GESTURE CONTROLLED DRONE

FINAL YEAR PROJECT UG 2020

By

Maj Wasif Gulzar

Capt Usman Anis Khan (Leader)

Capt Nauman Siddique

Capt Baasim Mustafa

Project Supervisor

Asst Prof. Dr. Mir Yasir Umair

Submitted to the Faculty of Electrical Engineering, Military College of Signals, National University of Sciences and Technology, Rawalpindi in partial fulfillment for the requirements of a B.E. Degree in Telecom Engineering

June 2020

CERTIFICATE OF CORRECTIONS & APPROVAL

Certified that work contained in this thesis titled "**GESTURE CONTROLLED DRONE**", carried out by (1) Maj. Wasif Gulzar (2) Capt. Usman Anis Khan (3) Capt. Baasim Mustafa (4) Capt. Nauman Siddique under the supervision of Asst. Prof. Dr. Mir Yasir Umair for partial fulfillment of Degree of Bachelors of Electrical Engineering, in Military College of Signals, National University of Sciences and Technology, Islamabad during the academic year 2019-2020 is correct and approved. The material that has been used from other sources has been properly acknowledged/ referred.

Approved by

Supervisor

Asst. Prof. Dr. Mir Yasir Umair

____

Date: __________________

DECLARATION

No portion of work presented in this thesis has been submitted in support of another award or qualification in either this institute or anywhere else.

Plagiarism Certificate (Turnitin Report)

This thesis has been checked for Plagiarism. We hereby declare that no content and variety of work bestowed during this thesis has been submitted in support of another award of qualification or degree either during this course or anyplace else. Percentage of plagiarism found in document as per software Turnitin available on LMS NUST comes out to be **11 %**

Signature of Supervisor

Asst. Prof. Dr. Mir Yasir Umair

____

Date: __________________

Acknowledgements

We are thankful to our Creator Allah Subhana-Watala to have guided us throughout this work at every step and for every new thought which You setup in our mind to improve it. Indeed, we could have done nothing without Your priceless help and guidance. Whosoever helped us throughout the course of our thesis, whether our parents or any other individual was Your will, so indeed none be worthy of praise but You.

We are profusely thankful to our beloved parents who raised us when we were not capable of walking and continued to support us throughout in every department of our life.

The accomplishment of this project would not have been possible without the contribution of different people the names of which cannot be reckoned with. Their assistances are sincerely respected and thankfully recognized. We would like to thank following: -

- To Allah Almighty who is the most Powerful and Merciful and everything is in His obedience.
- Asst Prof Dr. Mir Yasir Umair for his support and guidance during the entire project.
- We acknowledge our sincere indebtedness and gratitude to our parents for their love, dream and sacrifice throughout our life. We are thankful for their sacrifice, patience, and understanding that were inevitable to make this work possible. Their sacrifice had inspired us from the day we learned how to read and write until what we have become now. We cannot find the appropriate words that could properly describe our appreciation for their devotion, support and faith in our abilities to achieve our dreams.
- Lastly, we would like to thank any person who contributed in our final year project directly or indirectly. We would like to acknowledge their comments and suggestions, which were crucial for the successful completion of this study.

Dedicated to our exceptional parents and adored siblings whose tremendous support and cooperation led us to this wonderful accomplishment.

Abstract

In today's world, surveillance and monitoring of a place as well as delivery of pay loads for variety of reasons is of utmost importance. A drone will help in surveying a place and to carry a pay load to be dropped on site by monitoring through a display/ VR gear. Hand gesture is used to control the movement of the drone and an onboard camera is used for streaming the aerial view of the place. This paper describes how the drone is being controlled using hand gesture for aerial surveillance and how a payload can be aerially dropped via a drone. The flight controller board has been built using the Arduino Nano. Flex sensors and gyroscope have been used for the implementation of hand gesture; user will be able to control the forward, backward, leftward, rightward movements as well as Yaw and Pitch using the gyroscope sensor. This paper initially describes about how the glove assembly is built and then discusses about its components and its working. The main goal of this project is to provide efficient and convenient situational awareness and payload delivery system, at an affordable cost that can be used for both civil and military purposes.

Table of Contents

LIST OF FIGURES

ABBREVIATIONS

1.	FC	Flight Controller
2.	FCB	Flight Controller Board
3.	PDB	Power Distribution Board
4.	ESC	Electronic Speed Controller
5.	VTX	Video Transmitter
6.	FPV	First Person View
7.	RPV	Remote Person View
8.	RC	Remote Control
9.	BLDC	Brushless DC
10.	MCU	Micro Controller Unit
11.	Li-Po	Lithium Polymer
12.	MEMS	Micro Electro-Mechanical Systems

Chapter 1

1.1 Introduction

1.1.1. Background

Gesture controlled drone is fundamentally a flying multi-rotor that can be maneuvered remotely with proficiencies to transmit real time video to a user. In the last decade, drones have found wide applications ranging from Civilian to Military, from Archeology to Agriculture and from Environmental to Medical emergency management. Crop assessment, Water resources identification, couriering of parcels and Aerial survey of excavations for archaeological assessments are some of the Civilian applications. Drones help in inland and coastal patrolling that guarantees safety and security of community. Drones offer ease and speed in delivering help during natural disasters as they can travel and deliver food and medicines to disaster-stricken area where human resources cannot reach for the first aid help. The gesture-controlled drone can be used for aerial surveillance by the military personnel to patrol dangerous areas to monitor for any potential threat or illegitimate activities including detection of explosive mines that helps our citizens to live peacefully and safely. Drones don't suffer fatigue and can sustain unhindered for days and at a speed that humans cannot do. Thus, drones are becoming an integral part of Civilian, Medical and Military lives as a reliable, safe and efficient method of reconnaissance, surveillance and administration.

1.1.2. Problem statement

Traditional drones require a Remote Controller, Joystick or an application to control movements and functions of drone which keeps both hands and focus of a person busy in just flying a drone. Our goal is to develop an autonomous glove assembly with sensors installed on it for easy control and multitasking of the drone.

1.2. Project description

The proposed work of this project is to develop a drone controlling system that can be used via hand gestures. The system, which is designed and implemented through microcontroller, other features of the drone can be accessed through incorporation of flex sensors, gyroscope and a push button for dropping of pay load. Operation of Quad Copters or drones is subject to understanding difficult applications and gadgets whereas the purpose of hand or gesture-controlled drone is to make its functions easy for common man as well as for operational requirements of military personnel. The proposed system will have following features: -

a. Features of Gesture Controlled Drones

Following features will be incorporated in the Gesture Controlled Drone: -

(1) Left, Right, Up, Down, Forward, Backward Movement

a) Left - Moving the hand towards left

b) Right - Moving the hand towards right

c) Up – Moving flex sensors up

d) Down – Moving flex sensors down

e) Forward - Moving the hand forward

f) Backward - Moving the hand backwards

(2) Dropping of Payload

a) Using the button on the glove assembly to drop the payload being carried

1.3. Prospective Application Areas

The applications of our project can be can categorized into the following categories for ease of understanding and to highlight how versatile the application of the drone technology can be, following are the applications:

b. Military

(1) Remote Surveillance & Aerial Images.

(2) Can be equipped with shot gun and grenades.

(3) Crowd control drones equipped with non-lethal weapons such as tear gas or sound cannons.

(4) Through payload dropping mechanism payload can be dropped at any desired location.

c. Medical

(1) Deliver blood and medical samples to hospital.

(2) Aerial First Aid Responder.

(3) Medicine Delivery.

d. Commercial

(1) Customer parcels & shore to ship drone delivery.

(2) Save drowning lives in sea (dropping life jackets).

(3) Automated data collection for agriculture, livestock, and forestry.

(4) Hand gesture controlled industrial grade robotic arms can be developed.

(5) Most videogames today are played either on game consoles, arcade units or PCs and all these require a combination of input devices. Gesture recognition can be used along with Virtual Reality goggle to truly immerse a player in the game world.

1.4 Scope, Objectives, Specifications and Deliverables

1.4.1. Scope and Objectives

The objective of the project is to develop a system of drone controls on a glove for controlling the drone through hand gestures. The gyroscope and flex sensors detect the change in the gestures and transmit valid hand gestures commands to the drone's flight controller.

Scope includes implementation of our knowledge encompassing Microcontroller based Electronic and Sensor circuits to design and develop "Glove for Gesture Controlling" capable of operating effectively given all types of Environment.

1.4.2. Deliverables

Gesture recognition technology helps us to communicate or control the drone with hand gestures. From this technology, we can control the drone simply by moving our hands and assigning drone functions to the fingers by incorporating flex sensors. Now we can control the drone by doing the gesture where we don't need to have the controller in our hands for a flying drone.

Chapter 2

2.1. Literature review

2.1.1. Overview of existing literature

1. Wings is the first company awarded FAA certification to begin drone deliveries in the US in 15 different states.
2. UPS is now using drones to deliver blood to the hospital , UPS launched a new service using drones to transport blood and other medical samples between the various buildings at WakeMed Raleigh’s medical campus in North Carolina — and the speed at which the drones deliver the samples could literally be the difference between life and death
3. Drone Delivery Canada’s Sparrow Cargo Drone Earns Compliant UAV Status with Transport Canada, during recent research and development flights, the X1000 (The Sparrow) has demonstrated a history of safe operation in some of the harshest environmental conditions in Canada’s north.
4. Airbus Skyways Drone system trials world’s first shore to ship deliveries, airbus has begun shore-to-ship trials in Singapore with its Skyways parcel delivery drone. This marks the first-time drone technology has been deployed in real port conditions to deliver a variety of small, time-critical maritime essentials to working vessels at anchorage.
5. Parrot's new drones could save lives (and crops), a new drone from Parrot has the potential to help. Outfitted with a thermal camera, the Bebop Pro Thermal can fly over devastated areas and feed a live image back to an app, providing up-to-date status information and identifying hot spots -- including the heat given off by a survivor's body.
6. Drones could be 'lifesaving' in an emergency , researchers at William Carey University in Mississippi are studying how disaster drones could carry medical kits to victims in a mass casualty event, before an ambulance arrives. Bystanders could use the kits to help victims, or first responders on the scene could use them when multiple victims are injured.
7. Little Ripper UAV -- flew into action after someone spotted the swimmers in distress in a nearly 10-foot (3-meter) swell more than a half mile (about 1 kilometer) off a patrolled area, Surf Life Saving NSW.
8. Amazon's delivery drones drop packages via parachute; Amazon’s drones

would use magnets, parachutes or spring coils to release the delivery while in mid-flight. Once the package is released, the drone would then monitor the descending box to make sure it's dropping properly onto the desired landing patch.

9. The Russian defense contractor, Almaz Antey has created a drone equipped with a Vepr-12 shotgun with a 10-round magazine. The drone can take off vertically but then flies like an airplane hunting down fellow UAVs.

10. The US Army aims to give almost all its ground combat units tiny drones which can spy on other forces from the sky. The drones, described as "highly capable Nano-unmanned aerial vehicle (UAV) systems," measure only 6.6-inches across and weigh less than 33 grams. The drones have a range of 1.24 miles at speeds of up to 13.35 mph and can fly for up to 25 minutes on a single charge.

Chapter 3

3.1. Detailed Design

The project is divided into two major parts, the hardware and the programming/ coding. The hardware consists of parts of quadcopter, different sensors and the glove assembly. The software part consists of the coding which will be Arduino programming.

The sensors would send the data collected through hand gestures/ movements that they pick up, to the receiver/ Arduino nano.

3.2. Hardware design

The hardware part consists mainly of the sensors. The sensors are mounted on a glove that will be attached with an acrylic box and will be worn on hand. The details about each hardware component are illustrated below.

1. Equipment Required

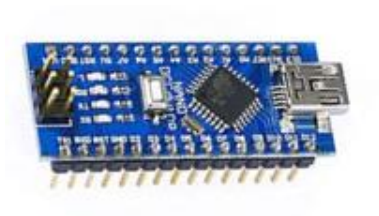
Equipment required for the implementation and completion of the project is as following: -

a. Arduino Nano Microcontroller
b. MPU 6050 Gyroscope
c. Flex Sensors
d. Glove Assembly
e. Lithium Polymer (LiPo) Battery Pack
f. Power Distribution Board
g. Drone / Quad copter
h. Payload Assembly
i. Payload
j. Push Button
k. Acrylic Box

2. Hardware Description

a. Arduino Nano

The Arduino Nano is a small, complete, and breadboard-friendly board based on the ATmega328P. Arduino Nano is half the size of Arduino UNO and uses a mini USB cable instead of a standard one.



b. MPU6050 Gyroscope Sensor

The MPU6050 is a Micro Electro-Mechanical Systems (**MEMS**) which consists of a 3-axis Accelerometer and 3-axis Gyroscope inside it. This helps us to measure acceleration, velocity, orientation, displacement and many other motions related parameter of a system or object.

c. Flex Sensor

Flex sensors are usually available in two sizes. One is **2.2 inch,** and another is **4.5 inch**. Although the sizes are different the basic function remains the same. We are using a 2.2-inch flex sensor in our project

d. **16 Channel Analog Multiplexers**

It can operate on voltages between 2 and 6V DC which allows it to be used with microcontrollers such as Arduino and Raspberry Pi

e. **Glove Assembly**

Glove assembly consists of a glove attached to an acrylic box containing the microcontroller and the sensors.

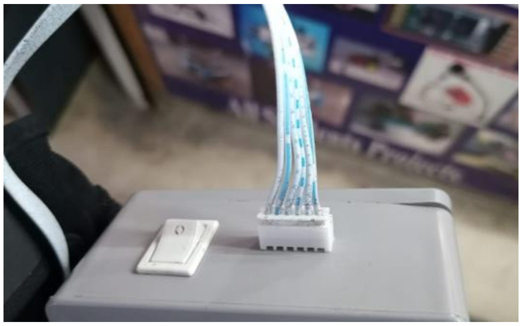
Joy Sticks used for Arming and Dis-Arming of drone

Start - Throttle down and yaw left

Stop - Throttle down and yaw right

Connector used to connect acrylic box with glove assembly.

Button used to shift control from joystick to glove and vice versa.



Charging Pin used for charging the microcontroller battery.

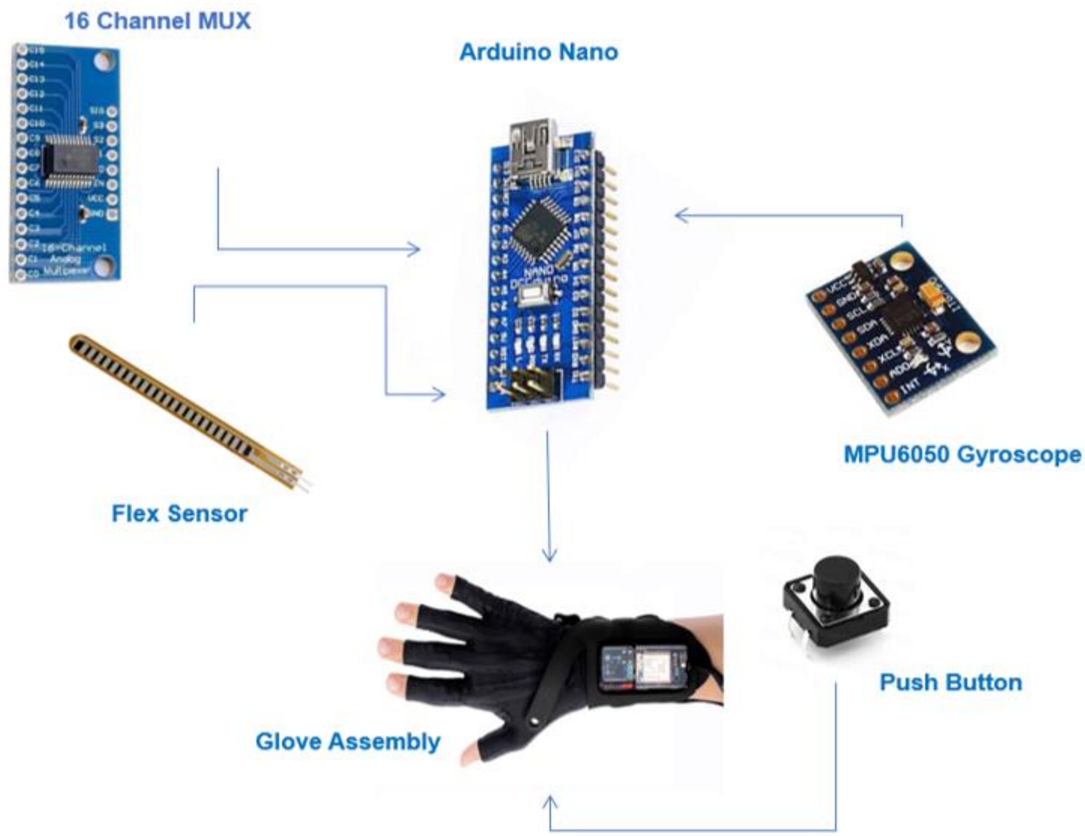
f. Customized Drone

(1) Flight time: 10-12 min

(2) 300 Meters Remote Distance

g. Push Button

Push button is a generic button assembly which is used within the glove assembly for initiating the command to release the payload carried by the drone



GESTURE CONTROLLED DRONE BLOCK DIAGRAM

Chapter 4

4.1 Recommendations and future work

The application of our project can be expanded further. Drones can be used for multiple purposes such as:

a. Pipeline inspection using thermo-graphic systems which can reduce the risk of losses caused by the defects in piping system. The UAV with the thermal camera can fly above the whole length of the pipeline while filming and displaying the monitored footage in real time.

b. This project can also be used for applications other than surveillance.

c. Drone technology will revolutionize the agriculture industry by providing precise analysis with the real time data that is being gathered and processed. Drones can be used throughout the yield cycle for various applications such as soil and field analysis, irrigation, crop spraying

d. Monitoring and health assessment, this results in higher yield productivity and in the case of catastrophe, the farmer will be able to file the losses more proficiently for insurance claims.

4.2 Conclusion

From the above designed project, it can be concluded that we will be able to use the gesture-controlled drone for surveillance/payload delivery/reconnaissance in civil and military areas. This can be equipped in the border for monitoring ambiguous situation thereby ensuring the safety of not only citizens but also soldiers. This project has many advantages. Some of them are enumerated below: -

a. Firstly, drones are faster than human being and can work without breaks. Drones are cheaper and cost effective as it reduces the human resources needed for this task.

b. Secondly, it is much more convenient for the soldiers to check for the potential threats in hostile terrains and use the payload delivery system to engage the target

c. Thirdly, glove controller allows precise and instinctive ways to maneuver the complicated machine. The person can just direct the aerial vehicle with hand movements.

d. This method will be much more efficient and less erroneous because human beings are more prone to committing errors than machines.

4.3 Drawbacks

1. Additional cost of glove assembly which makes the project expensive.
2. Drones are easily affected by weather such as heavy winds & rain.
3. Gesture movements for a longer time can cause fatigue

4.4 Advantages

1. Gesture control technology will ease the pilot as now we can control the drone with the help of gestures, and we don't need to have the controller in our hands for flying a drone
2. Gestures feel very natural to perform since they mirror our experiences in the real world.
3. No Training involved to understand the complex controls of a drone

References

a. Hand Gesture-based Wearable Human-Drone Interface for Intuitive Movement Control, 2016, Autonomous takeoff and landing of a quad copter," in Unmanned Aircraft Systems (ICUAS), International Conference, IEEE, Yong-Won Kang, Yong-Guk Kim & Sang-Yun Shin.
b. Solving the Perspective-2-Point Problem for Flying-Camera Photo Composition, 2018, IEEE/ CVF Conference on Computer Vision and Pattern Recognition, Ziquanlan & David Hsu.
c. Deep Learning Based Hand Gesture Recognition and UAV Flight Controls, 7 September 2018, 24th International Conference on Automation & Computing, Newcastle University, Newcastle upon Tyne, UK, Bin Hu & Jiacun Wang.
d. Hand Gesture Controlled Drones: An Open Source Library, 2018, 1st International Conference on Data Intelligence and Security, Mutlu Mete, Truong-Huy D. Nguyen & Kathiravan.
e. Natural User Interfaces for Human-Drone Multi-Modal Interaction, 2016, International Conference on Unmanned Aircraft Systems (ICUAS) June 7-10, 2016. Arlington, VA USA, Carlos Sampedro & Pascual Campoy.
f. Integrating Multi-Modal Interfaces to Command UAVs, 2014, Feature-rich path planning for robust navigation of MAVs with mono-SLAM. In Int. Conf. on Robotics and Automation (ICRA), Seyed Abbas, Sadat Shokoofeh & Pourmehr.
g. Ergonomic Quadcopter Control Using the Leap Motion Controller, 2018, Pazmany Peter Catholic University, Faculty of Information Technology and Bionics, Budapest, Hungary, Tamas Zsedrovits Gergely Gubcsi.
h. ROS-based UAV Control Using Hand Gesture Recognition, 2017, 29th Chinese Control and Decision Conference (CCDC), Yongwei Zhang & Zhiwei Zhong.
i. Drones and the Future of Domestic Aviation, 3 March 2014, University of California Los Angeles (UCLA), Los Angeles, CA, USA Brookings Institution, Washington, DC, USA, John Villasenor.

Appendix A

Timeline

	May 19	Jun 19	Jul 19	Aug 19	Sep 19	Oct 19	Nov 19	Dec 19	Jan 20	Feb 20	Mar 20	Apr 20
Proposal Defense	Completed											
Literature Research			Completed									
Acquisition of Resources						Completed						
Hardware Design & Initial Testing								Completed				
Software Integration									Completed			
Testing & Debugging										Completed		

Appendix B

Cost Breakdown

Component	Quantity	Unit Price	Amount	Remarks
Customized Drone	1	20,000	20,000	
RF 433 Transmitter/Receiver Module	1	700	700	
MPU 6050 6-Axis Accelerometer/ Gyroscope Sensor	1	900	900	
Arduino Nano	1	1500	1500	
Pay Load Assembly	1	5000	5000	
Flex Sensors	2	1500	3000	
Power Distribution Board	1	500	500	
Glove Assembly	1	1500	1500	
16 Channel Analog Multiplexer	1	500	500	
Miscellaneous	-	5000	5000	
Total			38600	

Appendix C

TRANSMITTER SIDE ARDUINO CODE

```
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>
#include <Wire.h>
#include <I2Cdev.h>
#include <MPU6050.h>

MPU6050 mpu;

int16_t ax, ay, az;
int16_t gx, gy, gz;

int timer=0, total=0;

long val1, val2, val3;


int up = 6;
int down =7;
int mode = 4;
int stop=0;

const uint64_t pipeOut = 0xE8E8F0F0E1LL;

RF24 radio(9, 10); // select  CSN  pin

// The size of this struct should not exceed 32 bytes
// This gives us up to 32 8 bits channels
struct MyData {
  byte throttle;
  byte yaw;
  byte pitch;
  byte roll;
  byte AUX1;
  byte AUX2;
};
```

```
MyData data;

void resetData()
{
  //This are the start values of each channel
  // Throttle is 0 in order to stop the motors
  //127 is the middle value of the 10ADC.

  data.throttle = 0;
  data.yaw = 127;
  data.pitch = 127;
  data.roll = 127;
  data.AUX1 = 0;
  data.AUX2 = 0;
}

void setup(){ //Start everything up
   Serial.begin(9600);
   pinMode(mode ,INPUT);
   pinMode(up ,INPUT);
   pinMode(down ,INPUT);
   Serial.print("OK1");
   Wire.begin();
   mpu.initialize();
   Serial.print("OK2");
   radio.begin();
   radio.setAutoAck(false);
   radio.setDataRate(RF24_250KBPS);
   radio.openWritingPipe(pipeOut);
   resetData();
   Serial.print("OK3");
}

/**************************************************/

// Returns a corrected value for a joystick position that takes into account
// the values of the outer extents and the middle of the joystick range.
int mapJoystickValues(int val, int lower, int middle, int upper, bool reverse)
{
```

```
  val = constrain(val, lower, upper);
  if ( val < middle )
    val = map(val, lower, middle, 0, 128);
  else
    val = map(val, middle, upper, 128, 255);
  return ( reverse ? 255 - val : val );
}

void loop(){


  // The calibration numbers used here should be measured
  // for your joysticks till they send the correct values.

if (digitalRead (2) == LOW){ stop=1;}
if (digitalRead (3) == LOW){ stop=0;}

  if (digitalRead (mode) == LOW){
  mpu.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);
  val1 = map(ax, -17000, 17000, 0, 1023);
  val2 = map(ay, -17000, 17000, 0, 1023);
  val3 = map(az, -17000, 17000, 0, 100);

if(digitalRead (up) == 1){
  if(timer==0){ timer = 10;
   if(total<1023){total = total+1;}
    }
  }

if(digitalRead (down) == 1){
  if(timer==0){ timer = 10;
   if(total>0){total = total-1;}
    }
  }

  data.throttle = mapJoystickValues( total, 0, 127, 255, true );  //Change this values and calibrate the
sent data
  data.yaw      = mapJoystickValues( analogRead(A1), 0, 127, 255, true );   // true or false in order to
invert the joystick direction
```

```
  data.pitch    = mapJoystickValues( val1, 0, 127, 255, true );     //0 low value, 127 middle value, 255
top value
  data.roll     = mapJoystickValues( val2, 0, 127, 255, true );
  data.AUX1     = stop;    //These are digital values so 0 and 1 maped in the receiver code to 0 to 255
  data.AUX2     = digitalRead (3);
  }else{
  data.throttle = mapJoystickValues( analogRead(A0), 0, 127, 255, true );  //Change this value and
calibrate the sent data
  data.yaw      = mapJoystickValues( analogRead(A1), 0, 127, 255, true );   // true or false in order to
invert the joystick direction
  data.pitch    = mapJoystickValues( analogRead(A2), 0, 127, 255, true );     //0 low value, 127 middle
value, 255 top value
  data.roll     = mapJoystickValues( analogRead(A3), 0, 127, 255, true );
  data.AUX1     = stop;    //These are digital values so 0 and 1 maped in the receiver code to 0 to 255
  data.AUX2     = digitalRead (3);
  }


  radio.write(&data, sizeof(MyData));

 Serial.print("T = ");
 Serial.println(data.throttle);
 Serial.print("Y = ");
 Serial.println(data.yaw);
 Serial.print("P = ");
 Serial.println(data.pitch);
 Serial.print("R = ");
 Serial.println(data.roll);


if(timer>0){timer=timer-1;}
}
```

RECEIVER SIDE ARDUINO CODE

```
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>
#include <Servo.h>      //To create PWM signals we need this library

const uint64_t pipeIn = 0xE8E8F0F0E1LL;     //Remember that this code is the same as in the
                                                                          transmitter
RF24 radio(9, 10);  //CSN and CE pins

// The sizeof this struct should not exceed 32 bytes
struct Received_data {
  byte ch1;
  byte ch2;
  byte ch3;
  byte ch4;
  byte ch5;
  byte ch6;
  byte ch7;
};

Received data received data;

Servo channel_1;
Servo channel_2;
Servo channel_3;
Servo channel_4;
Servo channel_5;
Servo channel_6;
Servo channel_7;

int ch1_value = 0;
int ch2_value = 0;
int ch3_value = 0;
int ch4_value = 0;
int ch5_value = 0;
int ch6_value = 0;
```

```
void reset_the_Data()
{
  // 'safe' values to use when NO radio input is detected
  received_data.ch1 = 0;      //Throttle (channel 1) to 0
  received_data.ch2 = 127;
  received_data.ch3 = 127;
  received_data.ch4 = 127;
  received_data.ch5 = 0;
  received_data.ch6 = 0;

}



/***********************************************/

void setup()
{
  //Attach the servo signal on pins from D2 to D8
  channel_1.attach(2);
  channel_2.attach(3);
  channel_3.attach(4);
  channel_4.attach(5);
  channel_5.attach(6);
  channel_6.attach(7);


  //We reset the received values
  reset_the_Data();

  //Once again, begin and radio configuration
  radio.begin();
  radio.setAutoAck(false);
  radio.setDataRate(RF24_250KBPS);
  radio.openReadingPipe(1,pipeIn);

  //We start the radio communication
  radio.startListening();
```

```
}

/**************************************************/

unsigned long lastRecvTime = 0;

//We create the function that will read the data each certain time
void receive_the_data()
{
  while ( radio.available() ) {
  radio.read(&received_data, sizeof(Received_data));
  lastRecvTime = millis(); //Here we receive the data
}
}

/**************************************************/

void loop()
{
  //Receive the radio data
  receive_the_data();

//////////This small if will reset the data if signal is lost for 1 sec.
/////////////////////////////////////////////////////////////////////////
  unsigned long now = millis();
  if ( now - lastRecvTime > 1000 ) {
    // signal lost?
    reset_the_Data();
    //Go up and change the initial values if you want depending on
    //your applications Put 0 for throttle in case of drones so it won't
    //fly away
  }


  ch1_value = map(received_data.ch1,0,255,1000,2000);
  ch2_value = map(received_data.ch2,0,255,1000,2000);
  ch3_value = map(received_data.ch3,0,255,1000,2000);
  ch4_value = map(received_data.ch4,0,255,1000,2000);
```

```
  ch5_value = map(received_data.ch5,0,1,1000,2000);
  ch6_value = map(received_data.ch6,0,1,1000,2000);


  //Create the PWM signals
  channel_1.writeMicroseconds(ch1_value);
  channel_2.writeMicroseconds(ch2_value);
  channel_3.writeMicroseconds(ch3_value);
  channel_4.writeMicroseconds(ch4_value);
  channel_5.writeMicroseconds(ch5_value);
  channel_6.writeMicroseconds(ch6_value);



}//Loop end
```