45, you get a full 6 degrees of freedom. The sensors communicate over I2C and one INT output pin from each sensor is broken out. If you need a simple and tiny board that gives you 6 degrees of freedom, this would be a good choice.

#### 

Figure 45: IMU Digital Combo Board.

Research indicates that a combination of accelerometers and gyroscopes is a common approach used by hobbyists to measure and stabilize the flight of RC planes, helicopters and quad-copters. This combination of different sensors working together to establish an accurate orientation measurement relative to the ground can referred to as an inertial measurement unit (IMU). IMU's are available to hobbyists that vary in the degree of complexity according to the number of axes that are measured by various sensors. For example, a 3 degrees of freedom IMU combo board might measure 3 axes with one type of sensor whereas a 6 degrees of freedom IMU would measure each axis with three different sensor types, the third in this example perhaps being a magnetometer. Pertaining to monitoring the tilt of the quad-copter it is necessary that, at a minimum, we measure the rotation about the X and Y horizontal axes, or 2 degrees of freedom. These axes could be measured either with gyroscopes, accelerometers or a combination of both. A combination of accelerometers and gyroscopes measuring the same axis may seem redundant but, the methods of the two sensors differ and thus a more complete picture can be relayed to the microcontroller for analysis. For these reasons a 5 degrees of freedom IMU will be integrated into the design of the quad-copter. This entails a triple axis accelerometer and a triple axis gyroscope measuring the X and Y horizontal axes. With this configuration each of the 2 critical horizontal axis will be monitored by 2 different sensor types.

#### Accelerometer (ADXL345)

#### The ADXL345 is a small, thin, low power, three-axis accelerometer with high resolution (13-bit) measurement up to ±16g. Digital output data is formatted as 16-bit twos complement and is accessible through either a SPI (3- or 4-wire) or I2C digital interface. Figure 45 shows the Pin configuration.

#### The ADXL345 would be sufficient to measure the roll and pitch of the quad-copter. It measures the static acceleration of gravity in tilt-sensing applications, as well as dynamic acceleration resulting from motion or shock. Its high resolution (4mg/LSB) enables resolution of inclination changes of as little as 0.25°.

#### Several special sensing functions are provided. Activity and inactivity sensing detect the presence or lack of motion and if the acceleration on any axis exceeds a user-set level. Tap sensing detects single and double taps. Free-Fall sensing detects if the device is falling. These functions can be mapped to interrupt output pins. An integrated 32 level FIFO can be used to store data to minimize host processor intervention.

#### Low power modes enable intelligent motion-based power management with threshold sensing and active acceleration measurement at extremely low power dissipation.

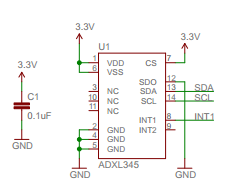


Figure 45: ADXL345 Pin Configuration.

#### Gyroscope (ITG-3200)

#### At first glance it may seem that accelerometers alone might be sufficient to monitor the X and Y axes of the quad-copter however, further research indicates that accelerometers have a tendency to measure more than just gravity. Accelerometers also are influenced by vibration and centripetal acceleration. As the quad-copter is not intended to achieve fast lateral movement or banking turns, like an airplane, the effects of centripetal acceleration on the accelerometer could be minimal relative to other applications. Vibration, on the other hand, could be a serious factor affecting the accelerometers owing to the four engines rotating at differing speeds and connected by a lightweight carbon fiber frame. For these reasons a dual axis gyroscope will be integrated into the quad-copter design to measure rotation about the X and Y axes. Figure 46 shows the orientation of axes of sensitivity and polarity of rotation.

#### The ITG-3200 is a groundbreaking triple-axis, digital output gyroscope. The ITG-3200 features three 16-bit analog-to-digital converters (ADCs) for digitizing the gyro outputs, a user-selectable internal low-pass filter bandwidth, and a Fast-Mode I2C (400kHz) interface. Additional features include an embedded temperature sensor and a 2% accurate internal oscillator. Figure 46 shows the Pin configuration.

#### The ITG-3200 can be powered at anywhere between 2.1 and 3.6V. For power supply flexibility, the ITG-3200 has a separate VLOGIC reference pin (labeled VIO), in addition to its analog supply pin (VDD) which sets the logic levels of its serial interface. The VLOGIC voltage may be anywhere from 1.71V min to VDD max. For general use, VLOGIC can be tied to VCC. The normal operating current of the sensor is just 6.5mA.

#### Communication with the ITG-3200 is achieved over a two-wire (I2C) interface. The sensor also features an interrupt output, and an optional clock input.

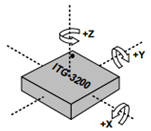


Figure 46: ITG-3200 Orientation of Axes of Sensitivity and Polarity of Rotation.

#### 

Figure 45: ITG-3200 Pin Configuration.

* 1. **RF Link Receiver - 434MHz**

This wireless receiver, shows in figure 54, provides a simple, straight-forward receiver for all of low-cost wireless project and work with our 434MHz transmitters. They can easily fit into a breadboard and work well with microcontrollers to create a very simple wireless data link. Since these are only receivers, they will only work communicating data one-way, you would need two pairs (of different frequencies) to act as a transmitter/receiver pair.

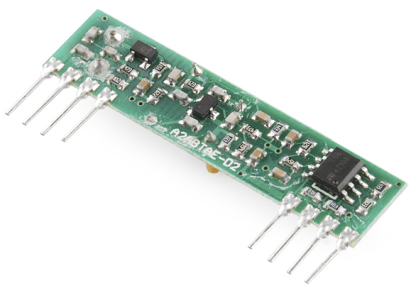


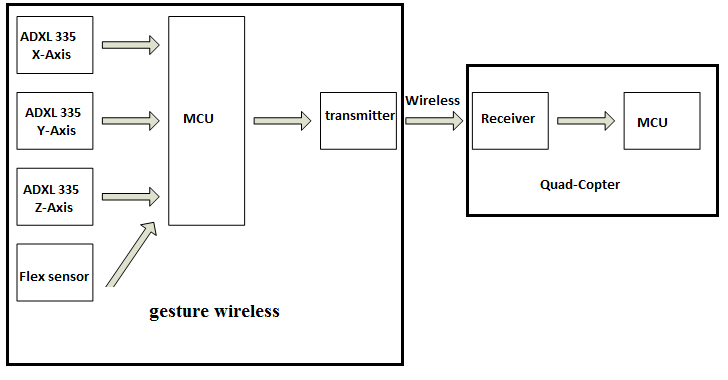
Figure 54: RF Link Receiver - 434MHz

These modules are indiscriminate and will receive a fair amount of noise. Both the transmitter and receiver work at common frequencies and don't have IDs. Therefore, a method of filtering this noise and pairing transmitter and receiver will be necessary.

**Features**:

* 434 MHz
* 150m range (given perfect conditions)
* 4800bps data rate
* 5V supply voltage

1. **Glove Component Modeling**



**7.1 Microcontroller**

The Arduino Pro Mini is a microcontroller board based on the ATmega168. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, an on-board resonator.

The Arduino Pro Mini, shows in figure 50, is intended for semi-permanent installation in objects or exhibitions. The board comes without pre-mounted headers, allowing the use of various types of connectors or direct soldering of wires.

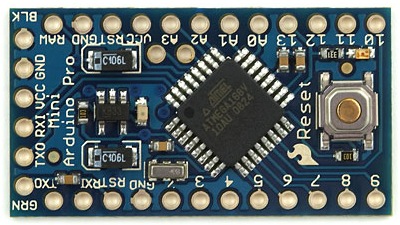


Figure 50: Arduino Pro Mini.

There are two version of the Pro Mini. One runs at 3.3V and 8 MHz, the other at 5V and 16 MHz. Table 51 shows the specification of Arduino Pro Mini.

|  |  |
| --- | --- |
| Parameter | Value |
| Microcontroller | ATmega168 |
| Operating Voltage | 3.3V or 5V (depending on model) |
| Input Voltage | 3.35 -12 V (3.3V model) or 5 - 12 V (5V model) |
| Digital I/O Pins | 14 (of which 6 provide PWM output) |
| Analog Input Pins | 6 |
| DC Current per I/O Pin | 40 mA |
| Flash Memory | 16 KB (of which 2 KB used by bootloader) |
| SRAM | 1 KB |
| EEPROM | 512 bytes |
| Clock Speed | 8 MHz (3.3V model) or 16 MHz (5V model) |

Table 51: Arduino Pro Mini Specification.

**ATmega168**

The high-performance, low-power Atmel 8-bit AVR RISC-based microcontroller combines 16KB ISP flash memory, 1KB SRAM, 512B EEPROM, an 8-channel/10-bit A/D converter (TQFP and QFN/MLF), and debugWIRE for on-chip debugging. The device supports a throughput of 20 MIPS at 20 MHz and operates between 2.7-5.5 volts. Figure 32 shows the Pinout configuration.

By executing powerful instructions in a single clock cycle, the device achieves throughputs approaching 1 MIPS per MHz, balancing power consumption and processing speed.

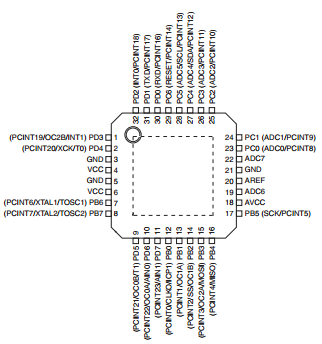


Figure 32: Pinout Configuration of ATmega168

* 1. **Triple Axis Accelerometer Breakout (ADXL335)**

The ADXL335 is a small, thin, low power, complete 3-axis accelerometer with signal conditioned voltage outputs. The product measures acceleration with a minimum full-scale range of ±3 g. Figure 52 shows the Pin Configuration.

It can measure the static acceleration of gravity in tilt-sensing applications, as well as dynamic acceleration resulting from motion, shock, or vibration.

We select the bandwidth of the accelerometer using the CX, CY, and CZ capacitors at the XOUT, YOUT, and ZOUT pins.

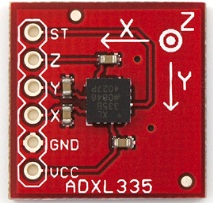
****

Figure 55: ADXL335.

Bandwidths can be selected to suit the application, with a range of 0.5 Hz to 1600 Hz for the X and Y axes, and a range of 0.5 Hz to 550 Hz for the Z axis.

The ADXL335 is available in a small, low profile, 4 mm×4 mm×1.45 mm, 16-lead, plastic lead frame chip scale package (LFCSP\_LQ).

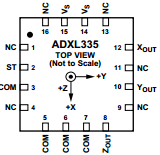
****

Figure 52: ADXL335 Pin Configuration.

* 1. **Flex Sensor**

One side of the sensor is printed with a polymer ink that has conductive particles embedded in it. When the sensor is straight, the particles give the ink a resistance of about 30k Ohms. When the sensor is bent away from the ink, the conductive particles move further apart, increasing this resistance (to about 50k Ohms when the sensor is bent to 90º as in the diagram below). When the sensor straightens out again, the resistance returns to the original value. By measuring the resistance, you can determine how much the sensor is being bent. Figure 52 shows the principle work.

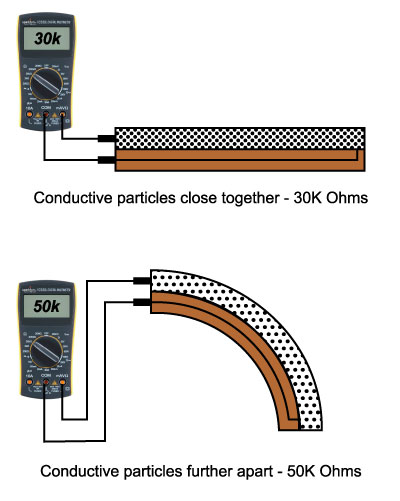


Figure 52: Flex Sensor Principle Work.

Although the active portion of the sensor (the area between the black squares) is quite sturdy, the pin-end of the sensor is susceptible to kinking and eventual failure. We recommend reinforcing or securing this area (for example, clamping or gluing down the sensor at the black square nearest the pins) to ensure that this area doesn't flex along with the rest of the sensor.

The simplest way to incorporate this sensor into our project is by using it in a voltage divider. This circuit requires one resistor. Many values from 10K to 100K will work, we'll use a 10K resistor here. The flex sensor connects to our microcontroller using the following circuit in figure 53.

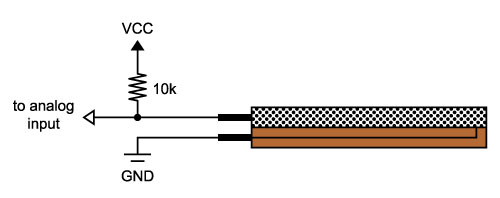


Figure 53: Connection Between The Flex Sensor And Microcontroller.

The resistor and the flex sensor form a voltage divider, which divides VCC by a ratio determined by the two resistances. When the sensor is straight, the 10K resistor and the 30K flex sensor will cause the output voltage to be about 75% of VCC. When the sensor is bent, the voltage will increase to about 83% of VCC. Because of we are using 5V for VCC, we see about 3.75V when the sensor is straight and about 4.17V when the sensor is bent by 90º.

* 1. **RF Link Transmitter - 434MHz**

This wireless transmitter, shows in figure 47, provides a simple, straight-forward transmitter for all of low-cost wireless project and work with the 434MHz receivers. They can easily fit into a breadboard and work well with microcontrollers to create a very simple wireless data link. Since these are only transmitters, they will only work communicating data one-way.

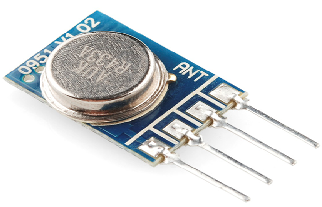


Figure 47: RF Link Transmitter - 434MHz

These modules are indiscriminate and will receive a fair amount of noise. Both the transmitter and receiver work at common frequencies and don't have IDs. Therefore, a method of filtering this noise and pairing transmitter and receiver will be necessary.

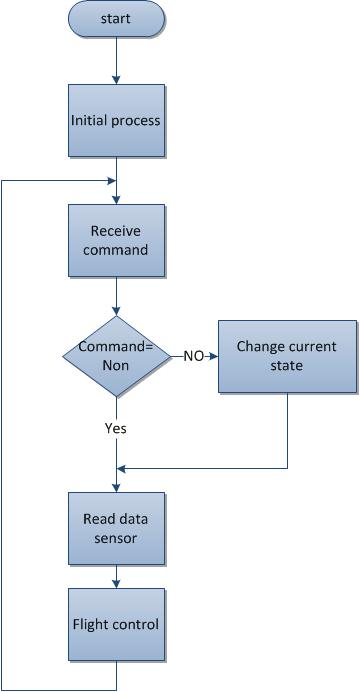
**Features:**

* 434 MHz
* 150m range (given perfect conditions)
* 4800bps data rate
* 5V supply voltage

1. **Software Implementation:**

5.1 Quad-copter processes:

To maintain the stability of the aircraft and control the direction of movement, there are many stages to be implemented and shown in flowchart:

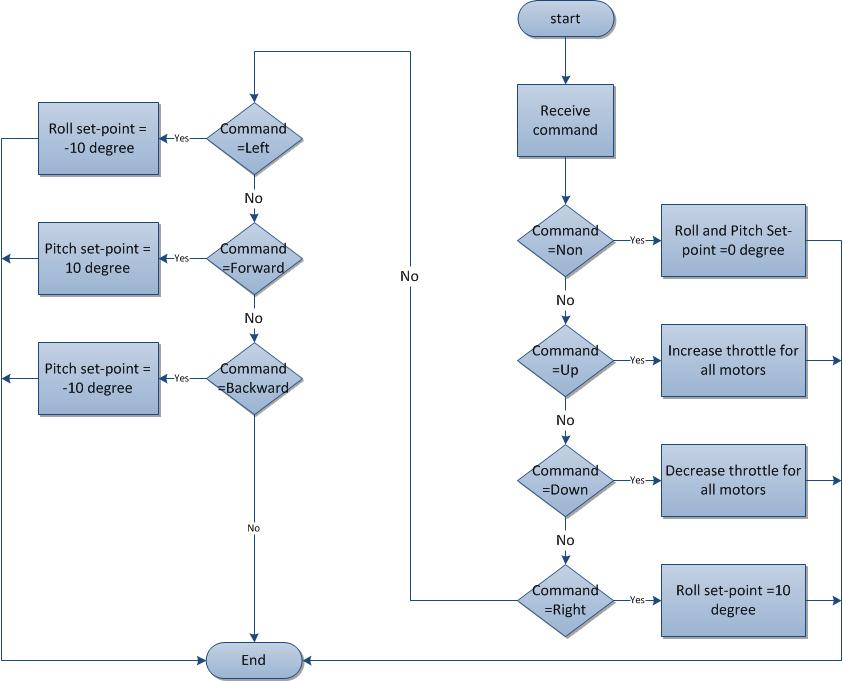


**1-intial process:**

This stage performs at once when the quad-copter starts working to calibrate accelerometer, gyroscope and calibrate ESC for all motors.  
To get the biggest possible accuracy we take 50 samples from accelerometer and gyroscope and then take the average for all of these samples to use in next stage as input. This is very important stage because of all next processes depend on these results.

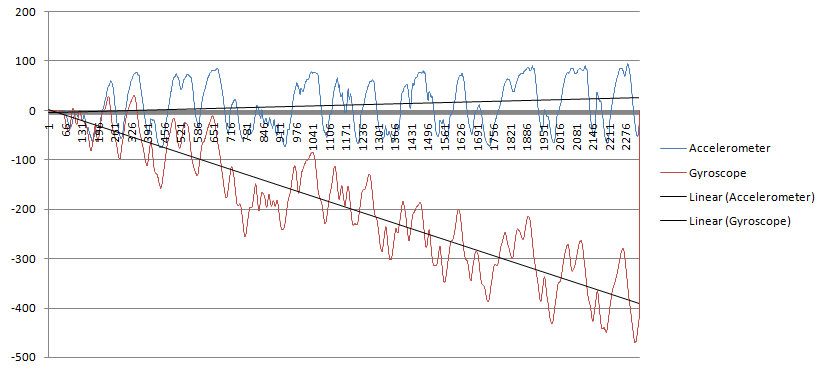
2-**Receive Process**  
in our project we choose RF as medium to send and receive commands from user to quad-copter.  
In this process we receive different commands to control the behavior of quad-copter and the commands are:  
1-up and down   
2-left and right  
3-forward and backward.

The following flowchart illustrates this process.



3- **Read data from sensor**  
IMU contains two sensors, the first being an Accelerometer and the second being a Gyroscope. Read data from IMU, by sending addresses for each acc. and gyro., using fast I2C mode (400 KHz).

Our challenge is the combination both of the sensor values to correct for the Gyroscope's drift over time. Below is an example graph we created from actual data to demonstrate the drift from the gyroscope:

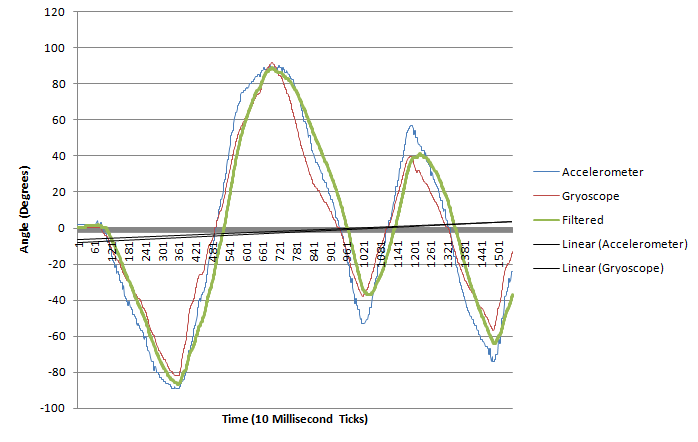


The most commonly used approach we've seen to make combining these sensors rock solid is by using a Kalman filter.

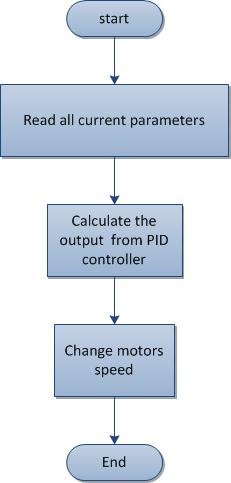
The Kalman filter can be summed up as an optimal recursive computation of the least-squares algorithm. It is a subset of a Bayes Filter where the assumptions of a Gaussian distribution and that the current state is linearly dependent on the previous state are imposed. In other words, the Kalman filter is essentially a set of mathematical equations that implement a predictor-corrector type estimator that is optimal in the sense that it minimizes the estimated error covariance when the condition of a linear Gaussian system is met.

We've successfully filtered out noise from both sensors and derived angles for both in a range between -90 and 90 degrees, with 0 degree being perfectly balanced.

The next figure shows the filtering for data combination from both sensors.



4- Flight control :  
This is a very important stage because it is responsible to keep the quad-copter stable.

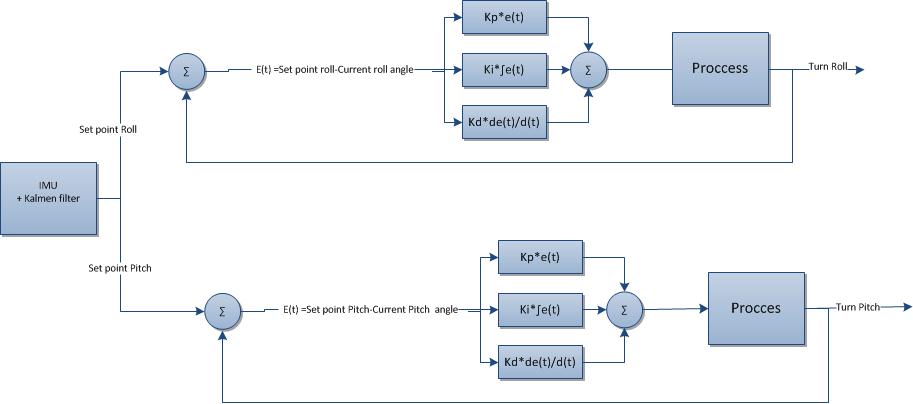


Read current parameters:

Read the set-point for roll direction(X-axis) and set-point for pitch direction(Y-axis) and the current angles for roll and pitch. Also, read the values of Kp,Ki,Kd and then use these data as input to PID controller which calculate the correct output depend on current state.

PID controller :

Use PID conttroller to correct the error dynamicly



The output from PID controller (turn) adds to current throttle to make quad-copter very stable.

The quad rotor will use a Proportional-Integral-Derivative control system, which will be tuned to determine the optimum response and settling time. The PID controller Eq. (4.9) is a closed-loop feedback system which will output a control signal *u* and receive feedback from the inertial sensors. The controller then calculated the difference between the desired position and orientation and the current position and orientation and adjusts *u* accordingly. The equation for a PID controller is as follows:

𝑢=𝑃+𝐼+𝐷 (4.1)

Define:

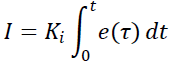
𝑒 𝑡 =(𝑡)−𝑒𝑎(𝑡) (4.2)

where ed is the desired condition, ea is the actual (measured) condition and e(t) is the difference (error) between the two at each individual time step. First, the proportional term Kp is defined as a whole number greater than 0 (for a stable system) which is simply some fraction of the error.

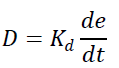
As an example set kp to .5. Our system is traveling from e0 to ed. At time *t* it is halfway between the two, so *e(t)* is .5\*ed. The proportional term *P* is defined as follows:

𝑃=(𝑡) (4.3)

After plugging in .5 for Kp and .5ed for e(t), u becomes .25ed. This is a fourth of the original control input, and thus will lead to a quick exponential convergence to the desired position. This system, while theoretically plausible, doesn‟t lend itself well to the quad rotor for several reasons. First, there is always a lag between the measured state and the corrective action; combined with the response time of the motors and the inertia of the system, the helicopter could become extremely unstable unless all of those factors are accounted for. The proportional and integral terms in the PID controller are what compensate for rate at which the error is changing and the rate at which the system is changing. Feed-forward control, which has been discussed in previous sections, accounts for the behavioral dynamics of the helicopter, such as its momentum and motor response time. The integral term determines the magnitude of the accumulated error by summing the instantaneous error over time. This value is “uncorrected error” that was not dealt with during the previous time step. Notice in this equation *t* (instantaneous time) is replaced by *τ* which is the past time. Adding the integral term increases the overshoot but decreases the settling time.



The derivative term determines the rate at which the error is changing, and by decreasing the rate of change near the set point it reduces overshoot and increases the settling time.



The determination of Kp, KI and Kd is determined by tuning the controller until reaching the desired settling and overshoot is reached

Change motor speed:

We use the output from PID, which it is the correction value, to change the motor speed through changing the duty cycle of PWM to each motor.

The next equations show how to change the motors speed.

motor[font\_right]=throttle +turn\_roll +turn\_pitch (66)

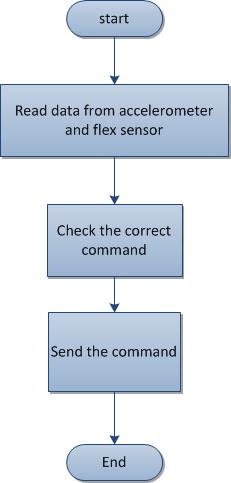
motor[font\_left]=throttle -turn\_roll +turn\_pitch (67)

motor[rear\_right]=throttle +turn\_roll -turn\_pitch (68)

motor[rear\_left]=throttle -turn\_roll -turn\_pitch (69)

* 1. gesture wireless

We use accelerometer and flex sensors to detect the correct hand motion. The bellow flowchart shows the sequence of process.

****

1. Read accelerometer and flex sensor data

The output for each accelerometer and flex sensor is analog. Read these output analog data and treat them with specific technic to get the correct value.

1. Check the correct command

The command depends on the received data from the sensors. We consider the priority for the command on confliction.

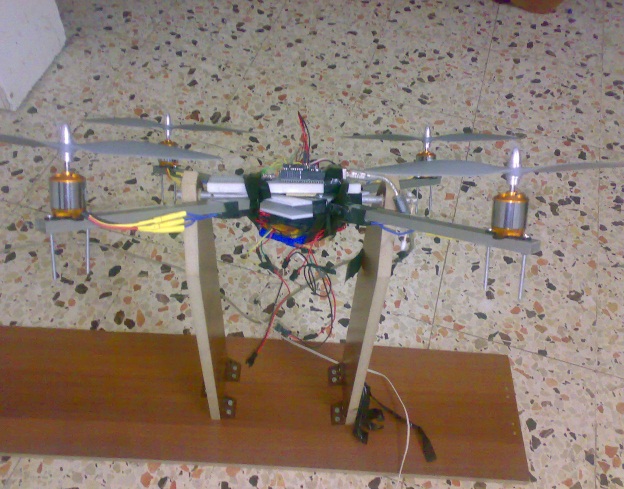
1. Transmit the command

The microcontroller sends the command through the RF transmitter after adding check sum to allow the receiver ensures that the data not change.

# Testing

A quad-copter falling down from a few meters can be a dangerous and costly thing. To avoid danger and minimize costs a serious testing effort has to be made. Therefore a realistic but safe environment is needed. Below you see how we test the quad-copter. Beside testing and debugging with the development tools each control algorithm is analyzed on the quad-copter with an adequate test stand.

On the image you see the pitch and roll angle test stand. One axis is mounted on the frame. The quad-copter can move around one axis. With this construction we are able to test different control algorithm and/or different control parameters.

 jjjjjj 

**Risks:**

The quad-copter as envisioned is a complex project with multiple potential points of failure. The assumption that a known risk is preferable to an unknown risk justifies further analysis of the risks associated with the quad-copter. According to a text on software engineering, risks can be quantified by equating risk exposure with the product of risk probability and risk impact. This approach is sufficient as a basis for assessing risks and ultimately avoiding negative consequences pertaining to the project. The risks involved in such a project can come in many forms ranging from issues of personal safety, a high impact risk, all the way down to losing a letter grade, a lower impact, higher probability risk.

There are issues of personal safety involved with the quad-copter such as the potential combustion of the LiPo (Lithium Polymer) battery if charged incorrectly. This risk can be overcome through researching safety precautions, implementing the precautions, and by buying the most suitable equipment without being overly swayed by price, i.e. by not buying a cheap charger. Another, lower impact, risk associated with the LiPo battery is that if the battery is drained too much then the battery could become un-rechargeable and therefore a time/money risk. This risk can be addressed through power regulation and emergency shutdown procedures but, it could still be considered as having a significant degree of probability and thus, having a significant total modified risk exposure. Other personal safety concerns include working with the substantial current of the LiPo batteries and potential injury from the propellers. Also to be considered is that if the quad-copter were to get out of control and cause injury to the public then liability would be a factor. Another risk that should be mentioned is that some motor/propeller testing was undertaken using a car battery as a power source, which has a potentially fatal level of amperage. There are other risks more uniquely associated with the quad-copter as the choice of a project. Previous senior design groups have had great difficulty in achieving flight stability with a quad-copter. From their mistakes it is learned that early prototyping should be undertaken in order to reduce this risk. If the quad-copter cannot maintain a steady hover in a time frame compatible with the milestone chart then the risk of not having a successful project increases. Another high probability project completion risk is that the wireless communication system, for direct control of the quad-copter, is to be an original, custom designed system undertaken by a group member with substantially more enthusiasm than experience on the subject. Again, previous senior design groups have had difficulties with similar systems. Although this custom wireless system will undoubtedly be rigorously attempted, there remains the risk of failure. This risk could be partially mitigated by maintaining a backup plan of substituting a predesigned system at the last minute.

There are numerous high probability project completion type risks associated with the quad-copter: MCU code development, parsing I2C serial data, power distribution and regulation, Aeronautics, and the list goes on. The quantity multiplied by the probability multiplied by the impact level of these project completion risks would therefore seem to generate a substantial level of risk exposure according to the risk assessment rubric. In conclusion, while high impact risks should not be underrated, lower impact risks can accumulate to threaten a project. The quad-copter is a technically demanding endeavor that will require all members of the group to function on a steep learning curve.

1-software for i2c ,imu , kalmen, filter , pid , pwm frequency ,hand

2-comincation RF (check sum )

3-