ORNITHOPTER

Mr. A. Raghu Ramachandra[1], M. Rishikesh[2], S. Sai Teja[3], V. Vishal Kumar[4].

[1]*Assistant Professor, Dept. of Electrical and Electronics Engineering, GCET College, Telangana, INDIA,*

[2,3,4]*Graduating Student, Dept. of Electrical and Electronics Engineering, GCET College, Telangana, INDIA.*

ABSTRACT: *In recent years the subject of flying vehicles propelled by flapping wings, also known as ORNITHOPTERS, has been an area of interest because of its application to micro aerial vehicles (MAVs). These miniature vehicles seek to mimic small birds and insects to achieve never before seen agility in light. This renewed interest has raised a host of new problems in vehicle dynamics and control to explore. In order to better study the control of flapping wing light we have developed a large scale ornithopter. It is capable of carrying a heavy (400 gram) computer and sensor package and is designed especially for the application of controls research. The design takes special care to optimize payload capacity, crash survivability, and field repair abilities. This thesis covers the design process of both the mechanical and electrical systems of the ornithopter and initial control experiments.*

Keywords: *ORNITHOPTER, MAV, Payload Capacity.*

1. INTRODUCTION

An Ornithopter (from Greek *ornithos* “bird” & *pteron* “wing”) is an aircraft that flies by flapping its wings. Designers seek to imitate the flapping-wing flight of birds, bats. Though machines may differ in form, they are usually built of the same scale as these flying creatures. An ornithopter is an aircraft that flies by flapping its wings. Inspired by nature, we intend to make a remote controlled ornithopter as our project. This is the beginning of a new era in flapping-wing flight. The wings had a set range of motion and we could only control the speed. This system mimics the real bird in the nature. In this project we used arduino controlled motor and gears to drive the flapping mechanism of the wings.

2. COMPONENTS USED

2.1. On Board

1. <u>Arduino UNO / NANO</u> : Arduino uno is an open source microcontroller board based on microchip ATmega328P microcontroller and developed by Arduino.cc. It is used to encode and decode data.

2. <u>Servo Motors</u> : Servo motors are DC motors that allow for precise control of the angular position. They are DC motors whose speed is slowly lowered by the gears. The servo motors usually have a revolution cut off from 90° to 180°. A few servo motors also have a revolution cutoff of 360° or more. It is used to control the motion of the tail.

3. <u>BLDC Motor</u> : A brushless DC electric motor (BLDC motor or BL motor), also known as electronically commutated motor (ECM or EC motor) and synchronous DC motors, are synchronous motors powered by direct current (DC) electricity via an inverter or switching power supply which produces electricity in the form of alternating current (AC) to drive each phase of the motor via a closed loop controller. The controller provides pulses of current to the motor windings that control the speed and torque of the motor. It is used to create flapping mechanism.

4. <u>Electronic speed control</u> : An electronic speed control or ESC is an electronic circuit that controls and regulates the speed of an electric motor. It may also provide reversing of the motor and dynamic braking. Miniature electronic speed controls are used in electrically powered radio controlled models. Full-size electric vehicles also have systems to control the speed of their drive motors. It is used to control the speed of BLDC motor on the command of slider potentiometer.

5. <u>Lithium polymer battery</u> : A lithium polymer battery, or more correctly lithium-ion polymer battery (abbreviated as LiPo, LIP, Li poly, lithium-poly and others), is a rechargeable battery of lithium-ion technology using a polymer electrolyte instead of a liquid electrolyte. High conductivity semisolid (gel) polymers form this electrolyte. These batteries provide higher specific energy than other lithium battery types and are used in applications where weight is a critical feature, like mobile devices and radio-controlled aircraft.

6. nRF24L01 : nRF24L01 is a single chip radio transceiver for the world wide 2.4 - 2.5 GHz ISM band. The transceiver consists of a fully integrated frequency synthesizer, a power amplifier, a crystal oscillator, a demodulator, modulator and Enhanced Shock Burst protocol engine.

2.2. On Remote

1. Slider Potentiometer : A potentiometer is a simple slider that provides a variable resistance logarithmically, which we can read into the Arduino board as an analog value. We connect three wires to the Arduino board. The first goes to ground from one of the outer pins of the potentiometer. The second goes from 5 volts to the other outer pin of the potentiometer. The third goes from analog input 2 to the middle pin of the potentiometer. It is used to vary the speed of the BLDC motor.

2. Joystick Module : The X and Y axes are two ~10k potentiometers which control 2D movement by generating analog signals. The joystick also has a push button that could be used for special applications. When the module is in working mode, it will output two analog values, representing two directions. Compared to a normal joystick, its output values are restricted to a smaller range (i.e. 200~800), only when being pressed that the X value will be set to 1023 and the MCU can detect the action of pressing.

3. nRF24L01 : nRF24L01 is a single chip radio transceiver for the world wide 2.4 - 2.5 GHz ISM band. The transceiver consists of a fully integrated frequency synthesizer, a power amplifier, a crystal oscillator, a demodulator, modulator and Enhanced Shock Burst protocol engine.

4. Arduino UNO / NANO : Arduino uno is an open source microcontroller board based on microchip ATmega328P microcontroller and developed by Arduino.cc. It is used to encode and decode data.

3. PARTS AND MATERIALS USED FOR BODY

3.1 Wingspan, Weight and Flapping Frequency

The flapping rate of bird wings is determined by wing area. For example, for a stork It is enough to flap wings with a frequency of 2 strokes per second, a sparrow has to make 13 strokes per second, and a hummingbird - up to 80 strokes per second. I wanted to make a large ornithopter. Therefore the wing area should be large too. To calculate the area of the wing you should know the wingspan. So, wingspan became the first parameter to chose. I decided to make an ornithopter with a wingspan in the range of 1200-1400 mm.

Wingspan » 1200 - 1400 mm;
Flapping rate » 5 - 7 Hz;
Flight weight » 400 g.

3.2 Preparation. Motor, ESC, and Battery Motor:

The motor should be small in size. Big size motors weight a lot and weight can be very critical for the design. At the same time, the electric motor should be sturdy to provide enough torque to overcome air resistance. To increase the torque and reach the necessary flapping frequency, I'm going to use a gearbox. In this case, I can take a weaker motor with a higher revolution per minute (rpm) value.

Main motor characteristics:
Output shaft diameter: 2.3 mm;
Max current: 22A / 20S;
Voltage: 2 - 3S;
Dimension: 26mm x 27mm, 41mm;
Weight: » 39g;

Power:

The battery is the most massive component by weight, so it's critical to choose the right one. To power the motor I use a Li-Po battery. The capacity-to-mass coefficient of such cells is really high. Also, they are able to output a high current value which is so required for brushless motors.

Main battery characteristics:
2 Cells, 7.4V;
Capacity: 900mAh;
Discharge Rate: 30C;
Weight: » 56g;

Electronic Speed Control (ESC)

Main characteristics:
Lipo: 2-3 cells;
Continuous current: 20A;
Peak current: 25A;
BEC: Yes/No;
Weight: » 19g;

3.3 Materials used for mechanical design

Body frame: Glass Proxy
Skeleton of wings: Carbon fiber Rods
Gears: Powdered Nylon (3D Printed)
Wings: Nylon Fabric

4. BLOCK DIAGRAM

4.1. On Board

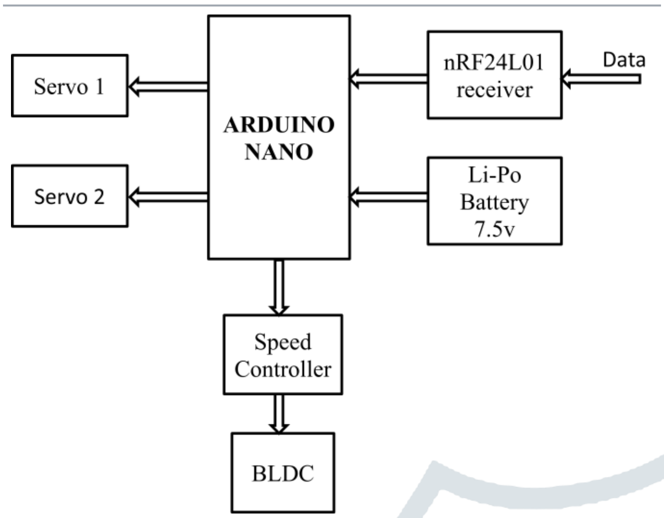

Fig1. Block diagram of On Board Circuit.

4.2. On Remote Control

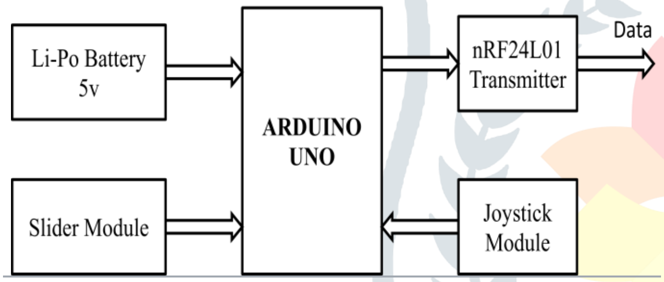

Fig2. Block diagram of On Remote Control Circuit.

5. CIRCUIT DIAGRAM

5.1. On Board

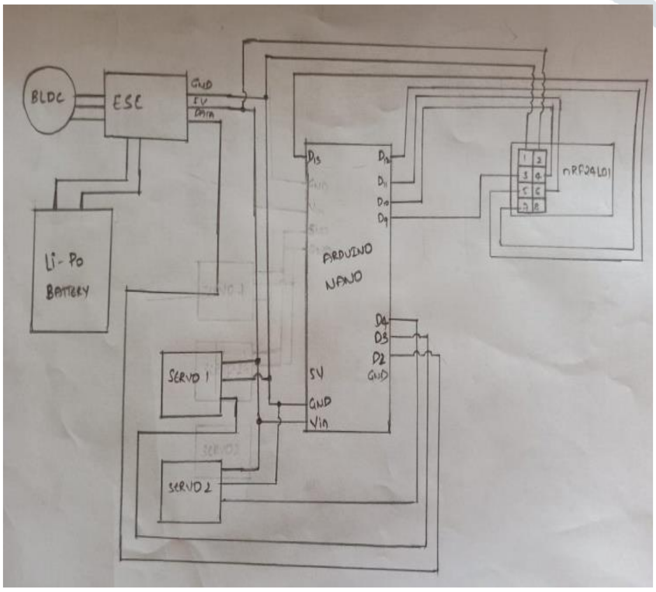

Fig.3 Circuit diagram of On Board Connections

5.2. On Remote Control

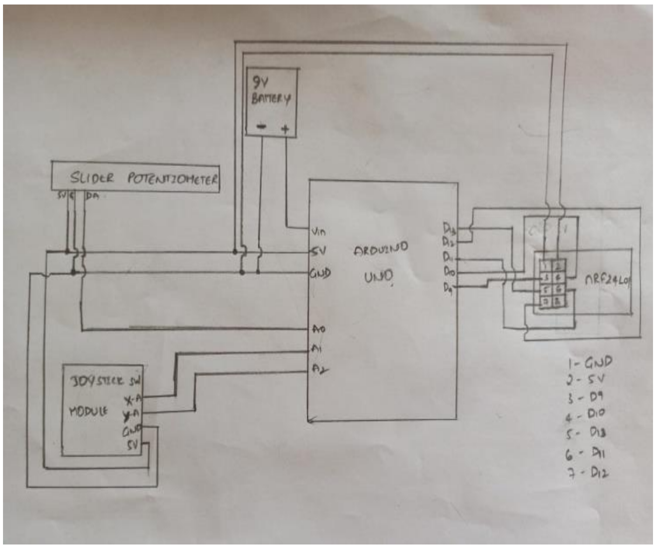

Fig.4 Circuit diagram of On Remote Control Connections

6. WORKING

In this project we operate ornithopter using remote control. The slider module and joystick module gives input to the arduino uno in RC. Arduino Uno decodes and sends the data to the flight using RF transmitter. The transmitted data is received by RF receiver and the data is encoded by Arduino Nano in the flight and then the BLDC motor and Servos are operated accordingly for the movement of bird.

7. ARDUINO CODE

Generally, Arduino microcontroller needs a code for its working which is mentioned below..,

7.1. Transmitter Code:

```
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>

const uint64_t pipeOut = 0xE9E8F0F0E1LL;
//IMPORTANT: The same as in the receiver
0xE9E8F0F0E1LL
RF24 radio(9, 10); // select CE,CSN pin
struct Signal {
byte throttle;
byte x_pos;
byte y_pos;
};
Signal data;
void ResetData()
{
data.throttle = 0;   // slider start position
data.x_pos = 127;    // Center of joystick x_axis
```

```
data.y_pos = 127;     // Center of joystick y_ axis
}
void setup()
{
//Start everything up
radio.begin();
radio.openWritingPipe(pipeOut);
radio.stopListening(); //start the radio
comunication for Transmitter
}
// Joystick center and its borders
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 (val);
}
void loop()
{
// Control Stick Calibration
// Setting may be required for the correct values of
the control levers.
data.throttle = mapJoystickValues(
analogRead(A0), 0, 512, 1023, true );   // "true" or
"false" for signal direction
data.x_pos    = mapJoystickValues(
analogRead(A1), 0, 512, 1023, true );      // "true"
or "false" for servo direction
data.y_pos    = mapJoystickValues(
analogRead(A2), 0, 512, 1023, true );     // "true"
or "false" for servo direction
radio.write(&data, sizeof(Signal));
}
```

7.2. Receiver Code :

```
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>
#include <Servo.h>
int ch_width_1 = 0;
int ch_width_2 = 0;
int ch_width_3 = 0;
Servo ch1;
Servo ch2;
Servo ch3;
struct Signal {
byte throttle;
byte x_pos;
byte y_pos;
};
Signal data;
const uint64_t pipeIn = 0xE9E8F0F0E1LL;
RF24 radio(9, 10);
void ResetData()
{
// Define the inicial value of each data input.
// The middle position for Potenciometers.
(254/2=127)
data.x_pos = 127;   // joystick centre
data.y_pos = 127;  // joystick centre
data.throttle = 0; // Motor Stop
}
void setup()
{
  //Set the pins for each PWM signal
  ch1.attach(2);
  ch2.attach(3);
  ch3.attach(4);
Serial.begin(9600);
  //Configure the NRF24 module
  ResetData();
  radio.begin();
  radio.openReadingPipe(1,pipeIn);
 radio.startListening(); //start the radio
comunication for receiver
}
unsigned long lastRecvTime = 0;
void recvData()
{
while ( radio.available() ) {
radio.read(&data, sizeof(Signal));
lastRecvTime = millis();   // receive the data
}
}
void loop()
{
recvData();
Serial.print(data.throttle);
Serial.print("      ");
Serial.print(data.x_pos);
Serial.print("      ");
Serial.println(data.y_pos);
ch_width_1 = map(data.throttle, 0, 255, 1000,
2000);     // pin D2 (PWM signal)
ch_width_2 = map(data.x_pos,    0, 255, 1000,
2000);     // pin D3 (PWM signal)

ch_width_3 = map(data.y_pos,    0, 255, 1000,
2000);     // pin D4 (PWM signal)
/*Serial.print(ch_width_1);
Serial.print("      ");
Serial.print(ch_width_2);
Serial.print("      ");
Serial.println(ch_width_3);*/
// Write the PWM signal
ch1.writeMicroseconds(ch_width_1);
ch2.writeMicroseconds(ch_width_2);
ch3.writeMicroseconds(ch_width_3);
```

}

8. PICTURE OF THE PROJECT

Fig.5 Picture of Ornithopter Project

9. ADVANTAGES

1. Used for army purposes.
2. Light in weight.
3. We can control the ornithopter through remote.
4. Camera is fitted to view the targeted area.
5. Compact in size.

10. APPLICATIONS

1. Ornithopters can be made to resemble Birds or insects, they could be used for military applications without alerting the enemies that they are under surveillance.
2. Several ornithopters have been flown with video cameras on board, some of which can hover and maneuver in small spaces.

11. FUTURE SCOPE

The further extension of the project can be done in the way of aerial vehicles to spy or to monitor the areas where there are chances of occurrence of any hazard by attaching a camera and displaying the visuals in smart phone or screen. As Ornithopter is something that appears to be a part of nature there are less chances of identifying that they are being spied.
Hence these can be used in army for many purposes.
Taking this as the prototype we can develop a aerial vehicles of huge size that can carry humans too.

12. CONCLUSION

- In this project, we made a prototype of an unmanned ornithopter, which looks appear as a bird in our locality.
- This will be helpful in keeping an eye on other countries or terrorist organisation which will improve our defence system.
- Operation of bird looks similar as a bird is flying which is common in nature and none can think of being spied.
- In these circumstances we as an engineer can help the country by making this kind of robots which are much helpful in supporting our defence.

REFERENCES

1. https://www.instructables.com/id/Opensource-Ornithopter-Prototype-Arduino-Powered-a/
2. https://www.youtube.com/watch?time_continue=1&v=ruWvzF1fGHw&feature=emb_logo
3. https://www.arduino.cc/
4. https://en.wikipedia.org/wiki/Brushless_DC_electric_motor
5. https://www.electronicshub.org/arduino-rf-transmitter-receiver-module/
6. https://people.eecs.berkeley.edu/~ronf/Ornithopter/index.html
7. Wei Shyy, Yongsheng Lian, Jian Teng, Dragos Viieru, and Hao Liu. Aerodynamics of low reynolds number flyers. 2008.