

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA ĐIỆN TỬ - VIỄN THÔNG



BÁO CÁO PBL 3:
Chuyên đề Hệ Thống Máy Tính
Tên đề tài: Hồ Cá Thông Minh

Người hướng dẫn: Hồ Viết Việt
Nhóm: Nguyễn Quốc Việt
Lớp HP: 19.40A

Họ và tên	Lớp	MSSV
Nguyễn Quốc Việt	19DTCLC2	106190092
Bùi Thành Nhân	19DTCLC2	106190073
Đinh Ngọc Tiên	19DTCLC2	106190085
Đoàn Thanh Tuấn	19DTCLC2	106190090
Trần Phúc Thịnh	19DTCLC1	106190037

Đà Nẵng, 29 tháng 06 năm 2022

MỤC LỤC

CHƯƠNG 1: CƠ SỞ LÝ THUYẾT	6
1.1. KỸ THUẬT NUÔI CÁ CẢNH	6
1.1.1. Tổng quan	6
1.1.2. Thức ăn cho cá	6
1.1.3. Ánh sáng	6
1.2. TỔNG QUAN VỀ WEBSITE	7
1.2.1. Các khái niệm về Internet	7
1.2.2. Giới thiệu tổng quan về website	7
1.2.3. Giới thiệu về ngôn ngữ HTML	8
1.2.4. Giới thiệu về ngôn ngữ CSS	8
CHƯƠNG 2: THIẾT KẾ VÀ THI CÔNG HỒ CÁ THÔNG MINH	9
2.1. GIỚI THIỆU VỀ PHẦN CỨNG	10
2.1.1. Chi tiết về các thành phần phần cứng	10
2.1.1.1. Đèn LED RGB	10
2.1.1.2. IRF520N	12
2.1.1.3. NodeMCU ESP8266	13
2.1.1.4. Động cơ Servo SG90	14
2.1.1.5. ESP32 - CAM AI - THINKER	14
2.1.2. Sơ đồ nối dây	18
2.1.1.1. Sơ đồ nối dây của LED RGB	18
2.1.1.2. Sơ đồ nối dây của Servo SG90	19
2.1.1.3. Sơ đồ nối dây của cả hệ thống	19
2.2. PHẦN MỀM	19
2.2.1. Giới thiệu phần mềm sử dụng	19
2.2.1.1. Giới thiệu về Arduino IDE	19
2.2.1.1.1. Arduino IDE là gì	19
2.2.1.1.2. Cách cài đặt ESP8266 và ESP32 Camera vào Arduino IDE	20
2.2.1.2. Giới thiệu về Ngrok	22
2.2.1.2.1. Ngrok là gì	22
2.2.1.2.2. Cách cài đặt Ngrok	22
2.2.2. Lưu đồ thuật toán	23

2.2.3. Lập trình cho hệ thống	24
2.2.3.1. Led RGB.....	24
2.2.3.2. ESP32 Camera	26
2.2.3.3. Servo	30
2.2.3.4. Code file .BAT.....	32
2.3. THI CÔNG ĐỒ ÁN	32
2.3.1. Thiết kế hồ cá.....	32
2.3.2. Thiết kế hộp bảo vệ	33
2.3.3. Thiết kế hộp đựng thức ăn cho cá.....	34
2.3.4. Thiết kế dải đèn LED RGB	34
2.3.5. Bố trí ESP 32 Camera.....	34
2.3.6. Mô hình tổng quát của hồ cá.....	35
CHƯƠNG 3: KẾT QUẢ MÔ PHỎNG VÀ HƯỚNG DẪN SỬ DỤNG	36
3.1. MÔ PHỎNG ĐỒ ÁN	36
3.2. HƯỚNG DẪN SỬ DỤNG	39
CHƯƠNG 4: KẾT QUẢ - NHẬN XÉT VÀ ĐÁNH GIÁ.....	43
4.1. KẾT QUẢ	43
4.2. NHẬN XÉT VÀ ĐÁNH GIÁ	43
4.2.1. Ưu điểm của hệ thống	43
4.2.2. Nhược điểm của hệ thống.....	44
CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	44
5.1. KẾT LUẬN.....	44
5.2. HƯỚNG PHÁT TRIỂN	45

LIỆT KÊ HÌNH

<i>Hình 2. 1 Cấu tạo của LED RGB</i>	<i>10</i>
<i>Hình 2. 2 Xung PWM điều chỉnh điện áp của LED RGB.....</i>	<i>11</i>
<i>Hình 2. 3 Sơ đồ chân của IRF520N</i>	<i>12</i>
<i>Hình 2. 4 Sơ đồ chân của ESP8266</i>	<i>13</i>
<i>Hình 2. 5 Động cơ Servo SG90</i>	<i>14</i>
<i>Hình 2. 6 Thành phần cấu tạo bo mạch ESP32-CAM AI-THINKER ở mặt trên và mặt dưới</i>	<i>15</i>
<i>Hình 2. 7 Sơ đồ chân của module AI-Thinker ESP32-CAM</i>	<i>15</i>
<i>Hình 2. 8 Sơ đồ nối dây của LED RGB.....</i>	<i>18</i>
<i>Hình 2. 9 Sơ đồ nối dây của Servo SG90</i>	<i>19</i>
<i>Hình 2. 10 Sơ đồ nối dây của cả hệ thống.....</i>	<i>19</i>
<i>Hình 2. 11 Giao diện phần mềm Arduino IDE.....</i>	<i>20</i>
<i>Hình 2. 12 Giao diện sau khi nhập dòng trên vào “Additional Board Manager URLs” .</i>	<i>21</i>
<i>Hình 2. 13 Giao diện tìm kiếm Esp8266.....</i>	<i>21</i>
<i>Hình 2. 14 Giao diện tìm kiếm Esp32 Camera.....</i>	<i>22</i>
<i>Hình 2. 15 Lưu đồ thuật toán chính của hệ thống.....</i>	<i>23</i>
<i>Hình 2. 16 Hộp nhựa cỡ lớn làm hồ cá</i>	<i>33</i>
<i>Hình 2. 17 Hộp bảo vệ mạch điều khiển</i>	<i>33</i>
<i>Hình 2. 18 Hộp đựng thức ăn cho cá</i>	<i>34</i>
<i>Hình 2. 19 Thiết kế LED RGB cho cá</i>	<i>34</i>
<i>Hình 2. 20 Lắp đặt ESP32 Camera.....</i>	<i>35</i>
<i>Hình 2. 21 Mô hình tổng quát của hồ cá.....</i>	<i>36</i>

LỜI NÓI ĐẦU

Ngày nay, sự phát triển nhanh chóng của khoa học công nghệ hiện nay đã làm cho cuộc sống của con người trở nên hiện đại hơn. Cùng với xu hướng phát triển của khoa học kỹ thuật ngày càng mạnh mẽ trên toàn thế giới, thì việc áp dụng chúng vào cuộc sống luôn mang lại nhiều lợi ích. Chúng ta phải nắm bắt và vận dụng nó một cách hiệu quả, nhằm góp phần vào sự phát triển nền khoa học kỹ thuật nói chung và trong sự phát triển kỹ thuật điện tử nói riêng để ứng dụng vào thực tiễn góp phần đưa cuộc sống con người ngày càng tốt hơn.

Bằng cách sử dụng ESP8266 có khả năng kết nối internet, chúng ta có thể dùng chúng để giám sát và điều khiển các thiết bị khác. Các dữ liệu giám sát và điều khiển đó được lưu trữ trên hosting giúp chúng ta có thể xem và điều khiển các thiết bị khác thông qua website.

Với việc áp dụng khoa học kỹ thuật thì giờ đây, người dùng có thể quan sát, điều khiển các thiết bị hỗ trợ nuôi cá và quản lý chúng từ xa. Họ có thể tự điều khiển bật tắt đèn, hay cho cá ăn ngay cả khi họ và thiết bị cách nhau nửa vòng Trái Đất. Để làm được điều đó thì họ chỉ cần có internet, thiết bị truy cập website và “HỒ CÁ THÔNG MINH”.

CHƯƠNG 1: CƠ SỞ LÝ THUYẾT

1.1. KỸ THUẬT NUÔI CÁ CẢNH

1.1.1. Tổng quan

Ngày nay thú chơi cá cảnh đã trở nên phổ biến. Tuy nhiên, để có một bể cá hoàn hảo để trong nhà thì không phải là điều dễ dàng, đòi hỏi người chủ cần có sự hiểu biết và kinh nghiệm cần thiết. Để hạn chế tới mức thấp nhất độ rủi ro khi nuôi cá, người mới nuôi cần quan tâm đến môi trường sống, chế độ ăn uống, nước nuôi cá, kỹ thuật thả cá,... Những điều đó sẽ giúp cho người nuôi tự tin hơn khi bắt đầu chăm sóc bể cá của riêng mình.

1.1.2. Thức ăn cho cá

Nên cho cá ăn vừa đủ không nên để thức ăn dư thừa tránh làm thừa nhiều thức ăn gây ảnh hưởng đến chất lượng nước bể cá.

Cá có tập tính thấy mồi là đói, nên nhiều người tưởng cá còn đói nên cho ăn nhiều sẽ làm cá bị đầy bụng mà chết. Vì thế nên cho ăn 2 lần/ngày (sáng và chiều). Nếu lỡ để cá đói vài ngày thì cá không chết nhưng cho ăn no quá thì cá lại rất dễ chết. Ngoài thức ăn khô, tùy loại cá mà ta nên bổ sung thêm thức ăn tươi như cá con, cá trâm, cá chép mồi,... tùy thuộc vào giống cá đang nuôi.

Thức ăn tổng hợp là một trong những loại thức ăn mà người nuôi cá hay sử dụng bởi tính tiện dụng và dễ bảo quản hơn các loại thức ăn tươi. Có 2 loại là thức ăn dạng bột và thức ăn dạng hạt. Đây là loại thức ăn thích hợp để sử dụng cho hệ thống nuôi cá tự động.

1.1.3. Ánh sáng

Ánh sáng là yếu tố cần thiết cho mỗi bể cá. Ánh sáng vừa giúp trang trí cho bể thêm lung linh nổi bật vừa giúp cho những loại cá phát triển các sắc tố cho chúng.

Cần đặt hồ cá nơi thoáng mát, tránh ánh sáng trực tiếp, nếu để bể cá nơi tối tăm, không thoáng khí lâu ngày dễ làm cá phát bệnh. Tùy tình huống có thể sử dụng đèn công suất nhỏ cho hồ cá với thời lượng khoảng vài giờ /ngày (bật ban ngày dưới 8 tiếng và tắt vào ban đêm cho cá nghỉ ngơi). Nếu đặt bể cá ngoài trời cần tạo bóng mát, hạn chế tác động từ nắng, mưa...

Chính vì thế ánh sáng cũng cần sự điều khiển, không thể bật liên tục cũng như không thể luôn tắt khi chăm sóc cho 1 bể cá được.

1.2. TỔNG QUAN VỀ WEBSITE

1.2.1. Các khái niệm về Internet

Internet là một hệ thống thông tin toàn cầu có thể được truy cập công cộng giữa các mạng máy tính được liên kết với nhau.

TCP/IP là bộ giao thức cho phép kết nối các hệ thống mạng không đồng nhất với nhau. Ngày nay TCP/IP được dùng rộng rãi trong các mạng cục bộ cũng như mạng toàn cầu. TCP/IP được xem như giản lược của mô hình tham chiếu OSI với 4 tầng như sau:

- Tầng liên kết (Datalink layer).
- Tầng mạng (Internet layer).
- Tầng giao vận (Transport layer).
- Tầng ứng dụng (Application layer).

Phương thức hoạt động của bộ giao thức TCP/IP:

Khi truyền dữ liệu, quá trình tiến hành từ tầng trên xuống tầng dưới (tầng ứng dụng truyền xuống tầng liên kết), qua mỗi tầng dữ liệu được thêm vào thông tin điều khiển được gọi là Header.

Khi nhận dữ liệu thì quá trình này sẽ xảy ra ngược lại. Dữ liệu được truyền từ tầng dưới lên tầng trên (tầng liên kết truyền lên tầng ứng dụng) và qua mỗi tầng thì phần Header tương ứng sẽ được lấy đi và khi đến tầng cuối cùng thì dữ liệu không còn phần Header nữa.

1.2.2. Giới thiệu tổng quan về website

WEBSITE, còn gọi là trang web, là một tập hợp các trang web con, bao gồm văn bản, hình ảnh, video, flash,... WEBSITE chỉ nằm trong một tên miền hoặc tên miền phụ lưu trữ trên các máy chủ chạy online trên đường truyền World Wide của Internet.

Một trang web là tập tin HTML hoặc XHTML có thể truy nhập dùng giao thức HTTP hoặc HTTPS.

WEBSITE có thể được xây dựng từ các tập tin HTML (trang mạng tĩnh) hoặc vận hành bằng các CMS chạy trên máy chủ (trang mạng động).

WEBSITE có thể được xây dựng bằng nhiều ngôn ngữ lập trình khác nhau (HTML, PHP, .NET, Java, Ruby on Rails...).

Một website có thể vận hành trên môi trường World Wide, cần bắt buộc có 3 phần chính:

- Tên miền (là tên riêng và duy nhất của website).
- Hosting (là các máy chủ chứa các tập tin nguồn).
- Source code (là các tập tin html, xhtml,...).

Website được tương tác và hiển thị đến với người dùng thông qua các phần mềm gọi là “ Trình duyệt web ” với các văn bản, hình ảnh, đoạn phim, nhạc, trò chơi, và các thông tin khác ở trên một trang web của một địa chỉ web trên mạng toàn cầu hoặc mạng nội bộ. Bất kì đâu có Internet thì ai cũng có thể truy cập website và tương tác một cách dễ dàng.

1.2.3. Giới thiệu về ngôn ngữ HTML

HTML (tiếng Anh, viết tắt cho HyperText Markup Language, hay là “Ngôn ngữ đánh dấu Siêu văn bản”) là một ngôn ngữ đánh dấu được thiết kế ra để tạo nên các trang web với các mẫu thông tin được trình bày trên World Wide Web. Cùng với CSS và JavaScript, HTML tạo ra bộ ba nền tảng kỹ thuật cho World Wide Web. HTML được định nghĩa như là một ứng dụng đơn giản của SGML và được sử dụng trong các tổ chức cần đến các yêu cầu xuất bản phức tạp. HTML đã trở thành một chuẩn Internet do tổ chức World Wide Web Consortium (W3C) duy trì. Phiên bản chính thức mới nhất của HTML là HTML 4.01 (1999). Sau đó, các nhà phát triển đã thay thế nó bằng XHTML. Hiện nay, HTML đang được phát triển tiếp với phiên bản HTML5 hứa hẹn mang lại dao diện mới cho web.

1.2.4. Giới thiệu về ngôn ngữ CSS

Cascading Style Sheets (CSS) là ngôn ngữ dùng để tìm và định dạng lại các phần tử cho website như việc tạo ra các đoạn văn bản, các tiêu đề, bảng ... và CSS sẽ giúp các phần tử HTML thay đổi cấu trúc rất nhiều như đổi màu sắc cho trang, đổi màu chữ, thêm hiệu ứng cho các đoạn văn bản...

CSS có vai trò trang trí thêm cho văn bản được viết bằng HTML trở nên sinh động và bắt mắt hơn.

CSS là một file có phần mở rộng là “.css”, file này có tác dụng tách riêng phần định dạng trong thẻ < style > ra khỏi nội dung HTML. Giúp người thiết kế dễ dàng quản lý được nội dung trong HTML, tiết kiệm được nhiều thời gian trong chỉnh sửa đặc biệt là các dự án lớn được kết cấu từ nhiều trang HTML.

Phương thức hoạt động: CSS sẽ tìm dựa vào các vùng chọn được quy định trong văn bản HTML. Vùng chọn này có thể là tên một thẻ HTML, tên một ID, class hay kiểu khác. Sau đó nó sẽ áp dụng các thuộc tính lên vùng được chọn.

1.2.5. Giới thiệu về ngôn ngữ JavaScript

JavaScript là một ngôn ngữ lập trình được sử dụng để tạo ra những trang web tương tác. Nó được tích hợp và nhúng trong HTML. JavaScript cho phép kiểm soát các hành vi của trang web tốt hơn so với khi chỉ có một mình HTML.

CHƯƠNG 2: THIẾT KẾ VÀ THI CÔNG HỒ CÁ THÔNG MINH

2.1. GIỚI THIỆU VỀ PHẦN CỨNG

2.1.1. Chi tiết về các thành phần phần cứng

2.1.1.1. Đèn LED RGB

LED RGB gồm 3 led rất nhỏ được gộp lại thành 1 bóng LED RGB lớn, 3 LED này lần lượt có màu đỏ, lục, lam.

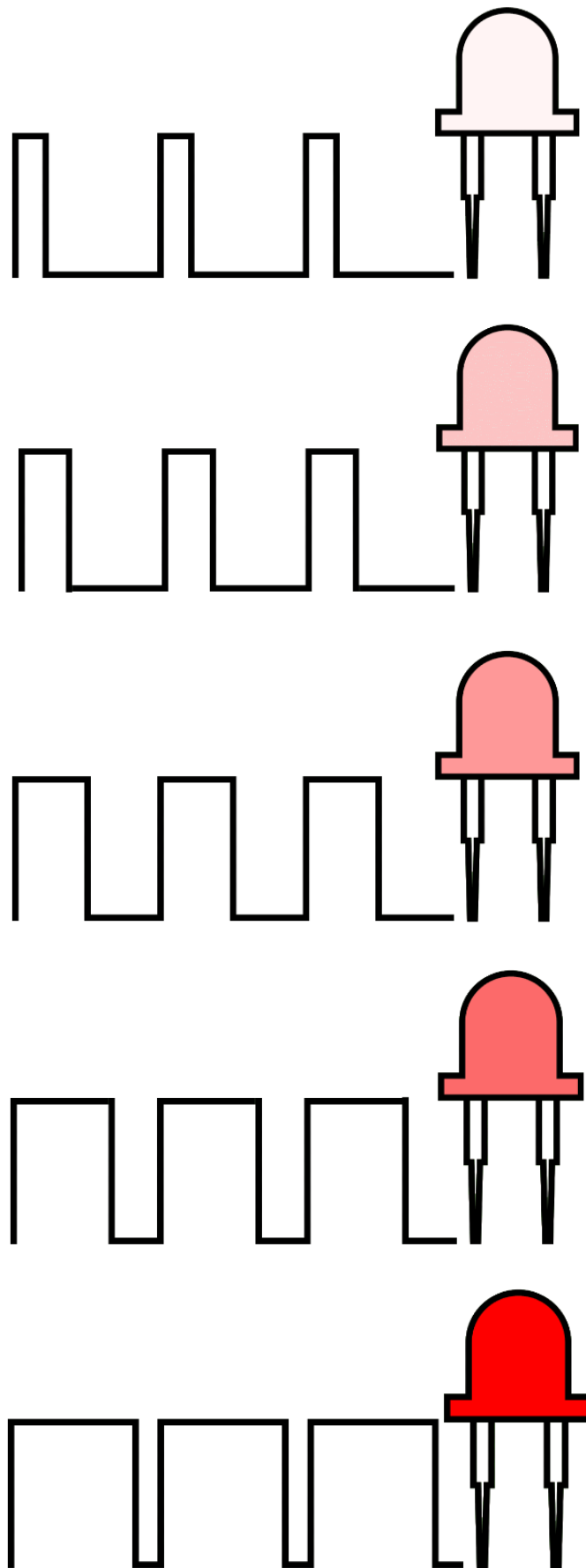


Hình 2. 1 Cấu tạo của LED RGB

LED RGB thay đổi màu sắc bằng cách điều chỉnh cường độ của 3 LED nhỏ bên trong theo thông số nhất định, và vì 3 led này đặt rất gần nhau nên khi chúng kết hợp lại sẽ làm cho mắt ta thấy một màu đã định thay vì là 3 màu riêng lẻ.

Để điều chỉnh cường độ của các LED nhỏ, ta có thể sử dụng tín hiệu PWM để điều khiển.

Khi dùng xung PWM để thay đổi các mức điện áp cao và thấp với tốc độ rất nhanh, mắt chúng ta sẽ không thể theo kịp tốc độ bật tắt của led và vì thế ta thấy được một cường độ màu nhất định.

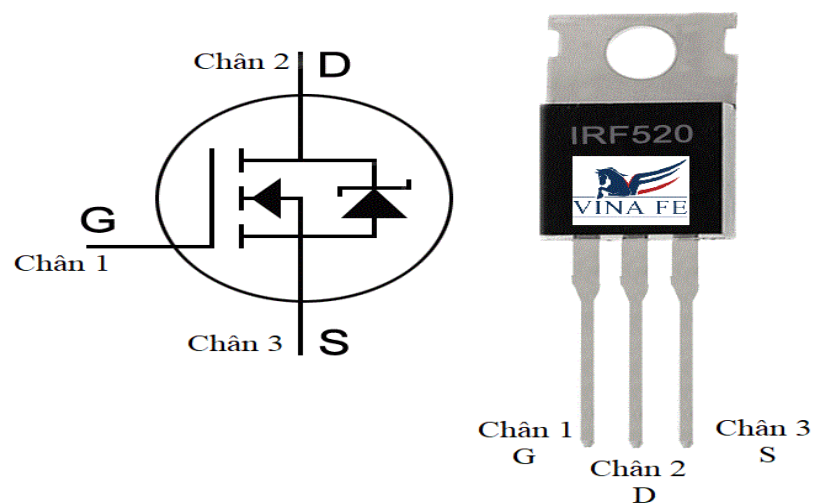


Hình 2. 2 Xung PWM điều chỉnh điện áp của LED RGB

2.1.1.2. IRF520N

IRF520 là một MOSFET điều khiển đầu ra của IC hoặc từ một nền tảng như Arduino hoặc Raspberry Pi. Tải tối đa mà transistor này có thể điều khiển là 9,2A với điện áp tải tối đa lên đến 100V. Transistor này cũng là một thiết bị chuyển mạch tốc độ cao do đó nó có thể được sử dụng trong bất kỳ ứng dụng nào cần tốc độ chuyển mạch, ví dụ như trong các mạch UPS. Hơn nữa, nó cũng có thể điều khiển tải lên đến 37A ở chế độ xung.

Sử dụng Mosfet trong mạch này với chức năng đóng ngắt điện áp điều khiển từ tín hiệu PWM đến các led.



Hình 2. 3 Sơ đồ chân của IRF520N

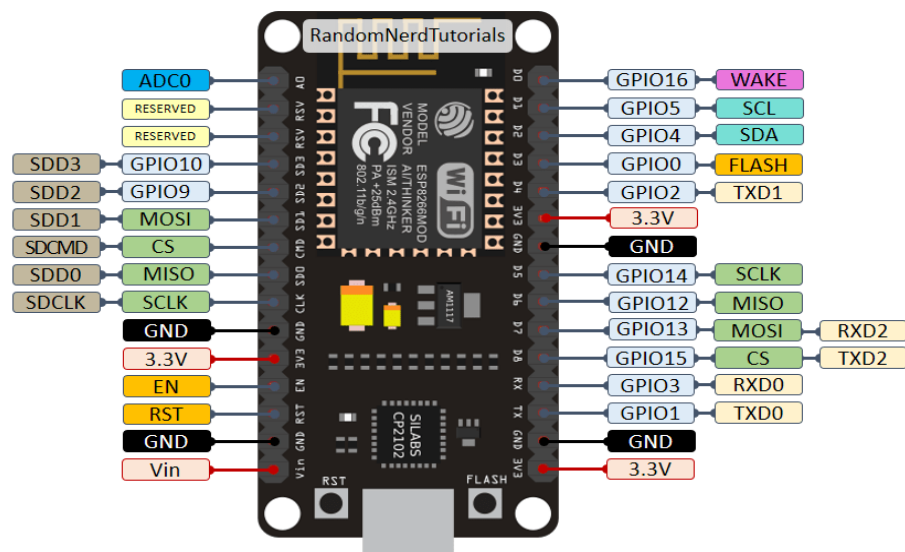
Thông số kỹ thuật:

- Loại gói: TO-220
- Loại transistor: Kênh N
- Điện áp tối đa từ cực máng đến cực nguồn: 100V
- Điện áp tối đa từ cực cổng đến cực nguồn: $\pm 20V$
- Dòng cực máng liên tục tối đa: 9.2A
- Dòng cực máng xung tối đa: 37A
- Công suất tiêu tán tối đa: 60W
- Điện áp tối thiểu cần thiết để dẫn: 2V đến 4V
- Nhiệt độ lưu trữ & hoạt động tối đa: -55 đến +175 độ C

2.1.1.3. NodeMCU ESP8266

Bảng phát triển NodeMCU ESP8266 đi kèm với mô-đun ESP-12E chứa chip ESP8266 có bộ vi xử lý Tensilica Xtensa 32-bit LX106 RISC. Bộ vi xử lý này hỗ trợ RTOS và hoạt động ở tần số xung nhịp có thể điều chỉnh từ 80MHz đến 160 MHz.

NodeMCU có 128 KB RAM và 4MB bộ nhớ Flash để lưu trữ dữ liệu và chương trình. Sức mạnh xử lý cao của nó với Wi-Fi / Bluetooth khiến nó trở nên thích hợp và lý tưởng cho các dự án IoT.



Hình 2. 4 Sơ đồ chân của ESP8266

Thông số kỹ thuật:

- WiFi: 2.4 GHz hỗ trợ chuẩn 802.11 b/g/n
- Điện áp hoạt động: 3.3V
- Điện áp vào: 5V thông qua cổng USB
- Số chân I/O: 11 (tất cả các chân I/O đều có Interrupt/PWM/I2C/One-wire, trừ chân D0)
- Số chân Analog Input: 1 (điện áp vào tối đa 3.3V)
- Bộ nhớ Flash: 4MB
- Giao tiếp: Cable Micro USB (tương đương cáp sạc điện thoại)
- Hỗ trợ bảo mật: WPA/WPA2
- Tích hợp giao thức TCP/IP
- Lập trình trên các ngôn ngữ: C/C++, Micropython,...

2.1.1.4. Động cơ Servo SG90

Động cơ Servo SG90 (Góc Quay 180) là Servo phổ biến dùng trong các mô hình điều khiển nhỏ và đơn giản như cánh tay robot. Động cơ có tốc độ phản ứng nhanh, được tích hợp sẵn Driver điều khiển động cơ, dễ dàng điều khiển góc quay bằng phương pháp điều độ rộng xung PWM.



Hình 2. 5 Động cơ Servo SG90

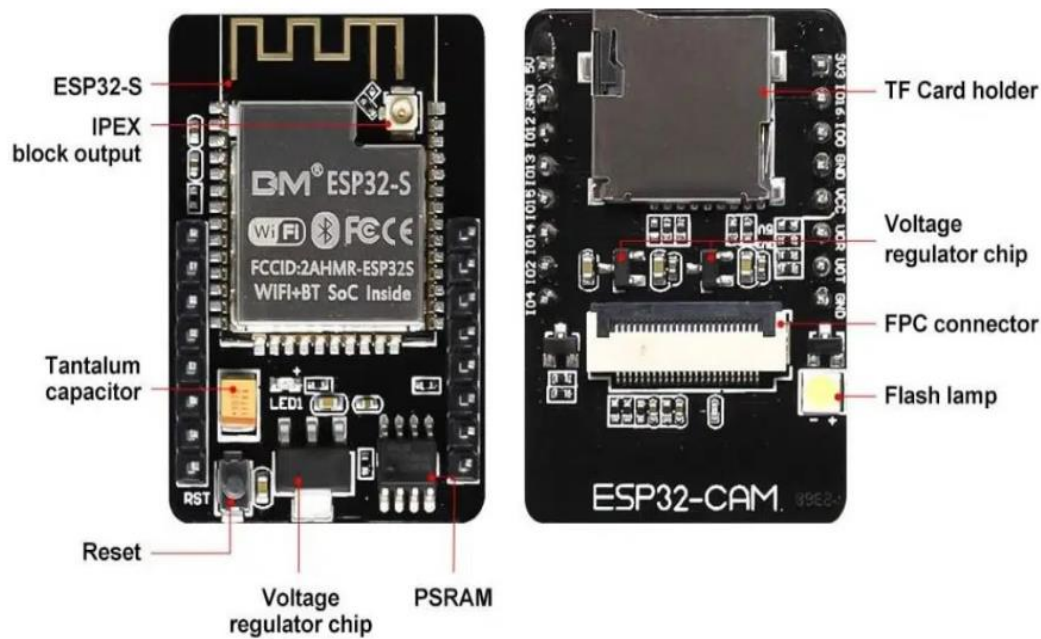
Thông số kỹ thuật:

- Điện áp hoạt động: 4.8V ~ 6V DC
- Tốc độ quay: 0.12 giây/60° (4.8V) , 0.1 giây/60° (6V)
- Mômen xoắn: 1.8kg/cm (4.8V) , 2.5kg/cm (6V)
- Góc quay: 180°
- Bánh răng: nhựa
- Kích thước: 22.5 * 11.8 * 30 mm
- Nhiệt độ hoạt động: 0°C ~ 55°C
- + Dây cam: Xung
- + Dây đỏ: Vcc (4.8V ~ 6V)
- + Dây đen: GND / 0V

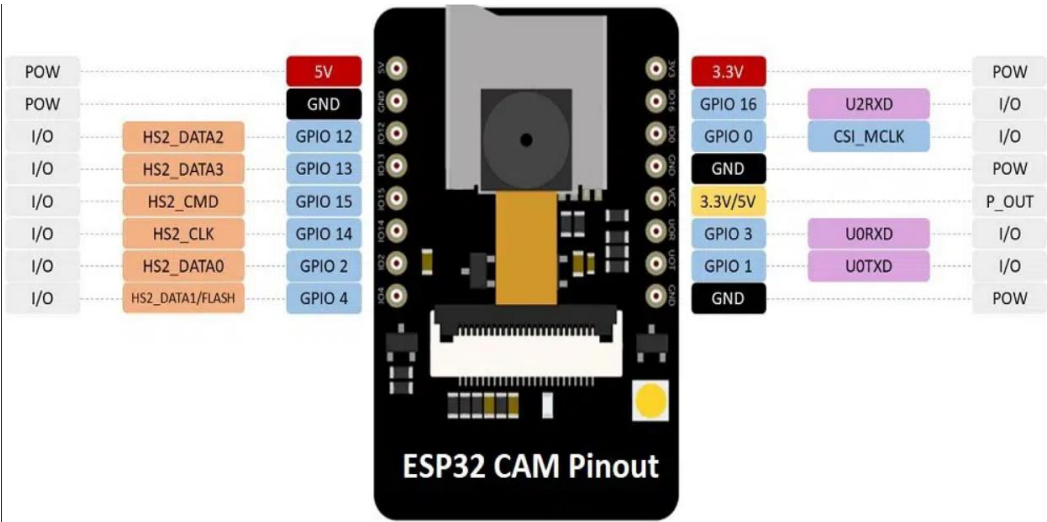
2.1.1.5. ESP32 - CAM AI - THINKER

ESP32-CAM AI-Thinker là phiên bản nâng cấp của ESP8266-01 được Espressif tung ra với nhiều tính năng. Module siêu nhỏ, công suất thấp có hai CPU LX6 32bit hiệu suất cao với kiến trúc vi mạch đường dẫn 7 lớp.

Các thành phần AI-THINKER của ESP32-CAM:



Hình 2. 6 Thành phần cấu tạo bo mạch ESP32-CAM AI-THINKER ở mặt trên và mặt dưới



Hình 2. 7 Sơ đồ chân của module AI-Thinker ESP32-CAM

Thông số kỹ thuật:

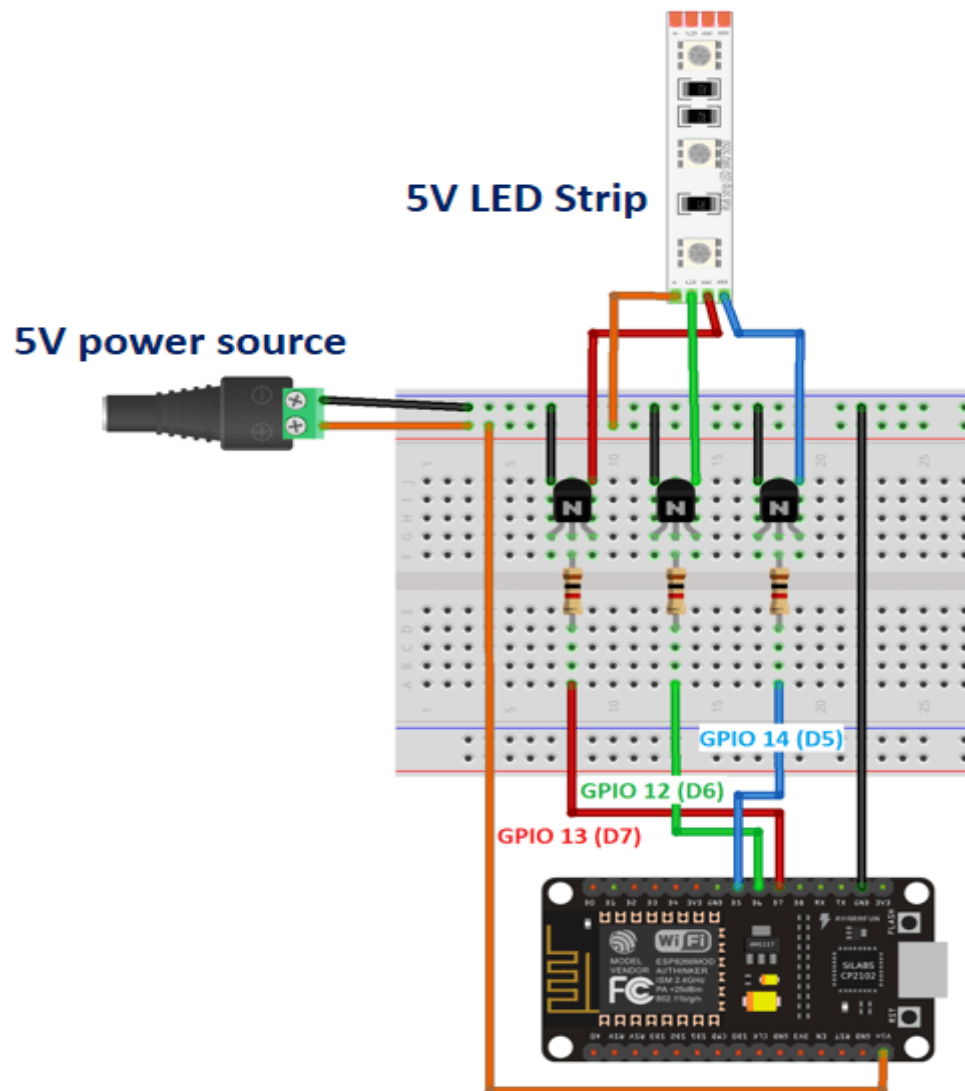
Đặc tính	Khả dụng
Module	ESP32-CAM AI-Thinker

Bộ xử lý	LX6 32-bit
Kiến trúc vi mạch	Đường dữ liệu 7 lớp
SPI Flash	32-bit
SRAM	520 KB
PSRAM	4MB
Dãy nhiệt độ hoạt động	- 20 ⁰ C – 85 ⁰ C
Điện áp hoạt động	5V
UART	1
Tốc độ truyền UART	115200 bps
SPI	Có
I2C	Có
PWM	Có
Wifi	802.11 b / g / n
Công suất truyền tải	<ul style="list-style-type: none"> • 802.11 b: 17 ± 2 dBm ở tốc độ 11Mbps. • 802.11 g: 14 ± 2 dBm ở tốc độ 54Mbps • 802.11 n: 13 ± 2 dBm ở MCS7
Bluetooth	Bluetooth 4.2 BR / EDR theo tiêu chuẩn BLE
Hỗ trợ thẻ TF	Hỗ trợ tối đa 4G
Bảo mật dữ liệu	WPA / WPA2 / WPA2-Enterprise / WPS
Định dạng hình ảnh đầu ra	BMP, GRAYSCALE, JPEG (chỉ hỗ trợ OV2640)
Dải quang phổ	2412 MHz – 2484 MHz
Độ lợi ăng ten trên bo mạch PCB	2dBi
Số cổng đầu vào / đầu ra	9
Độ nhảy	<ul style="list-style-type: none"> • CCK 1 Mb / s: -90dBm. • CCK 11 Mb / s: -85dBm • 6 Mb / s (1/2 BPSK): -88dBm • 54 Mb / s (3/4 64-QAM): -70dBm

	<ul style="list-style-type: none"> MCS7 (65 Mb / s, 72,2 Mb / s): -67dBm
Công suất tiêu thụ	<p>Khi tắt đèn flash tiêu tốn: 180mA ở 5V. Bật đèn flash với độ sáng tối đa: 310mA ở 5V. Chế độ ngủ sâu (sleep-mode): có mức tiêu thụ điện năng thấp nhất có thể đạt 6mA ở 5V. Modem-sleep mode: <20mA ở 5V, slight-sleep mode: <6.7mA ở 5V</p>
Tần số xung nhịp	240 MHz (tối đa)
Loại package	DIP-16
Kích cỡ package	27 40,5 x 4,5 (± 0,2) mm

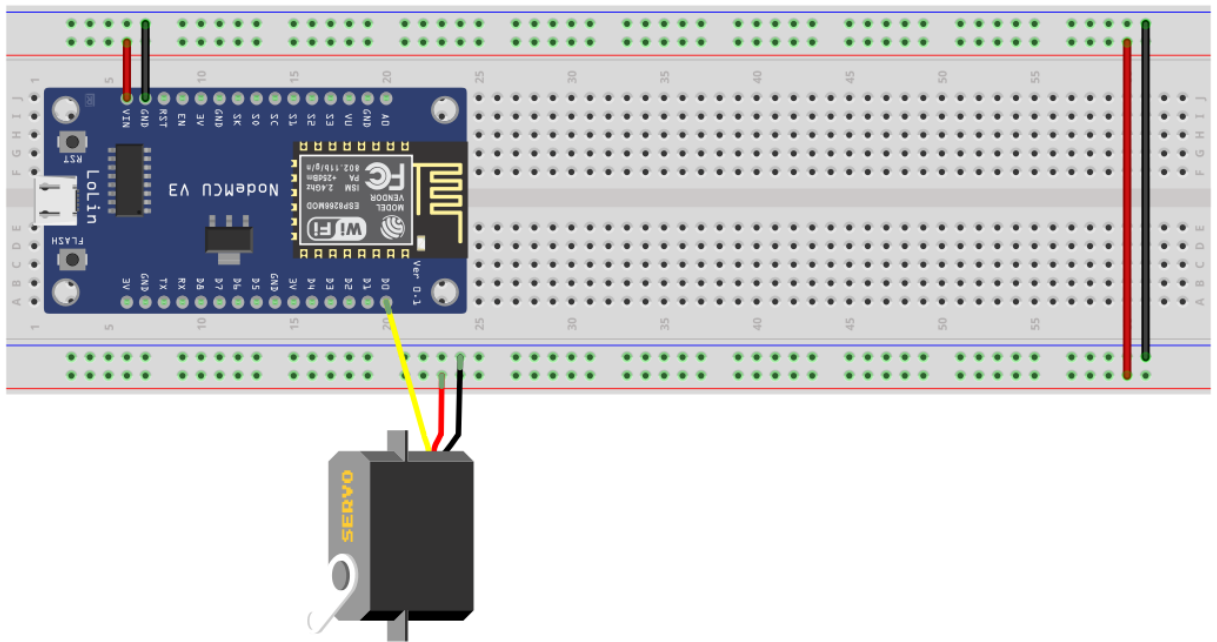
2.1.2. Sơ đồ nối dây

2.1.1.1. Sơ đồ nối dây của LED RGB



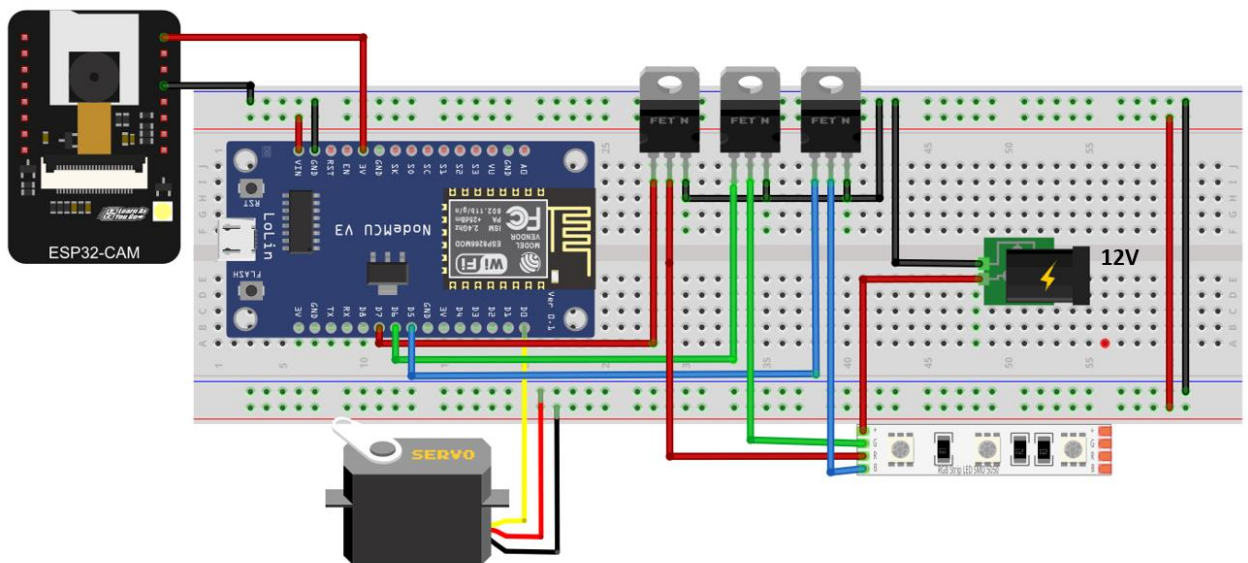
Hình 2. 8 Sơ đồ nối dây của LED RGB

2.1.1.2. Sơ đồ nối dây của Servo SG90



Hình 2. 9 Sơ đồ nối dây của Servo SG90

2.1.1.3. Sơ đồ nối dây của cả hệ thống



Hình 2. 10 Sơ đồ nối dây của cả hệ thống

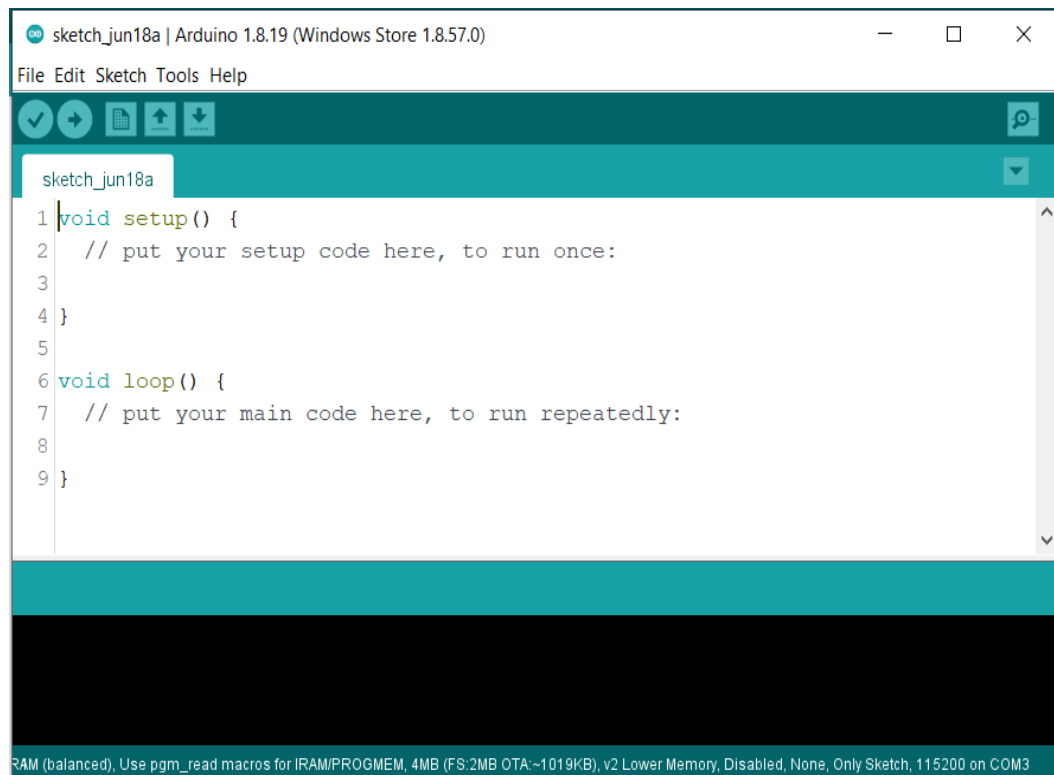
2.2. PHẦN MỀM

2.2.1. Giới thiệu phần mềm sử dụng

2.2.1.1. Giới thiệu về Arduino IDE

2.2.1.1.1. Arduino IDE là gì

Arduino IDE là một phần mềm mã nguồn mở chuyên lập trình cho các kit Arduino với giao diện trực quan, dễ sử dụng và hoàn toàn miễn phí.



Hình 2. 11 Giao diện phần mềm Arduino IDE

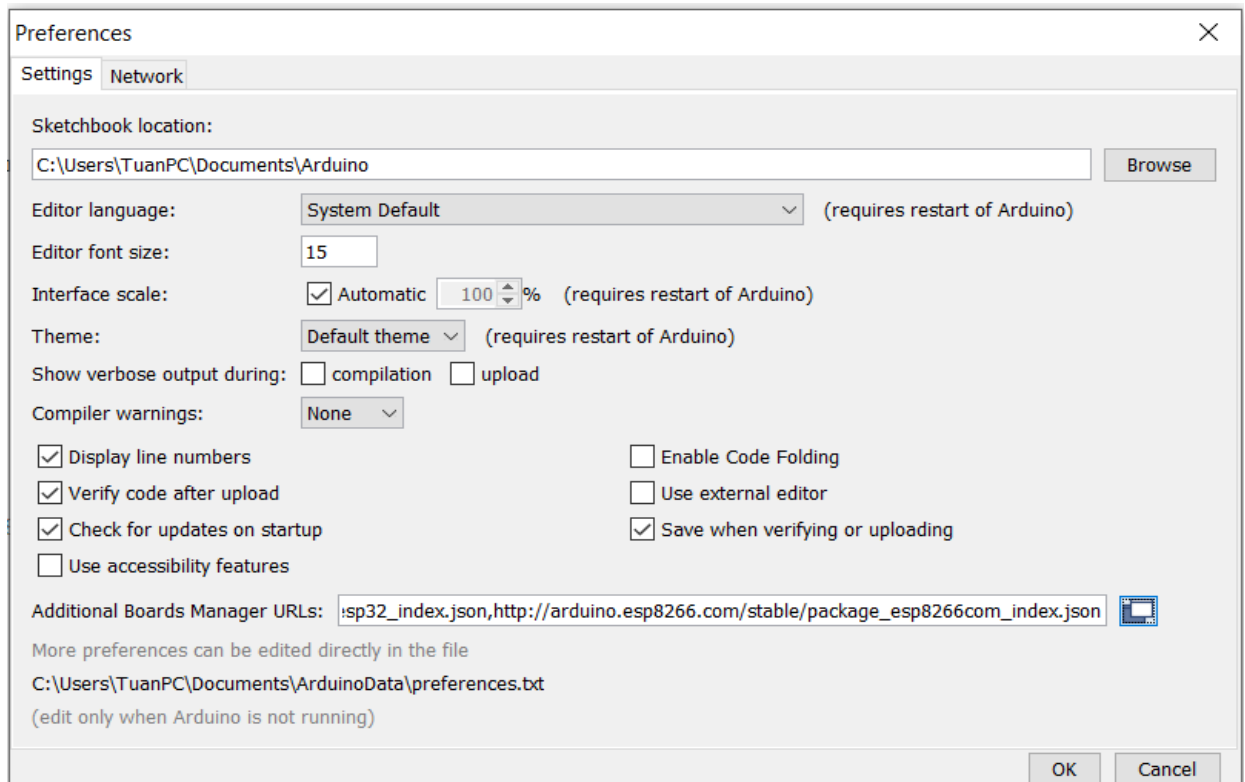
Chúng em sử dụng Arduino IDE phiên bản 1.8.57 để thực hiện lập trình cho hệ thống điều khiển.

2.2.1.1.2. Cách cài đặt ESP8266 và ESP32 Camera vào Arduino IDE

- Trong Arduino IDE chọn File -> Preferences.
- Nhập dòng sau vào trong “Additional Board Manager URLs”:

https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json

http://arduino.esp8266.com/stable/package_esp8266com_index.json



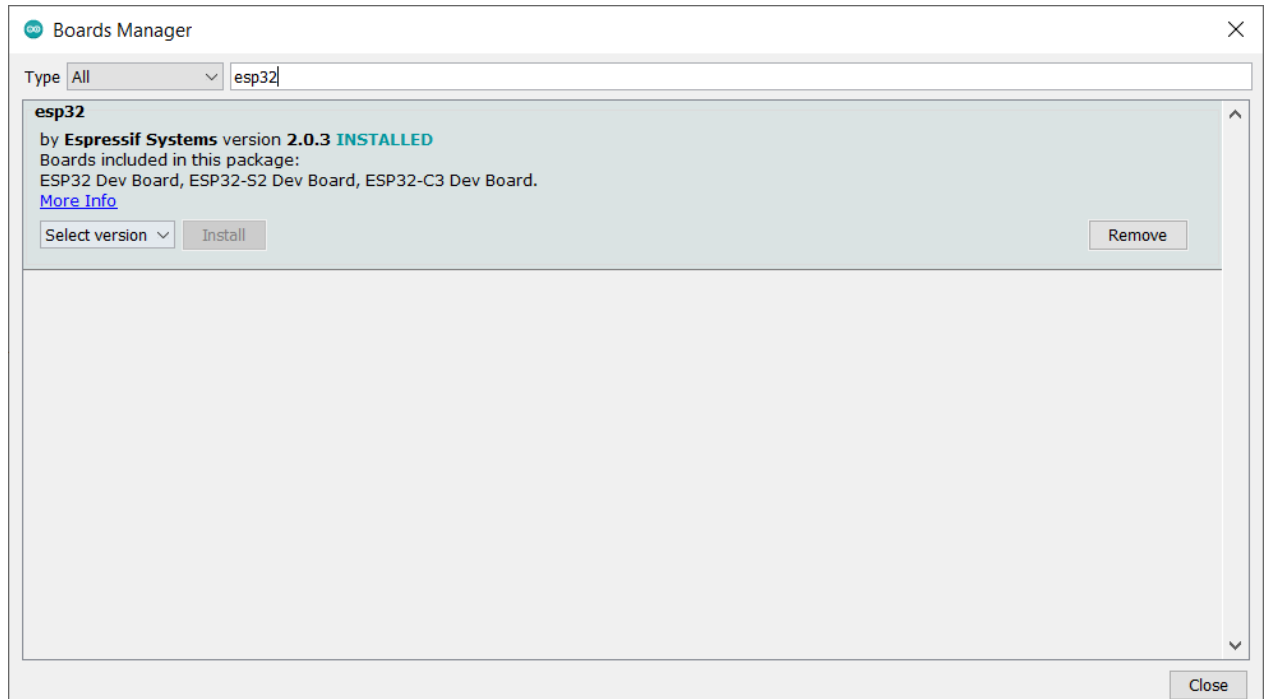
Hình 2. 12 Giao diện sau khi nhập dòng trên vào “Additional Board Manager URLs”

- Vào Tool -> Board -> Board Manager... tìm và cài đặt Esp8266 như hình bên dưới:



Hình 2. 13 Giao diện tìm kiếm Esp8266

- Cũng ở giao diện Board Manager... tìm và cài đặt Esp32 Camera như hình bên dưới:



Hình 2. 14 Giao diện tìm kiếm Esp32 Camera

2.2.1.2. Giới thiệu về Ngrok

2.2.1.2.1. Ngrok là gì

Ngrok là một công cụ tạo ra một đường hầm từ máy bạn (desktop, localhost) đi qua hệ thống Firewall, giúp từ internet có thể truy cập vào máy trạm.

Ngrok cho phép bạn để lộ máy chủ cục bộ sau bức tường lửa thông qua đường hầm bảo mật cho Internet công cộng.

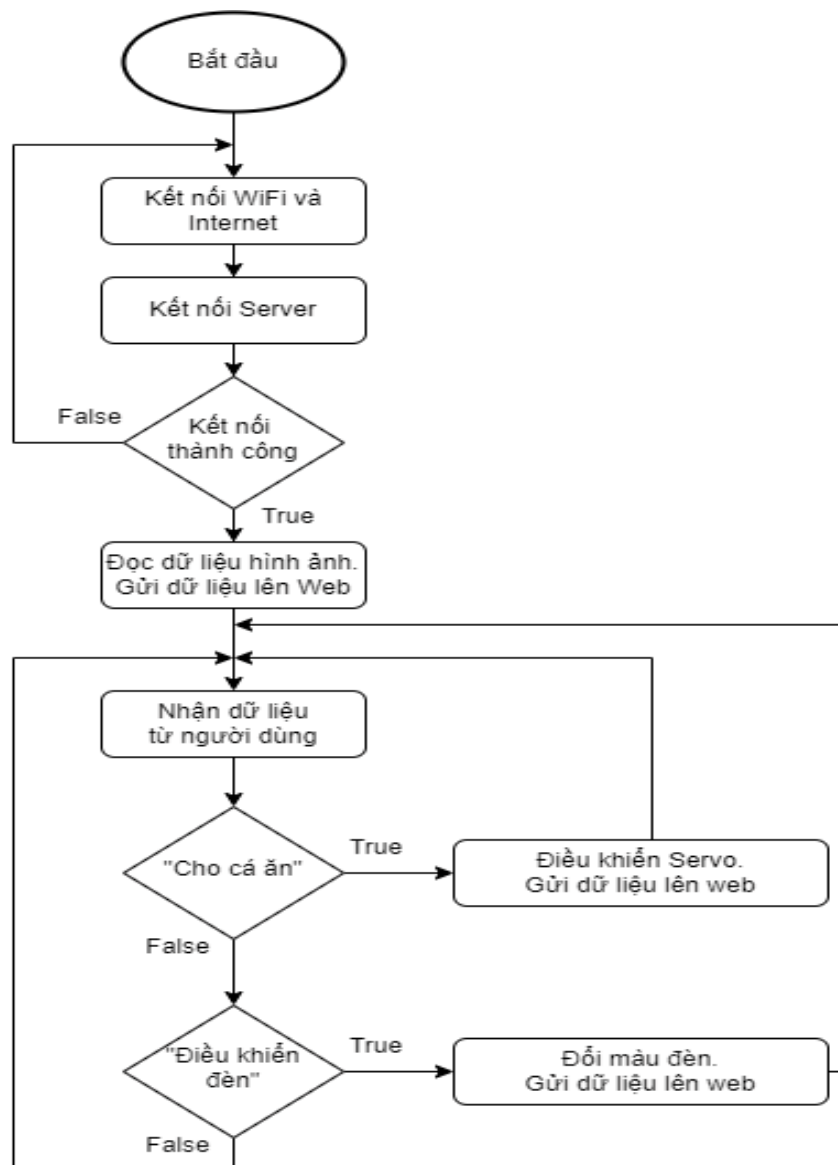
Bạn có thể dùng ngrok để giúp từ ngoài internet truy cập vào một trang web (máy chủ http) đang chạy thử trên máy của bạn, mà không nhất thiết phải triển khai web trên một server thực sự.

2.2.1.2.2. Cách cài đặt Ngrok

- Vào đường link sau: <https://dashboard.ngrok.com/get-started/setup>
- Đăng ký tài khoản Ngrok.
- Lựa chọn phiên bản tương thích với hệ điều hành và tải về. Ở đây nhóm mình sẽ tải phiên bản của Window10.
- Sau khi cài đặt file zip và giải nén, các bạn sẽ có một file "ngrok.exe".

- Ở đây nhóm sẽ sử dụng phiên bản miễn phí của ngrok. Vậy thì ở phiên bản miễn phí, chúng ta sẽ bị giới hạn ở các điểm:
 - Chỉ hỗ trợ HTTP và TCP.
 - Subdomain ngẫu nhiên.
 - Chỉ chạy được 1 process ngrok tại 1 thời điểm.
 - Giới hạn 4 tunnel / 1 process.
 - Giới hạn 40 connections / 1 phút.

2.2.2. Lưu đồ thuật toán



Hình 2. 15 Lưu đồ thuật toán chính của hệ thống

2.2.3. Lập trình cho hệ thống

2.2.3.1. Led RGB

Một số dòng code chính:

- Xác định các biến chuỗi để giữ các tham số R, G và B từ request:

```
String redString = "0";
String greenString = "0";
String blueString = "0";
```

- Gán các biến với các chân GPIO kiểm soát các tham số R, G và B (ở đây là GPIO12, GPIO13 và GPIO14):

```
const int redPin = 13;    // 13 corresponds to GPIO13
const int greenPin = 12;  // 12 corresponds to GPIO12
const int bluePin = 14;   // 14 corresponds to GPIO14
```

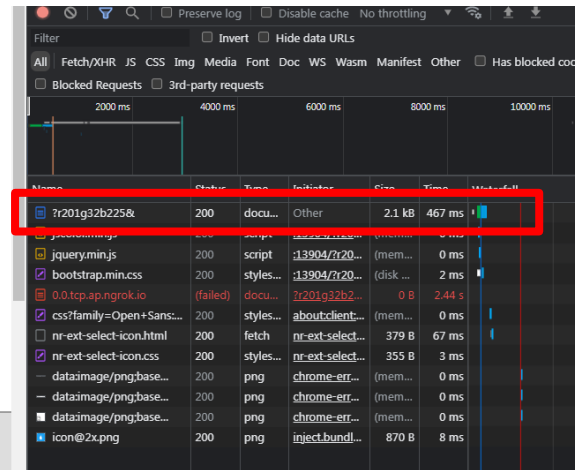
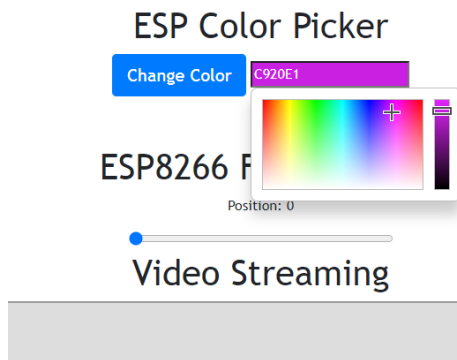
- Hiển thị giao diện bộ chọn màu trên trang web và tạo yêu cầu dựa trên màu đã chọn:

```
client.println("<!DOCTYPE html><html>");
client.println("<head><meta name='viewport' content='width=device-width, initial-scale=1'>");
client.println("<link rel='icon' href='data:;'>");
client.println("<link rel='stylesheet' href='https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css'>");
client.println("<script src='https://cdnjs.cloudflare.com/ajax/libs/jscolor/2.0.4/jscolor.min.js'></script>");
client.println("</head><body><h1><br>Project: Aquarium Control</h1>");
client.println("</head><body><h1><br></h1>");
client.println("</head><body><div class='container'><h1 style='text-align:center;'>ESP Color Picker</h1>");
client.println("<a class='btn btn-primary btn-lg' href='#' id='change_color' role='button'>Change Color</a> ");
client.println("<input class='jscolor {onFineChange:'update(this)'}' id='rgb'></div>");
client.println("<script>function update(picker) {document.getElementById('rgb').innerHTML = Math.round(picker.rgb[0]) + ', ' + Math.round(picker.rgb[1]) + ' ' + Math.round(picker.rgb[2]);}");
client.println("</script>");
```

- Sau khi chọn màu, bạn sẽ nhận được một yêu cầu với định dạng như sau (ở đây ví dụ ta đang chọn màu với Red = 201 | Green = 32 | Blue = 255):

```
/?r201g32b255&
```


Project: Aquarium Control



- Sau đó, ta cần chia chuỗi này để nhận các tham số R, G và B. Các tham số này sẽ được lưu trong các biến redString, greenString, blueString và có thể có giá trị từ 0 đến 255:

```
pos3 = header.indexOf('r');
pos4 = header.indexOf('g');
pos5 = header.indexOf('b');
pos6 = header.indexOf('&');
redString = header.substring(pos3+1, pos4);
greenString = header.substring(pos4+1, pos5);
blueString = header.substring(pos5+1, pos6);
```

- Biến header là biến lưu trữ các request HTTP.
- Hàm indexOf() lấy index tại 'r', 'g', 'b' và '&'.
- Hàm substring() lấy chuỗi con theo giới hạn đã định.

Vd: pos3 = 'r' => lấy tại pos3+1 = '2' cho đến pos4 = 'g' => lấy chuỗi con '201'.

- Để điều khiển dải màu với ESP8266, chúng ta sử dụng hàm analogWrite() để tạo ra tín hiệu PWM với giá trị được giải mã từ yêu cầu HTTP:

```
analogWrite(redPin, redString.toInt());
analogWrite(greenPin, greenString.toInt());
analogWrite(bluePin, blueString.toInt());
```

- Vì chúng ta nhận được các giá trị trong một biến chuỗi nên cần phải chuyển chúng thành số nguyên bằng phương thức toInt()

2.2.3.2. ESP32 Camera

Một số dòng code chính:

```
const char* ssid = "Tuan T2";
const char* password = "02363691412";
```

⇒ Nhập tài khoản và mật khẩu Wifi cần kết nối.

```
#elif defined(CAMERA_MODEL_AI_THINKER)
#define PWDN_GPIO_NUM    32
#define RESET_GPIO_NUM  -1
#define XCLK_GPIO_NUM    0
#define SIOD_GPIO_NUM    26
#define SIOC_GPIO_NUM    27

#define Y9_GPIO_NUM      35
#define Y8_GPIO_NUM      34
#define Y7_GPIO_NUM      39
#define Y6_GPIO_NUM      36
#define Y5_GPIO_NUM      21
#define Y4_GPIO_NUM      19
#define Y3_GPIO_NUM      18
#define Y2_GPIO_NUM       5
#define VSYNC_GPIO_NUM   25
#define HREF_GPIO_NUM    23
#define PCLK_GPIO_NUM    22
```

⇒ Định nghĩa các chân của esp32 camera AI_THINKER.

```
static esp_err_t stream_handler(httpd_req_t *req){
    camera_fb_t * fb = NULL;
    esp_err_t res = ESP_OK;
    size_t _jpg_buf_len = 0;
    uint8_t * _jpg_buf = NULL;
```

⇒ Tạo con trỏ fb có kiểu dữ liệu là camera_fb_t trở về NULL. con trỏ này trỏ tới vị trí của data.

- ⇒ Set kết quả là OK.
- ⇒ Khởi tạo độ dài của buffer chứa data là 0.

```

120 | res = httpd_resp_set_type(req, _STREAM_CONTENT_TYPE);
121 | if(res != ESP_OK) {
122 |     return res;

```

- ⇒ Dòng 120: setup http content type cho camera.
- ⇒ Dòng 121: nếu không được thì sẽ trả về res Error.
- ⇒ Dòng 122: sau đó return lại hàm.
- ⇒ Đây là hàm hanndler nên khi return lỗi thì camera sẽ báo lỗi.

```

126 |     fb = esp_camera_fb_get();

```

- ⇒ Nơi con trỏ fb trở đến.

```

131 | if (!fb) {
132 |     Serial.println("Camera capture failed");
133 |     res = ESP_FAIL;
134 | } else {
135 |     if(fb->width > 400){
136 |         if(fb->format != PIXFORMAT_JPEG){
137 |             bool jpeg_converted = frame2jpg(fb, 80, &_jpg_buf, &_jpg_buf_len);
138 |             esp_camera_fb_return(fb);
139 |             fb = NULL;
140 |             if(!jpeg_converted){
141 |                 Serial.println("JPEG compression failed");
142 |                 res = ESP_FAIL;
143 |             }
144 |         } else {
145 |             _jpg_buf_len = fb->len;
146 |             _jpg_buf = fb->buf;
147 |         }

```

- ⇒ Đoạn code từ 131-147 là 1 đoạn lệnh.
- ⇒ Lệnh esp sẽ chụp ảnh lại.
- ⇒ Sau đó con trỏ fb sẽ trở đến nơi lưu trữ những ảnh đã chụp.
- ⇒ Dòng 135: nếu con trỏ fb khác NULL tức là có dữ liệu, khi đó đoạn code từ dòng 136-138 sẽ thực hiện định dạng lại ảnh đã chụp.
- ⇒ Dòng 140: nếu con trỏ fb vẫn NULL, tức là không có dữ liệu, nghĩa là sẽ không chụp ảnh được.

```

150     if(res == ESP_OK){
151         size_t hlen = snprintf((char *)part_buf, 64, _STREAM_PART, _jpg_buf_len);
152         res = httpd_resp_send_chunk(req, (const char *)part_buf, hlen);
153     }
154     if(res == ESP_OK){
155         res = httpd_resp_send_chunk(req, (const char *)_jpg_buf, _jpg_buf_len);
156     }
157     if(res == ESP_OK){
158         res = httpd_resp_send_chunk(req, _STREAM_BOUNDARY, strlen(_STREAM_BOUNDARY));
159     }

```

⇒ Nếu res OK thì sẽ gửi request về server.

```

160     if(fb) {
161         esp_camera_fb_return(fb);
162         fb = NULL;
163         _jpg_buf = NULL;
164     } else if(_jpg_buf) {
165         free(_jpg_buf);
166         _jpg_buf = NULL;
167     }
168     if(res != ESP_OK) {
169         break;
170     }
171     //Serial.printf("MJPG: %uB\n", (uint32_t) (_jpg_buf_len));
172 }
173 return res;

```

⇒ Nếu như mà có ảnh thì sẽ reset lại hết vì đoạn code ở dòng 150-159 đã gửi ảnh đến server rồi nên không cần lưu ảnh nữa.

```

202     config.pin_d0 = Y2_GPIO_NUM;
203     config.pin_d1 = Y3_GPIO_NUM;
204     config.pin_d2 = Y4_GPIO_NUM;
205     config.pin_d3 = Y5_GPIO_NUM;
206     config.pin_d4 = Y6_GPIO_NUM;
207     config.pin_d5 = Y7_GPIO_NUM;
208     config.pin_d6 = Y8_GPIO_NUM;
209     config.pin_d7 = Y9_GPIO_NUM;
210     config.pin_xclk = XCLK_GPIO_NUM;
211     config.pin_pclk = PCLK_GPIO_NUM;
212     config.pin_vsync = VSYNC_GPIO_NUM;
213     config.pin_href = HREF_GPIO_NUM;
214     config.pin_sscb_sda = SIOD_GPIO_NUM;
215     config.pin_sscb_scl = SIOC_GPIO_NUM;
216     config.pin_pwdn = PWDN_GPIO_NUM;
217     config.pin_reset = RESET_GPIO_NUM;

```

⇒ Cấu hình chân ngõ ra của ESP32 camera AI_THINKER.

```

218     config.xclk_freq_hz = 20000000;
219     config.pixel_format = PIXFORMAT_JPEG;

```

⇒ Cấu hình tần số xung clock là 20MHz và định dạng ảnh đầu ra là JPEG.

```

222     config.frame_size = FRAMESIZE_SVGA;
223     config.jpeg_quality = 10;
224     config.fb_count = 2;

```

⇒ Cấu hình độ phân giải của video là SVGA và chất lượng ảnh là mức 10.

```

233     esp_err_t err = esp_camera_init(&config);
234     if (err != ESP_OK) {
235         Serial.printf("Camera init failed with error 0x%x", err);
236         return;

```

⇒ Gắn config vào camera, nếu config ok thì sẽ kết nối wifi không thì báo lỗi.

2.2.3.3. Servo

Các hàm servo sử dụng trong chương trình:

- myservo.attach();
- myservo.write();

Một số dòng code chính:

- static const int servoPin = 16;
⇒ định nghĩa chân điều khiển Servo là chân 16.
- String valueString = String(5);
⇒ Tạo biến valueString có kiểu String và giá trị khởi tạo là 5.
- Myservo.attach (servoPin);
⇒ Trước khi điều khiển được Servo cần khai báo hàm này, truyền tham số servoPin là chân 16, để cài đặt chức năng điều khiển cho Servo.
- myservo.write(valueString.toInt());
Serial.println(valueString);
⇒ Hàm này dùng để tùy chỉnh góc quay của Servo với tham số truyền vào là góc quay dao động từ 0 – 180.
⇒ valueString.toInt() là hàm chuyển đổi một chuỗi thành một số. Hàm này sẽ trả về số nguyên đầu tiên nó tìm được với yêu cầu ký tự đầu tiên phải là 1 ký tự số.
- if(header.indexOf("GET/ ?value=") >= 0) {
 pos1 = header.indexOf('=');
 pos2 = header.indexOf('&');
 valueString = header.substring(pos1+1, pos2);
⇒ header.indexOf("GET/ ?value=") sẽ trả về vị trí của chuỗi "GET /?value" từ vị trí thứ 0.
⇒ header.indexOf('='); sẽ trả về vị trí của ký tự =
⇒ header.indexOf('&'); sẽ trả về vị trí của ký tự &

Giao diện điều khiển Servo

```
client.println("<p>Position: <span id=\"servoPos\"></span></p>");
```

- ⇒ Hiển thị dòng text "Position" có Id = "servoPos".

```
client.println("<input type=\"range\" min=\"0\" max=\"180\" class=\"slider\"  
id=\"servoSlider\" onchange=\"servo(this.value)\" value=\""+valueString+"\"/>")
```

- ⇒ Tạo 1 thanh Slider chia giá trị từ 0 – 180, Class = “slider”, Id = “servoSlider” và value của thanh Slider là valueString. Khi kéo thanh slider thì giá trị valueString sẽ được thay đổi.

Tiếp theo là phần code JavaScript.

```
client.println("<script>var slider = document.getElementById(\"servoSlider\");");
```

- ⇒ Tạo biến Slider tìm và gán cho nó phần tử có Id = “servoslider”.

```
client.println("var servoP = document.getElementById(\"servoPos\");  
servoP.innerHTML = slider.value;");
```

- ⇒ Tạo biến servoP tìm và gán cho nó phần tử có Id = “servoPos”.
- ⇒ Thêm thuộc tính value có class “slider” vào servoPos.
- ⇒ Như vậy khi kéo thanh Slider thì giá trị valueString sẽ được cập nhật, đồng thời hiển thị giá trị đó tại “Position:”.

ESP8266 Fish Feeder

Position: 0



```
client.println("slider.oninput = function()  
{ slider.value = this.value; servoP.innerHTML = this.value; }");
```

- ⇒ Sự kiện **oninput** là một sự kiện trong javascript, sự kiện xảy ra dựa vào giá trị mà ta nhập vào có làm hàm function được thực thi hay không. Ở đây khi ta kéo thanh slider, hàm function sẽ thực thi và cập nhật giá trị value của slider ngay lập tức. Tức là con số hiển thị góc quay của Servo sẽ được cập nhật ngay lập tức khi ta kéo thanh slider.

2.2.3.4. Code file .BAT

- Mục đích: Giúp đơn giản hoá các bước sử dụng sản phẩm và thân thiện với người dùng.
- Vào nơi lưu thư mục ngrok. Tạo 2 file .txt và đặt tên là ESP32-Cam và ESP8266.
- Mở file ESP32-Cam và dán đoạn code sau:

ESP32-CAM - Notepad

File Edit Format View Help

```
@echo off
title Aquarium Config
echo ----- ESP32-CAM CONFIG -----
echo          Team Nguyen Quoc Viet
echo
set str=ngrok tcp xxx --authtoken 28kbnbqYD9U7HEE9NJraXjXVz40_3cpxZk2rTZ378JNCvmCPB
set /p var1="Input IP ESP32-CAM: "
call set str=%%str:xxx=%var1%%
::echo %str%
cd \ngrok
%str%
pause
```

- Mở file ESP8266 và dán đoạn code sau:

ESP8266 - Notepad

File Edit Format View Help

```
@echo off
title Aquarium Config
echo ----- ESP8266 CONFIG -----
echo          Team Nguyen Quoc Viet
echo

set str=ngrok tcp xxx --authtoken 27m3Vwa1Q6ZrIlxSXg9zdaiAlYm_7Bqe9Q7iTrD4DUtTyTB2J
set /p var1="Input IP ESP8266: "
call set str=%%str:xxx=%var1%%
::echo %str%
cd \ngrok
%str%
pause
```

- Lưu 2 file với đuôi là .bat

2.3. THI CÔNG ĐỒ ÁN

2.3.1. Thiết kế hồ cá

Để thiết một hồ cá không quá lớn, tiết kiệm chi phí và dễ dàng di chuyển, nhóm em đã quyết định dùng hộp nhựa cỡ lớn để làm hồ cá.



Hình 2. 16 Hộp nhựa cỡ lớn làm hồ cá

2.3.2. Thiết kế hộp bảo vệ

Thiết kế hộp bảo vệ là để bảo vệ mạch điều khiển, đảm bảo an toàn cho người sử dụng và giúp bộ điều khiển có tính thẩm mỹ hơn. Để tiết kiệm chi phí và đáp ứng những yêu cầu trên, nhóm em đã sử dụng bìa cứng màu trắng để làm hộp đựng bản mạch



Hình 2. 17 Hộp bảo vệ mạch điều khiển

2.3.3. Thiết kế hộp đựng thức ăn cho cá

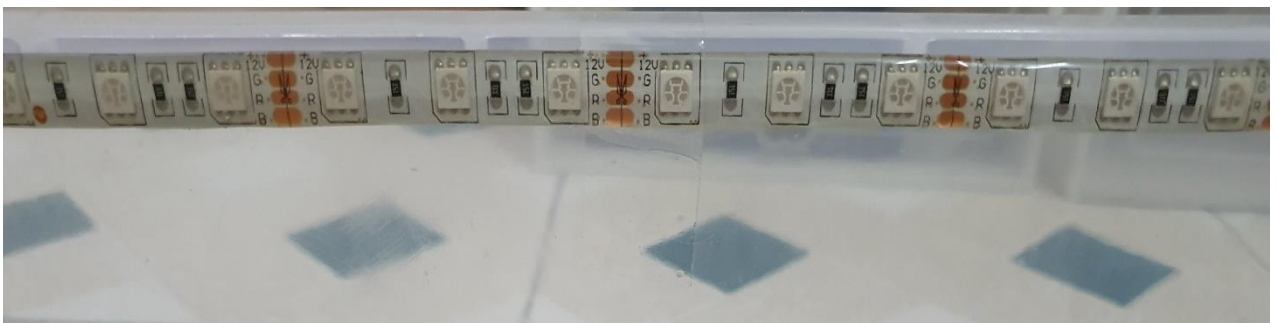
Cho cá ăn là một trong những mục đích quan trọng của đồ án này. Để lượng thức ăn có thể xuống một cách vừa phải, phù hợp với diện tích hồ và số lượng cá, nhóm em đã sử dụng hộp nhựa tái chế nhỏ và khoét đường rãnh để thức ăn có thể xuống không quá nhiều.



Hình 2. 18 Hộp đựng thức ăn cho cá

2.3.4. Thiết kế dải đèn LED RGB

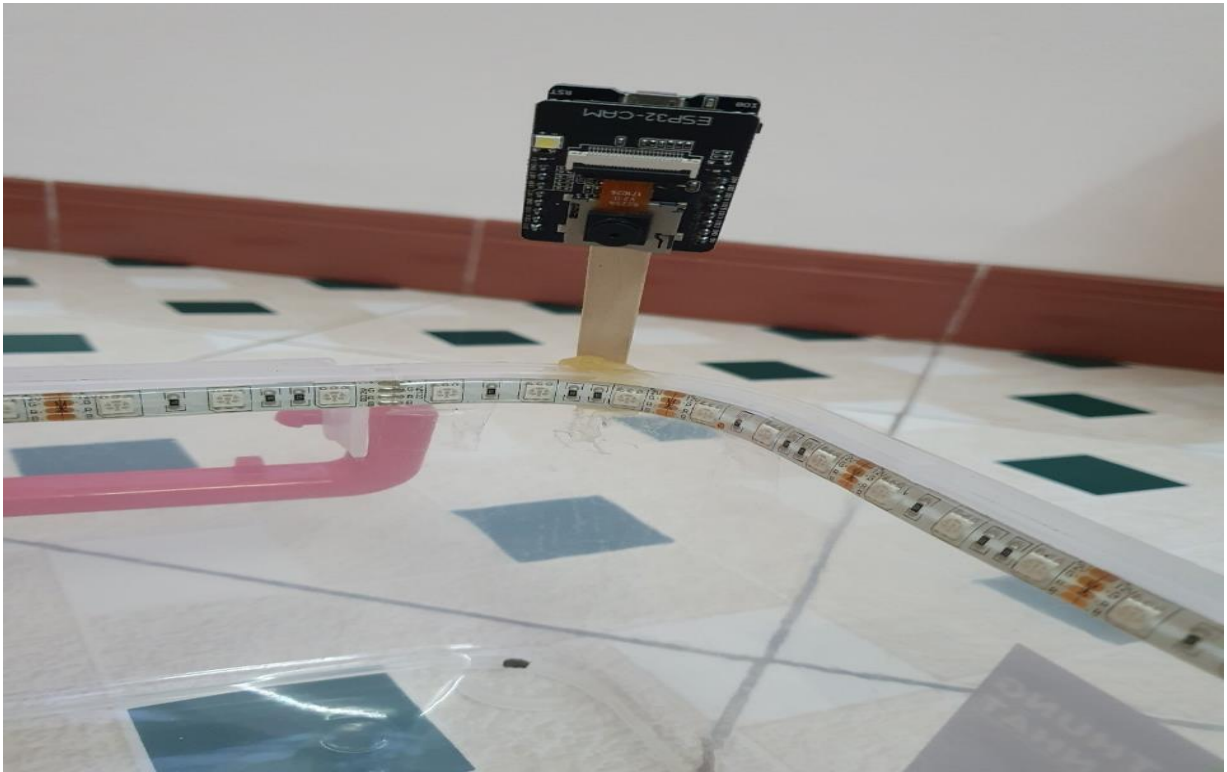
Đèn LED RGB được sử dụng trong đồ án lần này có mục đích vừa để trang trí và vừa để sưởi ấm cho cá nhờ tận dụng nhiệt toả ra từ các bóng đèn. Chúng em đã sử dụng 1m dây LED RGB và dán xung quanh mép viền của hồ cá.



Hình 2. 19 Thiết kế LED RGB cho cá

2.3.5. Bố trí ESP 32 Camera

ESP32 Camera được dùng để quan sát hồ cá, giúp người sử dụng có thể kiểm tra xem thức ăn có rơi xuống mỗi khi điều khiển cho cá ăn hay không. Ngoài ra sẽ giúp cho người sử dụng có thể ngắm những chú cá của mình từ xa. Để đáp ứng được 2 yêu cầu trên thì ESP32 Camera phải được bố trí sao cho người dùng có thể nhìn được hồ cá một cách bao quát nhất. Nhóm em đã lắp đặt ESP32 Camera ở vị trí trên cao so với hồ cá và được đặt ở phía góc của hồ cá giúp người dùng có thể quan sát một cách bao quát nhất



Hình 2. 20 Lắp đặt ESP32 Camera

2.3.6. Mô hình tổng quát của hồ cá

Dưới đây là mô hình hồ cá sau khi đã lắp đặt xong tất cả các linh kiện

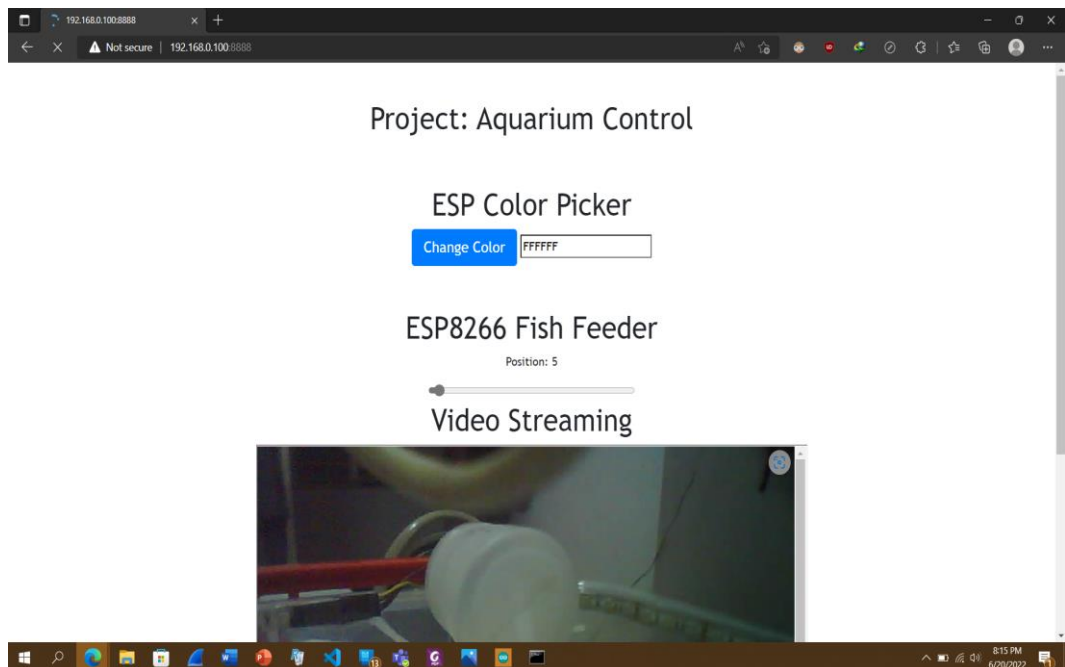


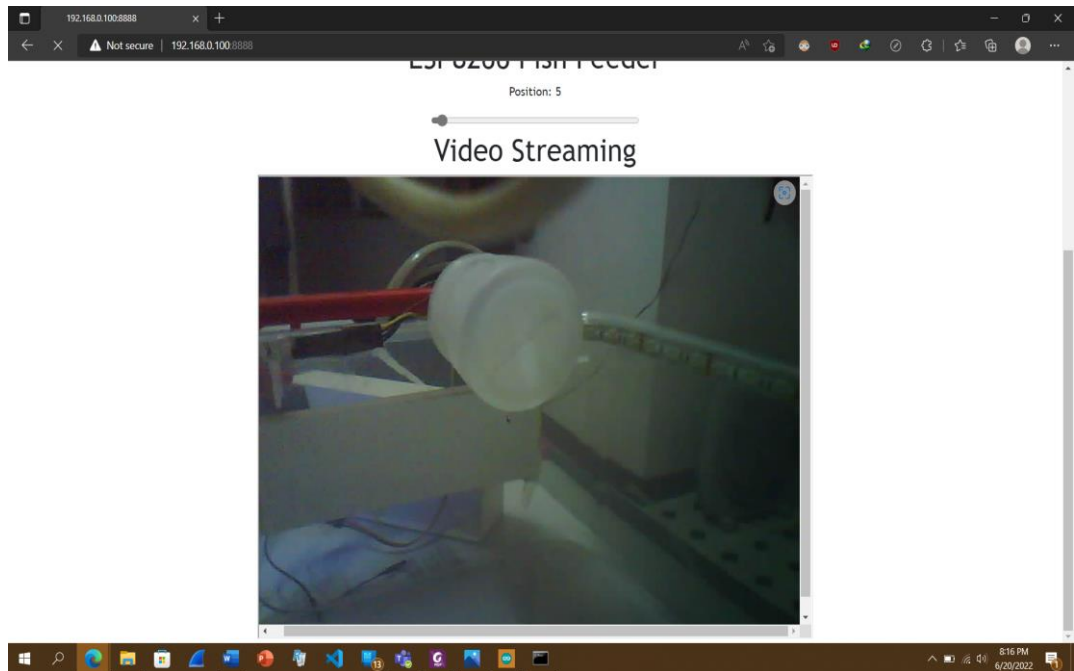
Hình 2. 21 Mô hình tổng quát của hồ cá

CHƯƠNG 3: KẾT QUẢ MÔ PHỎNG VÀ HƯỚNG DẪN SỬ DỤNG

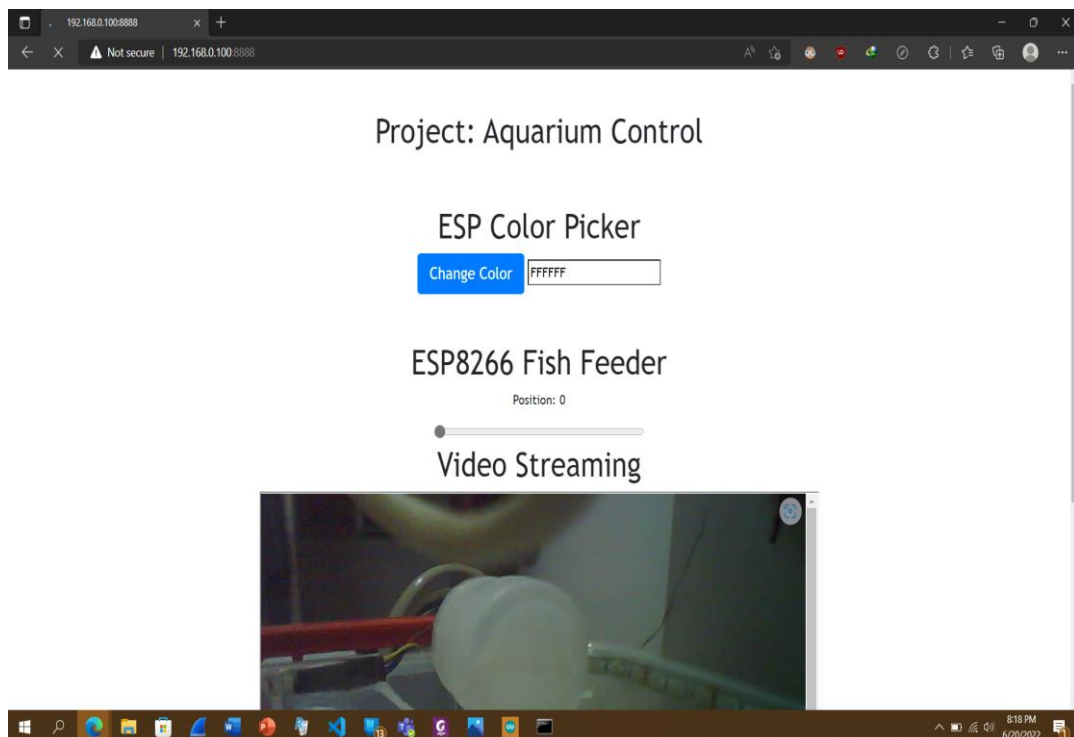
3.1. MÔ PHỎNG ĐỒ ÁN

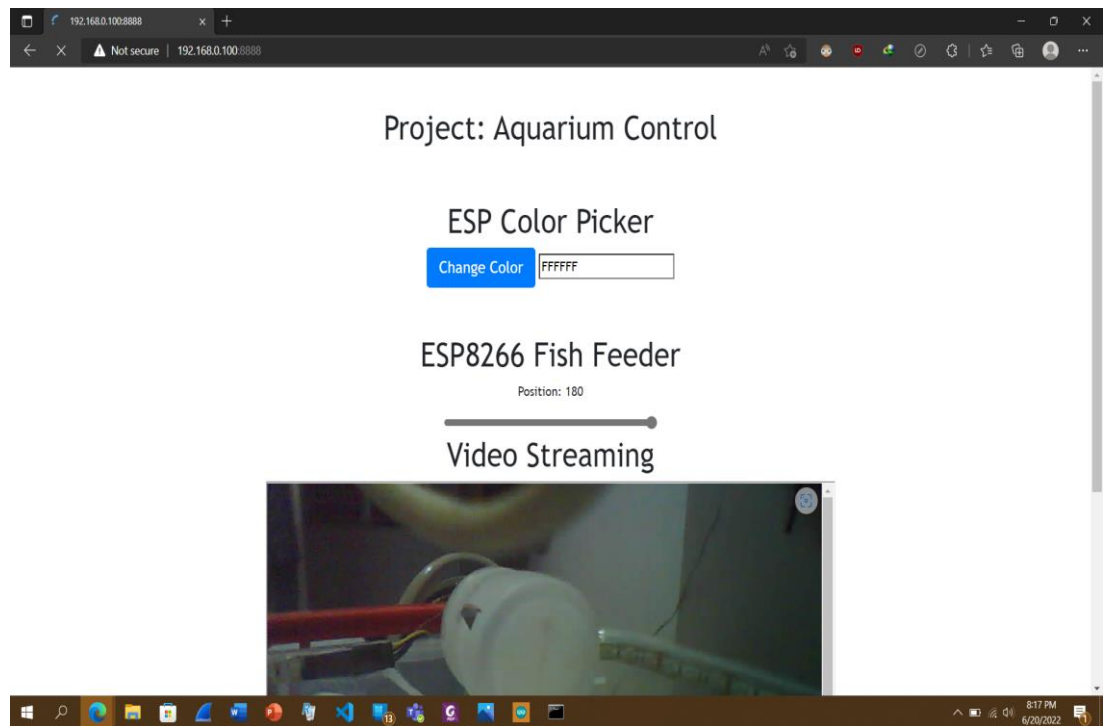
Trang web điều khiển hồ cá sẽ hiển thị như hình bên dưới:



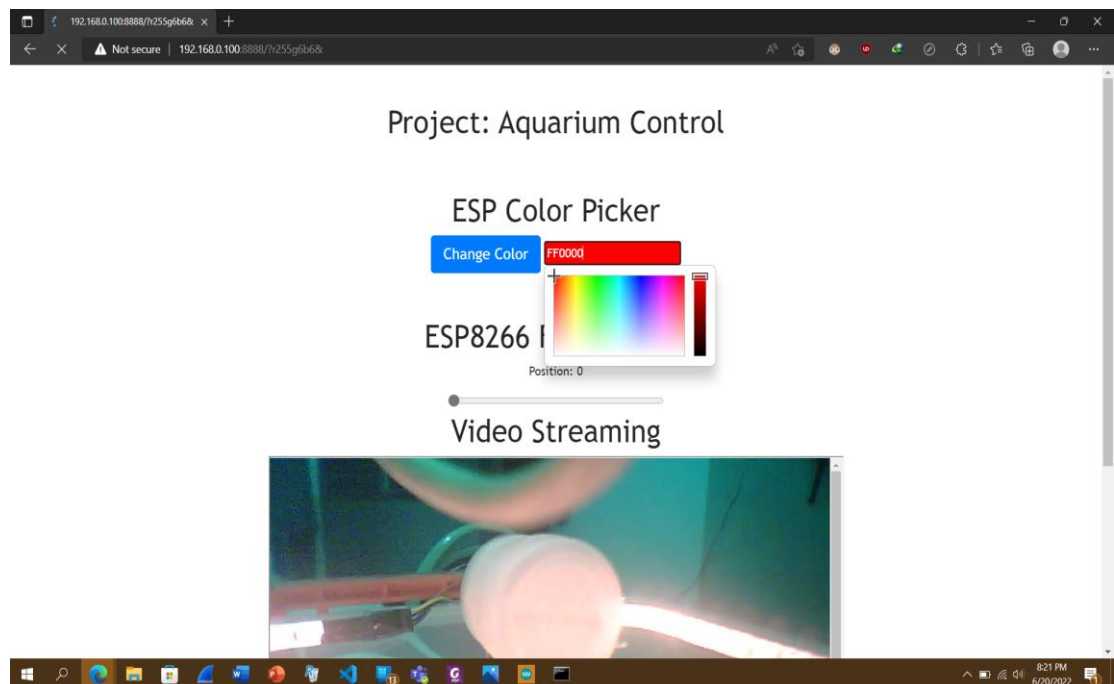


Nếu muốn cho cá ăn, ta chỉ cần kéo thanh slider từ 0° lên 180° :





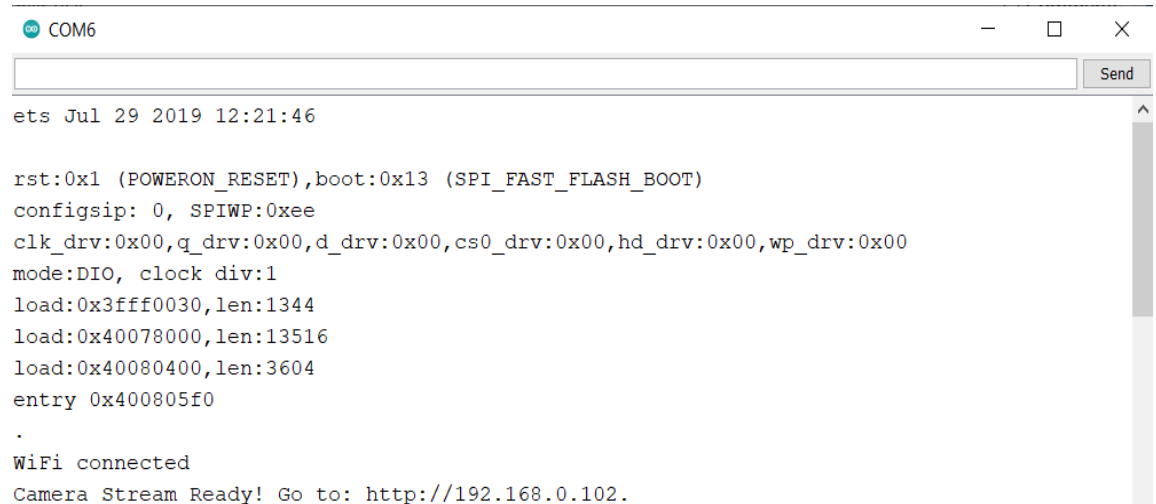
Nếu muốn bật đèn LED RGB, ta chọn màu ở ô FFFFFFFF sau đó nhấn Change Color:



3.2. HƯỚNG DẪN SỬ DỤNG

Để có thể nạp code và điều khiển hồ cá từ xa ta làm các bước sau:

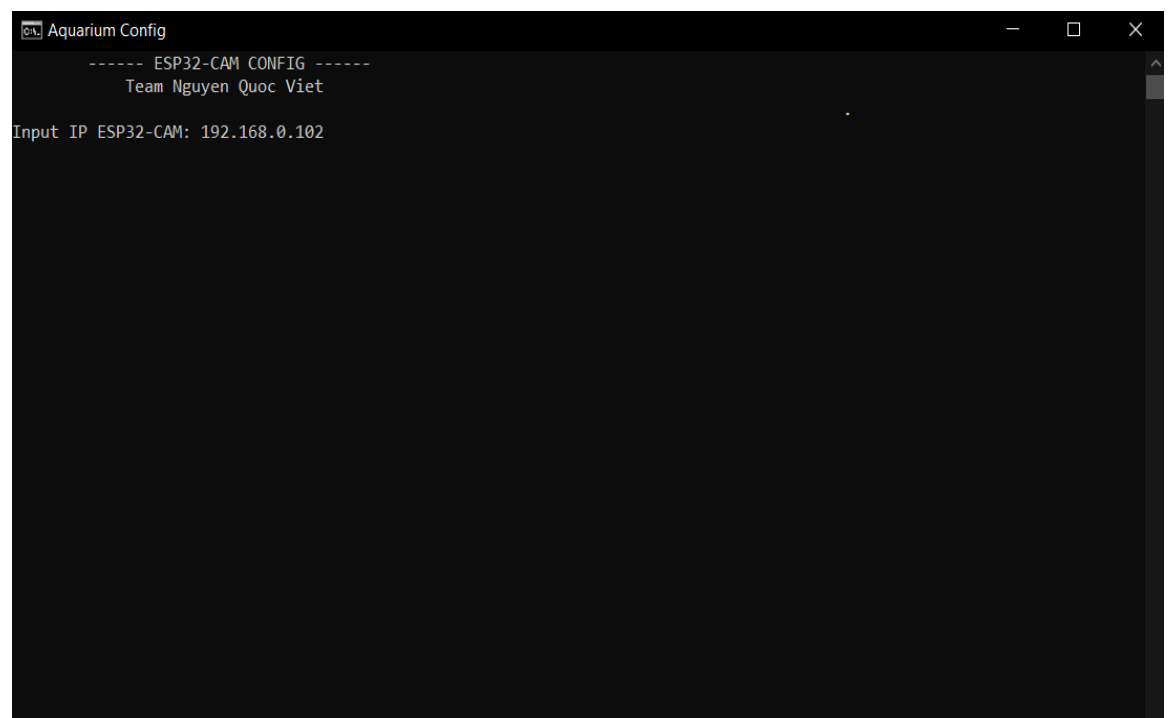
- Bước 1: Tiến hành nạp code ESP32 Camera, sau khi nạp xong sẽ xuất hiện 1 địa chỉ IP.



```
COM6
ets Jul 29 2019 12:21:46

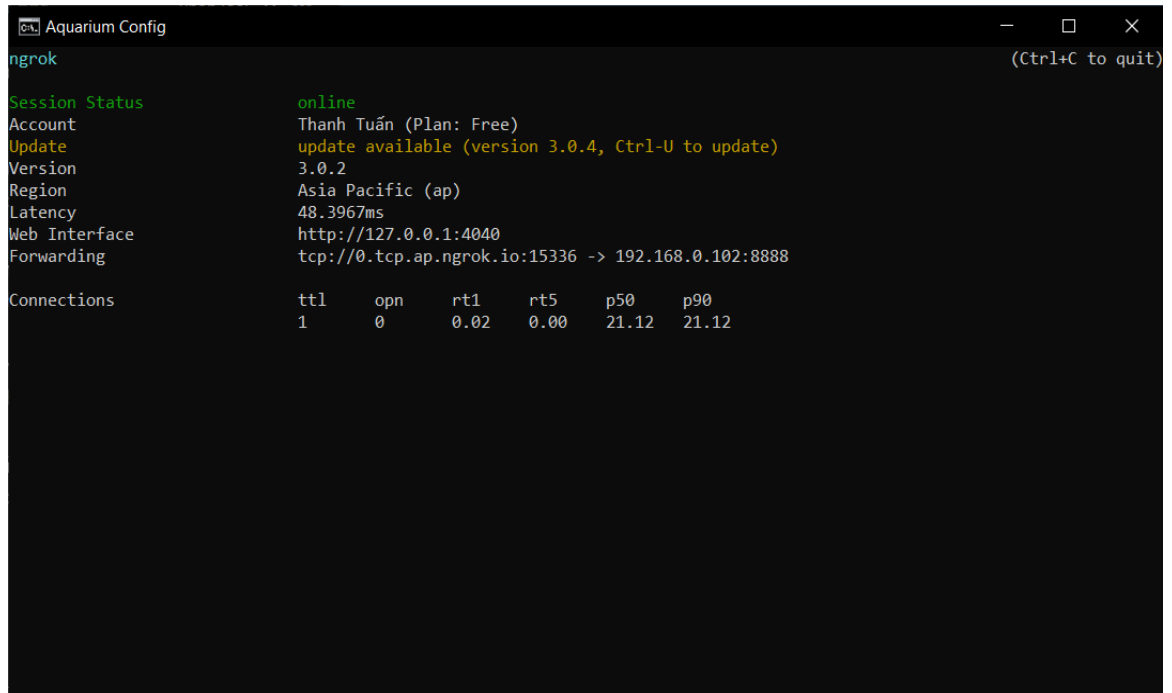
rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:1
load:0x3fff0030,len:1344
load:0x40078000,len:13516
load:0x40080400,len:3604
entry 0x400805f0
.
WiFi connected
Camera Stream Ready! Go to: http://192.168.0.102.
```

- Bước 2: Sao chép địa chỉ IP bỏ vào file ESP32-CAM rồi nhấn Enter.



```
Aquarium Config
----- ESP32-CAM CONFIG -----
Team Nguyen Quoc Viet
Input IP ESP32-CAM: 192.168.0.102
```

- Bước 3: Một địa chỉ IP mới sẽ xuất hiện, hãy sao chép địa chỉ IP đó.



The screenshot shows the 'ngrok Aquarium Config' window. It displays the following information:

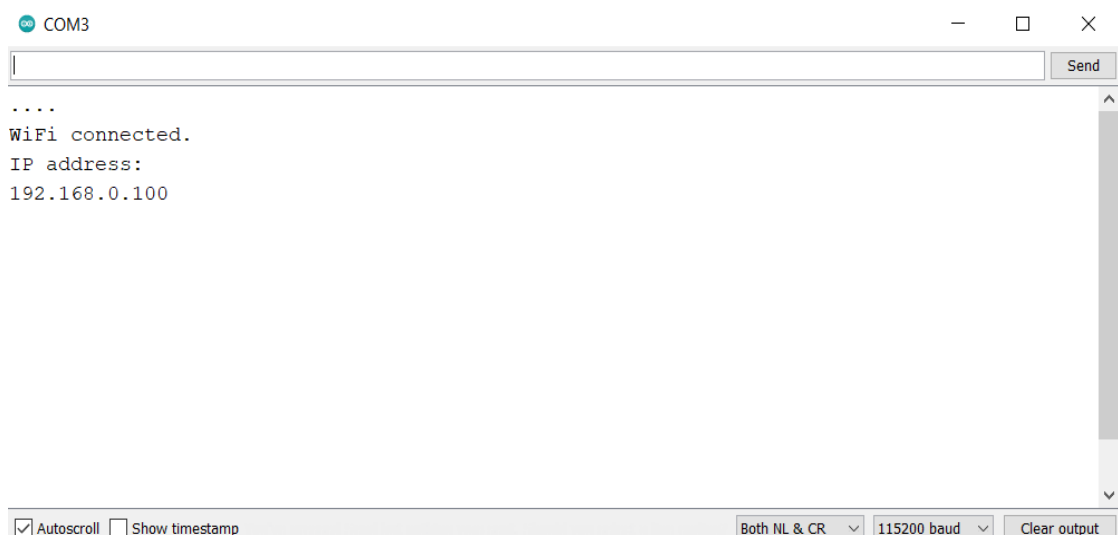
- Session Status:** online
- Account:** Thanh Tuấn (Plan: Free)
- Update:** update available (version 3.0.4, Ctrl-U to update)
- Version:** 3.0.2
- Region:** Asia Pacific (ap)
- Latency:** 48.3967ms
- Web Interface:** http://127.0.0.1:4040
- Forwarding:** tcp://0.tcp.ap.ngrok.io:15336 -> 192.168.0.102:8888

Connections	ttl	opn	rt1	rt5	p50	p90
1	0	0.02	0.00	21.12	21.12	

- Bước 4: Dán địa chỉ IP vừa sao chép bỏ vào file code của ESP8266.

'http://0.tcp.ap.ngrok.io:15336'

- Bước 5: Tiến hành chạy code ESP8266 sẽ xuất hiện được địa chỉ IP như hình bên dưới.

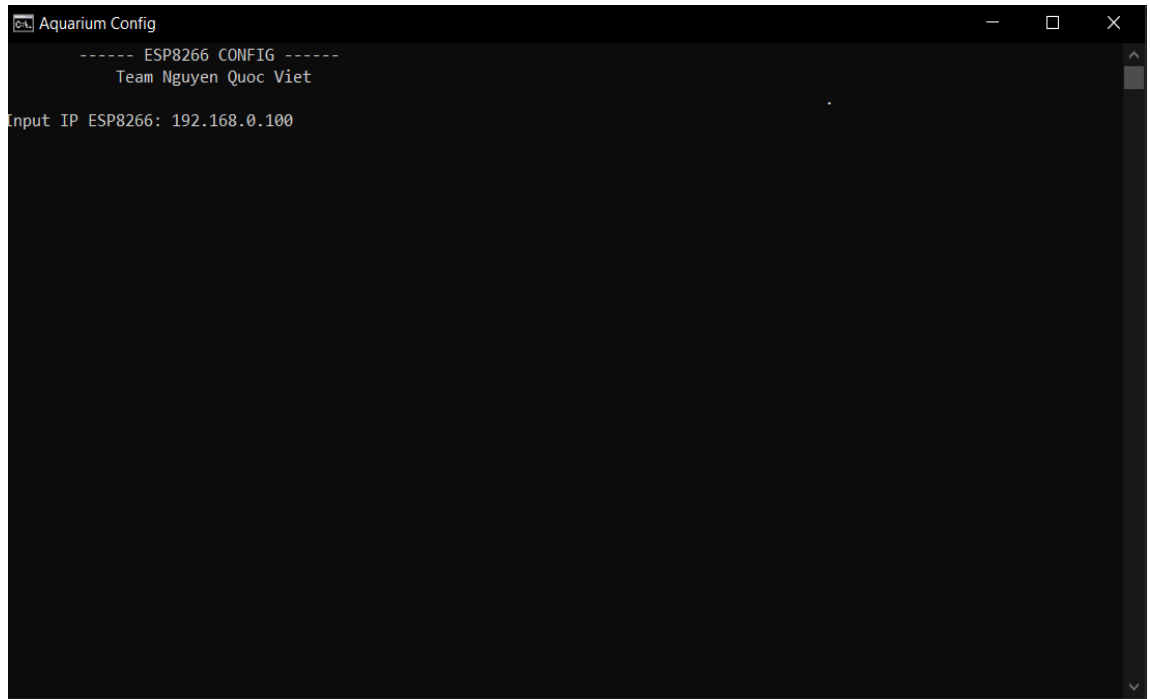


The screenshot shows a serial monitor window titled 'COM3'. It displays the following text:

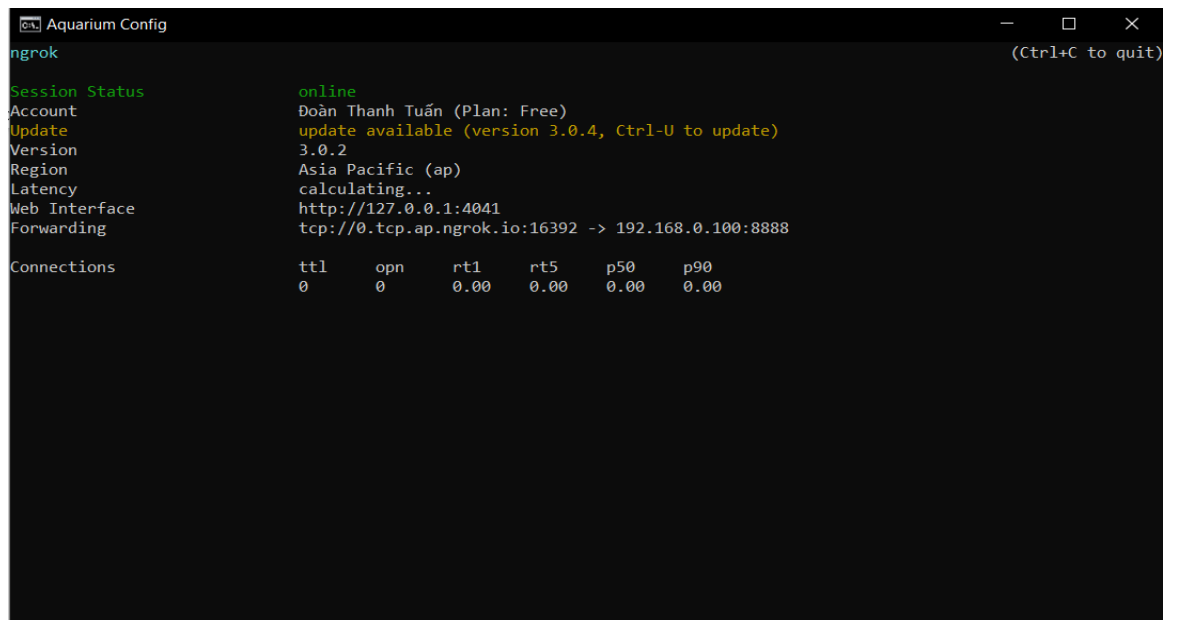
```
....  
WiFi connected.  
IP address:  
192.168.0.100
```

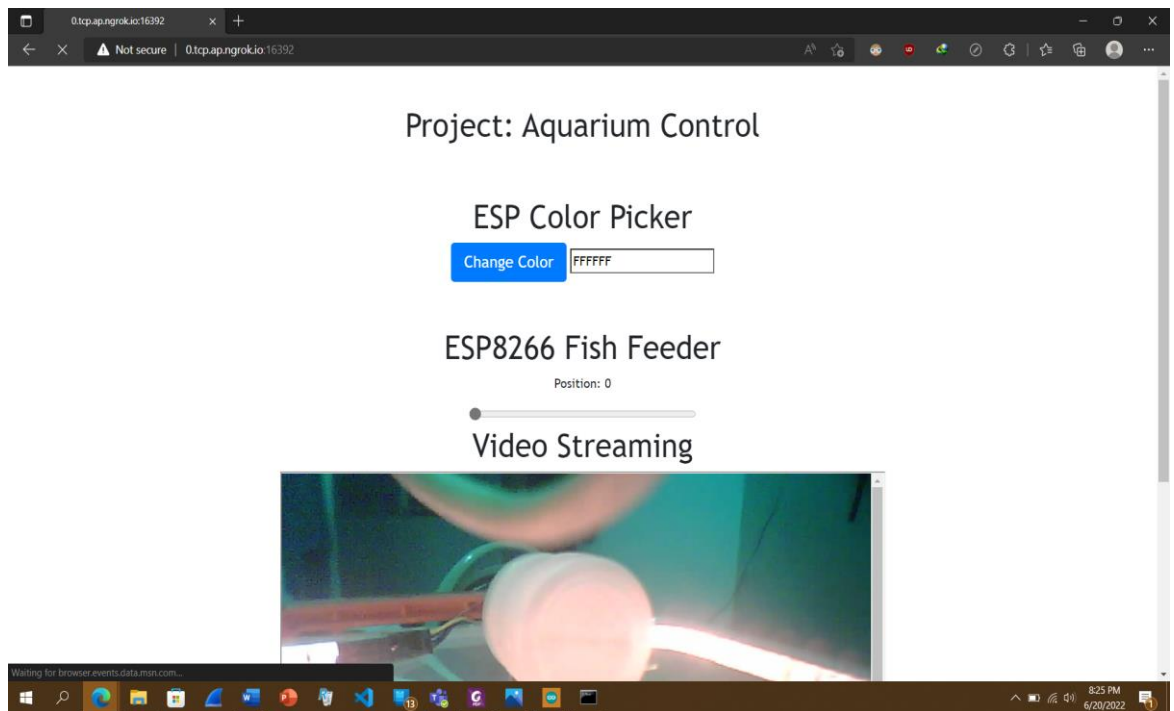
At the bottom, there are settings: ☒ Autoscroll, ☐ Show timestamp, Both NL & CR, 115200 baud, and a Clear output button.

- Bước 6: Sao chép địa chỉ IP đó rồi dán vào trong file ESP8266 Config.



- Bước 7: Nhấn Enter, một địa chỉ IP mới sẽ xuất hiện, đây là địa chỉ IP public nên chúng ta chỉ cần có địa chỉ IP đó là sẽ điều khiển được hồ cá cho dù ở bất cứ nơi nào chỉ cần có internet.





CHƯƠNG 4: KẾT QUẢ - NHẬN XÉT VÀ ĐÁNH GIÁ

4.1. KẾT QUẢ

Sau một khoảng thời gian thực hiện đề tài, nhóm chúng em đã đạt được một số kết quả sau:

- Đã tìm hiểu và sử dụng được ESP8266 và ESP32 Camera trong việc thiết kế mạch điều khiển.
- Đã kết hợp được motor và LED với ESP8266.
- Nhóm đã biết và sử dụng được thêm nhiều ngôn ngữ lập trình để lập trình WEB như html, css, javascript...
- Khả năng sáng tạo và khả năng thiết kế phần cơ khí trong việc tự thiết kế thiết bị cho cá ăn.
- Từ những ngôn ngữ lập trình web, nhóm đã biết được trang có thể biến IP Private thành IP public để điều khiển hồ cá từ xa.
- Kỹ năng tìm kiếm tài liệu, chọn lọc tài liệu nâng cao.

4.2. NHẬN XÉT VÀ ĐÁNH GIÁ

Thiết bị đã tự động điều khiển hỗ trợ việc nuôi cá. Đã giám sát và điều khiển thiết bị qua trang web. Mô hình hoạt động đạt yêu cầu, khả năng nhận dữ liệu và thực hiện bật tắt thiết bị thành công khi chúng ta điều khiển thiết bị.

Website hoạt động tốt, ổn định, dữ liệu luôn được cập nhật liên tục khi hệ thống được kết nối internet. Giao diện hiển thị trực quan, dễ điều khiển cũng như xem thông tin của thiết bị.

4.2.1. Ưu điểm của hệ thống

Phần cứng được thiết kế nhỏ gọn, dễ lắp đặt, sử dụng các linh kiện phổ biến, giá thành hợp lý, thiết bị được thiết kế hoạt động khá tốt và ổn định.

Các thiết bị được điều khiển từ xa qua website giúp thuận tiện hơn trong đời sống. Có thể điều khiển ở bất kỳ thời điểm nào, ở bất cứ nơi nào, chỉ cần có mạng internet và thiết bị truy cập website.

Điều khiển từ xa tốt, ổn định. Trang web hiển thị trực quan, dễ điều khiển và xem thông tin.

Thông tin bề cá được cập nhật liên tục khi thiết bị được kết nối internet, với độ trễ từ 1-5 giây.

Điều khiển thiết bị tốt, ổn định với độ trễ từ 2-6 giây.

4.2.2. Nhược điểm của hệ thống

Không thể xem thông tin và điều khiển thiết bị từ trang web khi thiết bị không có kết nối internet.

Tên wifi và mật khẩu để thiết bị kết nối internet là cố định, chỉ có thể thay đổi bằng cách nạp lại chương trình.

Máy chủ (laptop) khi tạo ra public IP không được tắt, nếu tắt thì địa chỉ IP đó sẽ biến mất và sẽ không điều khiển được hồ cá.

CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1. KẾT LUẬN

Sau thời gian tìm hiểu và thực hiện đề tài, nhóm chúng em đã vận dụng được kiến thức về Internet, giao thức TCP/IP, biết cách tìm kiếm phần cứng thích hợp... từ những môn học như Mạng thông tin máy tính, Truyền số liệu, Kỹ thuật vi xử lý vào trong đồ án lần này.

Mạch điều khiển hoạt động đã đạt yêu cầu đặt ra như:

- Khả năng nhận dữ liệu và điều khiển thiết bị qua web thành công.
- Các motor, LED, và Camera đều hoạt động ổn định.
- Hệ thống hoạt động ổn định sau thời gian dài hoạt động liên tục.
- Có thể xem thông tin và điều khiển thiết bị ở bất kì đâu thông qua website.

Qua thực tế làm việc, nhóm cũng đã hiểu sâu hơn về hệ thống thông tin web:

- Các ngôn ngữ thiết kế web như html, css, javascript...
- Biết cách đưa dữ liệu từ trang web này vào trang web khác.

Bên cạnh đó là các yếu tố liên quan giữa bài toán kỹ thuật và kinh tế, hiệu quả và giá thành.

Tuy nhiên, thiết bị vẫn còn có một số nhược điểm chưa được khắc phục như không thể xem thông tin và điều khiển thiết bị từ trang web khi thiết bị không có kết nối internet. Tên wifi và mật khẩu để thiết bị kết nối internet là cố định, chỉ có thể thay đổi bằng cách nạp lại chương trình. Máy chủ (laptop) khi tạo ra public IP không được tắt, nếu tắt thì địa chỉ IP đó sẽ biến mất và sẽ không điều khiển được hồ cá.

5.2. HƯỚNG PHÁT TRIỂN

Vì thời gian thực hiện đề tài có hạn nên vẫn còn nhiều phương án – thiết kế để thay đổi – bổ sung thêm cho hệ thống mà nhóm chưa thể tìm hiểu và thực hiện được. Những phương án bổ sung đó nhằm tăng thêm chức năng, mở rộng khả năng điều khiển, tính linh hoạt cho thiết bị hỗ trợ nuôi cá hơn. Cụ thể, hướng phát triển mà chúng em nghĩ đến như sau:

- Thêm chức năng đặt lịch/hẹn giờ cho cá ăn, bật tắt đèn từ xa mà không cần phải điều khiển thông qua website.
- Thêm cảm biến đo độ PH. Với một số loại cá khi nuôi cần quan tâm thêm độ PH của nước.
- Kết hợp module thu phát sóng RF. Giúp điều khiển bằng tay khi ở gần nhà mà không cần thông qua mạng.