

TRƯỜNG ĐẠI HỌC BÁCH KHOA HỒ CHÍ MINH  
KHOA ĐIỆN – ĐIỆN TỬ  
BỘ MÔN ĐIỀU KHIỂN TỰ ĐỘNG

LUẬN VĂN TỐT NGHIỆP  
ĐIỀU KHIỂN VÀ GIÁM SÁT QUADCOPTER SỬ  
DỤNG  
ARDUINO UNO R3 VÀ STM32F103C8T6

Giảng viên hướng dẫn: TS. Hoàng Minh Trí

Sinh viên thực hiện: Lương Cao Thuần

KỸ SƯ NGÀNH KỸ THUẬT ĐIỀU KHIỂN & TỰ ĐỘNG

TP. HỒ CHÍ MINH, NĂM 2022

**TRƯỜNG ĐẠI HỌC BÁCH KHOA HỒ CHÍ MINH  
KHOA ĐIỆN – ĐIỆN TỬ  
BỘ MÔN ĐIỀU KHIỂN TỰ ĐỘNG**

**LUẬN VĂN TỐT NGHIỆP  
ĐIỀU KHIỂN VÀ GIÁM SÁT QUADCOPTER SỬ DỤNG  
ARDUINO UNO R3 VÀ STM32F103C8T6**

**CONTROL AND MONITORING QUADCOPTER USED  
ARDUINO UNO R3 AND STM32F103C8T6**

**LƯƠNG CAO THUẦN-1814218  
KỸ SƯ NGÀNH KỸ THUẬT ĐIỀU KHIỂN & TỰ ĐỘNG**

**TP. HỒ CHÍ MINH, NĂM 2022**

## **DANH SÁCH HỘI ĐỒNG BẢO VỆ LUẬN VĂN**

Hội đồng chấm luận văn tốt nghiệp, thành lập theo Quyết định số .....  
ngày.....của Hiệu trưởng Trường Đại học Bách khoa TP.HCM.

1. .... Chủ tịch.
2. .... Thư ký.
3. .... Ủy viên.

TRƯỜNG ĐẠI HỌC BÁCH KHOA  
TP. HỒ CHÍ MINH  
**KHOA ĐIỆN – ĐIỆN TỬ**  
**BỘ MÔN: ĐIỀU KHIỂN TỰ ĐỘNG**

CỘNG HÒA XÃ HỘI CHỦ NGHĨA  
VIỆT NAM  
Độc Lập - Tự Do - Hạnh Phúc

TP. HCM ngày... tháng..... năm.....

**NHẬN XÉT LUẬN VĂN TỐT NGHIỆP  
(CỦA CÁN BỘ HƯỚNG DẪN/PHẢN BIỆN)**

**Tên đề tài  
ĐIỀU KHIỂN VÀ GIÁM SÁT QUADCOPTER**

**Sinh viên thực hiện**

Lương Cao Thuần

**MSSV**

1814218

**Cán bộ hướng dẫn**

Ts. Hoàng Minh Trí

**Đánh giá Luận văn**

**1. Về cuốn báo cáo:**

Số trang	-----	Số chương	-----
Số bảng số liệu	-----	Số hình vẽ	-----
Số tài liệu tham khảo	-----	Sản phẩm	-----

Một số nhận xét về hình thức cuốn báo cáo:

.....  
.....  
.....  
.....

**2. Về nội dung luận văn:**

.....  
.....

.....  
.....  
.....  
.....  
.....  
**3. Về tính ứng dụng:**

.....  
.....  
.....  
.....  
.....

.....  
.....  
.....  
.....  
**4. Về thái độ làm việc của sinh viên:**

.....  
.....  
.....  
.....  
.....

**Đánh giá chung:**

.....  
.....  
.....  
.....  
.....

**Điểm từng sinh viên:**

Lương Cao Thuần :...../10

**Người nhận xét**  
(Ký tên và ghi rõ họ tên)

TRƯỜNG ĐẠI HỌC BÁCH KHOA  
TP. HỒ CHÍ MINH  
**KHOA ĐIỆN – ĐIỆN TỬ**  
**BỘ MÔN: ĐIỀU KHIỂN TỰ ĐỘNG**

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM  
Độc lập - Tự do - Hạnh phúc

TP. HCM, ngày .... tháng ..... năm .....

**NHẬN XÉT LUẬN VĂN TỐT NGHIỆP  
CỦA CÁN BỘ PHẢN BIỆN**

Tên đề tài:

**ĐIỀU KHIỂN VÀ GIÁM SÁT QUADCOPTER**

**Sinh viên thực hiện**

Lương Cao Thuần

**MSSV**

1814218

**Cán bộ phản biện**

**Đánh giá Luận văn**

**1. Về cuốn báo cáo:**

Số trang	-----	Số chương	-----
Số bảng số liệu	-----	Số hình vẽ	-----
Số tài liệu tham khảo	-----	Sản phẩm	-----

Một số nhận xét về hình thức cuốn báo cáo:

**2. Về nội dung luận văn:**

**3. Về tính ứng dụng:**

**4. Về thái độ làm việc của sinh viên:**

**Đánh giá chung:**

**Điểm từng sinh viên:**

Lương Cao Thuần :...../10

**Người nhận xét**

(Ký tên và ghi rõ họ tên)

TRƯỜNG ĐẠI HỌC BÁCH  
KHOA TP. HỒ CHÍ MINH

**KHOA ĐIỆN – ĐIỆN TỬ**

**BỘ MÔN: ĐIỀU KHIỂN TỰ  
ĐỘNG**

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập - Tự do - Hạnh phúc

TP. HCM, ngày .... tháng ..... năm .....

### **ĐỀ CƯƠNG CHI TIẾT**

**TÊN ĐỀ TÀI: ĐIỀU KHIỂN VÀ GIÁM SÁT QUADCOPTER**

**Cán bộ hướng dẫn: Ts.Hoàng Minh Trí**

**Thời gian thực hiện:** Từ ngày 1/6/2022.....đến ngày 1/12/2022

**Sinh viên thực hiện:**

**Lương Cao Thuần-1814218**

### Nội dung đề tài:

**Mục tiêu, phạm vi đối tượng:** Xây dựng được hệ thống điều khiển và giám sát được máy bay 4 cánh quạt, có thể đáp ứng mục tiêu thương mại với các tính năng.

- Bộ điều khiển bay hoạt động ổn định với sai số nhỏ.
- Bộ giám sát, giám sát được được độ ổn định của mạch cân bay.
- Giám sát được vị trí máy bay trên bản đồ, dung lượng PIN.
- Thiết kế được màn hình giám sát từ máy tính.

### **Phương pháp thực hiện:**

- Thiết kế mạch điều khiển bằng ARDUINO UNO R3, và phần mềm ARDUINO IDE.
- Thiết kế mạch giám sát bằng STM32F103C8T6 và phần mềm Keil C V5.
- Thiết kế màn hình giám sát bằng công cụ Visual Studio 2019.

**Kết quả mong đợi:** thiết kế và thực hiện được mạch cân bằng quadcopter, thiết kế và thực hiện được hệ thống giám sát Quadcopter.

### **Kế hoạch thực hiện**

Sinh viên thực hiện	Công việc	Thời gian
Lương Cao Thuần	Tìm hiểu và khảo sát các Quadcopter	01/06 – 08/06/2022
Lương Cao Thuần	Tìm hiểu công nghệ và lên phương án thực hiện	08/06 – 15/06/2022
Lương Cao Thuần	Tìm hiểu, xác định phần cứng, linh kiện cần thiết và tiến hành mua	16/06 – 30/07/2022
Lương Cao Thuần	Tìm hiểu và nghiên cứu làm phần cứng	01/10 – 06/08/2022

Lương Cao Thuần	Tìm hiểu về Arduino UNO R3 và MPU6050	08/08 – 20/08/2022
-----------------	--	--------------------

Lương Cao Thuần	Tìm hiểu về STM32F103C8T6	21/08 – 30/09/2022
Lương Cao Thuần	Tìm hiểu về MC6C và Mc7RB	01/10 – 08/10/2022
Lương Cao Thuần	Tìm hiểu các giao thức truyền thông như UART, I2C, SPI	09/10 – 15/10/2022
Lương Cao Thuần	Nghiên cứu giải thuật điều khiển cho Quadcopter	15/10 – 20/10/2022
Lương Cao Thuần	Tìm hiểu và lập trình vi điều khiển STM32F103C8T6	21/10 – 1/11/2022
Lương Cao Thuần	Tìm hiểu về Visual Studio 2019 và Winform C#	01/11 – 05/11/2022
Lương Cao Thuần	Tìm hiểu về GPS Neo 6mM và NRF21L01	06/11 – 15/11/2022
Lương Cao Thuần	Thực hiện làm mạch in.	15/11 – 30/11/2022
Lương Cao Thuần	Thực hiện viết chương trình và debug	01/11 – 22/12/2022

<b>Xác nhận của Cán bộ hướng dẫn</b>	TP. HCM, ngày....tháng .....năm 2022

## Mục lục

Chương 0: TÓM TẮT LUẬN VĂN	19
Tóm tắt luận văn	19
CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN VỀ ĐỀ TÀI	21
1.1 Lý do chọn đề tài	21
1.2 Nhiệm vụ đề tài:	22
CHƯƠNG 2: TỔNG QUAN HỆ THỐNG VÀ PHƯƠNG PHÁP THỰC HIỆN	24
2.1 Khái niệm Quadcopter	24
2.2 Tổng quan của bộ điều khiển bay:	25
2.3 Tổng quan của bộ giám sát:	26
CHƯƠNG 3 CƠ SỞ LÝ THUYẾT	28
3.1 Nguyên lý cân bằng:	28
3.2 Nguyên lý di chuyển Quadcopter.	28
3.3 Bộ điều khiển PID.	32
3.4 Bộ lọc thông thấp (LPF):	33
3.5 Bộ lọc bù (complementary filter).	34
3.6 Bộ điều khiển PID cân bằng cho góc roll, pitch:	35
3.7 Nguyên lý hoạt động của GPS:	36
CHƯƠNG 4 QUÁ TRÌNH THỰC HIỆN	38
4.1 Tổng quan các phần cứng sử dụng trong luận văn.	38
4.2 Mạch cân bằng bay.	46
4.3 Mạch giám sát TX:	48
4.4 Mạch giám sát Rx:	50
4.5 Giới thiệu về các công cụ phần mềm sử dụng trong luận văn.	51
4.6 Phần mềm điều khiển và giám sát quadcopter:	54
4.7 Sơ đồ giải thuật phần mềm giám sát quadcopter:	55
4.8 Thực hiện màn hình giám quadcopter.	56
4.9 Kết nối giữa mạch cân bằng bay và mạch giám sát TX:	58
4.10 Kết nối phần cứng giữa GPS NEO 6M và SM32F103C8T6	60
4.11 Kết nối phần cứng giữa MPU6050 và Arduino UNO R3	61
4.12 Kết nối giữa STMF103C8T6 và NRF24L01:	62
CHƯƠNG 5 KẾT QUẢ THỰC HIỆN.	64
5.1 Tín hiệu bộ lọc thông thấp đọc gyroscope của MPU6050:	64
5.2 Tín hiệu góc pitch và roll khi quadcopter được cân bằng:	65
5.3 Kết quả thu được từ bộ giám sát.	70
	13

5.4. Video kết quả thực hiện:	73
-------------------------------	----

**CHƯƠNG 6 TỔNG KẾT** 74

6.1 Những kết quả đạt được trong đề tài luận văn:	74
---	----

6.2 Những hạn chế cần khắc phục trong đề tài:	74
---	----

## DANH HÌNH ẢNH

Hình 2.1.1 Hình ảnh minh họa của Quadcopter.	27
Hình 2.1.2 Hình ảnh minh họa tổng quan mô hình.	28
Hình 2.2 Hình ảnh minh họa tổng quan bộ điều khiển.	29
Hình 2.2 Hình ảnh minh họa tổng quan bộ điều khiển.	30
Hình 2.3 Hình ảnh minh họa tổng quan bộ giám sát.3	31
Hình 3.1 Hình ảnh minh họa lý thuyết cân bằng Quadcopter.33	32
Hình 3.2.1 Hình mô tả trạng thái quadcopter khi di chuyển lên xuống	33
Hình ảnh 3.2.2 Hình ảnh mô tả quadcopter khi di chuyển qua trái	34
Hình ảnh 3.2.3 Hình ảnh mô tả quadcopter khi di chuyển qua phải	35
Hình 3.2.5 Hình ảnh minh họa của mô hình quay cùng chiều kim đồng hồ và ngược chiều kim đồng hồ.	36
Hình 3.3.1 Hình ảnh minh họa bộ điều khiển PID	37
Hình 3.4.1 Hình ảnh minh họa cho bộ lọc thông thấp.	38
Hình 3.5.1 Hình ảnh minh họa bộ lọc thông bù.	39
Hình 3.6.1 Hình ảnh minh họa cho bộ điều khiển PID cân bằng góc Pitch và Yaw.	40
Hình 3.7.1 Hình ảnh minh họa cho nguyên lý hoạt động của GPS.	41
Hình 3.7.2 Hình ảnh minh họa cho nguyên lý hoạt động của GPS.	42
Hình 4.1 Hình ảnh minh họa Arduino UNO R3	43
Hình 4.1.2 Hình ảnh minh họa cho MPU6050	44
Hình 4.1.3 Hình ảnh minh họa Frame Kit F450	45
Hình 4.1.4 Hình ảnh minh họa bộ điều khiển động cơ không chổi than	46
Hình 4.1.5 Hình ảnh minh họa cho pin Lipo 3s.	47
Hình 4.1.6 Hình ảnh minh họa STM32F103C8T6	48
Hình 4.1.7 Hình ảnh mạch định vị GPS NEO 6M	49
Hình 4.1.8 Hình ảnh minh họa mạch phát MC6C	50
Hình 4.1.9 Hình ảnh minh họa board thu MC7RB	51
Hình 4.1.10 Hình ảnh minh họa mạch thu phát radio nrf24l01.	52

Hình 4.1.11 Hình ảnh minh họa mạch giảm áp Lm2569	52
Hình 4.2.1 Hình ảnh thể hiện sơ đồ nguyên lý của mạch điều khiển.	53
Hình 4.2.2 Hình ảnh mạch giám sát PCB được vẽ trên Proteus	53
Hình 4.2.3 Hình ảnh thực tế mạch điều khiển quadcopter (1).	54
Hình 4.2.3 Hình ảnh thực tế mạch điều khiển quadcopter (2).	57
Hình 4.3.1 Sơ đồ nguyên lý mạch giám sát Tx.	58
Hình 4.3.2 Hình ảnh vẽ mạch PCB trên proteus.	59
Hình 4.3.3 Hình ảnh thực tế của mạch giám sát TX	60
Hình 4.4.1 Sơ đồ nguyên lý mạch giám sát RX:	60
Hình 4.4.2 Hình minh họa mạch PCB RX	50
Hình 4.4.3 Hình ảnh thực tế mạch thu Rx	61
Hình 4.5.1 Hình ảnh minh họa giới thiệu phần mềm Keil C.	62
Hình 4.5.2 Hình ảnh minh họa cho giới thiệu phần mềm visual studio.	63
Hình 4.5.3 Hình ảnh minh họa cho WINFORM.	64
Hình 4.6.1 Hình ảnh sơ đồ giải thuật chương trình điều khiển quadcopter	65
Hình 4.7.1 Hình ảnh sơ đồ giám sát của chương trình phát dữ liệu TX (1)	65
Hình 4.7.2 Hình ảnh sơ đồ giải thuật của chương trình phát dữ liệu TX (2)	66
Hình 4.7.3 Hình ảnh sơ đồ giải thuật của chương trình thu RX	67
Hình 4.8.1 Hình ảnh minh họa giao diện monitor quadcopter (1)	68
Hình 4.8.2 Hình ảnh minh họa giao diện monitor quadcopter (2)	69
Hình 4.8.3 Hình ảnh minh họa giao diện monitor quadcopter (3)	70
Hình 4.8.4 Hình ảnh minh họa giao diện monitor quadcopter (4)	71
Hình 4.8.5 Hình ảnh minh họa giao diện monitor quadcopter (5)	71
Hình 4.9.1 Hình ảnh kết nối phần cứng giữ stm32f103 và arduino uno r3	72
Hình 4.9 Hình ảnh minh họa gói tin truyền từ arduino to stm32	72
Hình 4.10.1 Hình ảnh minh họa kết nối phần cứng giữa GPS NEO 6M và SM32F103C8T6	73
Hình 4.10.2 Hình ảnh minh họa gói tin truyền từ GPS NEO 6M.	74
Hình 4.11.1 Hình ảnh minh họa kết nối giữa MPU6050 và Arduino UNO R3	75

Hình 4.12.1 Hình ảnh minh họa kết nối giữa stm32 và nrf24l01	76
Hình 5.1.1 Hình ảnh ghi nhận tín hiệu gyro roll	77
Hình 5.1.3 Hình ảnh xử lý của bộ lọc thông thấp cho tín hiệu gyro pitch.	78
Hình 5.2.1 Hình ảnh đo được góc pitch và roll khi throttle = 1200 (motor 1, motor 3).	79
Hình 5.2.2 Hình ảnh đo được góc pitch và roll khi throttle = 1500 (motor 1, motor 3).	79
Hình 5.2.3 Hình ảnh đo được góc pitch và roll khi throttle = 1700 (motor 1, motor 3).	80
Hình 5.2.4 Hình ảnh đo được góc pitch và roll khi throttle = 1200 (motor 2, motor 4).	80
Hình 5.2.5 Hình ảnh đo được góc pitch và roll khi throttle = 1500 (motor 2, motor 4).	81
Hình 5.2.6 Hình ảnh đo được góc pitch và roll khi throttle = 1700 (motor 2, motor 4).	82
Hình 5.2.7 Hình ảnh đo được góc pitch và roll khi throttle = 1200 (4 motor).	82
Hình 5.2.8 Hình ảnh đo được góc pitch và roll khi throttle = 1500 (4 motor).	83
Hình 5.2.9 Hình ảnh đo được góc pitch và roll khi throttle = 1700 (4 motor).	83
Hình 5.3.1 Hình ảnh báo cáo kết quả giám sát quadcopter(1)	83
Hình 5.3.3 Hình ảnh báo cáo kết quả giám sát quadcopter(2)	84
Hình 5.3.4 Hình ảnh kết quả từ bộ giám sát LCD (1)	84
Hình 5.3.5 Hình ảnh kết quả từ bộ giám sát LCD (2)	85

### DANH MỤC TỪ VIẾT TẮT

<b>ESC</b>	Bộ điều tốc (Electronic Speed Controller)
<b>ADC</b>	Analog to Digital Convert
<b>RX</b>	Bộ thu sóng radio (Receiver)
<b>TX</b>	Bộ phát sóng radio (Transmitter).
<b>PWM</b>	Điều chỉnh độ rộng xung (Pulse Width Modulation)
<b>IDE</b>	Integrated Development Environment
<b>BLDC</b>	Động cơ một chiều không chổi than (Brushless DC Motor).
<b>QUADCOPTER</b>	Máy bay 4 động cơ

## Chương 0: TÓM TẮT LUẬN VĂN

### Tóm tắt luận văn

Luận văn tốt nghiệp tìm hiểu về đề tài điều khiển quadcopter, bài viết nêu lên các phương pháp cơ sở lý thuyết, quá trình thực hiện, kết quả thực hiện, cũng như kết luận và hướng phát triển

**Chương 1:** Trong chương này nói về lý do chọn đề tài, mục tiêu của đề tài là gì, nhiệm vụ của đề tài cũng như đối tượng và phạm vi nghiên cứu.

**Chương 2:** Tổng quan và phương pháp

Nhóm em nêu lên Quadcopter, mô hình của Quadcopter, bàn về các mục tiêu chính của đề tài, nêu các tính năng và thiết kế mô hình. Bên cạnh đó là giải thích một số công cụ và phần mềm hỗ trợ cho đề tài, các chuẩn giao tiếp trong mô hình, v.v...

**Chương 3:** Quá trình thực hiện

Ở chương này em nêu những công việc mà nhóm đã thực hiện trong suốt học kỳ 212 để cho thấy khối lượng công việc cũng như những công sức của em bỏ ra.

**Chương 4 :** Kết quả thực hiện

Trong chương này nhóm em đưa ra những kết quả đạt được trong suốt quá trình làm luận văn.

**Chương 5 :** Kết luận và hướng phát triển

Chương này nói về những kết quả đạt được cũng như hạn chế và hướng phát triển.

**Chương 6:** Tổng kết

Ở chương này em nêu lên những điều đạt được, những hạn chế và hướng phát triển của luận văn.

## LỜI CẢM ƠN

Lời đầu tiên, em xin gửi lời cảm ơn chân thành và sự tri ân sâu sắc đến quý thầy cô của Trường Đại học Bách Khoa Tp HCM, và chúng em cũng xin cảm ơn chân thành đến thầy Hoàng Minh Trí là người đã giúp em hoàn thành khóa luận này một cách trọn vẹn nhất, em chúc thầy luôn mạnh khỏe để tiếp tục dẫn dắt những thế hệ đàn em tiếp tục trở thành những kỹ sư tương lai.

Trong suốt thời gian học tập tại trường Đại học Bách Khoa TP.HCM, Ban giám hiệu nhà trường, Khoa Điện – Điện tử và Bộ môn Tự Động đã tạo điều kiện cho chúng em học tập và thực hành cũng như các thầy cô đã tận tình giảng dạy truyền đạt kiến thức và kinh nghiệm quý báu cho chúng em. Xin chân thành cảm ơn thầy cô, đây chính là những hành trang vững chắc để em tiếp tục học tập và làm việc.

Đi qua những năm tháng Bách Khoa, ta mới biết tuổi trẻ đáng trân trọng đến mức nào, đã bao lần những khó khăn muôn gục ngã, nhưng vẫn cố gắng để vượt qua những thử thách.

Cảm ơn Bách Khoa!!!.

## CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN VỀ ĐỀ TÀI

### 1.1 Lý do chọn đề tài

Với sự tiến bộ của khoa học kỹ thuật, bầu trời ngày càng được mọi người quan tâm và chú ý đến, hàng năm có hàng trăm đề tài nghiên cứu khoa học về UAV(máy bay không người lái). UAV (Unmanned aerial vehicle) là tên gọi chỉ chung cho các loại máy bay mà không có người lái ở buồng lái, hoạt động tự lập và thường được điều khiển từ xa từ trung tâm hay máy điều khiển. Theo sự phát triển công nghệ hiện có các dạng UAV: Máy bay theo nghĩa truyền thống được trang bị hệ thống điều khiển và lái tự động, được gọi là UAS (unmanned aircraft system), xuất hiện từ những năm 1950 và đã từng phục vụ việc do thám và trinh sát chiến trường. Loại tổ hợp máy bay này có khả năng tự động hóa các hoạt động của máy bay cao, không đòi hỏi những trang thiết bị hàng không đặc chủng, giá thành khai thác sử dụng và bảo trì hệ thống để phục vụ lâu dài rẻ, trong quân sự loại máy bay này có đặc tính tần công chớp nhoáng.

Phương tiện bay kiểu mới, được chế tạo rất đa dạng, có kích thước và công suất động cơ nhỏ đến trung bình, được gọi là Drone.

Các drone có lắp camera để quan sát, và thường được gọi là flycam. Để thuận tiện điều khiển thao tác thì drone có nhiều cánh quạt, thường là 4.

Ngày nay Quadcopter được ứng dụng rất nhiều trong cuộc sống, các ứng dụng của Quadcopter trong cuộc sống là:

#### 1. Chụp ảnh, quay phim giải trí từ trên cao

Ứng dụng nổi bật nhất mà khi nhắc đến máy bay điều khiển Drone hay Flycam đó chính là tính năng chụp ảnh, quay phim từ trên cao. Vượt trội hơn so với phương thức chụp ảnh truyền thống, chụp ảnh bằng flycam mang lại cho người chụp và người xem những góc chụp mới lạ và độc đáo. Công dụng của máy bay điều khiển Drone đem lại cho những người có đam mê với công nghệ và nghệ thuật quay phim, chụp ảnh những shoot hình hay thước phim vô cùng độc đáo mà không có bất kỳ thiết bị quay phim, chụp ảnh truyền thống nào có thể làm được.

## 2. Vận chuyển đồ cứu trợ, cứu người

Ở bất kì đâu cũng vậy, thiên tai luôn là mối hiểm họa không thể lường trước được, và đôi khi địa hình khó khăn khiến đội cứu hộ không thể tiếp cận được với những người đang mắc kẹt và thiếu thốn về lương thực, điều kiện trang thiết bị y tế. Đã có rất nhiều trận động đất, lũ lụt khiến cho một khu vực bị cô lập hoặc những vụ sập mỏ khiến người ngoài không thể chui vào bên trong để đưa đồ tiếp tế. Vậy nên, sử dụng thiết bị drone sẽ khiến cho việc hỗ trợ hiệu quả hơn và an toàn hơn cho đội cứu hộ. Công ty hàng không Matternet hiện đang xây dựng một mạng lưới để vận chuyển thuốc đến các khu vực gặp nguy hiểm. Google mới đây cũng tiết lộ về Project Wing dùng drone để thực hiện mục đích tương tự.

## 3. Giúp đỡ nông dân

Thông thường việc đồng áng luôn được những người nông dân ngày ngày kiểm tra, chăm sóc, nhưng không phải lúc nào thời tiết cũng hỗ trợ những người nông dân thực hiện công việc này. Chính vì vậy, đã có rất nhiều nông dân sử dụng máy bay điều khiển drone để kiểm tra ruộng vườn trong mọi điều kiện thời tiết khác nhau. Ngoài việc kiểm tra, một số thiết bị đo hồng ngoại hay những cảm biến đặc biệt khác có thể được lắp trên drone hỗ trợ theo dõi sự tăng trưởng của động, thực vật. Bên cạnh đó, Drone có thể thay người nông dân phun thuốc sâu, phân bón, tưới nước,... giúp người dân hạn chế tiếp xúc với hóa chất gây hại sức khỏe. Với những nông trại có diện tích lớn, điều này là vô cùng hữu ích.

### 1.2 Nhiệm vụ đề tài:

Với đề tài này em có 2 nhiệm vụ là nhiệm vụ thứ nhất là nghiên cứu chế tạo được mạch điều khiển của 1 quadcopter hoạt động tốt, và nhiệm vụ thứ 2 là nghiên cứu chế tạo được mạch giám sát có tính năng giám sát được vị trí, dung lượng Pin, đ áp ứng của bộ điều khiển, ngày giờ của Quadcopter.

### Về Phần điều khiển:

- 1.Tìm hiểu về board mạch Arduino UNO R3.
- 2.Tìm hiểu về mạch cảm biến góc quay và gia tốc MPU6050.
- 3.Tìm hiểu về động cơ không chổi than A2112.
- 4.Bộ điều khiển động cơ không chổi than.
- 5.Tìm hiểu về tay cầm MC6C và MR7RB.

5.Tìm hiểu về thuật toán PID, và thuật toán điều khiển Quadcopter.

**Về Phần giám sát:**

- 1.Tìm hiểu về board mạch stm32f103c8t6.
- 2.Tìm hiểu về mạch GPS NEO 6M.
- 3.Tìm hiểu về NRF24L01.
- 4.Tìm hiểu về lập trình winform C#.
- 5.Tìm hiểu về LCD 1602.
- 6.Tìm hiểu về mạch giảm áp L2596.
- 7.Tìm hiểu về mạch UART to USB.

## CHƯƠNG 2: TỔNG QUAN HỆ THỐNG VÀ PHƯƠNG PHÁP THỰC HIỆN

### 2.1 Khái niệm Quadcopter

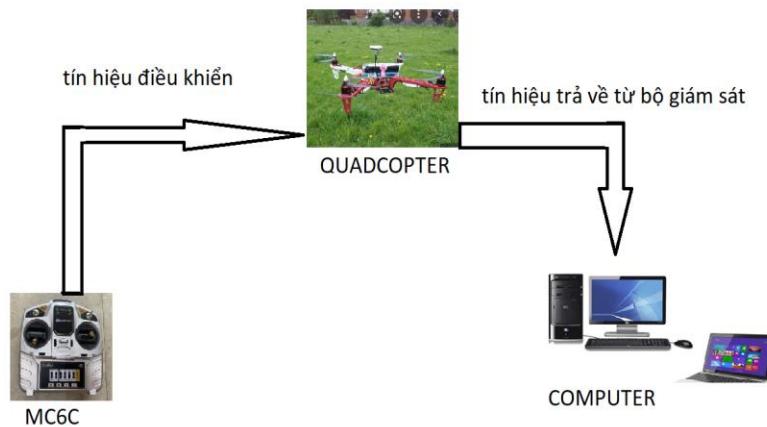
Quadcopter và Multicopter là một dạng máy bay sử dụng lực quay của motor để tạo dòng khí đẩy xuống, qua đó xuất hiện lực nâng khiến máy bay bay lên. Không giống như dòng máy bay cánh bằng, các loại máy bay phản lực sử dụng độ nghiêng của cánh để tạo lực nâng. Quadcopter phổ biến nhất là loại có bốn động cơ, bằng cách điều chỉnh lực quay từng motor, quadcopter có thể lật nhào di chuyển cực kỳ linh động. Trực thăng cũng được gọi là một multicopter khi nó sử dụng một cánh quạt làm động cơ chính tạo lực nâng, và một cánh quạt còn lại giúp cân bằng. Nhưng nó không phải là quadcopter do Quad có nghĩa là 4 - 4 động cơ.



Hình 2.1.1 Hình ảnh minh họa của Quadcopter

Mô hình tổng quan của Quadcopter sử dụng trong đề tài:

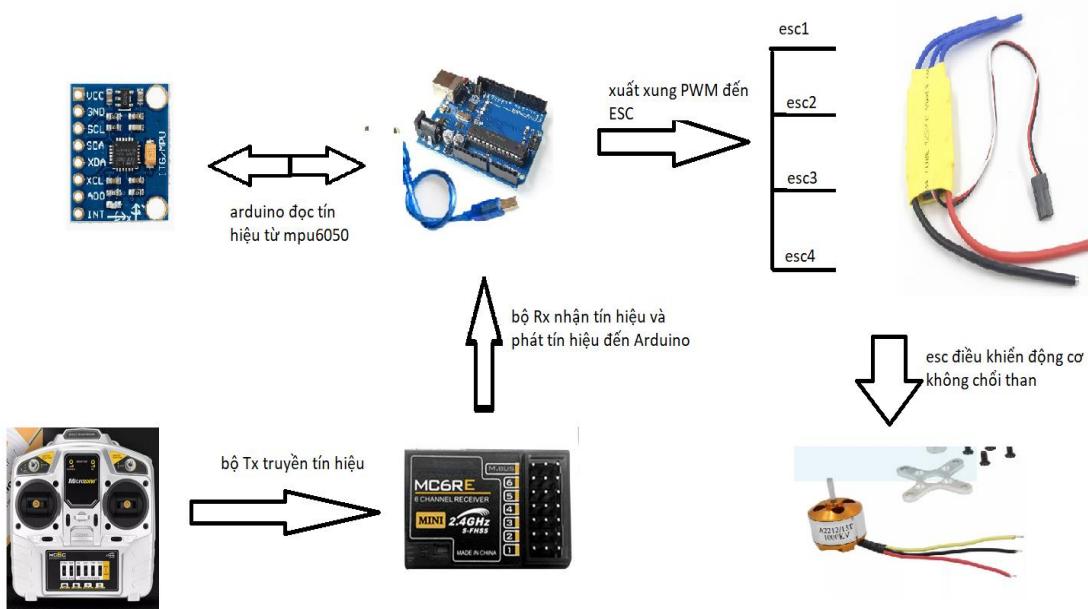
Sau đây em xin giới thiệu tổng quan đề tài này.



Hình 2.1.2 Hình ảnh minh họa tổng quan mô hình.

Với tín hiệu điều khiển được gửi từ bộ điều khiển MC6C đến bộ MC7C thông qua sóng vô tuyến với tần số 2.400 GHz đến 2.483 GHz. Bộ MC7C được gắn trực tiếp trên bộ điều khiển cân bằng bay. Bộ điều khiển cân bằng bay gửi tín hiệu đến bộ giám sát TX thông qua chuẩn UART. Bộ giám sát TX gửi tín hiệu đến bộ giám sát RX, và sau đó được hiển thị lên máy tính thông qua chuẩn UART.

## 2.2 Tổng quan của bộ điều khiển bay:

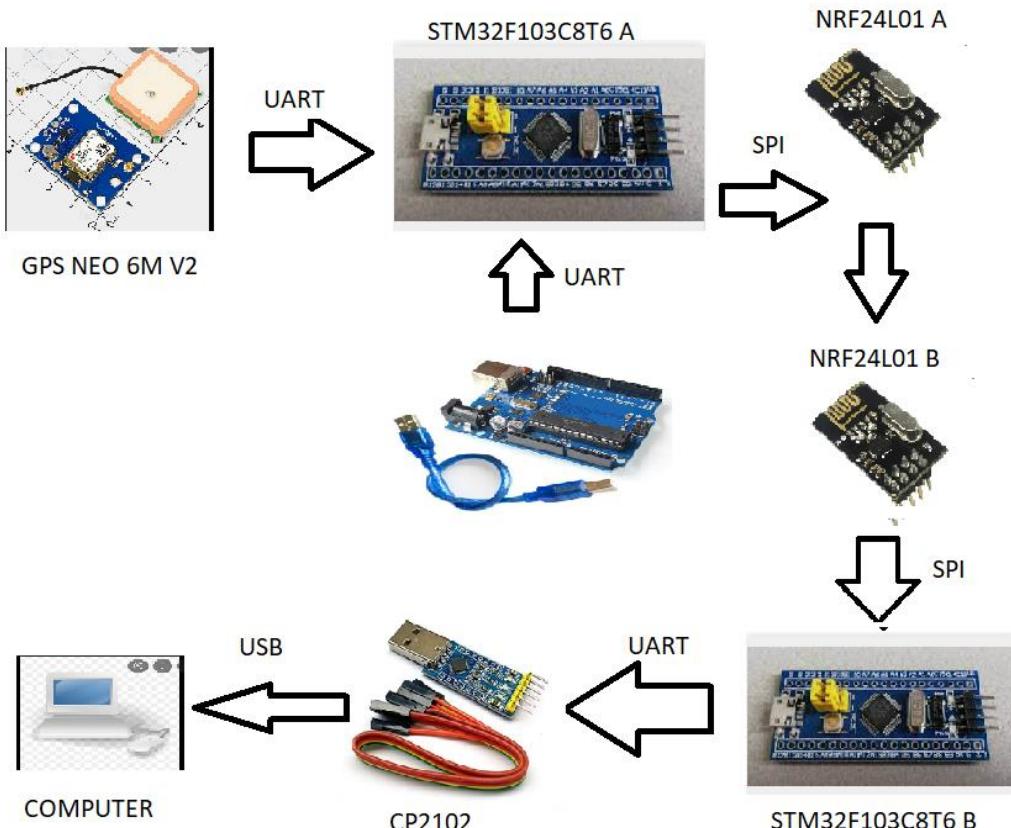


Hình 2.2 Hình ảnh minh họa tổng quan bộ điều khiển.

Trong đề tài này để chế tạo mạch điều khiển cho Quadcopter em sử dụng ARDUINO UNO R3. ARDUINO đọc tín hiệu từ MPU6050 qua giao tiếp I2C. Bộ truyền TX MC6C truyền tín hiệu đến bộ nhận RX MC7C. Board Arduino UNO đọc tín hiệu từ bộ RX

MC7C, đọc tín hiệu từ MPU6050 xử lý 2 tín hiệu đó và xuất xung PWM đến ESC. ESC có nhiệm vụ điều khiển động cơ không chổi than.

### 2.3 Tổng quan của bộ giám sát:



Hình 2.3 Hình ảnh minh họa tổng quan bộ giám sát.

Đối với bộ giám sát, em sử dụng board STM32f103c8t6 làm chức năng đọc giữ liệu từ mạch điều khiển qua chuẩn UART, đọc tín hiệu GPS qua chuyển UART, và truyền giữ liệu qua mạch NRF21L01A. Mạch NRF21L01B nhận dữ liệu xuống board STM32F103C8T6 B. Sau đó board này truyền dữ liệu lên máy tính theo chuẩn UART. Dữ liệu nhận được xuất ra màn hình được lập trình bằng WinForm C#.

Phương pháp thực hiện:

Để thực hiện mô hình trên em đã nghiên cứu những phương pháp sau đây:

1. Mạch điều khiển cân bằng bay.
2. Nghiên cứu cơ sở lý thuyết về cân bằng bay.
3. Nghiên cứu về bộ điều khiển PID, áp dụng PID trong cân bằng bay.

4. Nghiên cứu về Arduino IDE, các câu lệnh để viết chương trình ARDUINO.
5. Nghiên cứu về bộ lọc thông thấp để lọc tín hiệu từ MPU6050.
6. Nghiên cứu về bộ lọc bổ sung.
7. Nghiên cứu về mô hình toán học của Quadcopter.
8. Nghiên cứu về cách làm mạch in, và cách hàn linh kiện.
9. Mạch giám sát quadcopter.

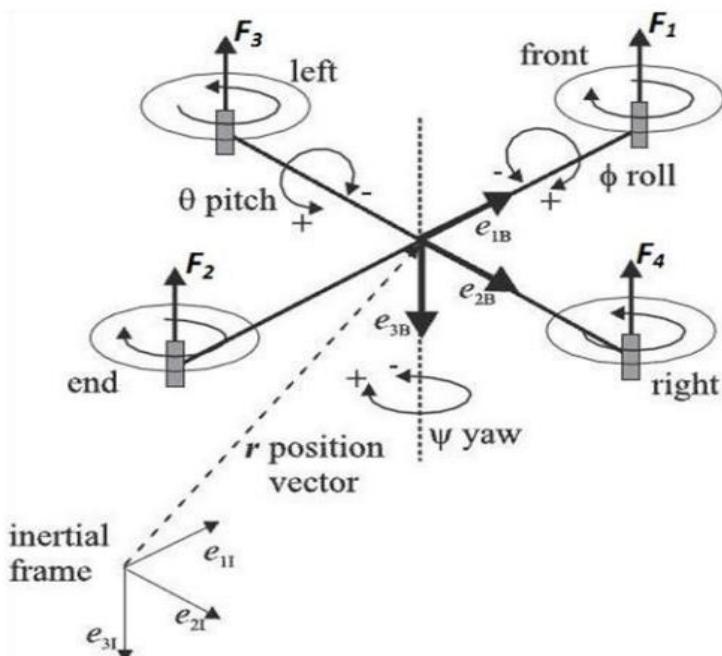
Để thực hiện bộ giám sát ta nghiên cứu các phương pháp sau đây:

1. Nghiên cứu về Keil C V5, và thư viện HAL để viết chương trình cho STM32F103C8T6.
2. Nghiên cứu về Visual Studio 2019, và Winform C# để lập trình giao diện.
3. Nghiên cứu về GMap.NET.Windows Form làm giao diện bản đồ.
4. Nghiên cứu về 1 số chuẩn giao tiếp SPI, UART để giao tiếp giữa các thiết bị.

## CHƯƠNG 3 CƠ SỞ LÝ THUYẾT

### 3.1 Nguyên lý cân bằng:

Máy bay Quadcopter có dạng chữ thập hay dấu cộng với bốn động cơ đặt ở bốn góc. Hai trục chéo của máy bay được đặt theo hai trục X, Y của hệ trục tọa độ Descartes. Mỗi động cơ kết hợp với cánh quạt trong Quadcopter sẽ tạo ra một lực đẩy và momen xoắn nhất định, bốn cánh quạt được chia thành hai nhóm có chiều quay ngược nhau, hai cánh đối diện nhau quay cùng chiều. Kết quả là momen xoắn bị triệt tiêu nếu 4 cánh quạt đều có cùng một vận tốc góc, do đó có thể làm cho máy bay không bị xoay tròn khi bay. Để cân bằng mô hình thì các động cơ phải được điều khiển sao cho mô hình có góc lệch so với trực chuẩn trong phạm vi cho phép.



Hình 3.1 Hình ảnh minh họa lý thuyết cân bằng Quadcopter.

### 3.2 Nguyên lý di chuyển Quadcopter.

- *Trạng thái lên, xuống theo phương thẳng đứng:*

Mô hình bay lên hoặc xuống theo phương thẳng đứng. Để mô hình bay lên theo

phương thẳng đứng, ta phải tăng đồng thời tốc độ của bốn động cơ và ở cùng một tốc độ, khi đó sẽ sinh ra tổng lực có phương trùng với trục z và giúp nâng mô hình bay sẽ bay lên dọc theo trục z, để máy bay hạ xuống theo phương thẳng đứng ta cần giảm tốc độ của bốn động cơ và tốc độ bốn động cơ là như nhau.



Hình 3.2.1 Hình mô tả trạng thái quadcopter khi di chuyển lên xuống

#### **Trạng thái nghiêng qua trái, nghiêng qua phải:**

Mô hình bay sẽ nghiêng bên trái hoặc nghiêng bên phải. Để mô hình nghiêng bên trái ta thực hiện đồng thời tăng tốc độ của cặp động cơ phía bên phải và giảm tốc độ của cặp động cơ bên trái sao cho tổng lực nâng không thay đổi. Khi đó tổng lực nâng mô hình của các cánh quạt không còn theo phương thẳng đứng mà sẽ nằm theo phương chuyển động về phía bên trái. Để mô hình nghiên về phía bên phải ta là tương tự.



Hình ảnh 3.2.2 Hình ảnh mô tả quadcopter khi di chuyển qua trái

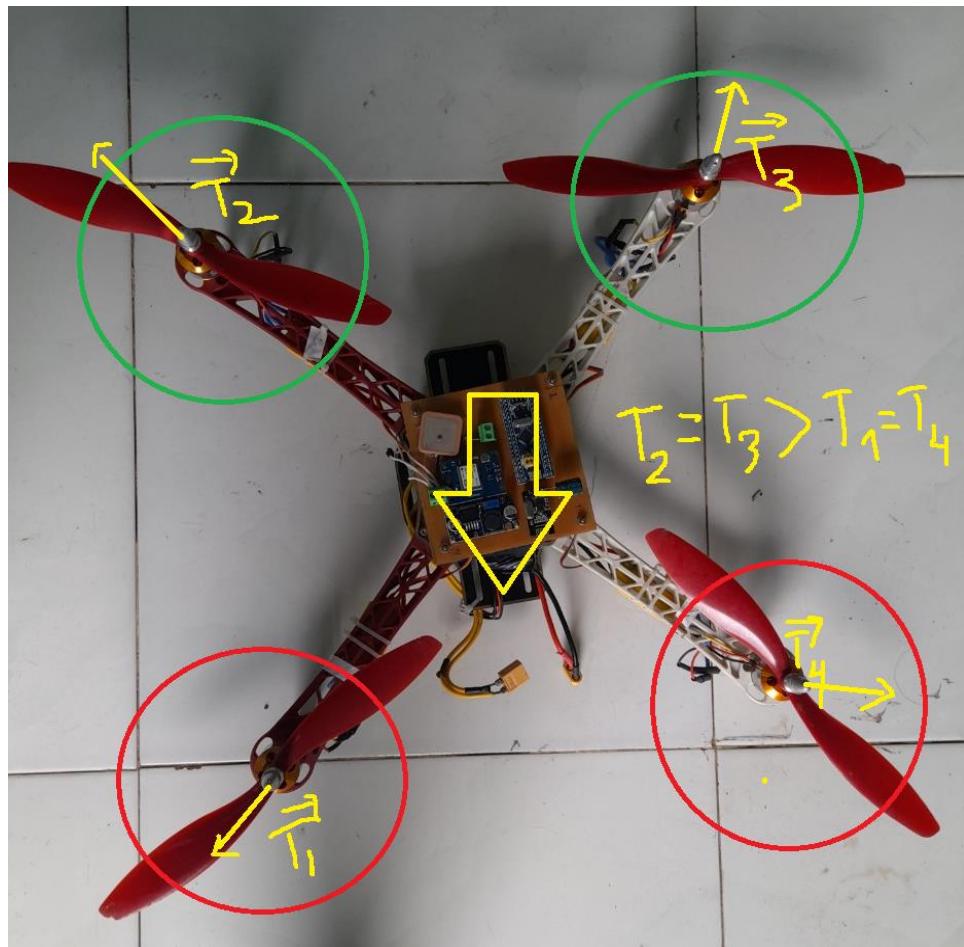


Hình ảnh 3.2.3 Hình ảnh mô tả quadcopter khi di chuyển qua phải

- **Trạng thái nghiêng về trước và sau:**

Mô hình bay sẽ bay về phía trước hoặc lùi về phía sau. Để mô hình bay về phía trước Ta thực hiện đồng thời tăng tốc độ của cặp động cơ phía sau và giảm tốc độ của động

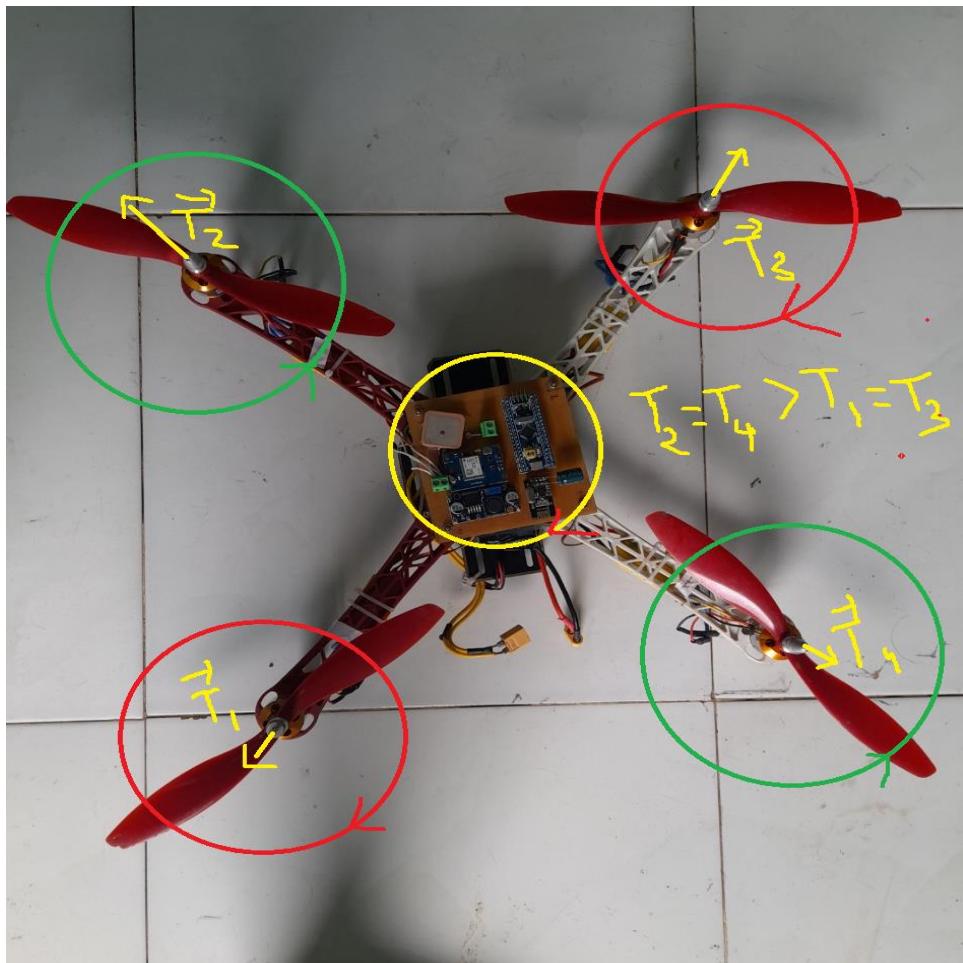
cơ của cặp động cơ phía trước sau cho tổng lực nâng không đổi. Khi đó tổng mô hình của cánh quạt không còn theo phương thẳng đứng mà sẽ nằm theo phương chuyển động về phía trước. Để mô hình lùi về phía sau ta làm tương tự.



Hình 3.2.4 Hình ảnh minh họa động cơ chuyển động về sau.

- **Trạng thái mô hình quay cùng chiều kim đồng hồ và ngược chiều kim đồng hồ.**

Mô hình sẽ quay ngược chiều kim đồng hồ, ta cần thực hiện tăng tốc độ của cặp động cơ quay cùng chiều kim đồng hồ và giảm tốc độ của động cơ quay ngược chiều kim đồng hồ sao cho tổng lực nâng không đổi. Để quay theo chiều kim đồng hồ ta làm ngược lại.

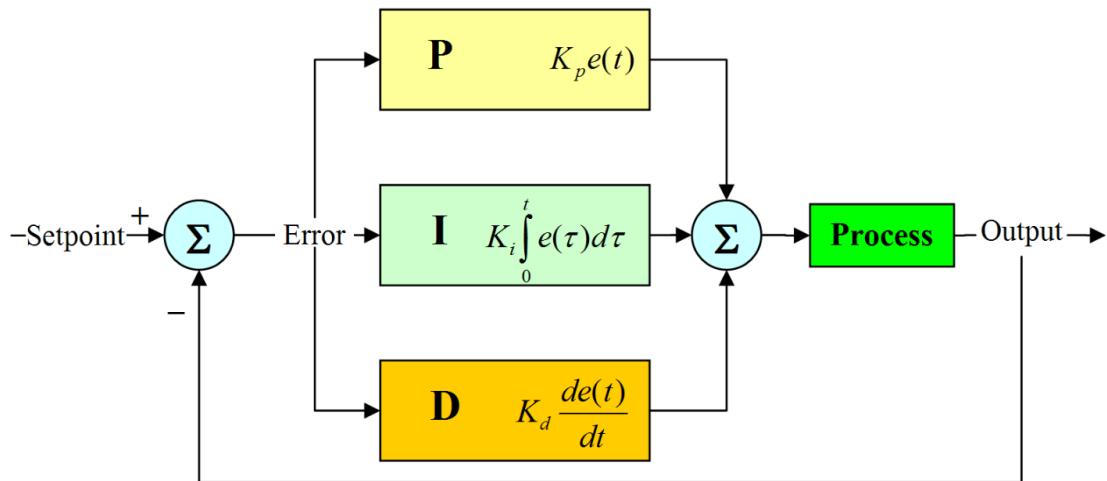


Hình 3.2.5 Hình ảnh minh họa của mô hình quay cùng chiều kim đồng hồ và ngược chiều kim đồng hồ.

### 3.3 Bộ điều khiển PID.

PID là một bộ điều khiển có cơ chế phản hồi vòng điều khiển, được viết tắt từ Proportional Integral Derivative. PID được kết hợp từ 3 bộ điều khiển tỉ lệ, tích phân, vi phân. Được sử dụng rộng rãi trong các hệ thống điều khiển công nghiệp, hệ thống điện, tự động hóa, điện tử – là bộ điều khiển được sử dụng phổ biến nhất.

Bộ điều khiển PID có khả năng tính toán giá trị sai số giữa thông số biến đổi và giá trị đặt. Điều chỉnh sai số thấp nhất có thể, đáp ứng vọt lố và sự dao động.



Hình 3.3.1 Bộ điều khiển PID

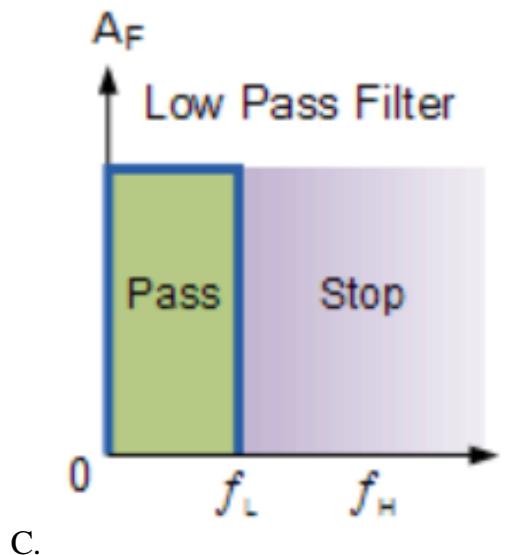
Trước tiên chúng ta phải xác định với nhau rằng, điều chỉnh vòng điều khiển chính là thay đổi các thông số kiểm soát của nó (Gồm có: độ lợi/dải tỉ lệ, độ lợi tích/reset, vi phân/tốc độ ) đến một mức giá trị đáp ứng điều khiển tối ưu của người dùng.

Để có thể điều chỉnh được PID là điều không phải dễ dàng. Ngay cả khi bạn đã có đầy đủ cả 3 thông số. Nói như vậy không có nghĩa là không có cách mà các phương pháp điều chỉnh này thường sẽ rất phức tạp. Dưới đây, có tổng hợp được một số cách thay đổi theo cách thông thường như sau:

- **Độ ổn định:** Sự bất ổn thường do dư thừa độ lợi đặc biệt khi có xuất hiện độ trễ lớn. Muốn có được độ ổn định cao bạn cần phải đảm bảo quá trình không xảy ra dao động.
- **Tối ưu hóa hành vi:** Để có thể làm được điều này, bạn nên thay đổi quá trình hoặc điểm đặt của bộ điều khiển. Hai yêu cầu cơ bản của phương pháp này chính là ổn định và tự điều chỉnh lệch nó sẽ phụ thuộc vào các tiêu chuẩn đặc biệt như: thời gian khởi động và thời gian xác lập. Các quá trình cần tối ưu hóa năng lượng bị tiêu hao.

### 3.4 Bộ lọc thông thấp (LPF):

Bộ lọc thông thấp là bộ lọc chỉ cho phép thành phần tần số thấp hơn tần số cắt đi qua mà tần số cao bị loại bỏ. Bộ lọc thông thấp được phát triển dựa trên lý thuyết của bộ lọc RC.

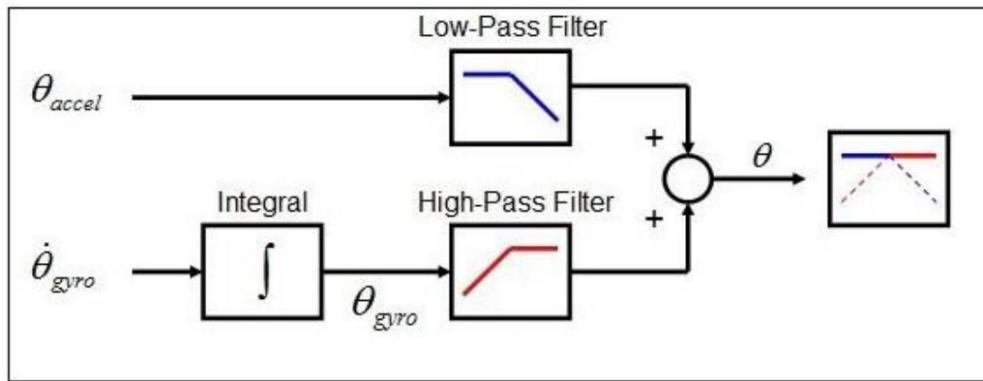


Hình 3.4.1 Hình ảnh minh họa cho bộ lọc thông thấp.

Khi sử dụng bộ lọc thông thấp thì tín hiệu ra có khuynh hướng bị chậm lại so với những thay đổi của tín hiệu vào. Nếu tín hiệu vào thay đổi càng nhanh (nghĩa là tần số càng cao) thì độ trễ tín hiệu ra càng thấy rõ và càng làm cho đầu ra khó đáp ứng thay đổi đó.

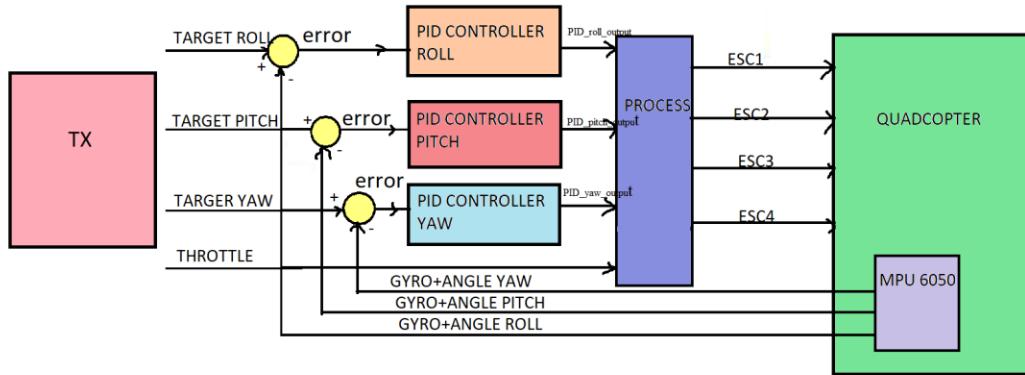
### 3.5 Bộ lọc bù (complementary filter).

Bộ lọc bù là một giải pháp hiệu quả cho IMU. Ta có thể hiểu một cách đơn giản bộ lọc bù bao gồm sự kết hợp giữa bộ lọc thông cao và bộ lọc thông thấp.



Hình 3.5.1 Hình ảnh minh họa bộ lọc thông bù.

### 3.6 Bộ điều khiển PID cân bằng cho góc roll, pitch:



Hình 3.6.1 Hình ảnh minh họa cho bộ điều khiển PID cân bằng góc Pitch và Yaw.

Tín hiệu được lên từ TX, sau đó sai số được tính dựa trên tín hiệu hồi tiếp về và tín hiệu gửi lên từ TX. Các sai số tương ứng được đưa vào các khối PID tương ứng. Các tín hiệu được tính ra từ khối PID được đưa vào khối PROCESS. Trong khối PROCESS ta thực hiện:

$$\text{ESC1} = \text{THROTTLE} + \text{PID\_roll\_output} - \text{PID\_pitch\_output} - \text{PID\_yaw}.$$

$$\text{ESC2} = \text{THROTTLE} + \text{PID\_roll\_output} + \text{PID\_pitch\_output} + \text{PID\_yaw}.$$

$$\text{ESC3} = \text{THROTTLE} - \text{PID\_roll\_output} - \text{PID\_pitch\_output} + \text{PID\_yaw}.$$

$$\text{ESC4} = \text{THROTTLE} - \text{PID\_roll\_output} + \text{PID\_pitch\_output} - \text{PID\_yaw}.$$

Các tín hiệu được phát ra từ khói process là các tín hiệu PWM để điều khiển động cơ.

### 3.7 Nguyên lý hoạt động của GPS:

Khi nói đến GPS (hệ thống định vị toàn cầu), người ta thường nghĩ tới máy thu, chẳng hạn máy thu GPS trên máy bay, tàu thủy, xe hơi, hay hầu hết smartphone cũng tích hợp GPS để định vị, và cùng với các ứng dụng bản đồ chúng trở thành công cụ dẫn đường hữu hiệu



Hình 3.7.1 Hình ảnh minh họa cho nguyên lý hoạt động của GPS.

GPS là gì? Thực tế GPS là một hệ thống định vị vận hành dựa vào 27 vệ tinh (trong đó có 3 vệ tinh dự phòng) chuyển động trên các quỹ đạo quanh trái đất, do Mỹ phát triển ban đầu cho mục đích quân sự, nay đã mở rộng ra cho cả dân sự. Các vệ tinh được bố trí sao cho tại bất kỳ thời điểm nào và ở nơi nào trên mặt đất, cũng có thể thấy ít nhất 4 vệ tinh.

Vệ tinh phát ra các tín hiệu gồm vị trí của chúng và thời điểm phát tín hiệu. Nhiệm vụ của máy thu GPS là xác định vị trí của 4 vệ tinh, tính toán khoảng cách tới các vệ tinh

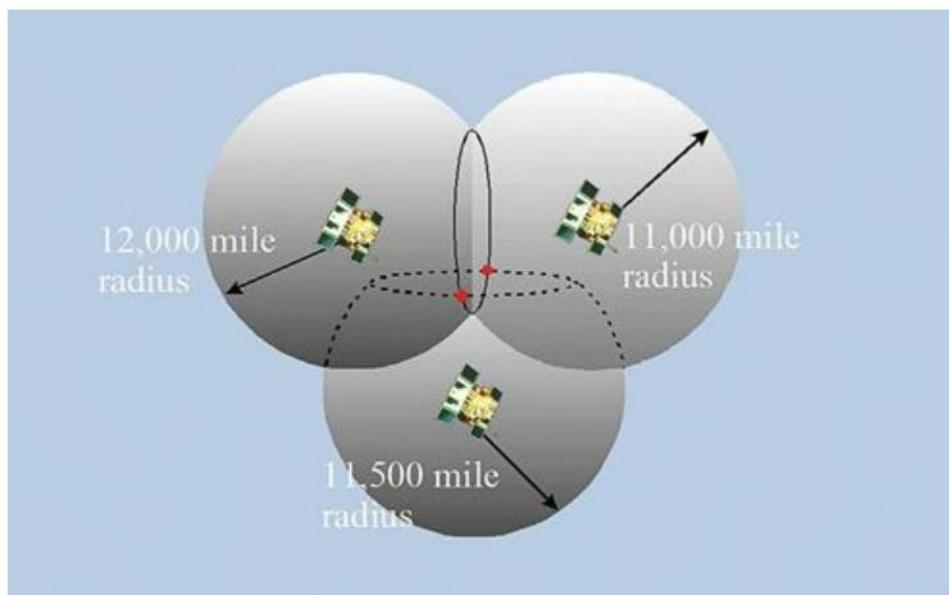
để từ đó tự xác định vị trí của chính nó theo công thức: Quãng đường = Vận tốc x Thời gian.

Nguyên lý hoạt động xác định vị trí bằng GPS

Giả sử bạn đang ở một nơi xa lạ và muốn biết mình đang ở đâu. Bạn hỏi thăm một người dân địa phương và được biết đang cách Vũng Tàu 60 km. Thông tin nhận được mới chỉ cho bạn biết đang ở đâu đó trên vòng tròn tâm Vũng Tàu, bán kính 50 km.

Một người khác nói bạn cách Biên Hòa 40 km. Giờ thì đã rõ hơn, bạn biết mình đang ở một trong hai vị trí là giao nhau của 2 vòng tròn. Người thứ ba cho biết, bạn đang cách TP.HCM 20 km. Và bạn đã có thể xác định vị trí hiện tại của mình là Nhơn Trạch – nơi giao của 3 vòng tròn.

Nguyên lý trên cũng được áp dụng tương tự để xác định vị trí trong hệ thống GPS, lấy điểm giao nhau của 3 mặt cầu trong không gian 3 chiều, thay vì là 3 đường tròn.



Hình 3.7.2 Hình ảnh minh họa cho nguyên lý hoạt động của GPS.

## CHƯƠNG 4 QUÁ TRÌNH THỰC HIỆN

### 4.1 Tổng quan các phần cứng sử dụng trong luận văn.

Giới thiệu các linh kiện và thông số:

1.Arduino UNO R3.

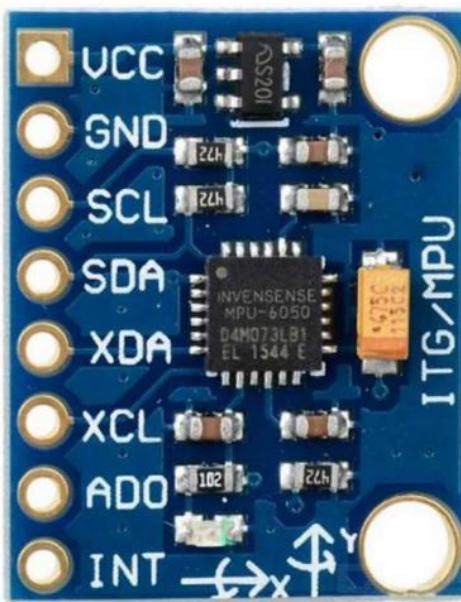


Hình 4.1.1 Hình ảnh minh họa Arduino UNO R3

Vì điều khiển	ATmega328 họ 8 bit
Điện áp hoạt động	5~12V DC (khuyên dùng)
Tần số hoạt động	16 MHz
Dòng tiêu thụ	Khoảng 30mA
Điện áp vào giới hạn	19V DC
Số chân Digital I/O	14 (6 chân PWM)
Số chân Analog	6 (độ phân giải 10bit)
Dòng tối đa trên mỗi chân I/O	30 mA
Dòng ra tối đa (5V)	500 mA
Dòng ra tối đa (3.3V)	50 mA

Bộ nhớ flash	32 KB (ATmega328) với 0.5KB dùng bởi bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Khối lượng	grams

### 1. Cảm Biến gia tốc GY-521 6DOF IMU MPU6050:



Hình 4.1.2 Hình ảnh minh họa cho MPU6050

Điện áp sử dụng: 3~5VDC

Điện áp giao tiếp: 3~5VDC

Chuẩn giao tiếp: I2C

Giá trị Gyroscopes trong khoảng: +/- 250 500 1000 2000 degree/sec

Giá trị Acceleration trong khoảng: +/- 2g, +/- 4g, +/- 8g, +/- 16g

## 2. Frame kit quadcopter F450:



Hình 4.1.3 Hình ảnh minh họa Frame Kit F450

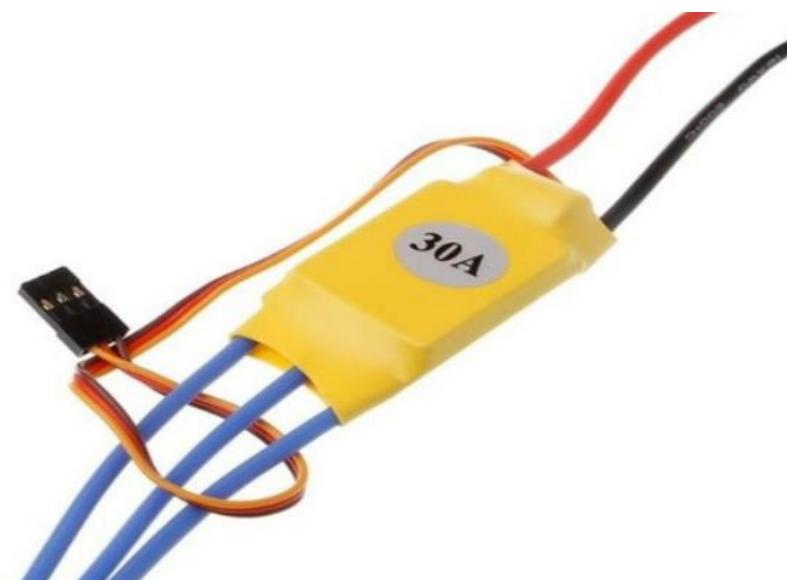
Khung: Flame Wheel f450 (F450)

Khung trọng lượng: 282g

Chiều dài đường chéo: 450mm

Trọng lượng cát cánh: 800g ~ 1600g.

## 3. Bộ điều khiển động cơ không chổi than ESC:

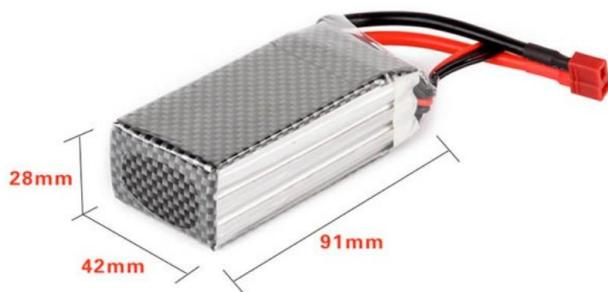


Hình 4.1.4 Hình ảnh minh họa bộ điều khiển động cơ không chổi than  
Thông số kỹ thuật  
Kích thước 32 x 24 x 7  
Điện áp hoạt động 4-16v

Dòng liên tục tối đa 30a

Dòng điện cực đại 40a.

#### 4. Pin lipo 3s:



Hình 4.1.5 Hình ảnh minh họa cho pin Lipo 3s.

Thông số kỹ thuật:

Điện áp: 11.1V / 3S (Pin sạc đầy 12.6V)

Dung lượng: 2200mAh.

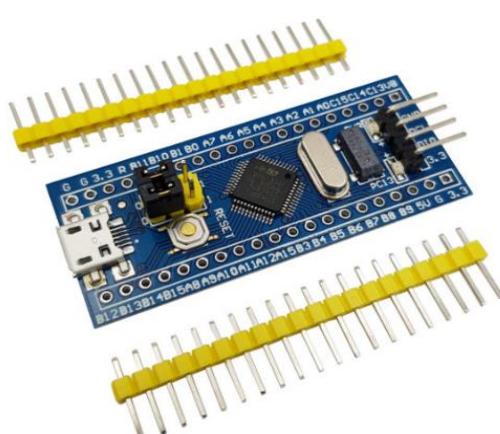
Dòng xả liên tục: 25C / 55A (Dòng xả tối đa  $25 \times 2200 / 1000 = 55A$ )

MAX ở ngưỡng: 50C / 110A.

Kích thước: 106mm x 33mm x 23mm (Dài x Rộng x Cao).

Trọng lượng: 166g.

#### 5. Board mạch STM32F103C8T6:



Hình 4.1.6 Hình ảnh minh họa STM32F103C8T6

Vì điều khiển: STM32F103C8T6.

Điện áp hoạt động: 3.3VDC

Điện áp cấp 5VDC qua cổng Micro USB sẽ được chuyển đổi thành 3v3 qua IC nguồn và cấp cho Vị điều khiển chính.

Có tích hợp sẵn thạch anh 8Mhz.

Ra chân đầy đủ tất cả các GPIO và giao tiếp: CAN, I2C, SPI, UART / USART, USB.

Có Led trạng thái nguồn.

Có led PC13.

Có nút Reset.

Kích thước: 5.3cm x 2.2cm.

## 6. Module định vị GPS NEO 6M.



Hình 4.1.7 Hình ảnh mạch định vị GPS NEO 6M

Mạch định vị GPS GY-NEO 6M, nguồn cung cấp 3V-5V

Mẫu: GY-GPS6MV2

Mô-đun với ăng-ten băng sứ, tín hiệu mạnh

EEPROM power-down lưu dữ liệu tham số cấu hình

Với pin dự phòng dữ liệu

Chỉ báo tín hiệu LED

Kích thước anten 12\*12mm

Kích thước module 23mm\*30mm

Lắp đặt khẩu độ 3mm

Tốc độ baud mặc định: 9600.

#### 7. Bộ phát MC6C:



Hình 4.1.8 Hình ảnh minh họa mạch phát MC6C

Khoảng cách truyền 400 – 800m.

Thông số tay điều khiển - Tần số: 2.400 GHz đến 2.483 GHz-

Công suất ≤ 100mW

Dòng hoạt động ≤ 120mA

Nguồn cung cấp : 8.4V – 12V

#### 8. Bộ thu MC7C



Hình 4.1.9 Hình ảnh minh họa board thu MC7RB

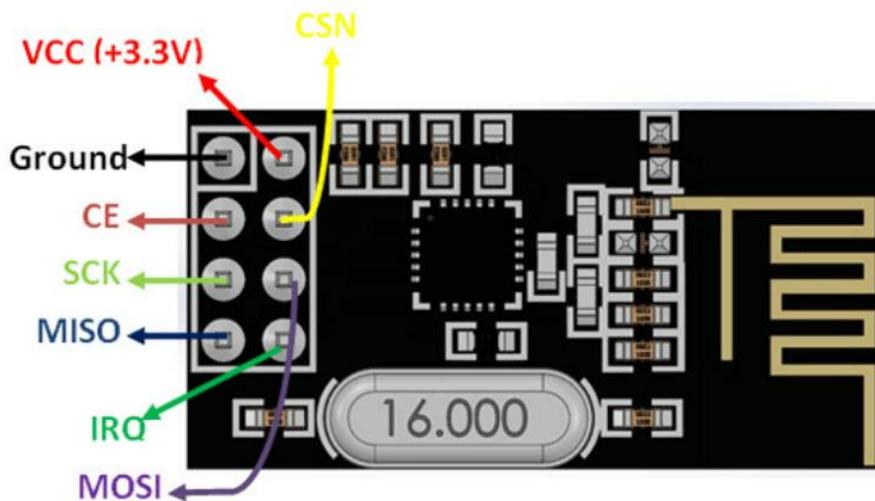
Tần số: 2.400 GHz đến 2.483 GHz

Nguồn 4.8V – 6V

Trọng lượng : 9.6g

Kích thước: 16x14x13cm

## 9. Module thu phát NRF24L01+ PA.



Hình 4.1.10 Hình ảnh minh họa mạch thu phát radio nrf24l01.

Điện thế hoạt động: 1.9V – 3.6V

Có sẵn athena sứ 2.4GHz.

Truyền được 100m trong môi trường mở với 250kbps baud.

Tốc độ truyền dữ liệu qua sóng: 250kbps to 2Mbps.

Tự động bắt tay (Auto Acknowledge).

Tự động truyền lại khi bị lỗi (auto Retransmit).

Multiceiver – 6 Data Pipes.

Bộ đệm dữ liệu riêng cho từng kênh truyền nhận: 32 Byte separate TX and RX FIFOs.

Các chân IO đều chịu được điện áp vào 5V.

Lập trình được kênh truyền sóng trong khoảng 2400MHz đến 2525MHz (chọn được 125 kênh).

Thứ tự chân giao tiếp: GND, VCC, CS, CSN, SCK, MOSI, MISO, IQR.

## 10. Mạch giảm áp Lm2569.



Hình 4.1.11 Hình ảnh minh họa mạch giảm áp Lm2569

Thông số kỹ thuật:

Điện áp đầu vào: Từ 3V đến 30V.

Điện áp đầu ra: Điều chỉnh được trong khoảng 1.5V đến 30V.

Dòng đáp ứng tối đa là 3A.

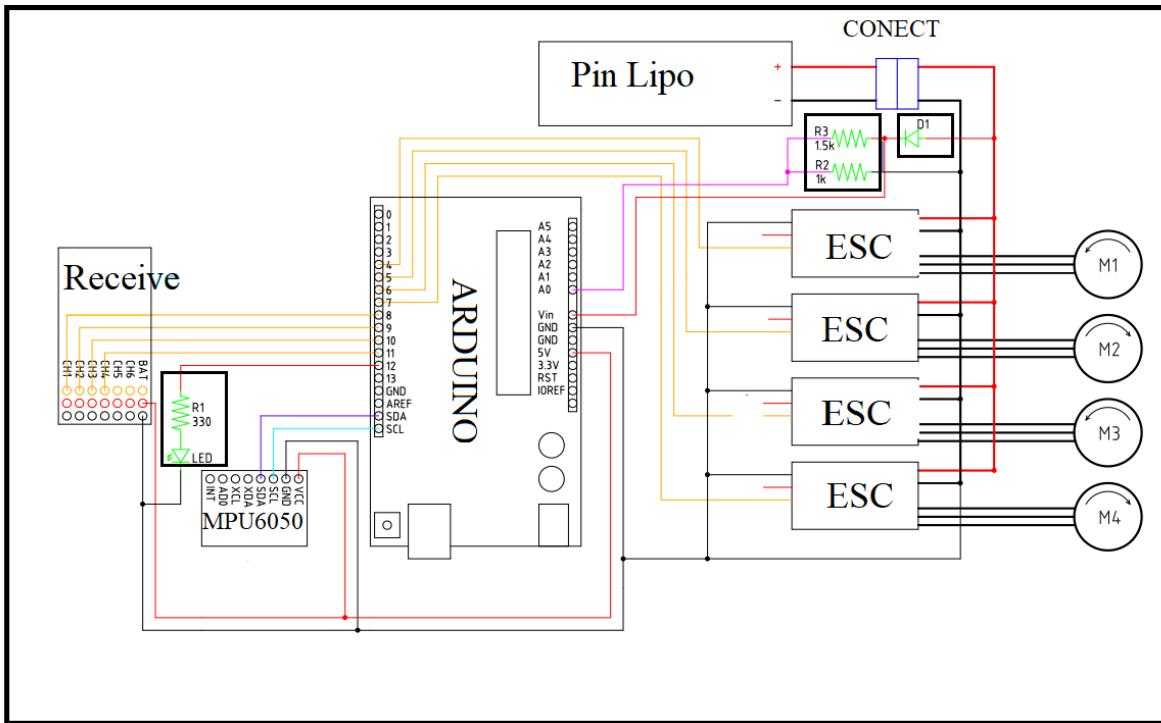
Hiệu suất: 92%

Công suất: 15W

Kích thước: 45 (dài) \* 20 (rộng) \* 14 (cao) mm

## 4.2 Mạch cân bằng bay.

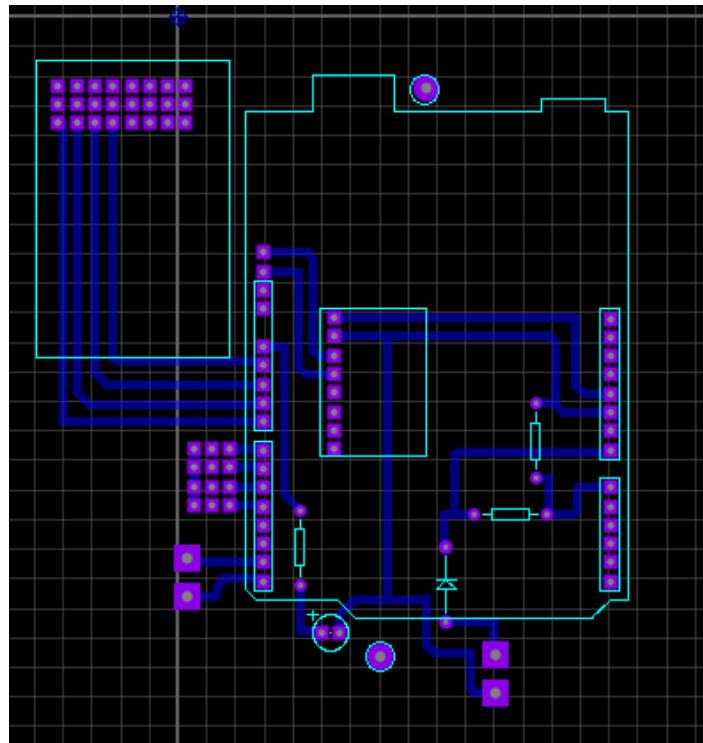
### 1. Sơ đồ nguyên lý:



Hình 4.2.1 Hình ảnh thể hiện sơ đồ nguyên lý của mạch điều khiển.

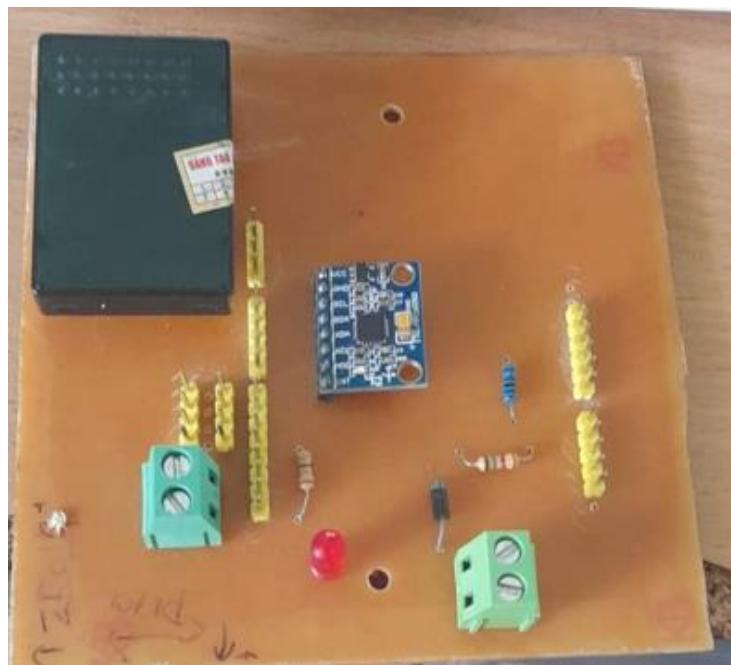
Bảng thiết kế sơ đồ nguyên lý được thể hiện như hình vẽ với bộ xử lý trung tâm arduino có chức năng nhận và xử lý thông tin tính toán các thuật toán và điều khiển các thiết bị chấp hành là 4 ESC và 4 Motor. Cảm biến MPU6050 là cảm biến đo góc nghiêng của quadcopter. Bộ nhận tín hiệu Receiver nhận tín hiệu từ người điều. Pin Lipo là loại Pin có công suất lớn và tải trọng nhẹ để cấp nguồn cho hệ thống, và các phần tử khác là đèn LED, và trở.

## 2.Mạch PCB:



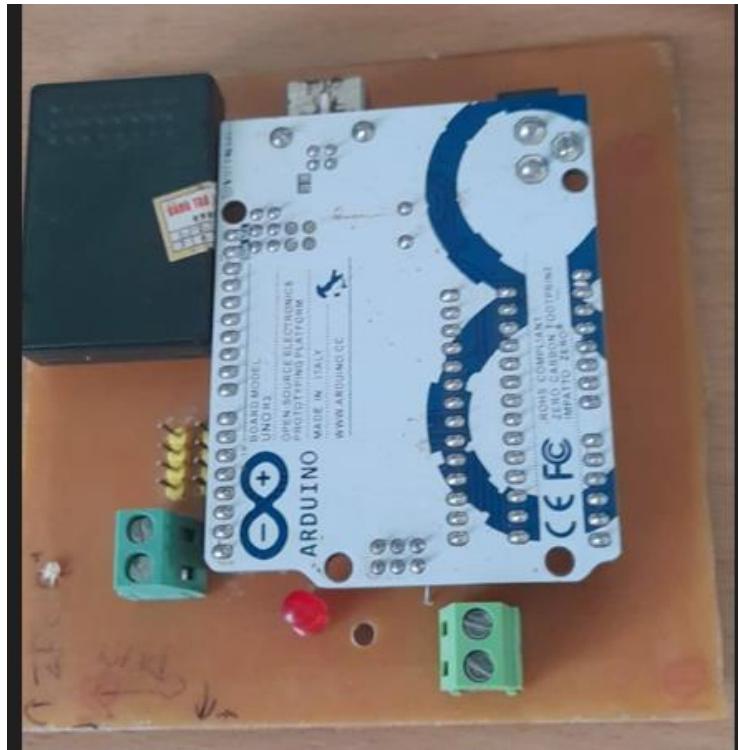
Hình 4.2.2 Hình ảnh mạch giám sát PCB được vẽ trên Proteus

Mạch PCB được thiết kế để tối ưu không gian khi đặt các phần tử lên 1 miếng đồng hình vuông 10cm x 10 cm.



Hình 4.2.3 Hình ảnh thực tế mạch điều khiển quadcopter (1).

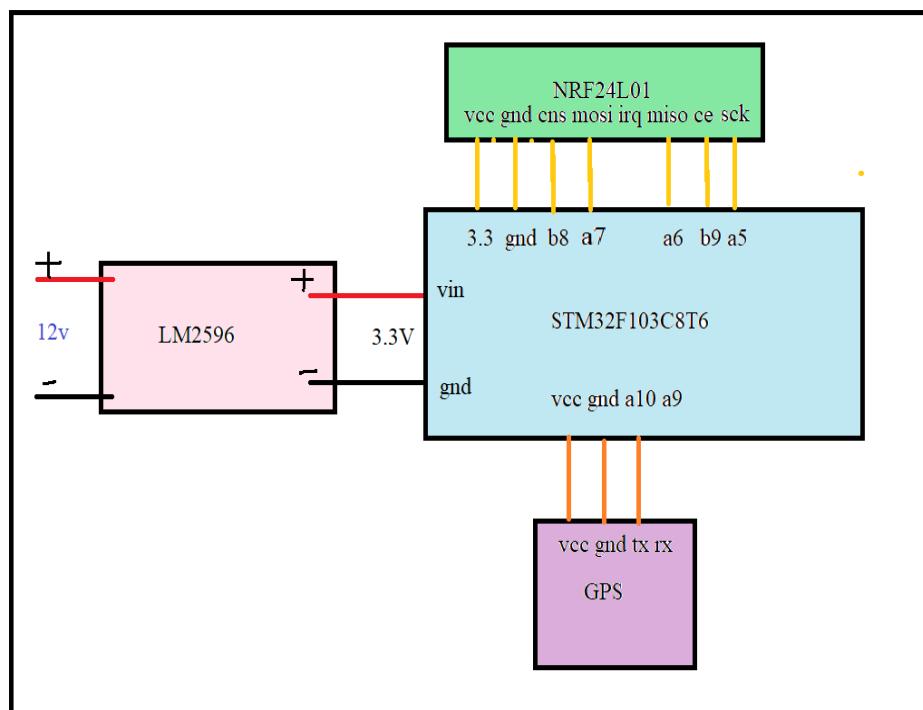
Thực tế sau khi chế tạo mạch điều khiển rất gọn nhẹ tối ưu được các phần tử.



Hình 4.2.3 Hình ảnh thực tế mạch điều khiển quadcopter (2).

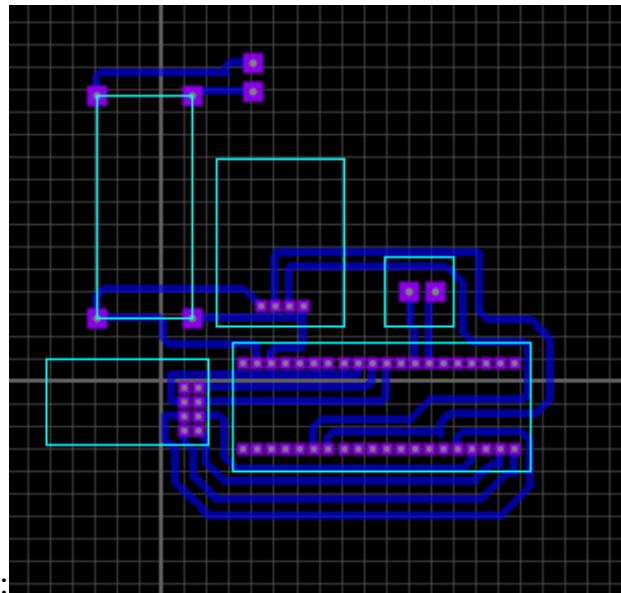
#### 4.3 Mạch giám sát TX:

Sơ đồ nguyên lý.



Hình 4.3.1 Sơ đồ nguyên lý mạch giám sát Tx.

Sơ đồ nguyên lý mạch giám sát Tx được thiết kế dựa trên ý tưởng cần 1 mạch có thể truyền thông số quadcopter về máy vi tính. Bao gồm stm32 là bộ vi điều khiển chính có chức năng nhận dữ liệu từ cảm biến GPS và truyền dữ liệu thông qua NRF. Mạch giảm áp LM2596 có chức năng hạ áp để cấp nguồn cho hệ thống.



Hình 4.3.2 Hình ảnh vẽ mạch PCB trên proteus.

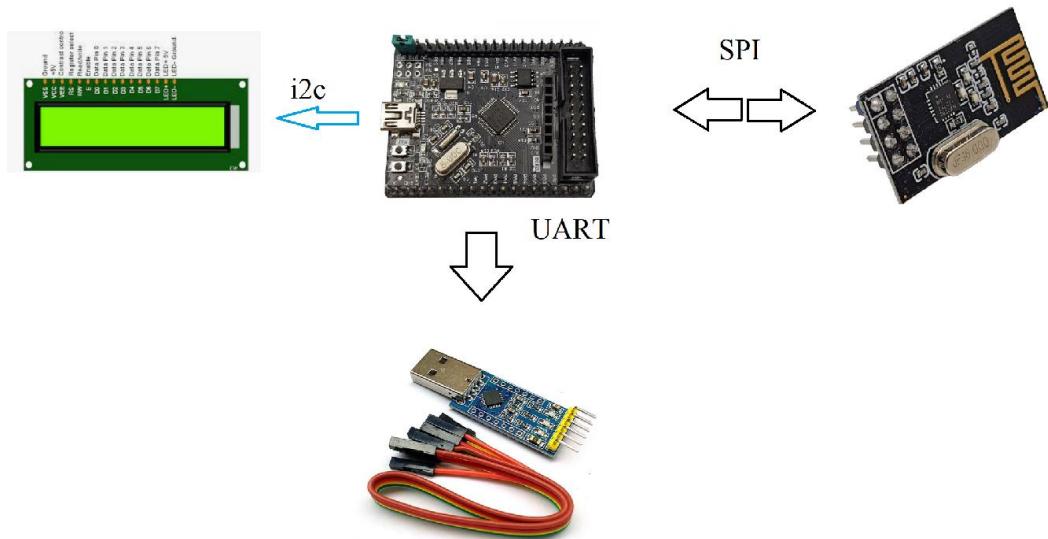
Mạch PCB được thiết kế tối ưu với các phần tử được đặt trên bảng mạch 10 cm x 10cm.



Hình 4.3.3 Hình ảnh thực tế của mạch giám sát TX

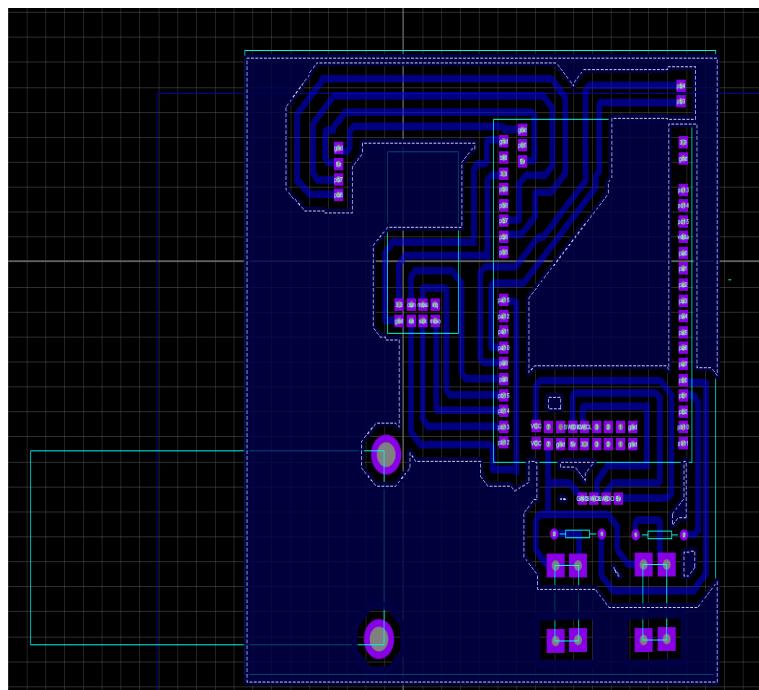
Hình ảnh mạch giám sát Tx gọn nhẹ sau khi được thực hiện xong

#### 4.4 Mạch giám sát Rx:



Hình 4.4.1 Sơ đồ nguyên lý mạch giám sát RX:

Mạch giám sát được thiết kế dựa trên ý tưởng cần 1 hệ thống nhận các dữ liệu truyền từ quadcopter và truyền chúng cho máy tính. Mạch được thiết kế có STM là vi điều khiển



Hình 4.4.2 Hình minh họa mạch PCB RX

Mạch PCB được thiết kế tối ưu với các phần tử được đặt trên bảng mạch 10 cm x 10cm



Hình 4.4.3 Hình ảnh thực tế mạch thu Rx

Mô tả: Dữ liệu giám sát được gửi về bộ thu sóng radio NRF24L01, tín hiệu được xử lý sau đó thể hiện trên LCD1602 và truyền data qua UART. Nút nhấn được sử dụng để tạo ngắt ngoài tác động lên vi điều khiển STM nhằm mục đích thay đổi dữ liệu thể hiện trên màn hình LCD.

#### 4.5 Giới thiệu về các công cụ phần mềm sử dụng trong luận văn.

##### *Phần mềm Arduino IDE là gì:*

Arduino IDE là một phần mềm mã nguồn mở chủ yếu được sử dụng để viết và biên dịch mã vào module Arduino.

Đây là một phần mềm Arduino chính thức, giúp cho việc biên dịch mã trở nên dễ dàng mà ngay cả một người bình thường không có kiến thức kỹ thuật cũng có thể làm được. Nó có các phiên bản cho các hệ điều hành như MAC, Windows, Linux và chạy trên nền tảng Java đi kèm với các chức năng và lệnh có sẵn đóng vai trò quan trọng để gỡ lỗi, chỉnh sửa và biên dịch mã trong môi trường.

Có rất nhiều các module Arduino như Arduino Uno, Arduino Mega, Arduino Leonardo, Arduino Micro và nhiều module khác.

Mỗi module chứa một bộ vi điều khiển trên bo mạch được lập trình và chấp nhận thông tin dưới dạng mã.

Mã chính, còn được gọi là sketch, được tạo trên nền tảng IDE sẽ tạo ra một file Hex,

sau đó được chuyển và tải lên trong bộ điều khiển trên bo.

Môi trường IDE chủ yếu chứa hai phần cơ bản: Trình chỉnh sửa và Trình biên dịch, phần đầu sử dụng để viết mã được yêu cầu và phần sau được sử dụng để biên dịch và tải mã lên module Arduino.

Môi trường này hỗ trợ cả ngôn ngữ C và C++.

### ***Arduino IDE hoạt động như thế nào:***

Khi người dùng viết mã và biên dịch, IDE sẽ tạo file Hex cho mã. File Hex là các file thập phân Hexa được Arduino hiểu và sau đó được gửi đến bo mạch bằng cáp USB. Mỗi bo Arduino đều được tích hợp một bộ vi điều khiển, bộ vi điều khiển sẽ nhận file hex và chạy theo mã được viết.

### ***Giới thiệu phần mềm KEIL C:***

Keil c uvision 5 là một phần mềm hỗ trợ cho người dùng trong việc lập trình cho vi điều khiển các dòng khác nhau (Atmel, AVR,...). Keil C giúp người dùng soạn thảo và biên dịch chương trình C hay cả ASM thành ngôn ngữ máy để nạp vào vi điều khiển giúp chúng ta tương tác giữa vi điều khiển và người lập trình.

### ***Keil c uvision 5 Full:***

Nếu các bạn tải bản Keil c uvision 5 Full tương ứng với nhu cầu của các bạn. Nếu dùng Keil c uvision 5 để lập trình cho ARM thì các bạn tải bản Keil c uvision 5 ARM, Nếu để lập trình cho Atmel thì tải bản Keil c uvision 5 8051.



Hình ảnh 4.5.1 Hình ảnh minh họa giới thiệu phần mềm Keil C.

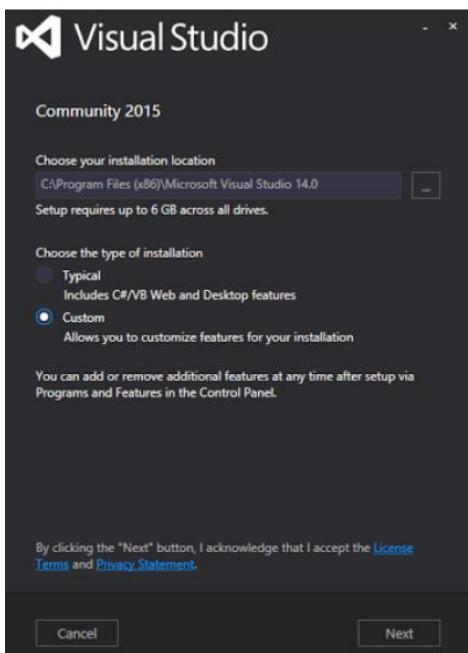
### ***Giới thiệu VISUAL STUDIO VÀ WINFORM C#.***

#### **Visual studio là gì?**

Visual studio là một phần mềm hỗ trợ đắc lực hỗ trợ công việc lập trình website. Công cụ này được tạo lên và thuộc quyền sở hữu của ông lớn công nghệ Microsoft. Năm 1997, phần mềm lập trình này có tên mã Project Boston. Nhưng sau đó, Microsoft đã

kết hợp các công cụ phát triển, đóng gói thành sản phẩm duy nhất.

Visual Studio là hệ thống tập hợp tất cả những gì liên quan tới phát triển ứng dụng, bao gồm trình chỉnh sửa mã, trình thiết kế, gỡ lỗi. Tức là, bạn có thể viết code, sửa lỗi, chỉnh sửa thiết kế ứng dụng dễ dàng chỉ với 1 phần mềm Visual Studio mà thôi. Không dừng lại ở đó, người dùng còn có thể thiết kế giao diện, trải nghiệm trong Visual Studio như khi phát triển ứng dụng Xamarin, UWP bằng XAML hay Blend vậy



Hình ảnh 4.5.2 Hình ảnh minh họa cho giới thiệu phần mềm visual studio.

### Winform là gì?

Winform là thuật ngữ mô tả một ứng dụng được viết dùng .NET FrameWork và có giao diện người dùng Windows Forms.

Mỗi màn hình windows cung cấp một giao diện giúp người dùng giao tiếp với ứng dụng. Giao diện này được gọi là giao diện đồ họa (*GUI*) của ứng dụng.

Là các ứng dụng windows chạy trên máy tính – mã lệnh thực thi ngay trên máy tính: Microsoft, Word, Excel, Access, Calculator, yahoo, Mail... là các ứng dụng Windows Forms.

### Ưu điểm các phần mềm trên Winform

Đa phần lập trình viên C#. NET nào cũng từng học/sử dụng Winform. Bởi vì: Giao diện kéo thả dễ sử dụng; Gắn các event cho các button chỉ cần double click, lại hỗ trợ

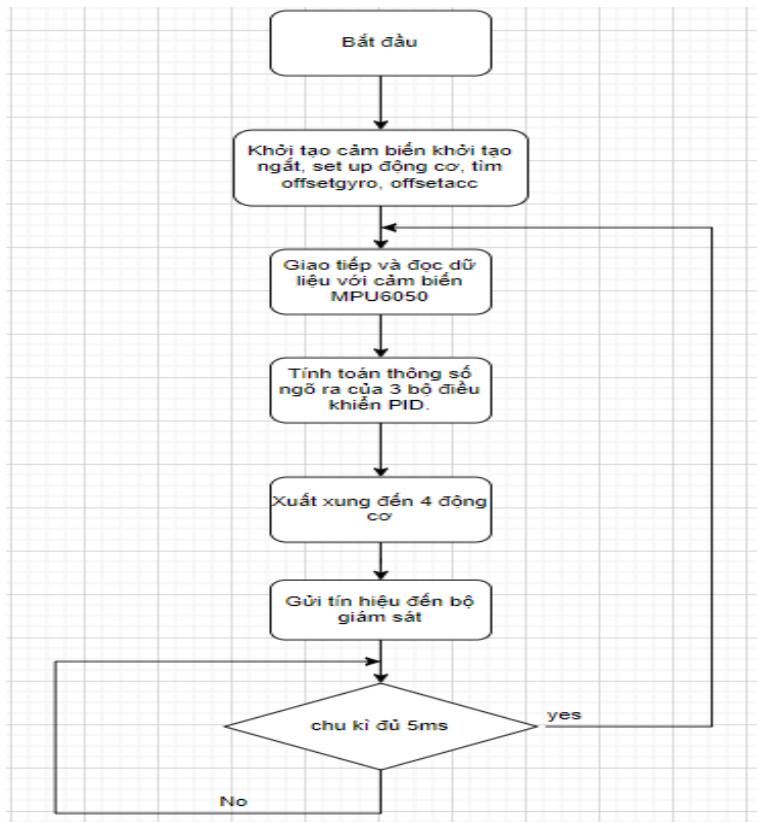
quá trội event như click. Việc viết code cũng vô cùng trực quan: từ việc lấy text từ TextBox cho tới show dữ liệu bằng MessageBox, hoặc dùng Grid để kết nối SQL. WinForm rất dễ học và dễ dạy.



Hình ảnh 4.5.3 Hình ảnh minh họa cho WINFORM.

#### 4.6 Phần mềm điều khiển và giám sát quadcopter:

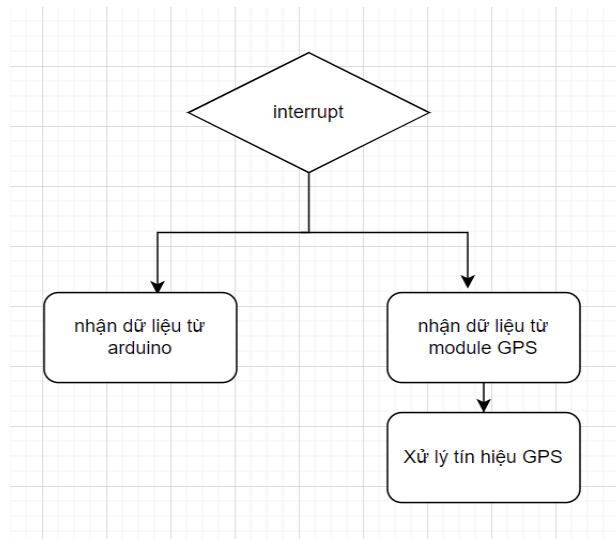
Sơ đồ giải thuật phần mềm điều khiển quadcopter:



Hình ảnh 4.6.1 Hình ảnh sơ đồ giải thuật chương trình điều khiển quadcopter

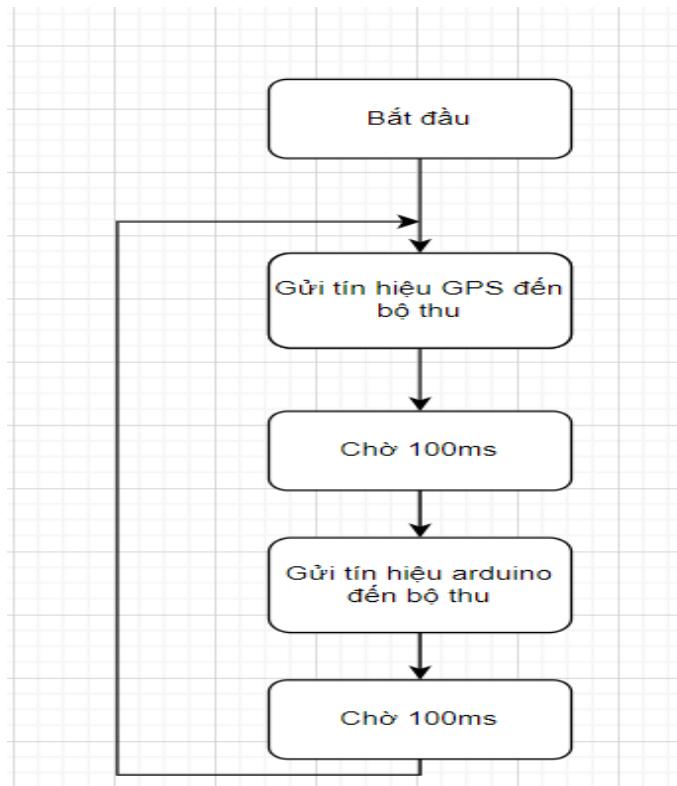
#### 4.7 Sơ đồ giải thuật phần mềm giám sát quadcopter:

Chương trình ngắn:



Hình ảnh 4.7.1 Hình ảnh sơ đồ giải thuật của chương trình phát dữ liệu TX (1)

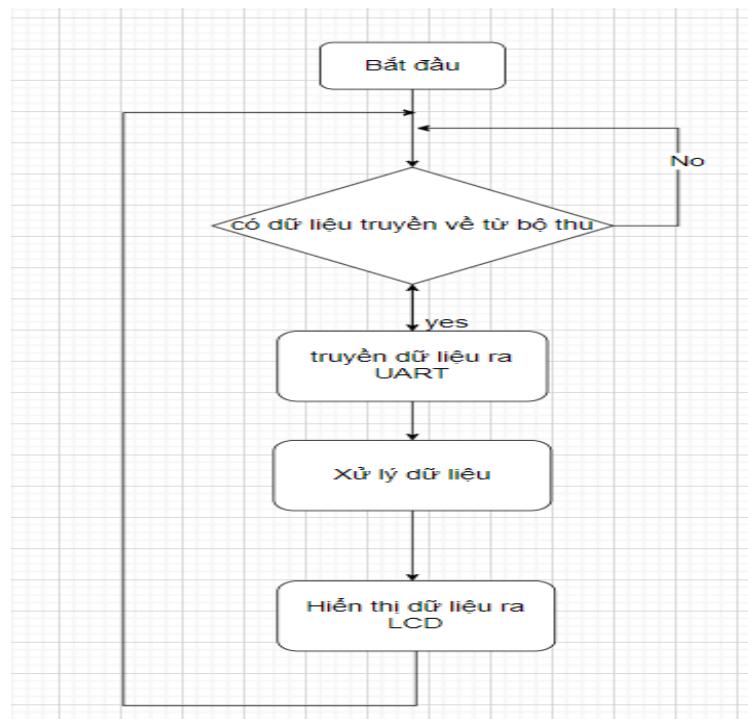
Chương trình chính:



Hình ảnh 4.7.2 Hình ảnh sơ đồ giải thuật của chương trình phát dữ liệu TX (2)

Sơ đồ giải thuật chương trình màn hình giám sát quadcopter.

### Sơ đồ giải thuật phần mềm nhận dữ liệu RX



Hình 4.7.3 Hình ảnh sơ đồ giải thuật của chương trình thu RX

### 4.8 Thực hiện màn hình giám quadcopter.

Phần màn hình C# gồm 5 phần:

**Phần I:** là phần tiêu đề chứa nội dung là, tên trường, tên bộ môn tên sinh viên làm luận văn, và giáo viên hướng dẫn.



Hình ảnh 4.8.1 Hình ảnh minh họa giao diện monitor quadcopter (1)

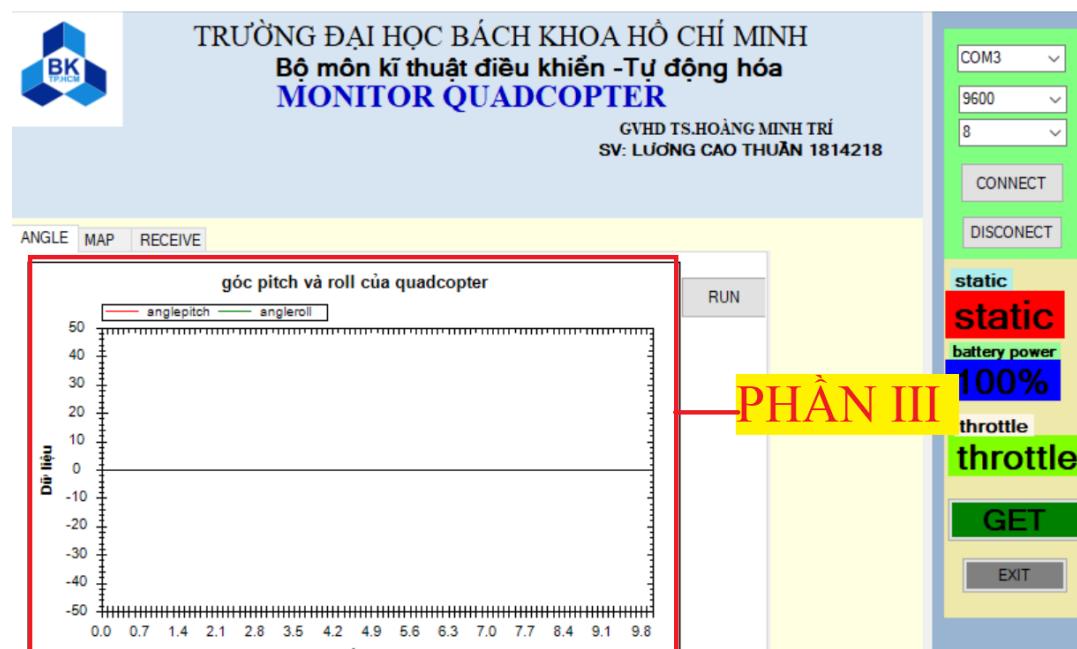
**Phần II:** là phần chứa các nội dung về giao tiếp serial port, tình trạng hoạt động của

quadcopter, dung lượng pin hiện tại, tình trạng buồm gas truyền từ bộ phát tín hiệu điều khiển MC6C.



Hình ảnh 4.8.2 Hình ảnh minh họa giao diện monitor quadcopter (2)

**Phần III:** chứa đồ thị góc roll, pitch của quadcopter.



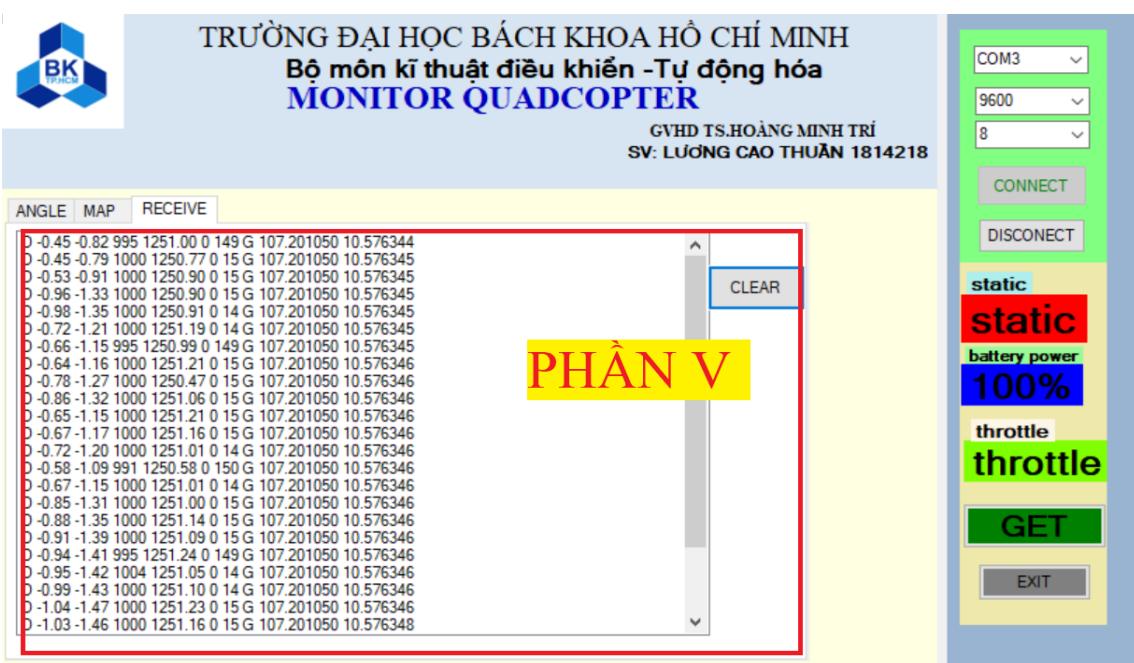
Hình ảnh 4.8.3 Hình ảnh minh họa giao diện monitor quadcopter (3)

**Phần IV:** chứa bản đồ để xác định vị trí quadcopter:



Hình ảnh 4.8.4 Hình ảnh minh họa giao diện monitor quadcopter (4)

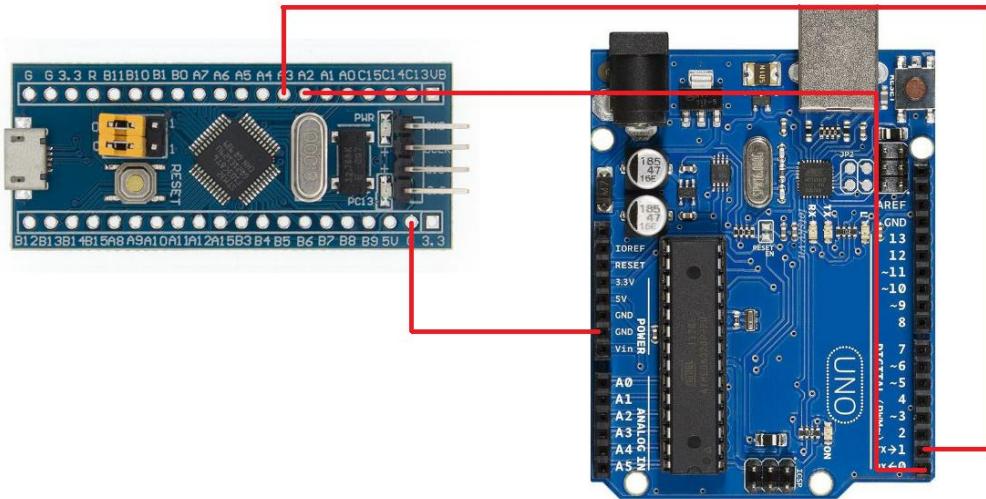
**Phần V:** chứa dữ liệu gửi về từ Serial port:



Hình ảnh 4.8.5 Hình ảnh minh họa giao diện monitor quadcopter (5)

#### 4.9 Kết nối giữa mạch cân bằng bay và mạch giám sát TX:

Chuỗi tin được truyền đi theo chuẩn UART với tốc độ là 200000 bit/s.



Hình 4.9.1 Hình ảnh kết nối phần cứng giữa stm32f103 và arduino uno r3

Sơ đồ nối chân:

PA3	TX
PA2	RX
GND	GND

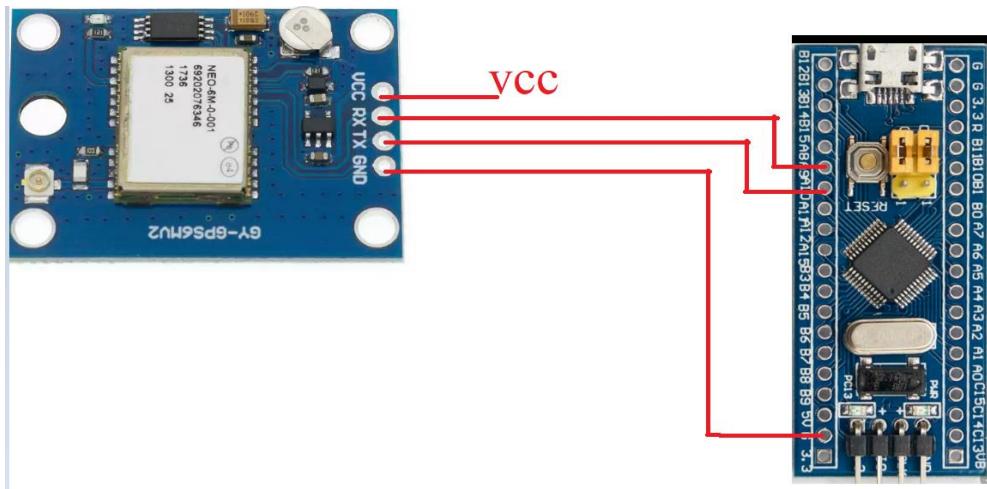
Chuỗi tin được truyền đi:

D	góc roll	góc pitch	throttle	Điện áp PIN	Tình trạng quadcopter
---	----------	-----------	----------	-------------	-----------------------

D 0.05 -0.09 995 1206 2
D 0.05 -0.09 1000 1206 2
D 0.05 -0.09 995 1207 2
D 0.05 -0.09 1000 1207 2
D 0.05 -0.09 995 1207 2
D 0.05 -0.09 1000 1207 2
D 0.05 -0.09 1000 1206 2
D 0.05 -0.09 1000 1206 2
D 0.05 -0.09 991 1206 2
D 0.05 -0.09 995 1206 2
D 0.05 -0.08 995 1206 2
D 0.05 -0.09 1000 1206 2
D 0.05 -0.08 1000 1207 2
D 0.05 -0.08 1000 1207 2
D 0.05 -0.08 1000 1206 2

Hình 4.9.2 Hình ảnh minh họa gói tin truyền từ arduino to stm32

#### 4.10 Kết nối phần cứng giữa GPS NEO 6M và STM32F103C8T6



Hình ảnh 4.10.1 Hình ảnh minh họa kết nối phần cứng giữa GPS NEO 6M và STM32F103C8T6

Sơ đồ nối chân:

GND	GND
TX	PA10
RX	PA9

Chuỗi tin từ GPS gửi cho ST32F103C8T6.

```
$GPGSA,A,3,01,30,07,09,,,,,,4.21,2.55,3.35*0E
$GPGSV,3,1,09,01,57,039,22,06,14,212,,07,78,225,35,08,08,044,*72
$GPGSV,3,2,09,09,16,187,15,14,25,334,,17,34,289,20,19,20,265,20*7A
$GPGSV,3,3,09,30,58,304,25*42
$GPGLL,1034.58558,N,10712.06781,E,021200.00,A,A*64
$GPRMC,021201.00,A,1034.58573,N,10712.06781,E,1.284,347.93,040622,,,A*6E
$GPVTG,347.93,T,M,1.284,N,2.380,K,A*31
$GPGGA,021201.00,1034.58573,N,10712.06781,E,1.04,2.55,29.2,M,-0.3,M,,*7A
$GPGSA,A,3,01,30,07,09,,,,,,4.21,2.55,3.35*0E
$GPGSV,3,1,09,01,57,039,21,06,14,212,,07,78,225,35,08,08,044,*71
$GPGSV,3,2,09,09,16,187,16,14,25,334,,17,34,289,20,19,20,265,19*73
$GPGSV,3,3,09,30,58,304,25*42
$GPGLL,1034.58573,N,10712.06781,E,021201.00,A,A*6E
$GPRMC,021202.00,A,1034.58613,N,10712.06825,E,0.818,349.25,040622,,,A*64
$GPVTG,349.25,T,M,0.818,N,1.516,K,A*36
$GPGGA,021202.00,1034.58613,N,10712.06825,E,1.04,2.55,29.2,M,-0.3,M,,*7D
$GPGSA,A,3,01,30,07,09,,,,,,4.21,2.55,3.35*0E
$GPGSV,3,1,09,01,57,039,22,06,14,212,,07,78,225,35,08,08,044,*72
$GPGSV,3,2,09,09,16,187,17,14,25,334,,17,34,289,20,19,20,265,19*72
$GPGSV,3,3,09,30,58,304,25*42
$GPGLL,1034.58613,N,10712.06825,E,021202.00,A,A*69
$GPRMC,021203.00,A,1034.58645,N,10712.06801,E,0.718,347.36,040622,,,A*63
$GPVTG,347.36,T,M,0.718,N,1.331,K,A*36
```

Hình ảnh 4.10.2 Hình ảnh minh họa gói tin truyền từ GPS NEO 6M.

\$ GPGSA - GPS DOP và các vệ tinh đang hoạt động

\$ GPGSV - Thông tin vệ tinh GPS chi tiết

\$ GPGLL - Kinh độ và Vĩ độ Địa lý

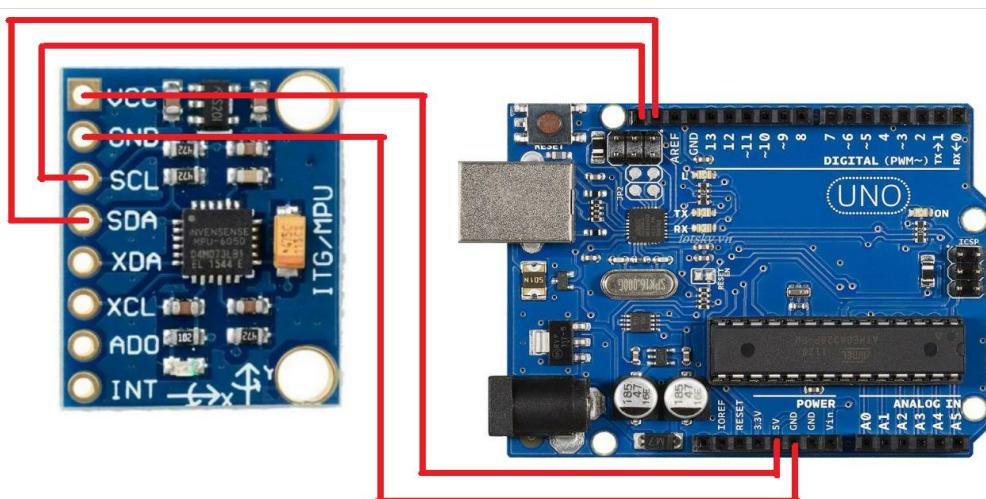
\$ GPRMC - Dữ liệu pvt GPS cần thiết (vị trí, vận tốc, thời gian)

\$ GPVTG - Vận tốc tốt

Dữ liệu được truyền từ module GPS NEO 6M với tốc độ là 9600 bit/s

Giao tiếp giữa Arduino và MPU6050.

#### 4.11 Kết nối phần cứng giữa MPU6050 và Arduino UNO R3



Hình ảnh 4.11.1 Hình ảnh minh họa kết nối giữa MPU6050 và Arduino UNO R3

VCC	5V
GND	GND
SCL	SCL
SDA	SDA

Trong giao tiếp i2c giữa mpu6050 và arduino thì arduino được cấu hình như là Master, còn mpu6050 là slave có địa chỉ là 0x68.

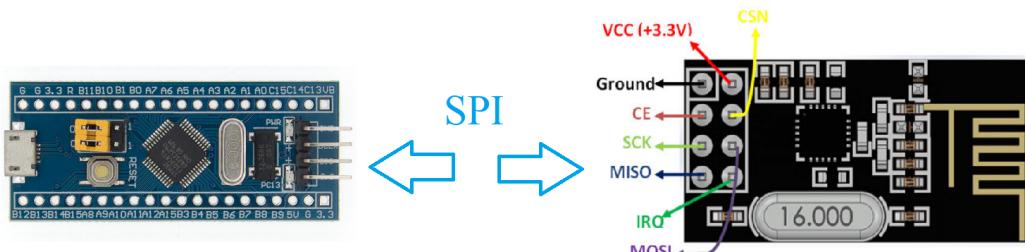
Các bước khởi tạo mpu6050 (quá trình này chỉ thiết lập 1 lần trong khi chạy chương trình):

- 1.Ban đầu gửi giá trị 0x00 để đánh thức mpu6050
- 2.Thanh ghi 0x1B ghi giá trị là 0x08 để cài đặt giá trị con quay hồi chuyển là 500 độ/s.
- 3.Thanh ghi 0x1C để cài đặt giá trị gia tốc kế là 8g.

Sau đó liên tục đọc giá trị từ mpu6050 (quá trình này thực hiện liên tục trong khi chạy chương trình):

- 1.Đọc giá trị gia tốc kế được lưu từ thanh ghi 0x3B đến 0x4B.
- 2.Đọc giá trị con quay hồi chuyển được lưu từ 0x43 đến 0x48.

#### 4.12 Kết nối giữa STMF103C8T6 và NRF24L01:



Hình 4.12.1 Hình ảnh minh họa kết nối giữa stm32 và nrf24l01

STM32F103C8T6	NRF24L01
PA8	CS

PB13	SCK
PB14	MISO
PB15	MOSI
PA15	CE
VCC	VCC
GND	GND

Cấu hình NRF24 L01 là TX:

Cấu hình kéo chân CE lên mức 1.

Khởi tạo địa chỉ Tx Address (0x11223344AA) qua thanh ghi TX ADDR (0x10).

Set true cho auto acknowledgement.

Cài đặt kênh 52 thông qua thanh ghi RF\_CH (0x05).

Cài đặt số byte truyền đi tối đa 32 byte

Cấu hình NRF24 L01 là RX:

Cấu hình kéo chân CE lên mức 1.

Khởi tạo địa chỉ Rx Address (0x11223344AA) qua thanh ghi RX\_ADDR (0x10).

Set true cho auto acknowledgement.

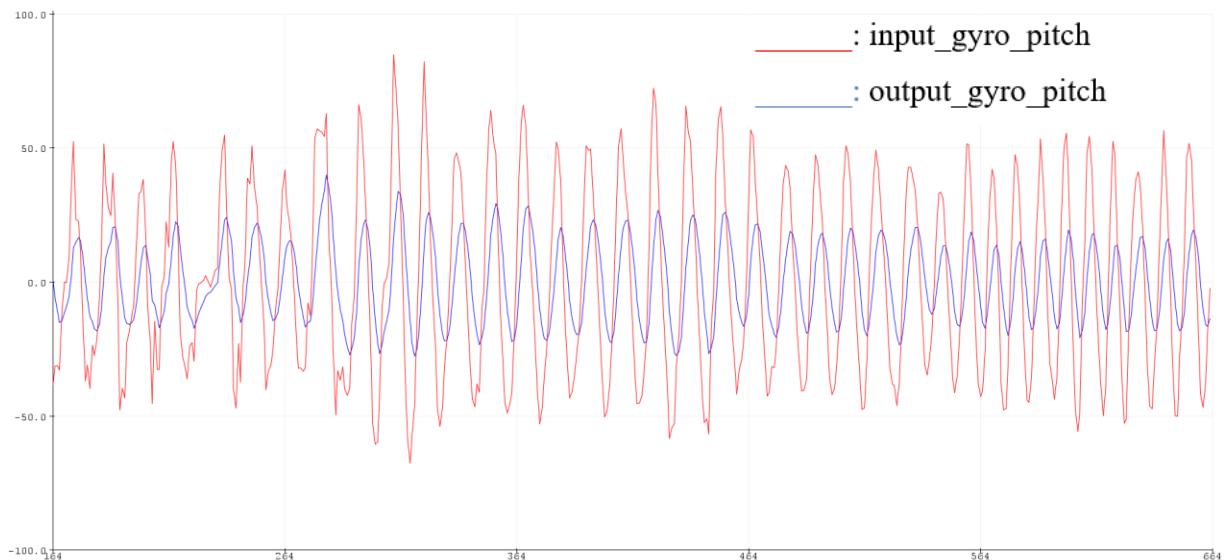
Cài đặt kênh 52 thông qua thanh ghi RF\_CH (0x05).

Cài đặt số byte truyền đi tối đa 32 byte

## CHƯƠNG 5 KẾT QUẢ THỰC HIỆN.

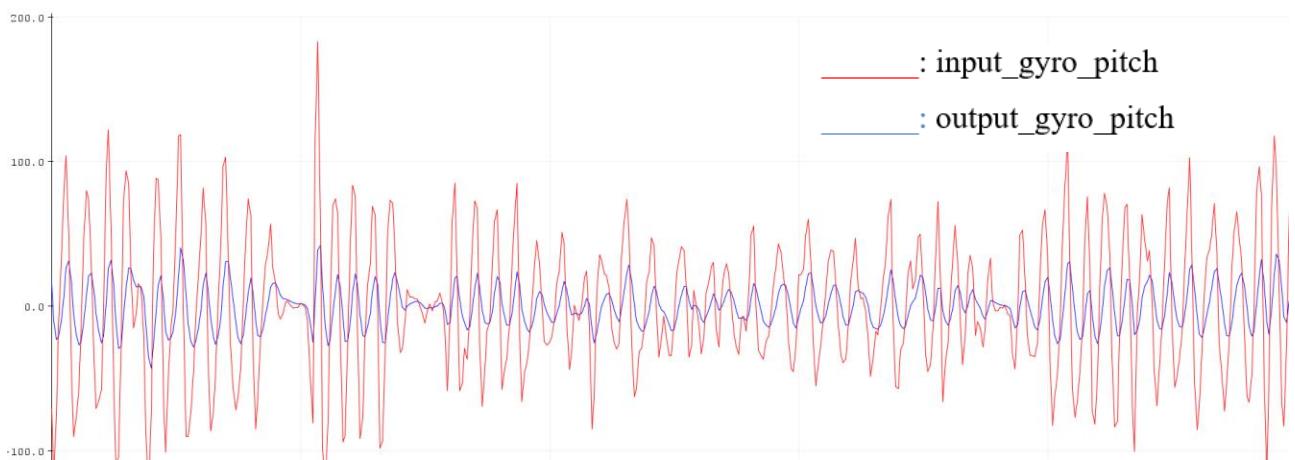
### 5.1 Tín hiệu bộ lọc thông thấp đọc gyroscope của MPU6050:

1. Đọc tín hiệu gyro roll:



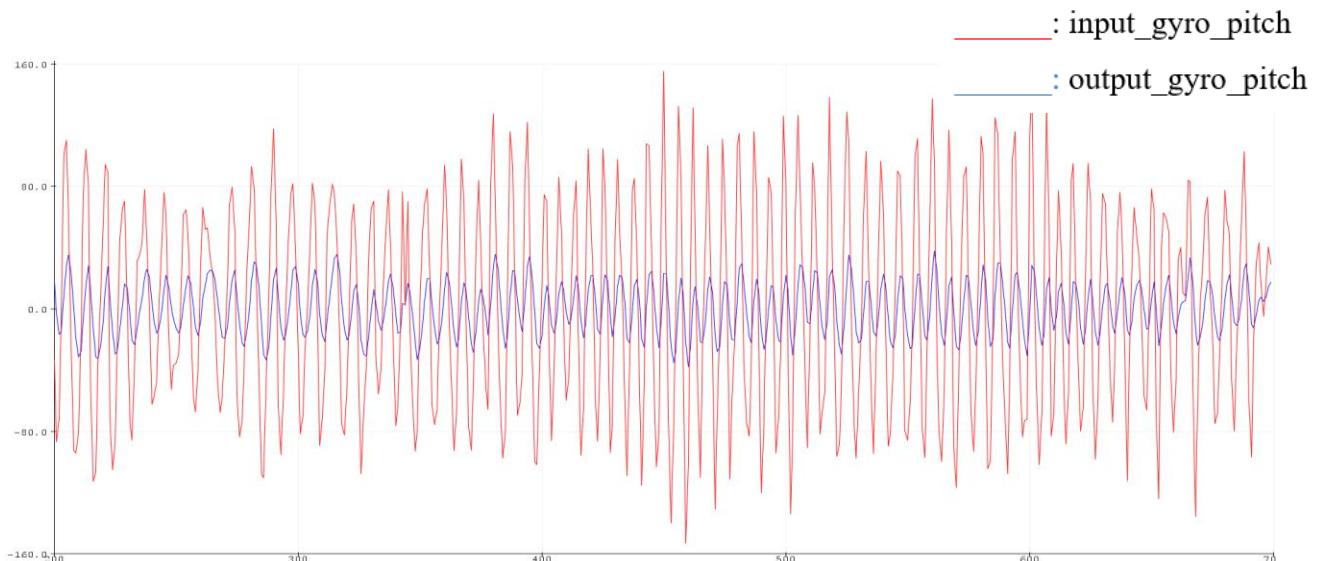
Hình 5.1.1 Hình ảnh ghi nhận tín hiệu gyro roll

2.Đọc tín hiệu gyro pitch:



Hình 5.1.2 Hình ảnh tín hiệu được xử lý của bộ lọc thông thấp cho tín hiệu gyro pitch.

3.Đọc tín hiệu gyro yaw:



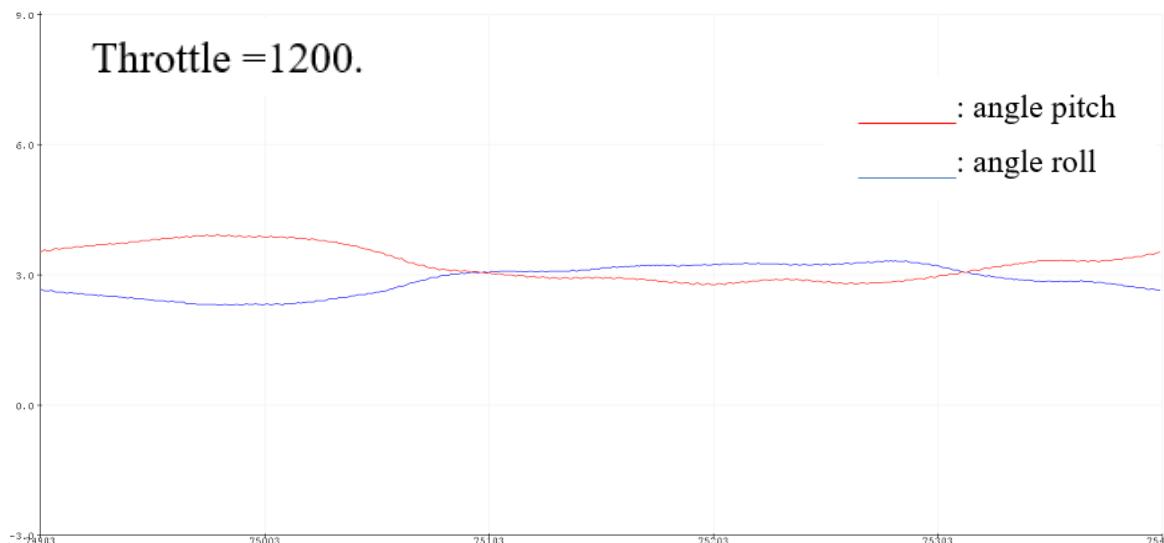
Hình 5.1.3 Hình ảnh xử lý của bộ lọc thông thấp cho tín hiệu gyro pitch.

Nhận xét:

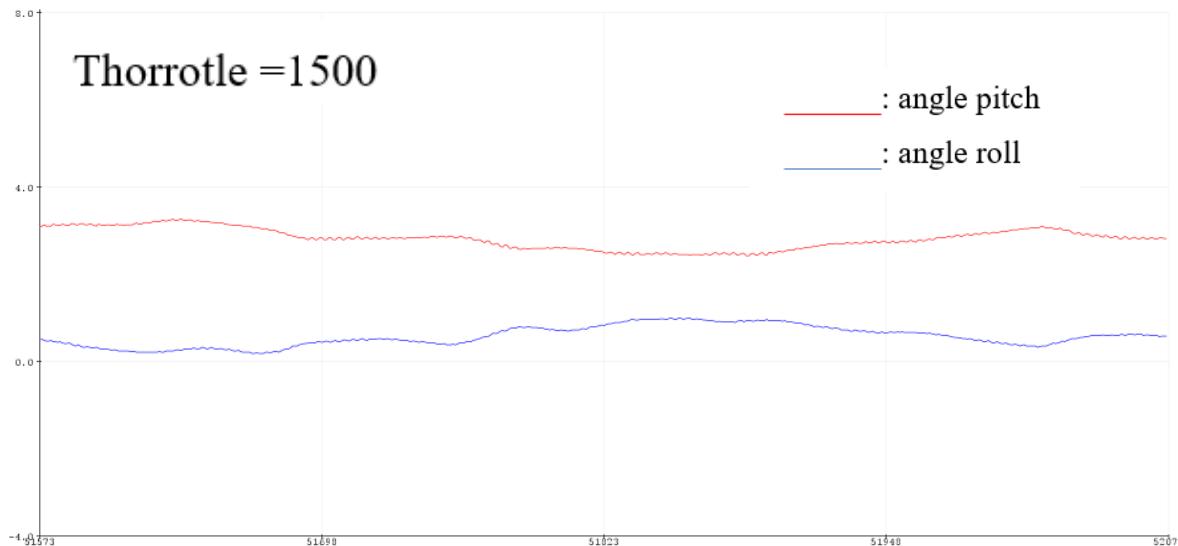
Sau khi áp dụng bộ lọc thông thấp, ta thấy tín hiệu output roll, pitch, yaw bị chậm trễ hơn so với tín hiệu đầu vào, do đó ta thấy tín hiệu đầu ra mượt hơn so với tín hiệu đầu vào.

## 5.2 Tín hiệu góc pitch và roll khi quadcopter được cân bằng:

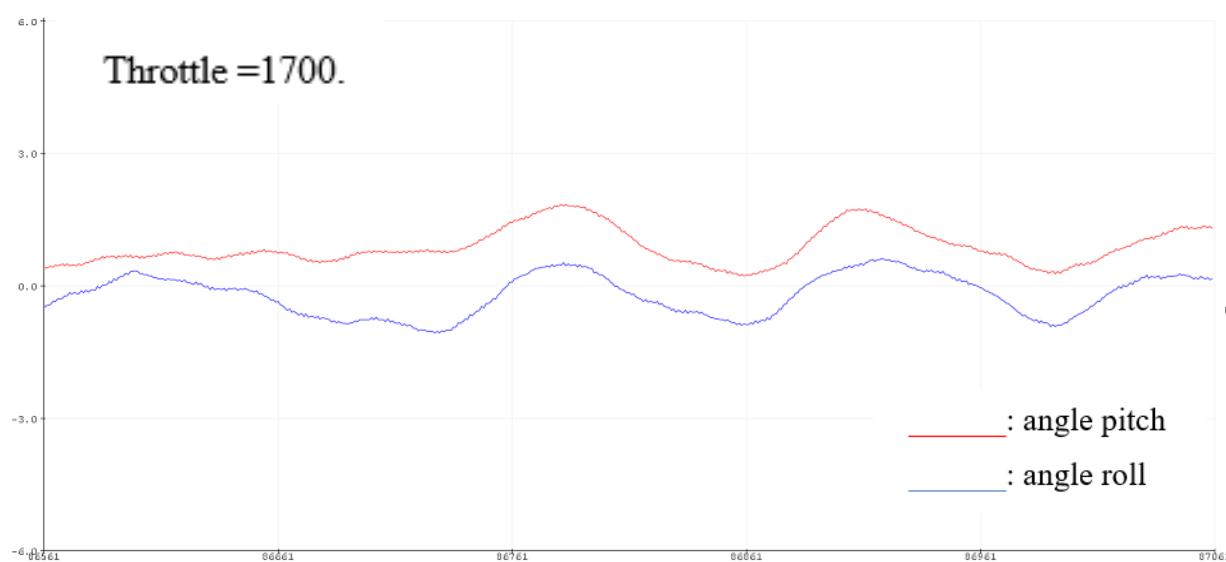
Góc pitch và roll khi cân 2 motor 1 và motor 3:



Hình 5.2.1 Hình ảnh đo được góc pitch và roll khi throttle = 1200 (motor 1, motor 3).



Hình 5.2.2 Hình ảnh đo được góc pitch và roll khi throttle = 1500 (motor 1, motor 3).



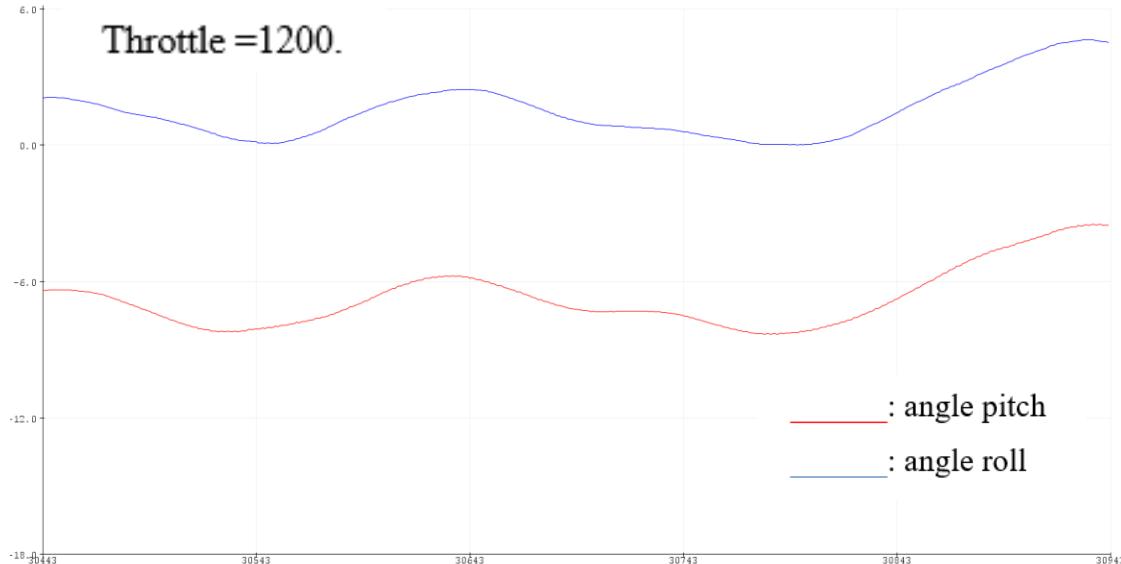
Hình 5.2.3 Hình ảnh đo được góc pitch và roll khi throttle = 1700 (motor 1, motor 3).

### Nhận xét:

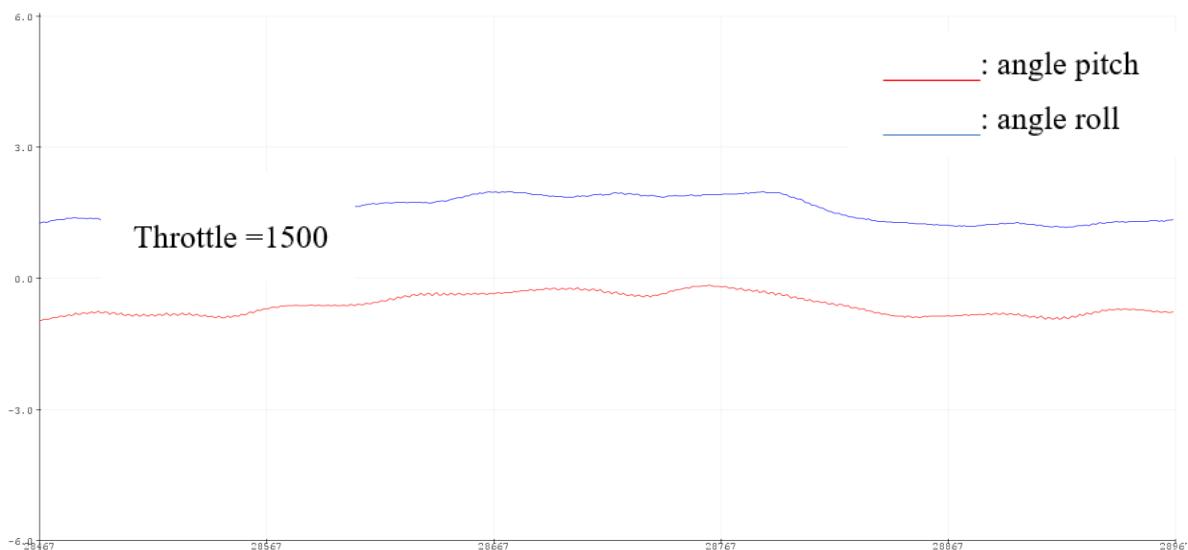
Nhìn vào đồ thị thu được ta thấy khi tăng throttle từ 1200 lên 1500 thì quadcopter ổn định hơn, nhưng khi tăng throttle lên 1700 thì quadcopter lại dao động.

Bộ điều khiển cân bằng 2 motor đối xứng nhau là motor 1 và motor 2 cho đáp ứng tốt, với sai số vọt lô dưới 4 độ.

Góc pitch và roll cho cân bằng motor 2 và motor 4.

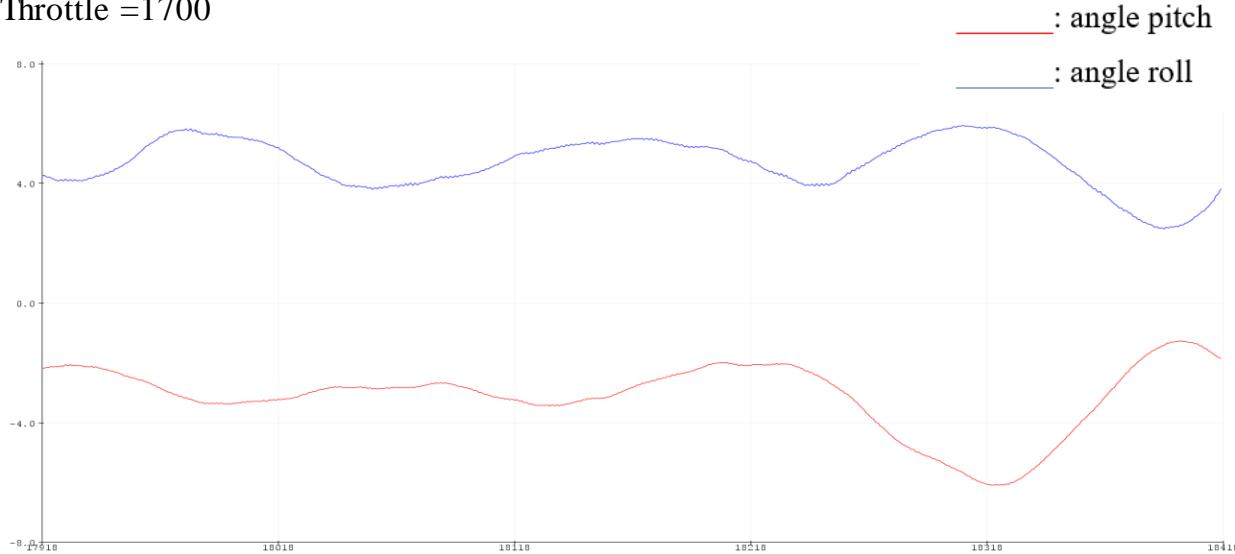


Hình 5.2.4 Hình ảnh đo được góc pitch và roll khi throttle = 1200 (motor 2, motor 4).



Hình 5.2.5 Hình ảnh đo được góc pitch và roll khi throttle = 1500 (motor 2, motor 4).

Throttle =1700



Hình 5.2.6 Hình ảnh đo được góc pitch và roll khi throttle = 1700 (motor 2, motor 4)

#### Nhận xét:

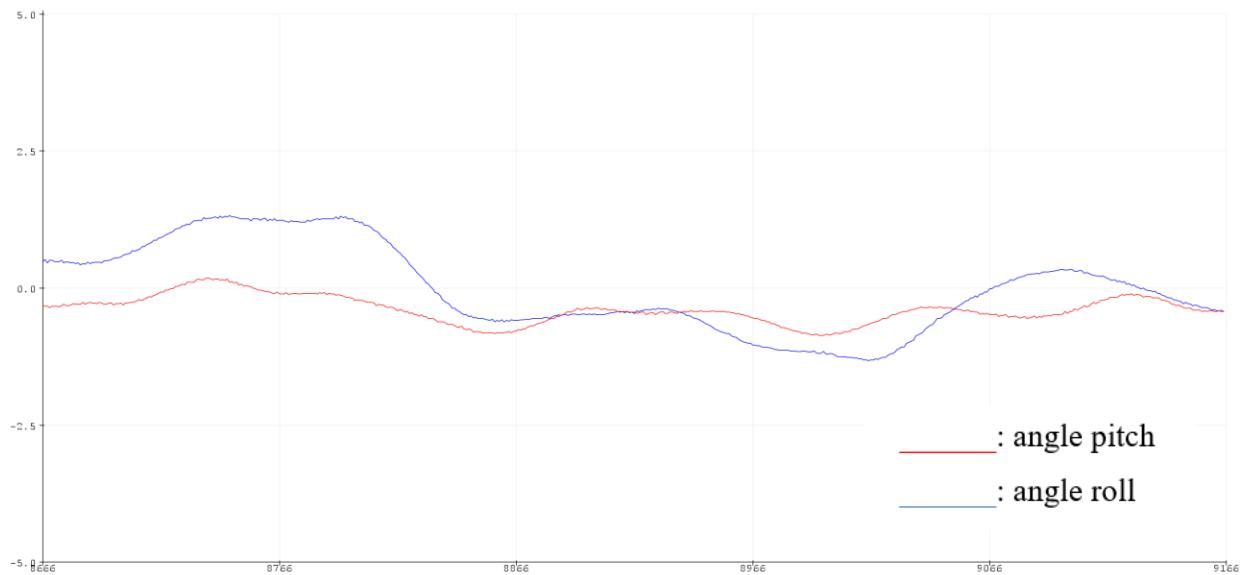
Đối với việc cân bằng 2 motor 2 và motor 4, khi throttle = 1200 thì, đáp ứng có sai số xác lập lớn, và dao động nhiều.

Khi tăng throttle = 1500, thì sai số xác lập giảm và hệ ổn định hơn, tuy nhiên đồ thị bắt đầu xuất hiện nhiều răng cưa.

Khi throttle = 1700, thì quadcopter lại dao động và xuất hiện răng cưa nhiều hơn, sai số trung bình tăng.

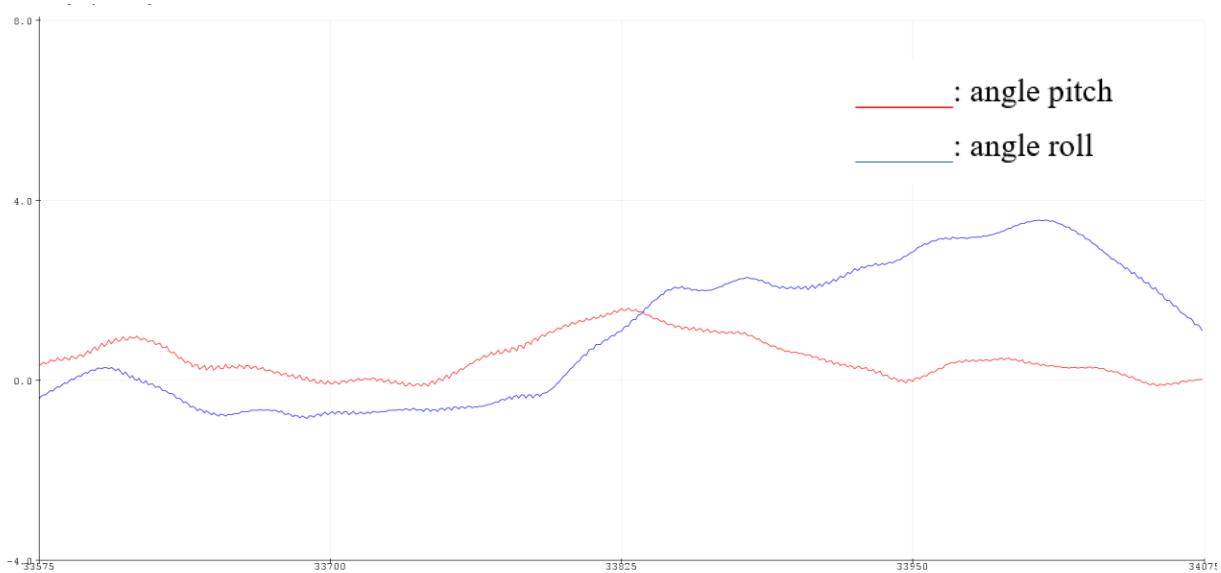
Cân bằng 4 motor của quadcopter

Throttle =1200



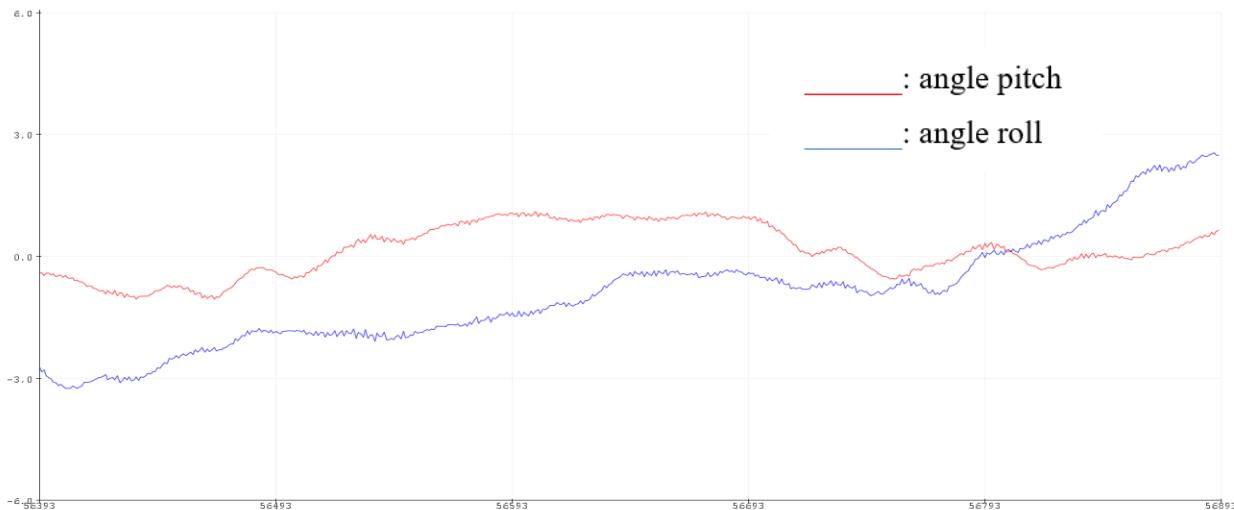
Hình 5.2.7 Hình ảnh đo được góc pitch và roll khi throttle = 1200 (4 motor)

Throttle = 1500



Hình 5.2.8 Hình ảnh đo được góc pitch và roll khi throttle = 1500 (4 motor).

Throttle =1700



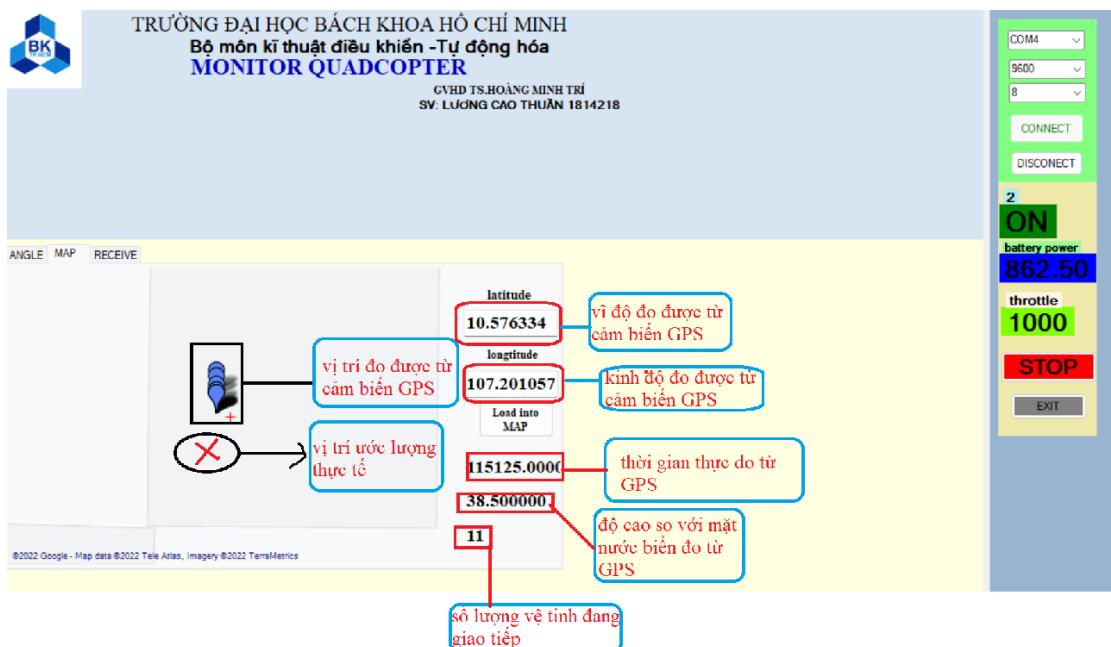
Hình 5.2.9 Hình ảnh đo được góc pitch và roll khi throttle = 1700 (4 motor).

**Nhận xét:**

Bộ điều khiển cân bằng quadcopter hoạt động ổn định với sai số nhỏ hơn 4 độ.

Khi càng tăng throttle thì đáp ứng quadcopter xuất hiện nhiều răng cưa.

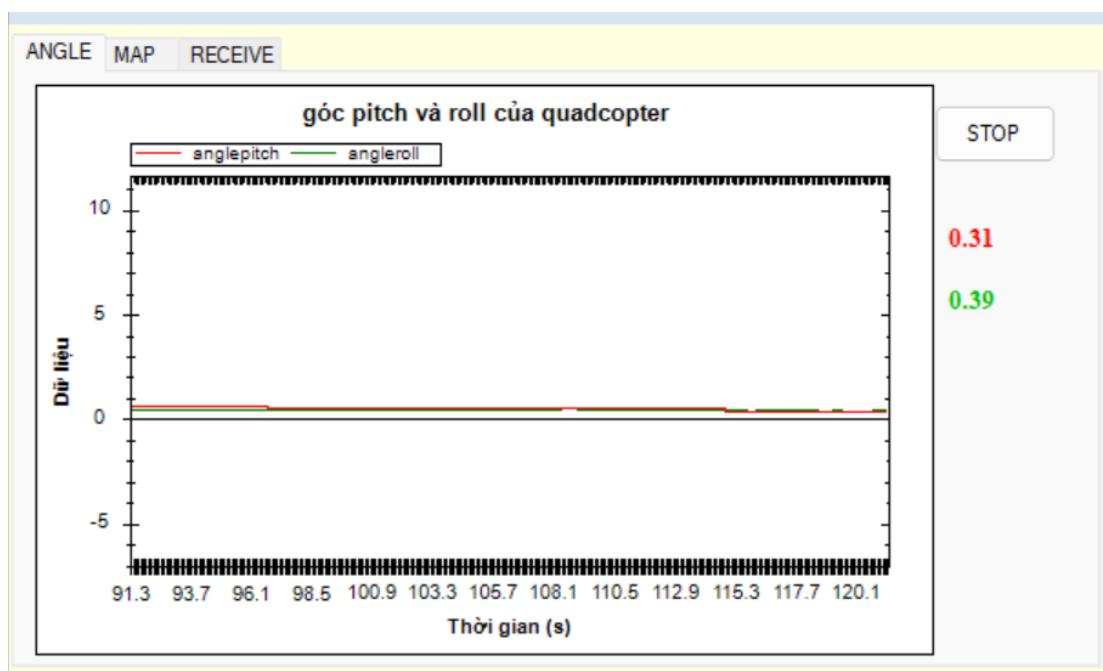
### 5.3 Kết quả thu được từ bộ giám sát.



Hình 5.3.1 Hình ảnh báo cáo kết quả giám sát quadcopter(1)

**Nhận xét:** Từ bản đồ được cung cấp bởi Google và vị trí xác định ta thấy hệ thống có sai số trong xác định vị trí. Tuy nhiên sai số xác định vị trí trên bản đồ là chấp nhận

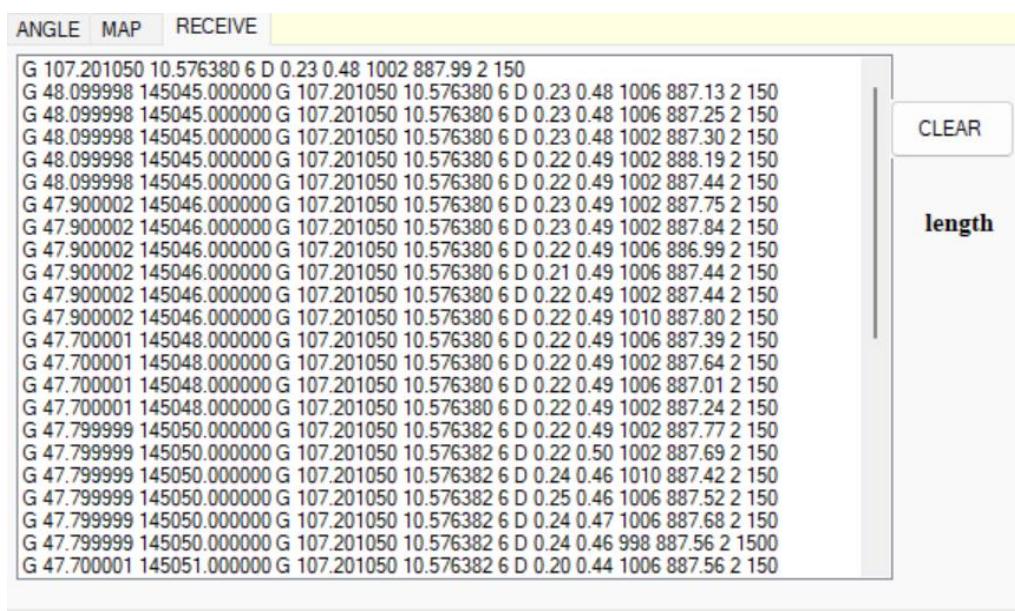
được.



Hình 5.3.2 Hình ảnh báo cáo kết quả giám sát quadcopter(2)

**Nhận xét:** Dữ liệu được nhận thông qua cổng serial port và được vẽ lên màn hình.

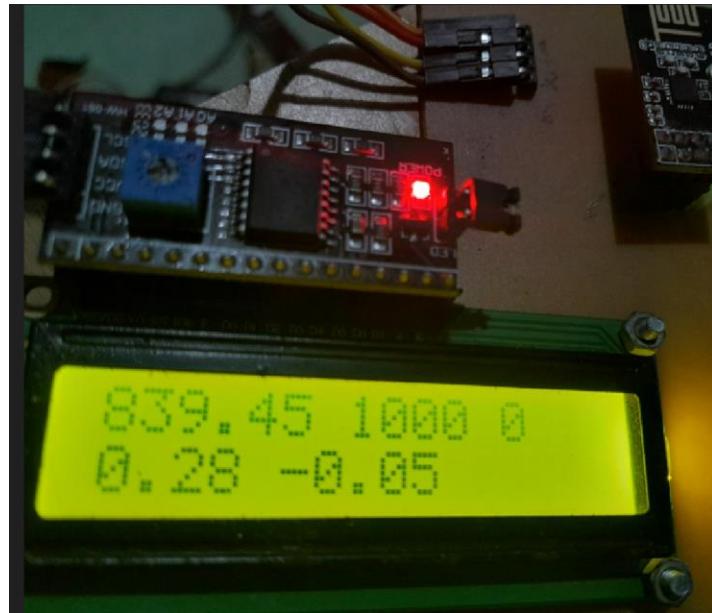
Hình ảnh minh họa vẽ góc pitch và roll trên winform



Hình 5.3.3 Hình ảnh báo cáo kết quả giám sát quadcopter(2)

Hình ảnh ghi nhận dữ liệu gửi lên Winform.

**Nhận xét:** Hệ thống giám sát có thể cơ bản giám sát được vị trí, góc roll, pitch, volt của pin và bướm ga, tuy nhiên hệ thống vẫn có 1 số lỗi trong quá trình thực hiện như bị đơ, bị thiếu hoặc tràn dữ liệu thì sẽ không nhận diện được dữ liệu.



Hình 5.3.4 Hình ảnh kết quả từ bộ giám sát LCD (1)



Hình 5.3.5 Hình ảnh kết quả từ bộ giám sát LCD (2)

Nhận xét: Màn hình LCD thể hiện được dung lượng pin, bướm ga, góc roll, góc pitch, vị trí kinh độ và vĩ độ của quadcopter.

Nhận xét tổng:

Bộ giám sát có thể đọc các tín hiệu từ quadcopter tuy nhiên nó hoạt động chưa ổn định, xảy ra một số lỗi trong quá trình hoạt động. Nguyên nhân là do thời gian nhận dữ liệu từ serial port và thời gian, cập nhật dữ liệu lên màn hình có 1 khoảng trễ nên có thời điểm dữ liệu bị thiếu và hệ thống bị lỗi.

#### **5.4. Video kết quả thực hiện:**

##### **Link video kết quả thực nghiệm:**

Thầy và các bạn có thể xem video theo như link sau để có kết quả trực quan hơn.

**1.Video tổng quan về quá trình đọc dữ liệu và cân bằng quadcopter, hiển thị vị trí trạng thái dung lượng pin trên winform và LCD.**

[https://www.youtube.com/watch?v=4l0j\\_hZRQaI](https://www.youtube.com/watch?v=4l0j_hZRQaI)

**2.Video điều khiển quadcopter trong nhà:**

<https://www.youtube.com/watch?v=T30jwDOJtIY>

**3.Video điều khiển quadcopter ngoài trời.**

<https://www.youtube.com/watch?v=VaFxiCoN-Ug>

## CHƯƠNG 6 TỔNG KẾT

### 6.1 Những kết quả đạt được trong đề tài luận văn:

Về cơ bản thiết kế, thực hiện được mạch điều khiển quadcopter, đã kiểm tra và cho ra kết quả hoạt động tốt ở nơi có diện tích lớn.

Về hệ thống giám sát có thể truyền nhận được liệu và hiển thị lên màn hình giám sát và màn hình LCD.

Trong quá trình nghiên cứu và thực hiện đề tài luận văn đã giúp em hiểu về nguyên lý điều khiển PID, nguyên lý cân bằng quadcopter, cải thiện khả năng thiết kế mạch in, làm mạch in, cải thiện khả năng lập trình ngôn ngữ C, hiểu biết nhiều về 2 dòng vi điều khiển Arduino và STM32, học được cơ bản về lập trình winform C#, và cách sử dụng các phần cứng đã được trong luận văn.

### 6.2 Những hạn chế cần khắc phục trong đề tài:

Kết quả tạo ra còn nhiều khiếm khuyết đó là mạch cân bằng hoạt động một thời gian bị đứng chương trình vẫn chưa tìm được nguyên nhân của lỗi trên.

Mạch cân bằng không thể điều khiển quadcopter trong phạm vi hẹp vì sai số điều khiển quá lớn và không có chức năng giữ vị trí.

Hệ thống giám sát xảy ra lỗi tràn dữ liệu vẫn chưa khắc phục được lỗi này.

### 6.3 Hướng phát triển của đề tài trong tương lai.

Đề tài điều khiển quadcopter là đề tài đã được nghiên cứu rất nhiều trong quá khứ và có rất nhiều hướng phát triển trong tương lai. Đề tài của em có thể được phát triển bằng cách thêm mạch cảm biến áp suất. Dựa trên cảm biến áp suất và cảm biến vị trí ta có thể điều khiển dữ vị trí quadcopter. Từ vị điều khiển được vị trí ta có thể điều khiển quỹ đạo quadcopter. Chúng ta có thể thay thế gắn thêm máy ảnh và các máy nhúng để thực hiện điều khiển dựa trên xử lý ảnh và trí tuệ nhân tạo. Trong tương lai nếu hệ thống internet toàn cầu được hoạt động chúng ta có thể điều khiển quadcopter trên mọi nơi trên thế giới.

Danh mục tài liệu tham khảo:

- [1] [http://www.brokking.net/JB-Serial\\_Monitor.html](http://www.brokking.net/JB-Serial_Monitor.html)
- [2] <https://www.youtube.com/watch?v=zD8ngmjMY2k&t=424s>
- [3] <https://www.youtube.com/watch?v=XqngC1M8uFo>

