

**TRƯỜNG ĐẠI HỌC QUỐC TẾ HỒNG BÀNG
KHOA KỸ THUẬT VÀ CÔNG NGHỆ
BỘ MÔN CÔNG NGHỆ THÔNG TIN**



ĐỒ ÁN MÔN HỌC LẬP TRÌNH NHÚNG

**THÁP SÚNG NERF ĐIỀU KHIỂN
BẰNG BLUETOOTH**

Giáo viên hướng dẫn: *ThS.Lê Văn Hạnh*

Sinh viên thực hiện: Nguyễn Trần Phước Lộc - 201107002

TP. Hồ Chí Minh, 2023

LỜI CẢM ƠN

Để hoàn thành đề tài khóa luận đồ án này, chúng em đã nhận được sự hướng dẫn, giúp đỡ và góp ý nhiệt tình từ quý thầy cô trường Đại Học Quốc Tế Hồng Bàng và thầy giáo viên **ThS.Lê Văn Hạnh**.

Chúng em xin gửi lời biết ơn sâu sắc đến thầy giáo viên **ThS.Lê Văn Hạnh** đã dành nhiều thời gian và tâm huyết hướng dẫn nghiên cứu và giúp chúng em hoàn thành khóa luận đồ án môn học.

Chúng em cũng xin chân thành cảm ơn đến quý thầy cô trường Đại học Quốc Tế Hồng Bàng, đặc biệt là những thầy cô đã tận tình dạy bảo chúng em suốt thời gian học tập tại trường.

Chúng em muốn bày tỏ lòng biết ơn đến Ban Giám Hiệu trường Đại Học Quốc Tế Hồng Bàng cùng quý thầy cô trong Khoa Kỹ thuật và Công nghệ đã tạo điều kiện thuận lợi cho chúng em học tập và hoàn thành tốt khóa học.

Mặc dù chúng em đã cố gắng hoàn thiện khóa luận đồ án bằng tất cả năng lực của mình, nhưng không thể tránh khỏi những thiếu sót. Chúng em rất mong nhận được những đóng góp quý báu từ quý thầy cô và các bạn để chúng em có thể hoàn thiện đề tài một cách tốt nhất.

Chân thành cảm ơn!

TP.HCM, Tháng 6 Năm 2023

Sinh viên thực hiện

Nguyễn Trần Phước Lộc

MỤC LỤC

LỜI CẢM ƠN	i
MỤC LỤC	ii
PHIẾU CHẤM ĐIỂM ĐỒ ÁN MÔN HỌC	iii
MỞ ĐẦU	iv
CHƯƠNG 1: NỘI DUNG CHÍNH ĐỀ TÀI.....	1
1.1 Giới thiệu đề tài	1
1.1.1 Giới thiệu đề tài cần xây dựng.....	1
1.1.2 Mục tiêu đề tài	1
1.2 Phân cứng và linh kiện	1
1.2.1 Giới thiệu về linh kiện sử dụng	1
1.2.2 Sơ đồ (Schematic) & Mạch (Circuit) được dùng trong đề tài	5
1.2.3 Thiết bị phân cứng: cấu tạo, bản vẽ phân cứng	6
1.3 Xây dựng phần mềm	10
1.3.1 Giải thuật	10
1.3.2 Chương trình điều khiển	11
1.3.3 Chương trình (mã lệnh - Sketch)	13
1.4 Tự đánh giá kết quả thực hiện	17
CHƯƠNG 2: KẾT LUẬN	18
TÀI LIỆU THAM KHẢO.....	19

PHIẾU CHẤM ĐIỂM ĐỒ ÁN MÔN HỌC

LẬP TRÌNH NHÚNG

Mã lớp học phần: 222210335801 (Lớp TH20DH-TH1)

Mã số sinh viên:

Họ tên sinh viên:

1. NỘI DUNG BÁO CÁO (4 điểm)

Tiêu chí chấm điểm	0đ	0.5đ	1.0đ
1-Hình thức trình bày báo cáo đúng mẫu theo quy định	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2-Đầy đủ các schema	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3-Đầy đủ các hướng dẫn minh họa	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4-Cài đặt hoàn chỉnh ứng dụng	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

2. NỘI DUNG VẤN ĐÁP (6 điểm) điểm

Tiêu chí chấm điểm	Câu hỏi 1 (3đ)				Câu hỏi 2 (3đ)			
	0 đ	0.5 đ	1.0 đ	1.5 đ	0 đ	0.5 đ	1.0 đ	1.5 đ
1.- Trả lời đúng nội dung câu hỏi	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2.- Thực hiện đúng yêu cầu của câu hỏi	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

ĐIỂM TỔNG CỘNG (sau khi thống nhất giữa 2 GV chấm)

Điểm số: Điểm chữ:

Giảng viên 1

(Ký và ghi rõ họ tên)

Lê Văn Hạnh

Giảng viên 2

(Ký và ghi rõ họ tên)

MỞ ĐẦU

Lập trình nhúng là một lĩnh vực trong lập trình máy tính, tập trung vào việc phát triển phần mềm cho các thiết bị điện tử nhúng vào các hệ thống khác nhau. Các thiết bị nhúng được sử dụng rộng rãi trong các ứng dụng như thiết bị y tế, thiết bị di động, ô tô, máy tính nhúng và nhiều hệ thống khác.

Lập trình nhúng yêu cầu kiến thức về lập trình và kiến thức về điện tử. Các ngôn ngữ lập trình phổ biến trong lĩnh vực nhúng bao gồm C, C++ và Assembly. Lập trình viên cần phải có kỹ năng phát triển phần mềm để tương tác với các thành phần phần cứng như vi điều khiển, cảm biến và các thiết bị khác.

Lập trình nhúng có thể được sử dụng để thiết kế và phát triển các hệ thống nhúng tùy chỉnh để giải quyết các vấn đề cụ thể. Nó cũng là một phần quan trọng của công nghiệp IoT (Internet of Things), cho phép các thiết bị kết nối với internet và với nhau để trao đổi dữ liệu và thực hiện các hoạt động thông minh.

CHƯƠNG 1:

NỘI DUNG CHÍNH ĐỀ TÀI

1.1 Giới thiệu đề tài

1.1.1 Giới thiệu đề tài cần xây dựng

Hệ thống tháp súng hoặc hệ thống tháp điều khiển từ xa được áp dụng rộng rãi trong quân sự, công nghiệp nặng, ... giúp người dùng có thể đứng ở khu vực an toàn và thao tác, từ đó sẽ chuyển dữ liệu để hệ thống tháp súng hoạt động

Với mục đích chính là thiết kế để cho phép người dùng điều khiển pháo bằng cách sử dụng bộ điều khiển từ xa được kết nối thông qua Bluetooth, Wifi, ...Giúp tăng tính linh hoạt và độ chính xác trong việc bắn đạn. Hơn nữa, với tính năng bắn liên thanh, laze, người dùng sẽ có thể tập trung vào việc di chuyển và định vị mục tiêu một cách dễ dàng hơn.

Vì thế nội dung đề tài xây dựng là “1 tháp súng Nerf có khả năng điều khiển từ xa bằng Bluetooth”, Nerf ở đây là tên gọi của đạn xốt, có đầu cao su, thường được dùng trong súng đồ chơi.

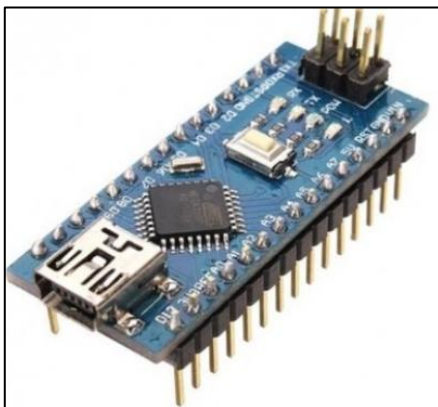
1.1.2 Mục tiêu đề tài

- Tháp súng có thể bắn được đạn Nerf
- Có khả năng quay ngang, quay dọc
- Có điều khiển từ xa bằng điện thoại hoặc trên máy tính thông qua Bluetooth

1.2 Phần cứng và linh kiện

1.2.1 Giới thiệu về linh kiện sử dụng

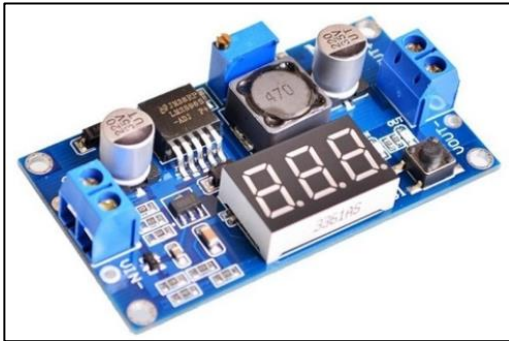
1.2.1.1 Arduino nano V3.0 ATmega328P



Mạch Ardunio Nano, mạch chính của hệ thống tháp súng, sử dụng chip Atmega328p, được nạp code chương trình bao gồm điều khiển các thiết bị sau:

- + HC-05 Bluetooth module
- + servo MG-90s xoay dọc
- + servo MG-90 dùng để đẩy đạn
- + servo MG996r xoay ngang
- + 2 động cơ RC 12V

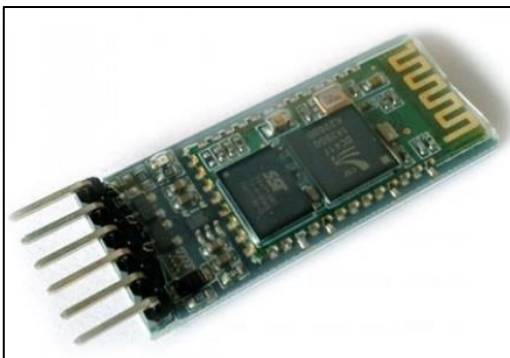
1.2.1.2 Mạch giảm áp Buck DC-DC LM2596 có hiển thị



Mạch giảm áp LM2596 dùng để hạ điện áp, giám sát nguồn điện luôn ở mức 5V nhằm làm tránh hỏng mạch điện, đồng thời cung cấp nguồn cho 3 servo hệ thống:

- + servo MG-90s xoay dọc
- + servo MG-90 dùng để đẩy đạn
- + servo MG996r xoay ngang

1.2.1.3 Module thu phát bluetooth HC-05



Module thu phát HC-05 dùng để thiết lập kết nối Serial giữa 2 thiết bị bằng sóng bluetooth, truyền xử lý về cho mạch Arduino Nano xử lý hoặc nhận ngược lại và truyền cho thiết bị đầu kia

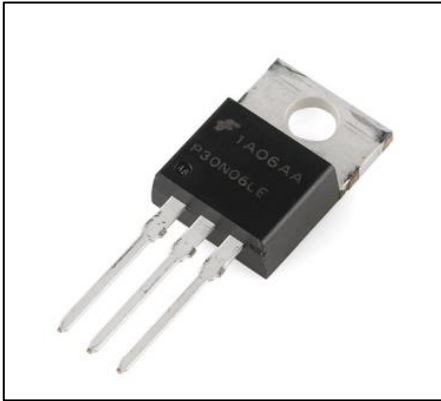
1.2.1.4 Diode chỉnh lưu FR207



Diode chỉnh lưu FR207 sử dụng để biến đổi một tín hiệu điện xoay chiều (AC) thành một tín hiệu điện một chiều (DC).

Diode chỉnh lưu được kết nối chéo với nguồn tín hiệu AC để chỉ cho dòng điện chạy tới một hướng duy nhất, loại bỏ phần dương hoặc phần âm của tín hiệu điện để không làm hỏng thiết bị điện tử

1.2.1.5 RFP30N06LE mosfet



RFP30N06LE là một loại MOSFET được sử dụng để điều khiển các và tải điện các động cơ

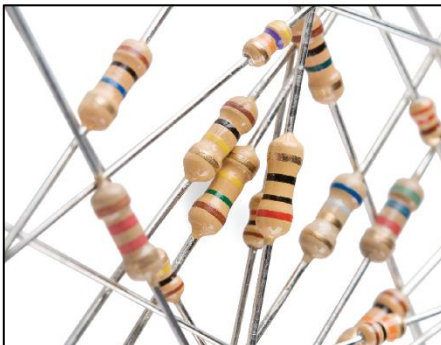
Ở đây là kết hợp với FR207 để điều khiển 2 động cơ DC 130 12V

1.2.1.6 Động cơ DC 130 12V



Động cơ DC loại 130 công suất tối đa 12V, được sử dụng để làm động lực bắn đạn Nerf nhờ lực cuốn sinh ra do 2 động cơ chạy ngược chiều nhau

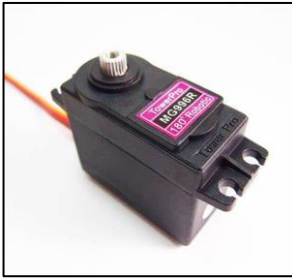
1.2.1.7 Điện trở



Hệ thống sử dụng 3 loại điện trở:

- 10k Ω , được kết nối với Mosfet và nguồn và giúp Mosfet luôn ở trạng thái tắt, khi nhận lệnh từ Arduino Nano sẽ mở và nhận dòng điện để chạy động cơ
- 330 Ω , giảm điện trở và kết nối chân RX của Module bluetooth HC-05 với chân TX của Arduino
- 680 Ω , giảm điện trở và kết nối chân GND và chân RX của Module bluetooth HC-05

1.2.1.8 Servo



Loại Servo MG996, có thể quay 360°, được sử dụng để làm thân trụ cho chân tháp súng, dùng để điều khiển quay ngang súng (giới hạn 120° cho mỗi bên)



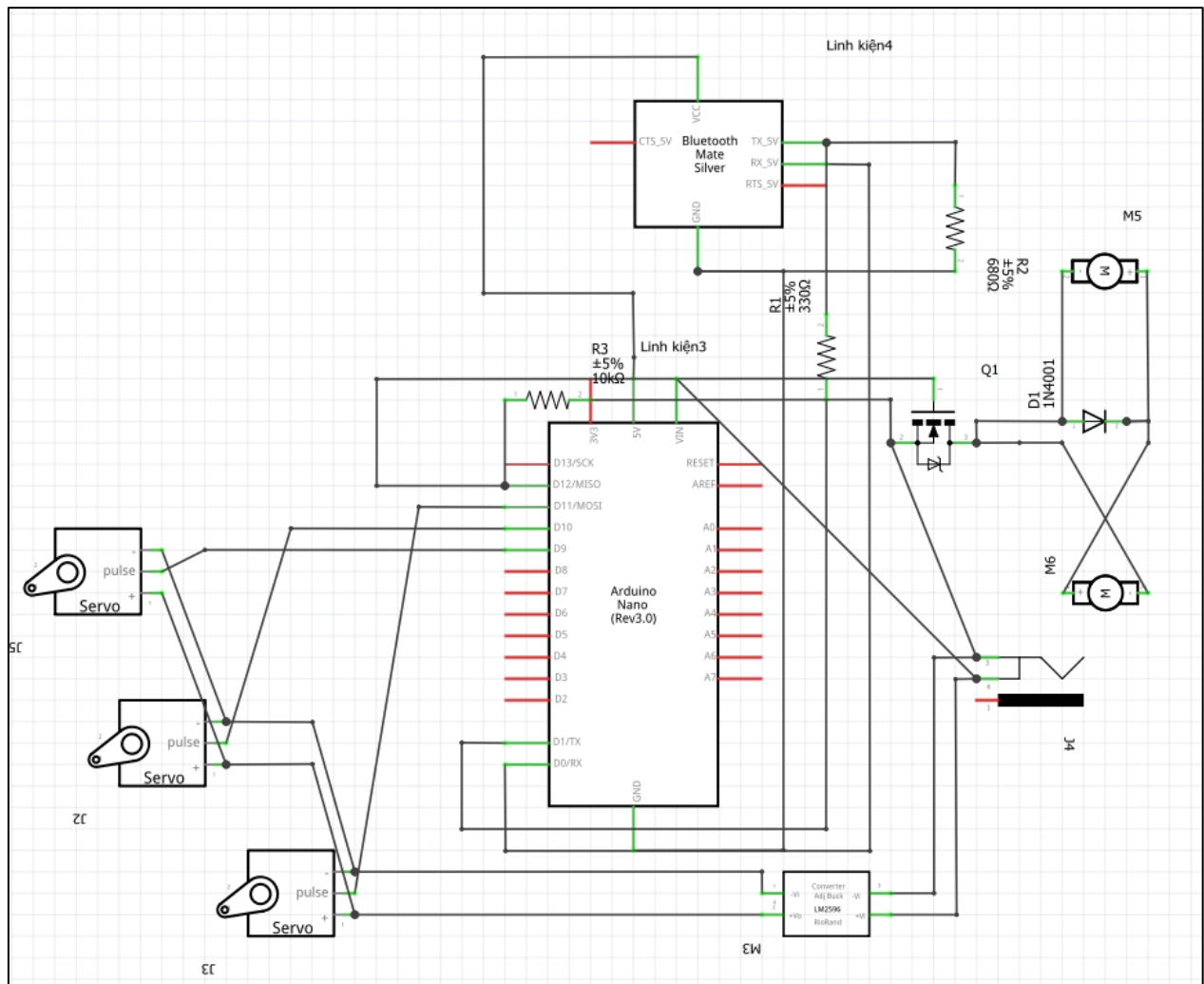
Loại Servo MG90S, có thể quay 360°, được sử dụng để làm hạ nòng, nâng nòng tháp súng (giới hạn ở mức 80° và -30°)



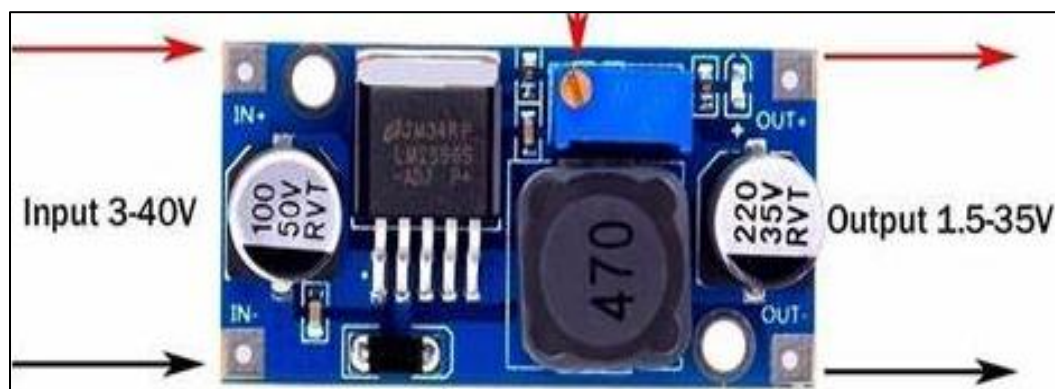
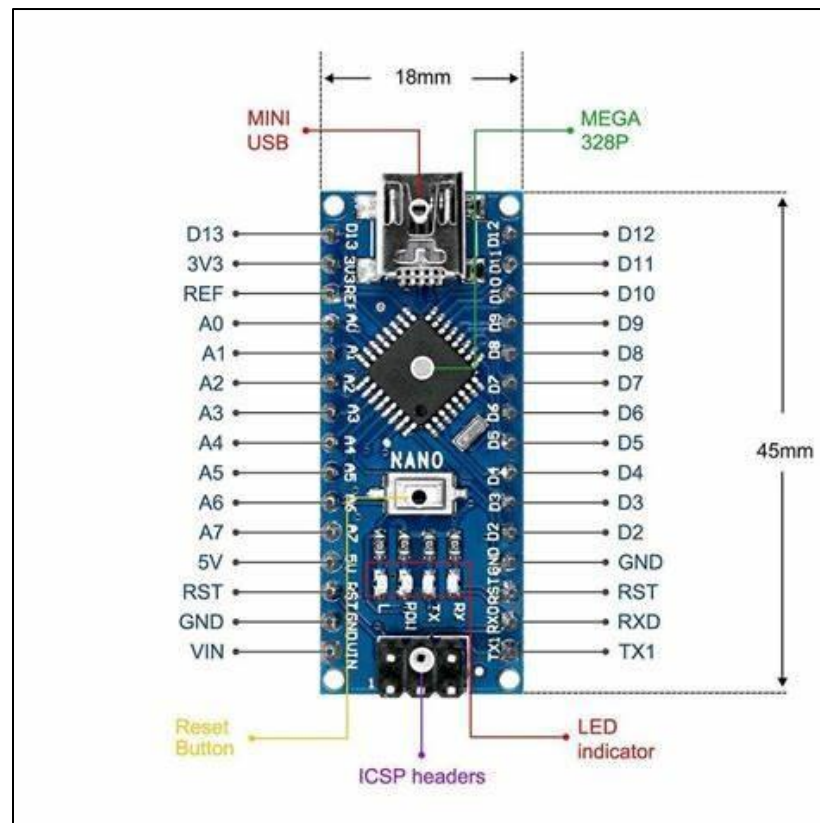
Loại Servo SG90, có thể quay 180°, được sử dụng để làm thiết bị gạt đạn (bằng cách chạy qua lại giữa 2 giá trị 120° và 60° mô phỏng thiết bị đẩy đạn liên hồi)

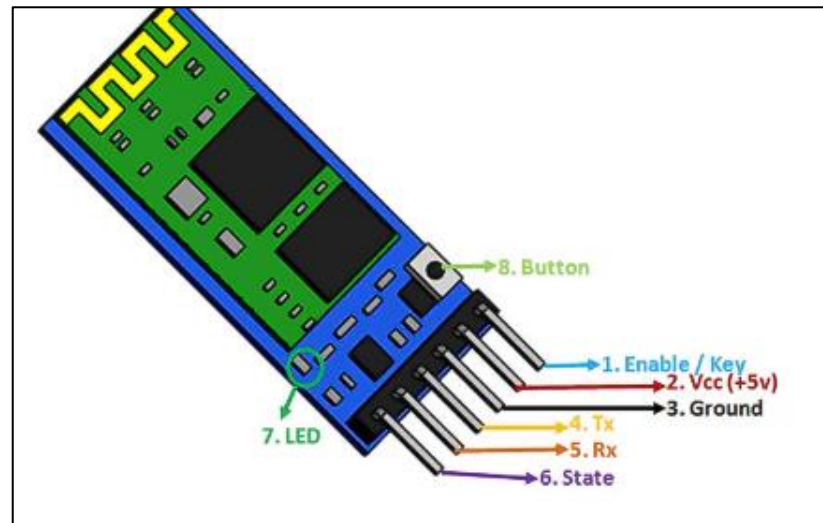
1.2.1.9 Các thành phần khác

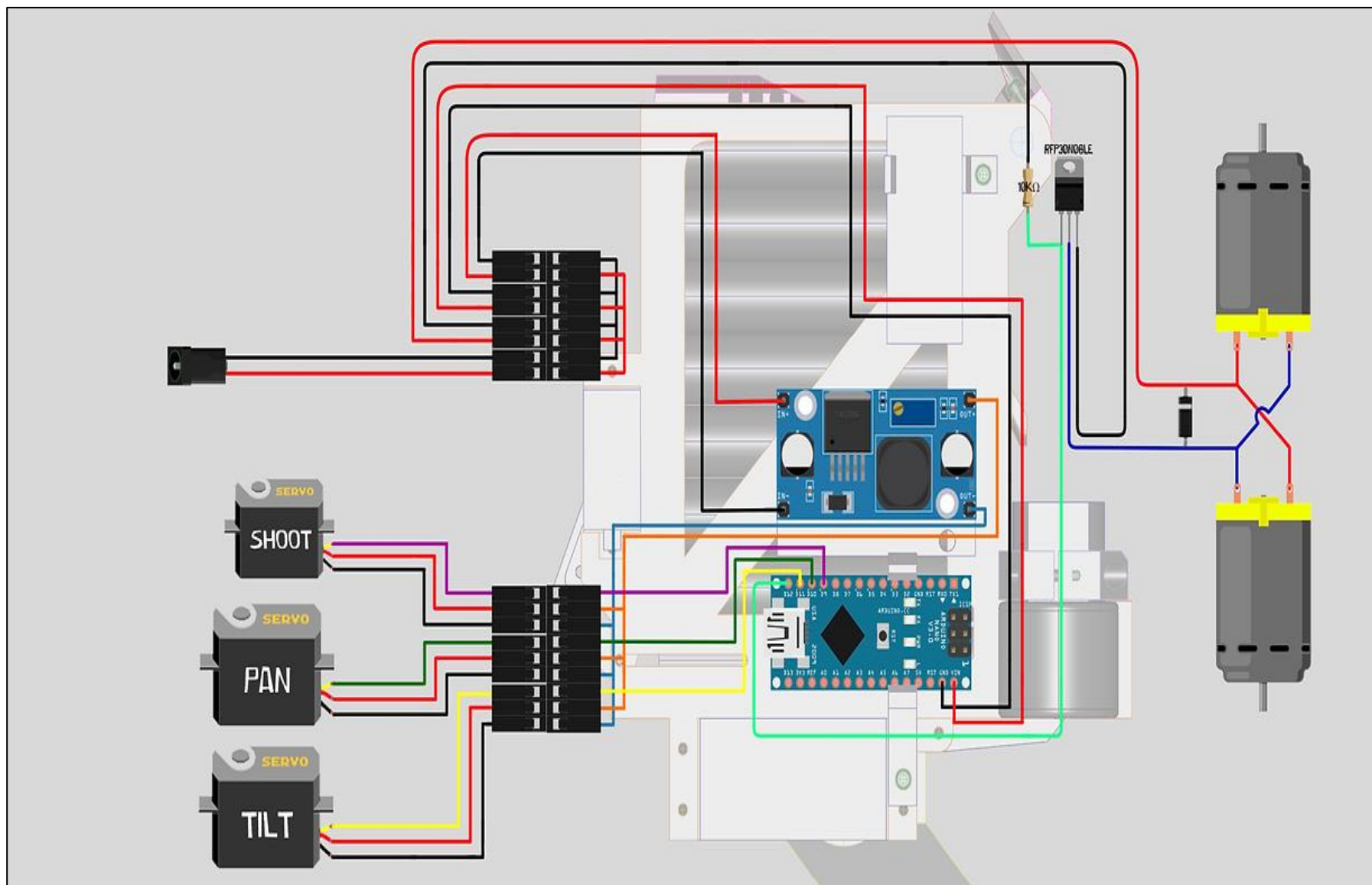
STT	Tên thành phần	Mô tả
1	Dây Bus	Dây nối điện
2	Đầu Cos	Đầu bấm dây thành dây bus
3	Chì hàn	Chì hàn mạch điện
4	Ống co nhiệt	Giúp bảo vệ mối hàn
5	ốc chìm M5x20	Ốc cố định phần đế tháp súng
6	Vít M2x10	Vít bắt cố định các linh kiện lên tháp súng
7	Nguồn 9V-2.5A	Cung cấp nguồn điện cho tháp súng

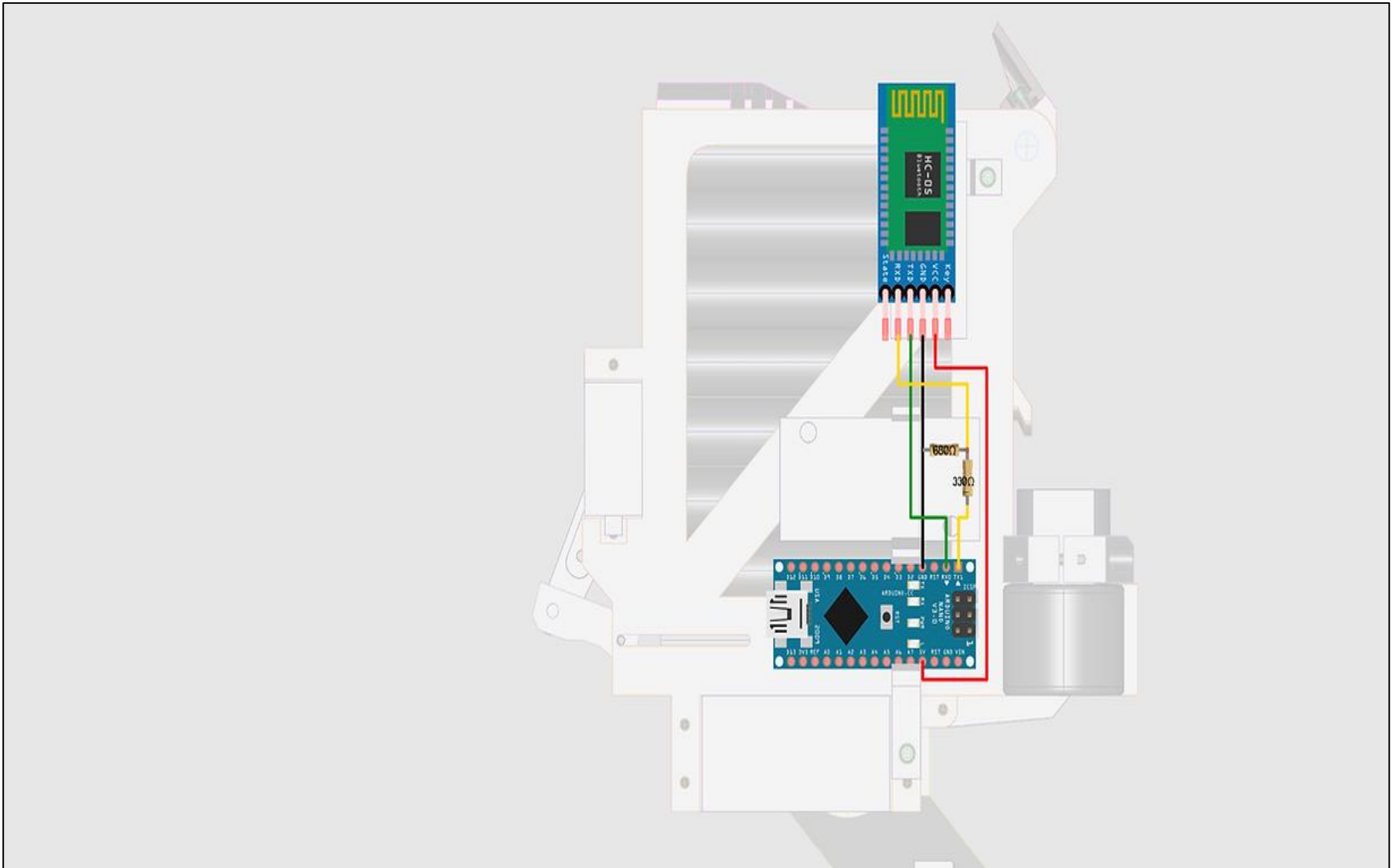


1.2.3 Thiết bị phần cứng: cấu tạo, bản vẽ phần cứng



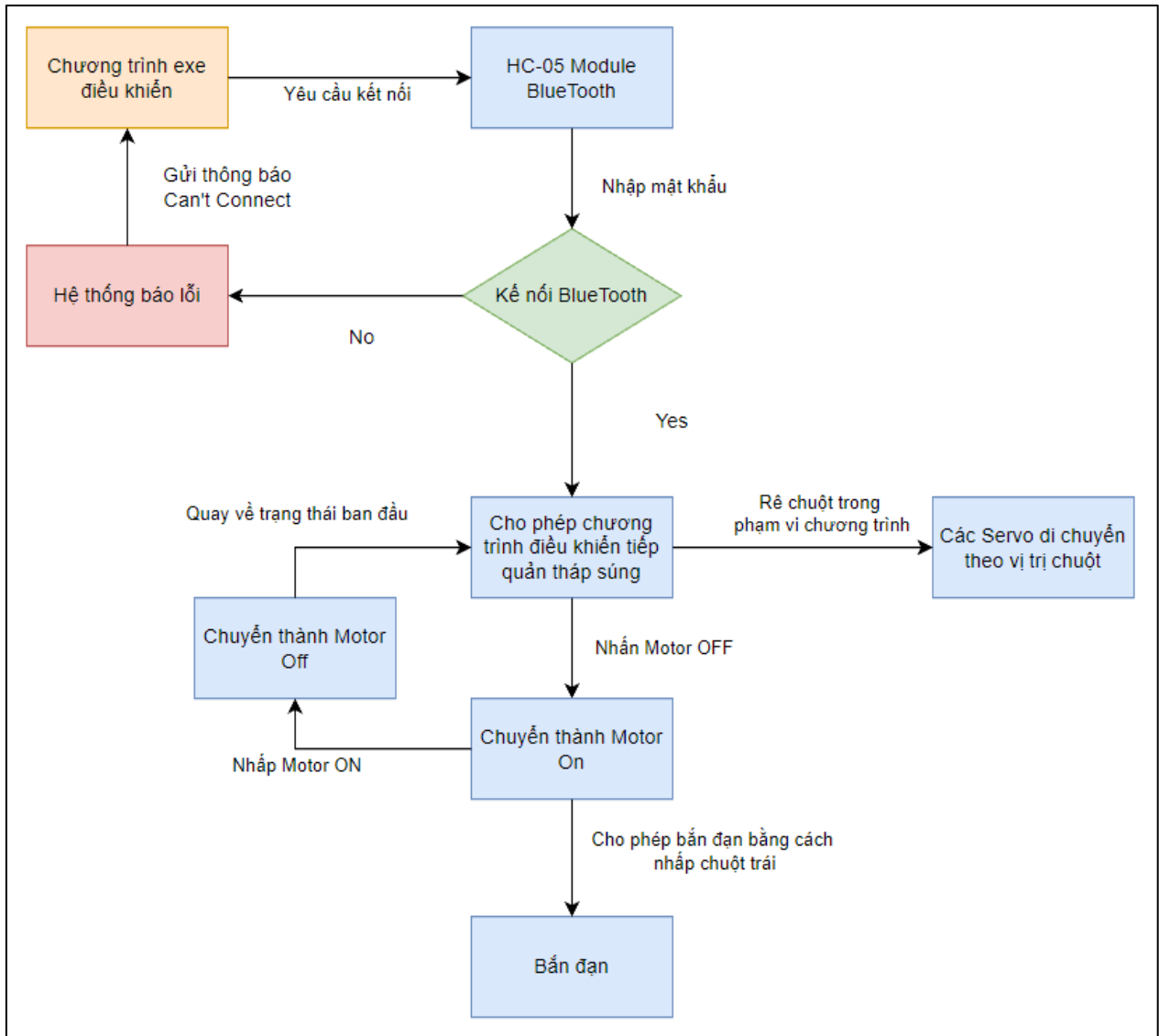








1.3 Xây dựng phần mềm

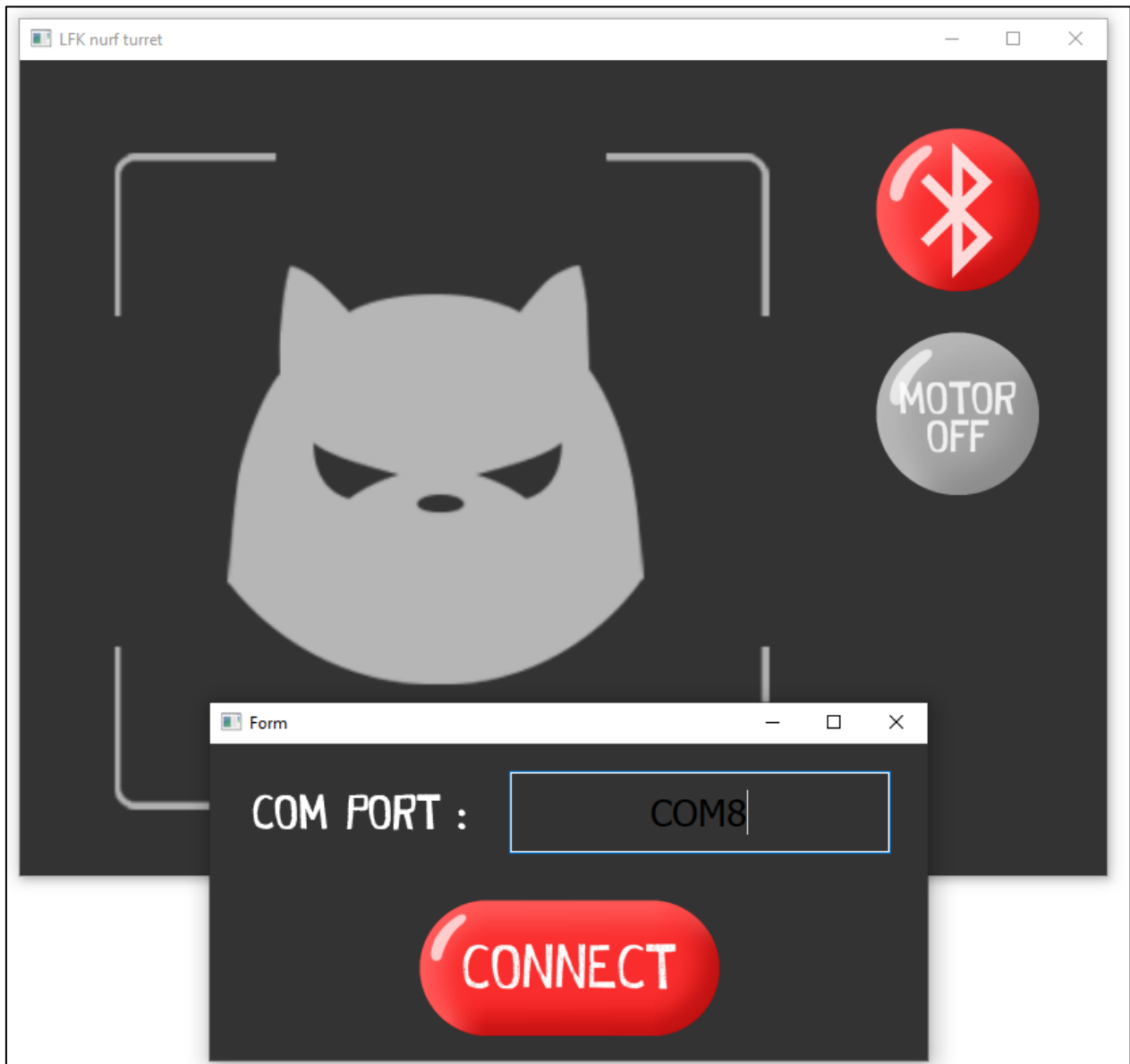
1.3.1 Giải thuật



1.3.2 Chương trình điều khiển

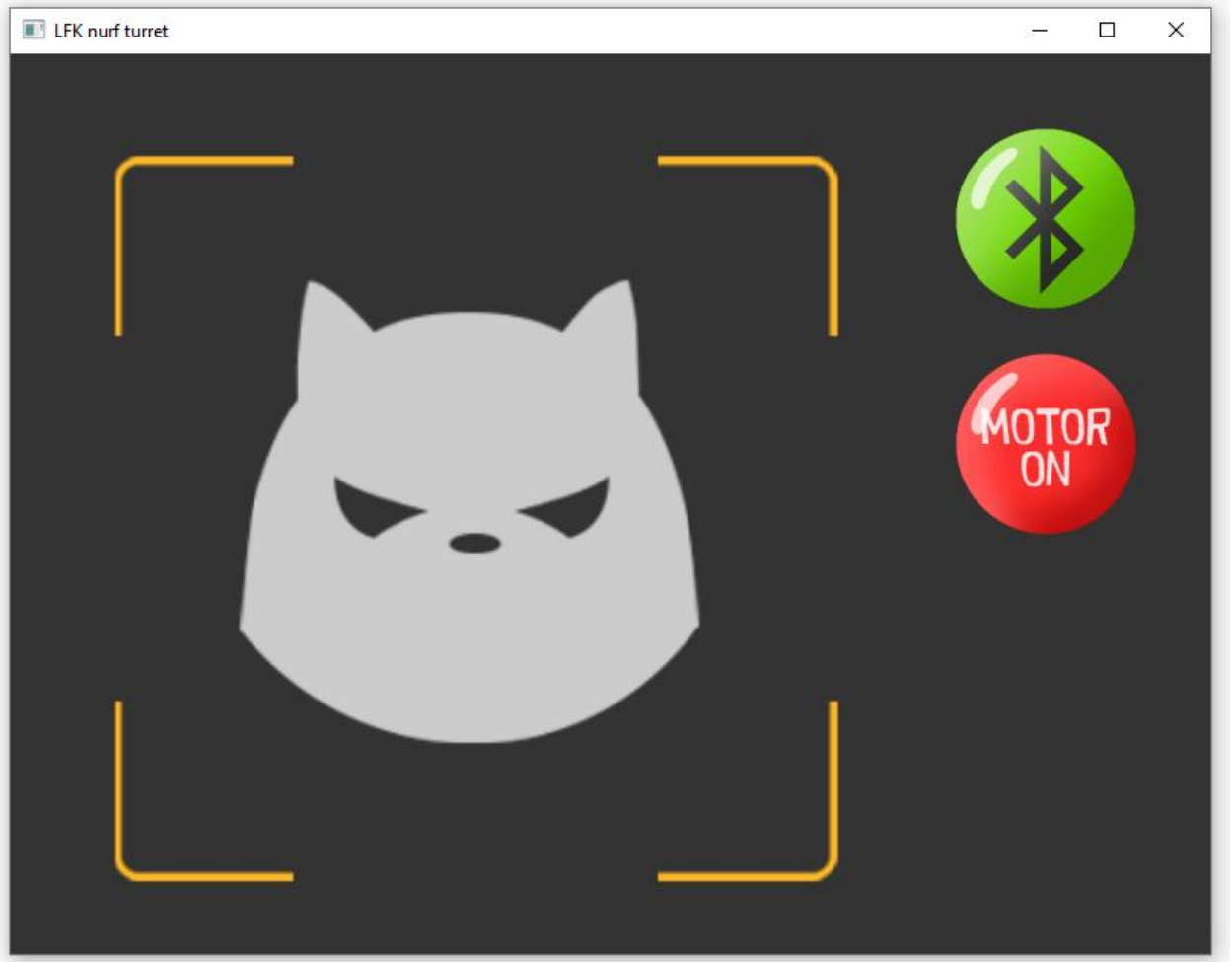
Hệ thống điều khiển bằng file exe

 GUI	19/10/2019 4:45 SA	File folder	
 Nerf_turret_pc	19/10/2019 4:43 SA	Application	34.573 KB



Cơ chế điều khiển tháp súng bằng cách rê chuột trên bản ứng dụng như trên (như cách rê touchpad laptop), tháp súng sẽ căn cứ vào vị trí con trỏ chuột sẽ điều khiển các Servo theo

Khi nhấn Motor On thì động cơ sẽ quay và lúc này có thể bắn, để bắn thì cần nhấp chuột trái vào khu vực điều khiển (có thể bắn liên thanh bằng việc nhấp chuột liên tục)



1.3.3 Chương trình (mã lệnh - Sketch)

```
#include <Servo.h>

//-----Declare servos and variables
Servo recoil_servo;
Servo pan_servo;
Servo tilt_servo;

const byte pan_limit_1 = 0;
const byte pan_limit_2 = 180;
const byte tilt_limit_1 = 65;
const byte tilt_limit_2 = 180;
const byte recoil_rest = 180;    // Angle of the servo when at rest
const byte recoil_pushed = 125;  // Angle the servo need to reach to push the
dart

//-----Variables related to serial data handling
byte byte_from_app;
const byte buffSize = 30;
byte inputBuffer[buffSize];
const byte startMarker = 255;
const byte endMarker = 254;
byte bytesRecvd = 0;
boolean data_received = false;

//-----Variable related to motor timing and firing
bool is_firing = false;
bool can_fire = false;
bool recoiling = false;

unsigned long firing_start_time = 0;
unsigned long firing_current_time = 0;
const long firing_time = 150;

unsigned long recoil_start_time = 0;
unsigned long recoil_current_time = 0;
const long recoil_time = 2 * firing_time;

const byte motor_pin = 12;
boolean motors_ON = false;

//8=====D
```

```
void setup()
{
    //-----define motor pin mode
    pinMode(motor_pin, OUTPUT);
    digitalWrite(motor_pin, LOW);

    //-----attaches servo to pins
    recoil_servo.attach(9);
    pan_servo.attach(10);
    tilt_servo.attach(11);

    //-----starting sequence
    recoil_servo.write(recoil_rest);
    pan_servo.write(90);
    //tilt_servo.write(tilt_limit_2);
    delay(1000);
    //tilt_servo.write(tilt_limit_2 + abs((tilt_limit_2 - tilt_limit_1)/2));
    tilt_servo.write(105);

    Serial.begin(9600); // begin serial communication
}

//8=====D

void loop()
{
    getDataFromPC();
    set_motor();
    if (data_received) {
        move_servo();
        set_recoil();
        set_motor();
    }
    fire();
}

//8=====D

void getDataFromPC() {

    //expected structure of data [start byte, pan amount, tilt amount, motor on,
    firing button pressed, end byte]
    //start byte = 255
```

```
//pan amount = byte between 0 and 253
//tilt amount = byte between 0 and 253
//motor on = 0 for off - 1 on
//firing button pressed = 0 for not pressed - 1 for pressed
//end byte = 254

if (Serial.available()) { // If data available in serial

    byte_from_app = Serial.read(); //read the next character available

    if (byte_from_app == 255) { // look for start byte, if found:
        bytesRecvd = 0; //reset byte received to 0(to start
        populating inputBuffer from start)
        data_received = false;
    }

    else if (byte_from_app == 254) { // look for end byte, if found:
        data_received = true; // set data_received to true so the
        data can be used
    }

    else { // add received bytes to buffer
        inputBuffer[bytesRecvd] = byte_from_app; //add character to input
        buffer
        bytesRecvd++; // increment byte received
        (this act as an index)
        if (bytesRecvd == buffSize) { // just a security in case the inputBuffer
        fills up (shouldn't happen)
            bytesRecvd = buffSize - 1; // if bytesReceived > buffer size set
            bytesReceived smaller than buffer size
        }
    }
}

//8=====D

void move_servo() {

    byte pan_servo_position = map(inputBuffer[0], 0, 253, pan_limit_2,
    pan_limit_1); //convert inputbuffer value to servo position value
    pan_servo.write(pan_servo_position); //set pan servo position
    byte tilt_servo_position = map(inputBuffer[1], 0, 253, tilt_limit_2,
    tilt_limit_1); //convert inputbuffer value to servo position value
```

```
    tilt_servo.write(tilt_servo_position); //set pan servo position
}

//8=====D

void set_recoil() {

    if (inputBuffer[3] == 1) {          //if fire button pressed
        if (!is_firing && !recoiling) { //and not already firing or recoiling
            can_fire = true;            //set can fire to true (see effect in void
fire())
        }
    }
    else {                              // if fire button not pressed
        can_fire = false;               //set can fire to false (see effect in void fire())
    }
}

//8=====D

void set_motor() {
    //-----start and stop motors using TIP120 transisitor .

    if (inputBuffer[2] == 1) {          //if screen touched
        digitalWrite(motor_pin, HIGH);  //turn motor ON
        motors_ON = true;
    }
    else {                              //if screen not touched
        digitalWrite(motor_pin, LOW);   //turn motor OFF
        motors_ON = false;
    }
}

//8=====D

void fire() { //if motor byte on, turn motor on and check for time it has been on

    if (can_fire && !is_firing && motors_ON) {
        //if (can_fire && !is_firing) {
        firing_start_time = millis();
        recoil_start_time = millis();
        is_firing = true;
    }
}
```

```
firing_current_time = millis();
recoil_current_time = millis();

if (is_firing && firing_current_time - firing_start_time < firing_time) {
    recoil_servo.write(recoil_pushed);
}
else if (is_firing && recoil_current_time - recoil_start_time < recoil_time) {
    recoil_servo.write(recoil_rest);
}
else if (is_firing && recoil_current_time - recoil_start_time > recoil_time) {
    is_firing = false;
}
}
```

1.4 Tự đánh giá kết quả thực hiện

Sau khi lựa chọn và hoàn thành đề tài thì bản thân em đã học hỏi thêm việc điều khiển thiết bị qua Bluetooth, biết giảm áp và tính toán điện năng cần cho hệ thống, điều khiển động cơ từ xa, và chạy các hệ thống servo riêng biệt. Theo đánh giá của em thì hệ thống tháp súng này có thể tùy chỉnh để hóa thân thành các thiết bị tưới tiêu từ xa trong nông nghiệp, các thiết bị giám sát khu vực nguy hiểm phục vụ cho khoa học,...

Về phần đồ án thì em đã có 1 trải nghiệm tuyệt vời khi thực hiện, đem lại cảm giác rất vui khi thực hiện nhưng cũng gặp không ít khó khăn thời gian đầu

Mặc dù đề tài này chức năng cũng như phần cứng đã được xử lý tốt nhưng nhìn chung thì cũng chỉ là mô phỏng nhỏ mọi điều khiển từ xa. Đây cũng là đề tài điện tử đầu tiên nên em thấy cũng khá thú vị về môn lập trình nhúng này. Vì vậy, em sẽ cố gắng tìm hiểu thêm để cải tiến tháp súng hơn, hy vọng ý tưởng này sẽ giúp các bạn mới có hứng thú thêm với môn học này, đồng thời cũng giúp người khác có thêm nhiều ý tưởng hay và biến thành hiện thực trong tương lai không xa

CHƯƠNG 2: KẾT LUẬN

Sau quá trình nghiên cứu và thực hiện đề tài "Tháp súng Nerf điều khiển bằng Bluetooth", nhóm em đã hoàn thành các yêu cầu chính đã đề ra trước đó. Trong quá trình thực hiện đề tài, em đã đúc kết được nhiều kinh nghiệm quý báu trong việc phân chia công việc và tìm tòi, đọc thêm các tài liệu ngoài để xây dựng nội dung chi tiết và viết báo cáo đồ án.

Tuy nhiên, phần báo cáo của em vẫn còn nhiều hạn chế, đặc biệt là về kiến thức chuyên sâu, vẫn chưa tìm hiểu được sâu về đề tài, do giới hạn khả năng chuyên môn, cũng như là lĩnh vực. Em sẽ tiếp tục nghiên cứu và cải tiến để cải thiện đề tài của mình tốt hơn trong tương lai

Một lần nữa, nhóm chúng em xin chân thành cảm ơn thầy **ThS.Lê Văn Hạnh** đã hướng dẫn nhóm chúng em rất nhiệt tình, cụ thể trong quá trình thực hiện đề tài này. Cảm ơn quý thầy cô và các bạn đã cho nhóm em những ý kiến đóng góp quý báu và tạo mọi điều kiện thuận lợi nhất để em học hỏi, nghiên cứu và thực hiện đề tài này. Mặc dù đã rất cố gắng, nỗ lực để hoàn thiện đề tài một cách tốt nhất nhưng do còn hạn chế về thời gian và kinh nghiệm nên đề tài vẫn còn nhiều thiếu sót. Kính mong thầy, cô và các bạn tiếp tục đóng góp thêm để đề tài được hoàn thiện hơn.

TÀI LIỆU THAM KHẢO

- [1] Arduino, "Getting Started with Arduino," [Online]. Available: <https://docs.arduino.cc/learn/starting-guide/getting-started-arduino>.
- [2] L. Frenchkev, "Bluetooth Nerf turret," [Online]. Available: <https://www.littlefrenchkev.com/bluetooth-nerf-turret>.