

BỘ THÔNG TIN & TRUYỀN THÔNG
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
Khoa Kỹ Thuật Điện Tử I



BÁO CÁO BÀI TẬP LỚN

MÔN HỌC: THỰC HÀNH CƠ SỞ

ĐỀ TÀI 1

Hệ thống điều khiển giám sát phòng học thông minh

Giảng viên: Nguyễn Đức Minh

Sinh viên thực hiện:

Lương Anh Đức- B19DCDT059

Phan Thị Thanh Thúy- B19DCDT240

Hà Nội, tháng 5/2022

LỜI NÓI ĐẦU

Lời đầu tiên chúng em xin được gửi lời cảm ơn sâu sắc tới thầy Nguyễn Đức Minh và nhà trường đã tạo điều kiện cho sinh viên được học tập và trải nghiệm môn học Thực hành cơ sở vô cùng bổ ích và nhiều kiến thức hay. Cảm ơn thầy đã dạy cho chúng em thêm nhiều kiến thức thú vị trong môn học!

Trong thời đại công nghiệp hóa - hiện đại hóa ngày nay, đã xuất hiện rất nhiều các thiết bị vô cùng thông minh và hiện đại đáp ứng cho nhu cầu cuộc sống của con người. Các nước đã và đang phát triển ngày càng củng cố, phát triển mạnh về khoa học kỹ thuật. Việt Nam tuy còn là một nước nhỏ, tuy nhiên cũng đã và đang dần lớn mạnh để hòa nhập vào công cuộc công nghiệp hóa hiện đại hóa toàn cầu.

Từ công cuộc phát triển và các kiến thức đã được học của môn học. Chúng em đã chọn đề tài **“Thiết kế lắp ráp hệ thống điều khiển giám sát phòng học thông minh sử dụng KIT Arduino ESP32 WROOM 32U”** để làm bài tập lớn cho môn học này. Ý tưởng đề tài được xuất phát từ việc quản lý phòng học từ xa.

Sau một thời gian học hỏi và tìm tòi, học tập và được sự giúp đỡ của thầy và các bạn trong lớp, chúng em đã hoàn thành xong bài tập lớn của mình. Trong quá trình làm bài và báo cáo không thể tránh được những thiếu sót, chúng em mong thầy sẽ góp ý để bài báo cáo của chúng em được hoàn thiện hơn nữa.

Chúng em xin chân thành cảm ơn!

MỤC LỤC

LỜI NÓI ĐẦU.....	1
Phần A: Giới thiệu linh kiện có trong mạch	3
I. KIT Arduino ESP32 WROOM 32U	3
1. Tổng quan về KIT Arduino ESP32 WROOM	3
2. Thông số kỹ thuật.....	4
3. Cấu hình chi tiết	5
II. Module relay 5VDC 2 kênh	9
1. Tổng quan	9
2. Sơ đồ chân.....	10
3. Sơ đồ mạch.....	10
4. Thông số kỹ thuật.....	11
III. Cảm biến nhiệt độ, độ ẩm DHT22	11
1. Tổng quan	11
2. Thông số kỹ thuật.....	12
3. Sơ đồ chi tiết	13
Phần B: Tìm hiểu về Firebase Realtime Database	17
I. Firebase Realtime Database là gì?	17
II. Các tính năng của Realtime Database.....	17
Phần C: Nguyên lý hoạt động của mạch	19
I. Cấu tạo mạch.....	19
II. Nguyên lý hoạt động	20
III. Các bước giao tiếp với Realtime Database Firebase sử dụng ESP32	20
IV. Code điều khiển	20
Phần D: Ưu điểm, nhược điểm và hướng phát triển.....	27

Phần A: Giới thiệu linh kiện có trong mạch

I. KIT Arduino ESP32 WROOM 32U

1. Tổng quan về KIT Arduino ESP32 WROOM

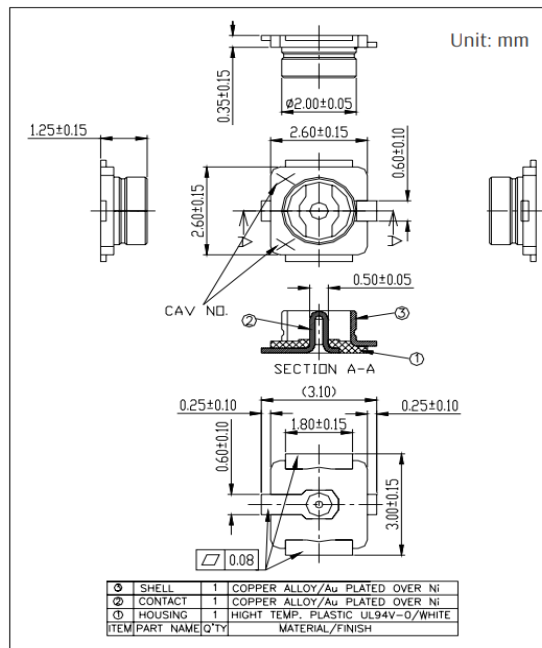
ESP32-WROOM-32D and ESP32-WROOM-32U là các module WiFi, BT, BLE mạnh mẽ, phổ biến phù hợp với nhiều dự án khác nhau, có thể đáp ứng cho các dự án đòi hỏi mạng, công suất thấp hoặc những đòi hỏi khắt khe khác chẳng hạn như: mã hóa giọng nói, phát nhạc và giải mã MP3. ESP32-WROOM-32U khác với ESP32-WROOM-32D ở chỗ có tích hợp đầu nối U.FL.

ESP32-WROOM-32U



Hình 1.1. KIT ESP32-WROOM-32U

ESP32-WROOM-32U sử dụng chip ESP32-D0WD là chip nhúng được thiết kế có thể mở rộng và thích ứng. Có 2 lõi CPU có thể điều khiển riêng lẻ và tần số xung nhịp CPU có thể điều chỉnh được từ 80MHz tới 240MHz.

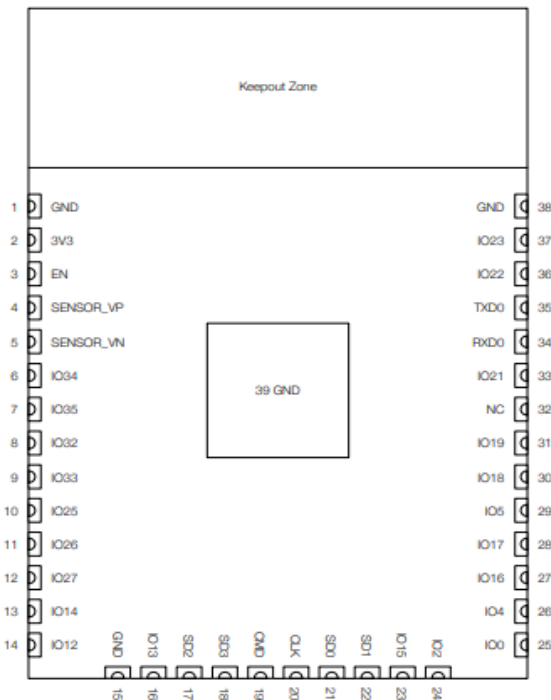


2. Thông số kỹ thuật

- Chip xử lí: ESP32-D0WD
- SPI flash: 32 Mbits, 3.3V
- Crystal: 40 MHz
- Antenna: đầu nối U.FL (giúp kết nối với anten IPEX bên ngoài)
- Network protocols: IPv4, IPv6, SSL, TCP/UDP/HTTP/FTP/MQTT
- User configuration: AT instruction set, cloud server, Android/iOS app
- WiFi:
 - Wi-Fi mode: Station/SoftAP/SoftAP+Station/P2P
 - Wi-Fi Security: WPA/WPA2/WPA2-Enterprise/WPS
 - Các giao thức:
 - 802.11 b/g/n (802.11n up to 150 Mbps)
 - A-MPDU and A-MSDU aggregation and 0.4 μ s guard interval support
 - Phạm vi tần số: 2.4 ~ 2.5 GHz
- Bluetooth:
 - Các giao thức: Bluetooth v4.2 BR/EDR and BLE specification
 - Radio:
 - NZIF receiver with -97 dBm sensitivity
 - Class-1, class-2 and class-3 transmitter
 - AFH
 - Audio: CVSD and SBC
- Encryption: AES/RSA/ECC/SHA
- Firmware upgrade: UART Download / OTA
- Software development: Supports Cloud Server Development / SDK for custom firmware development

- Module interface: SD card, UART, SPI, SDIO, I2C, LED PWM, Motor PWM, I2S, IR
- On-chip sensor: Hall sensor
- On-board clock: 40 MHz crystal
- Điện áp hoạt động: 2.7 ~ 3.6V
- Dòng điện hoạt động: trung bình: 80 mA
- Dòng điện tối thiểu lấy từ nguồn điện: 500 mA
- Nhiệt độ hoạt động: $-40^{\circ}\text{C} \sim +85^{\circ}\text{C}$

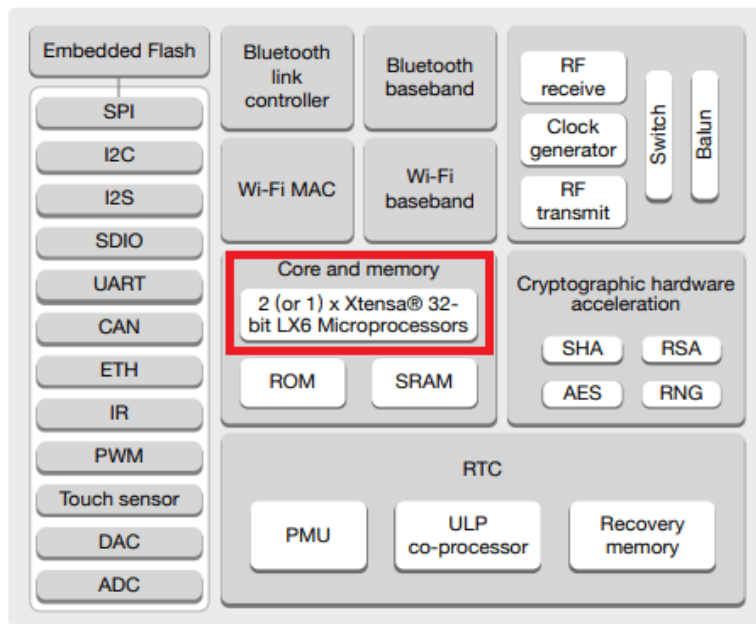
3. Cấu hình chi tiết



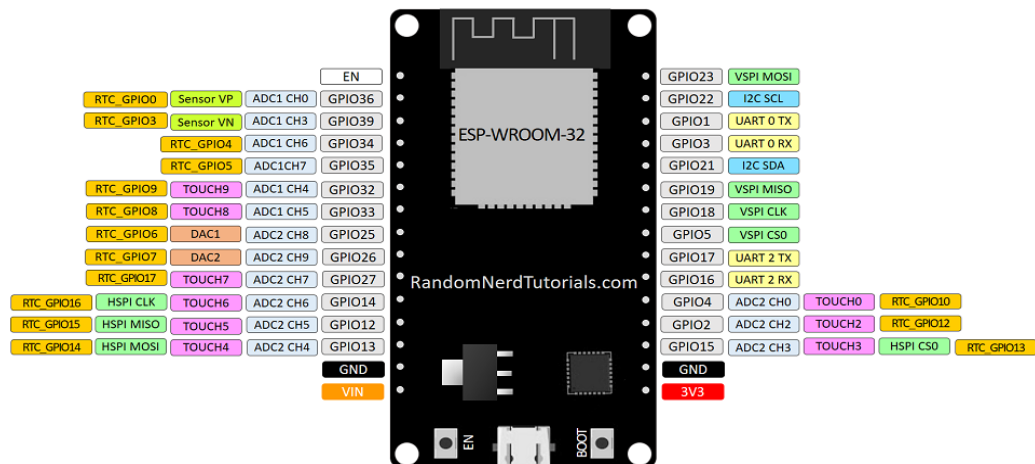
- **CPU**

Được đặt tên là “PRO-CPU” và “APP_CPU” Xtensa Dual- Core LX6. Khi chúng ta dùng FreeROS sẽ ứng với Core 0 và Core 1 (protocol cpu và applications cpu)

- 32bit
- Tốc độ xử lý 160MHZ-240 MHZ
- Tốc độ xung nhịp đọc flash chip 40mhz 80mhz (tùy chỉnh khi lập trình)
- ROM:448 Kbyte ROM
- 4MB external FLASH
- RAM: 520 KByte SRAM, 520 KB SRAM liền chip -(trong đó 8 KB RAM RTC tốc độ cao - 8 KB RAM RTC tốc độ thấp (dùng ở chế DeepSleep)



• Ngoại vi



GND: chân đất

3V3: chân cấp nguồn 3V3

EN: chân cho phép hoạt động, kích mức cao

SENSOR_VP: chân GPIO36, ADC1_CH0, RTC_GPIO0

SENSOR_VN: chân GPIO39, ADC1_CH3, RTC_GPIO3

IO34: chân GPIO34, ADC1_CH6, RTC_GPIO4

IO35: chân GPIO35, ADC1_CH7, RTC_GPIO5

IO32: chân GPIO32, XTAL_32K_P (kết nối thạch anh), ADC1_CH4, TOUCH9, RTC_GPIO9

IO33: chân GPIO33, XTAL_32K_N (kết nối thạch anh), ADC1_CH5, TOUCH8, RTC_GPIO8

IO25: chân GPIO25, DAC_1, ADC2_CH8, RTC_GPIO6, EMAC_RXD0

IO26: chân GPIO26, DAC_2, ADC2_CH9, RTC_GPIO7, EMAC_RXD1

IO27: chân GPIO27, ADC2_CH7, TOUCH7, RTC_GPIO17, EMAC_RX_DV

IO14: chân GPIO14, ADC2_CH6, TOUCH6, RTC_GPIO16, MTMS, HSPICLK, HS2_CLK, SD_CLK, EMAC_TXD2

IO12: chân GPIO12, ADC2_CH5, TOUCH5, RTC_GPIO15, MTDI, HSPIQ, HS2_DATA2, SD_DATA2, EMAC_TXD3

IO13: chân GPIO13, ADC2_CH4, TOUCH4, RTC_GPIO14, MTCK, HSPID, HS2_DATA3, SD_DATA3, EMAC_RX_ER

SHD/SD2*: chân GPIO9, SD_DATA2, SPIHD, HS1_DATA2, U1RXD

SWP/SD3*: chân GPIO10, SD_DATA3, SPIWP, HS1_DATA3, U1TXD

SCS/CMD*: chân GPIO11, SD_CMD, SPICS0, HS1_CMD, U1RTS

SCK/CLK*: chân GPIO6, SD_CLK, SPICLK, HS1_CLK, U1CTS

SDO/SD0*: chân GPIO7, SD_DATA0, SPIQ, HS1_DATA0, U2RTS

SDI/SD1*: chân GPIO8, SD_DATA1, SPID, HS1_DATA1, U2CTS
IO15: chân GPIO15, ADC2_CH3, TOUCH3, MTDO, HSPICS0, RTC_GPIO13, HS2_CMD, SD_CMD, EMAC_RXD3

IO2: chân GPIO2, ADC2_CH2, TOUCH2, RTC_GPIO12, HSPIWP, HS2_DATA0, SD_DATA0

IO0: chân GPIO0, ADC2_CH1, TOUCH1, RTC_GPIO11, CLK_OUT1, EMAC_TX_CLK

IO4: chân GPIO4, ADC2_CH0, TOUCH0, RTC_GPIO10, HSPIHD, HS2_DATA1, SD_DATA1, EMAC_TX_ER

IO16: chân GPIO16, HS1_DATA4, U2RXD, EMAC_CLK_OUT

IO17: chân GPIO17, HS1_DATA5, U2TXD, EMAC_CLK_OUT_180

IO5: chân GPIO5, VSPICS0, HS1_DATA6, EMAC_RX_CLK

IO18: chân GPIO18, VSPICLK, HS1_DATA7

IO19: chân GPIO19, VSPIQ, U0CTS, EMAC_TXD0

IO21: chân GPIO21, VSPIHD, EMAC_TX_EN

RXD0: chân GPIO3, U0RXD, CLK_OUT2

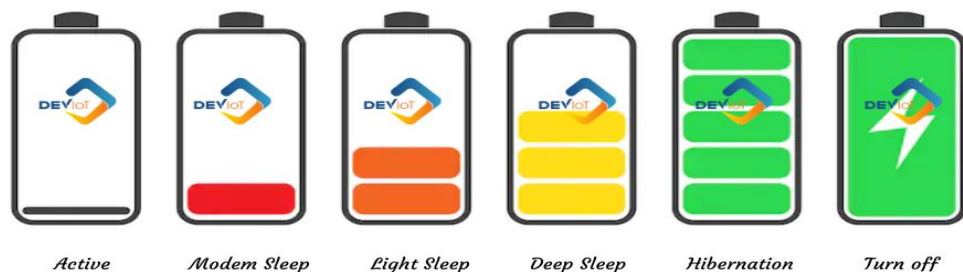
TXD0: chân GPIO1, U0TXD, CLK_OUT3, EMAC_RXD2

IO22: chân GPIO22, VSPIWP, U0RTS, EMAC_TXD1

IO23: chân GPIO23, VSPID, HS1_STROBE

5V: chân cấp nguồn 5V

- **Ultra -Low-Power**



- Sleep mode: Là trạng thái ESP32 tiết kiệm năng lượng của ESP32 khi không sử dụng. Năng lượng chỉ đủ truyền cho RAM để lưu dữ liệu

- Chế độ hoạt động: tắt cả các tính năng hoạt động. Dòng chip yêu cầu là 240mA, đôi khi nếu sử dụng cả Bluetooth và wifi có thể lên tới 790mA

- Light Sleep: tắt cả các wifi, BLE, RAM và CPU được chỉ định mức Clock, dòng tiêu thụ -0.8mA
- Deep Sleep: ở chế độ ngủ sâu, hầu hết các RAM và tắt cả các ngoại vi tắt. Các phần chip vẫn được bật là: Bộ điều khiển RTC, ngoại vi RTC (bộ đồng xử lý ULP), RTC memories, Dòng tiêu thụ 15uA 0.15uA
- Hibermate: mọi thứ khác đều bị tắt ngoại trừ chỉ một bộ đếm thời gian RTC và một số GPIO RTC đang hoạt động. Chúng chịu trách nhiệm đánh thức khỏi Hibermate

- **WIFI**

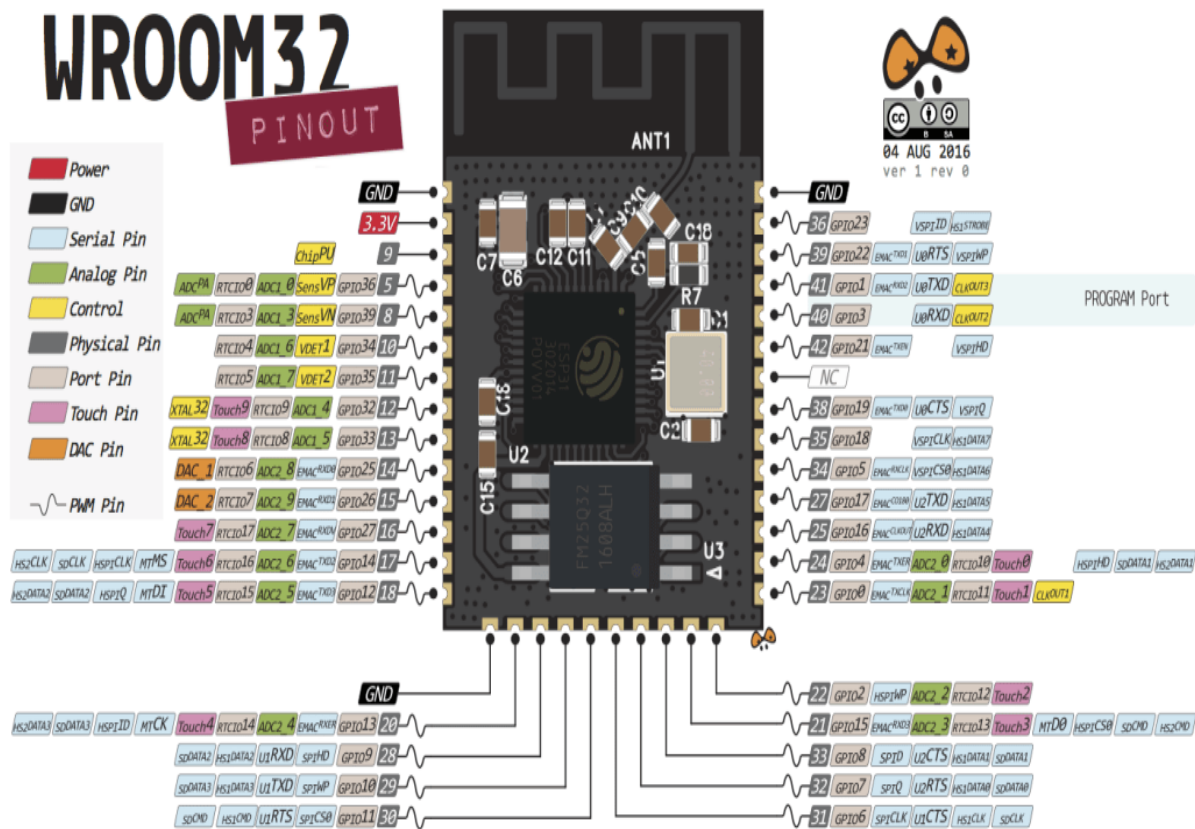


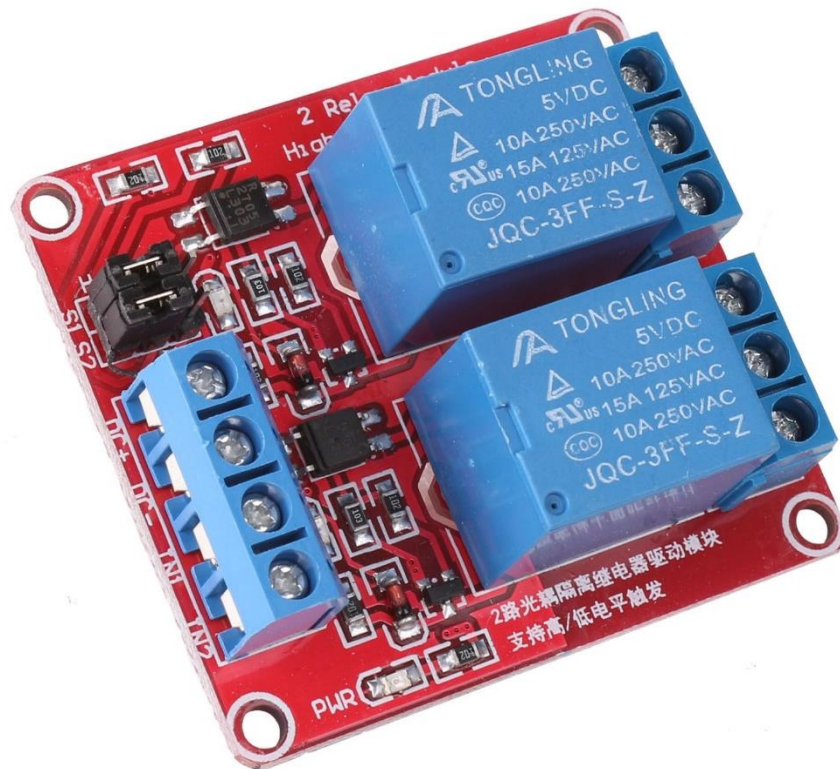
- 802.11 b/g/n/e/i (Wifi 2,4 GHZ)
- Station mode (STA hay Wi-Fi client). ESP32 sẽ kết nối tới các điểm tới các điểm truy cập
- Hoạt động như một điểm truy cập (Access Point Mode hay Soft-AP). Nó giống như trung tâm của mọi thông tin liên lạc
- AP-STA mode ESP32 sẽ đồng thời là điểm truy cập và truy cập đến điểm khác
- Các chế độ bảo mật khác nhau cho những điều kiện (WPA, WPA2, WEP...)

4. Sơ đồ chân

ESP32 WROOM là 1 dạng SoC (System on Chip) trên đó đã được thiết kế sẵn để cho chip ESP32 hoạt động.

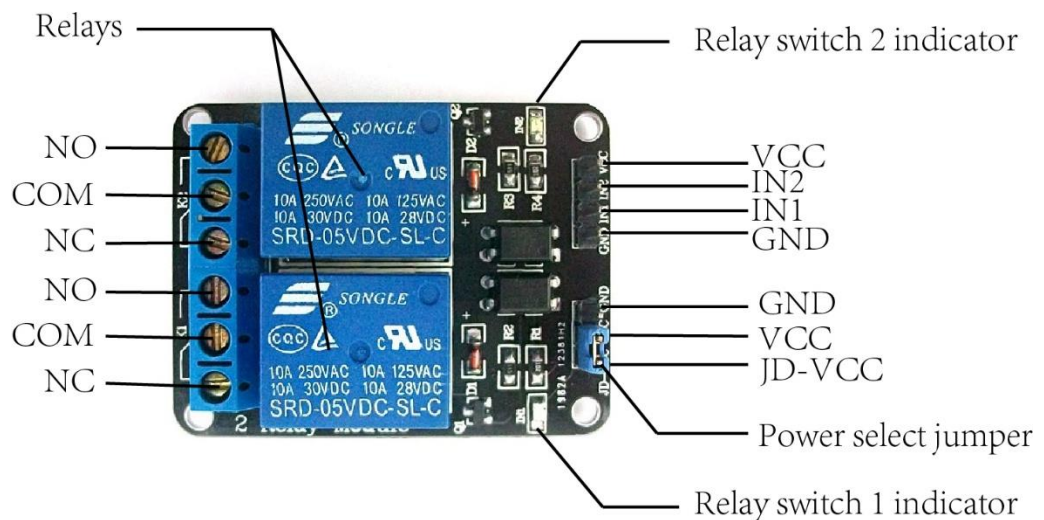
Đây là thành phần cơ bản của một sản phẩm IOT sử dụng ESP32.





Hình 1.2. Module relay 5VDC 2 kênh

2. Sơ đồ chân



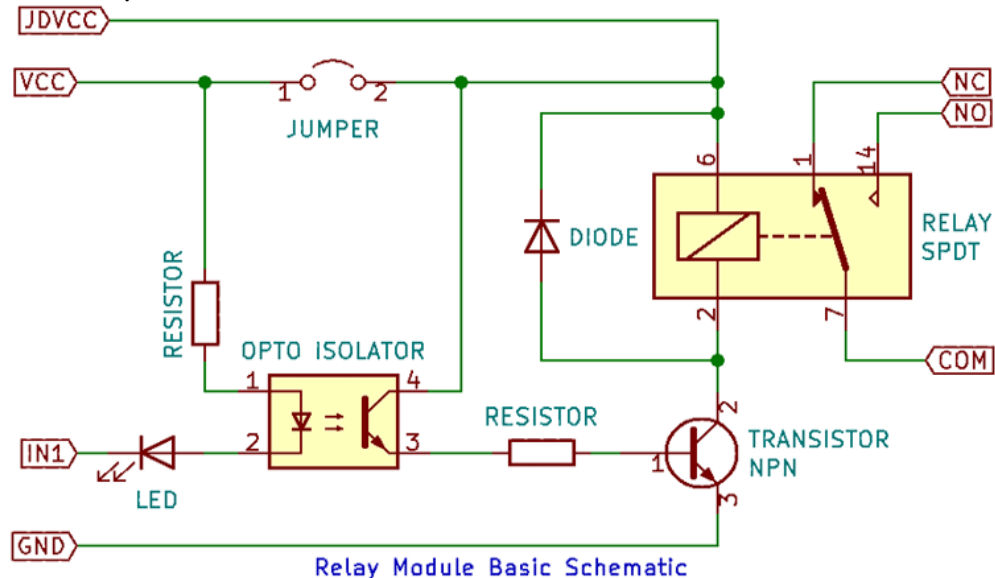
Hình 1.3. Sơ đồ chân của relay 2 kênh

Module được kết nối với các board điều khiển bằng 4 chân header như sau:

- VCC cung cấp nguồn cho các opto.
- GND kết nối với GND của board điều khiển.
- IN1 và IN2 dùng để điều khiển relay 1 và relay 2, tích cực mức thấp

Ngoài ra còn một 3 chân header được dùng để cấp nguồn cho relay, header này sẽ có một jumper dùng để kết nối chân VCC với chân RY_VCC mục đích dùng chung nguồn VCC (5V) từ header 4 chân cho relay, thông thường jumper được nối lại với nhau. Nếu như muốn cách ly tín hiệu điều khiển với nguồn cấp cho relay thì có thể bỏ jumper này ra và cấp nguồn riêng 5V cho chân RY_VCC.

3. Sơ đồ mạch



Mạch tương tự này được sao chép hai lần trên mô-đun. Có một số khác biệt khi so sánh với mô-đun chuyển tiếp đơn kênh. Đầu tiên là optocoupler và rơ le có thể có nguồn điện riêng biệt từ đầu vào. Điều này được chọn bằng cách sử dụng jumper JD-VCC - nếu jumper bị ngắn mạch, thì cuộn dây chuyển tiếp và đầu ra optocoupler được kết nối với tín hiệu VCC. Nếu jumper để hở, có thể cung cấp nguồn điện riêng cho cuộn dây rơ le thông qua chân JD-VCC.

Thứ hai là đầu vào hoạt động ở mức thấp, nghĩa là cuộn dây rơ le chỉ được kích hoạt khi đầu vào ở mức thấp. Điều này là do đầu vào optocoupler và đèn LED chỉ báo được liên kết với VCC. Khi đầu vào cao, sự chênh lệch điện áp trên chuỗi là 0V và không có dòng điện nào có thể chạy qua, nhưng khi đầu vào thấp, điện áp cung cấp xuất hiện trên chuỗi và đầu ra optocoupler dẫn, cung cấp dòng điện cho đế của bóng bán dẫn truyền động, quay tiếp điểm.

4. Thông số kỹ thuật

- Đóng ngắt được dòng điện cao: AC250V 10A, DC30V 10A
- 2 led báo trạng thái relay
- Điện áp điều khiển: 5V
- Mạch cách ly bằng opto
- Kích thước: 50x45 mm

III. Cảm biến nhiệt độ, độ ẩm DHT22

1. Tổng quan về Cảm biến DHT22

Với sự phát triển của công nghệ kỹ thuật số, ngày nay nhiều thiết bị dễ dàng có sẵn như chip Tích hợp. Những con chip này có kích thước nhỏ nhưng có hiệu suất cao hơn so với các phiên bản trước đó. Khi ngày càng nhiều thiết bị được tích hợp trên chip đơn, việc tản nhiệt của chúng trở thành một nguyên nhân đáng lo ngại. Nếu nhiệt cao hơn giá trị khuyến nghị, các kết nối nhỏ giữa các thành phần sẽ bị hỏng dẫn đến hư hỏng toàn bộ chip. Trong các tình huống như vậy, điều quan trọng là phải theo dõi các giá trị nhiệt độ và độ ẩm của các thiết bị. Đối với điều này, cảm biến nhiệt độ và độ ẩm được sử dụng. Một trong những cảm biến rất được ưa chuộng cho nhiệm vụ đó là cảm biến DHT22.

DHT22 là cảm biến nhiệt độ và độ ẩm có độ chính xác cao. Cảm biến này đo các giá trị độ ẩm tương đối. Nó sử dụng phần tử cảm biến điện dung để đo Độ ẩm. Để đo nhiệt độ, nó sử dụng nhiệt điện trở NTC. Cảm biến này cũng có thể được sử dụng trong các điều kiện khắc nghiệt. Nó có sẵn dưới dạng cảm biến cũng như Mô-đun.

Cảm biến độ ẩm nhiệt độ DHT22 ra chân được tích hợp sẵn điện trở 5.1 KOhm giúp người dung kết nối và sử dụng đơn giản hơn cảm biến DHT22 chưa ra chân. So với DHT11 thì DHT22 có độ chính xác cao và khoảng đo hoạt động rộng hơn. Module truyền dữ liệu thông qua giao tiếp 1 dây nên dễ dàng kết nối và lấy dữ liệu. Module được thiết kế hoạt động ở mức điện áp 5V.



Hình 1.4. Cảm biến nhiệt độ, độ ẩm DHT22

2. Thông số kỹ thuật

- DHT22 hoạt động với điện áp cung cấp DC từ 3,3V đến 5,5V.
- DHT22 yêu cầu dòng điện 1,5mA để hoạt động.
- Khi ở chế độ chờ, cảm biến này tiêu thụ dòng điện 0,02mA.
- Cảm biến này cung cấp các giá trị đầu ra kỹ thuật số.
- Phạm vi nhiệt độ hoạt động của cảm biến này là từ -40°C đến 80°C .

C.

- Nó có thể phát hiện độ ẩm từ 0-100% RH.
- Cảm biến này có thể đo độ ẩm với độ chính xác lên đến $\pm 2\%$ RH.
- Nhiệt độ chính xác lên đến $\pm 0.5^{\circ}\text{C}$ can be measured with DHT22.

DHT22.

• Cảm biến này đo độ ẩm với độ phân giải 0,1% RH và nhiệt độ với độ phân giải 0,1 độ C.

- DHT22 có độ trễ độ ẩm là $\pm 0,3\%$ RH
- Cảm biến này có thời gian phát hiện trung bình là 2 giây.
- Cảm biến này có hai kích thước - kích thước nhỏ $14 \times 18 \times 5.5\text{mm}$ và kích thước lớn $22 \times 28 \times 5\text{mm}$.

• Phần tử cảm biến điện dung được sử dụng làm phần tử cảm biến trong DHT22.

• Thiết bị đo nhiệt độ NTC ược sử dụng trong DHT22 để tính toán nhiệt độ.

- Cảm biến này có thể truyền tín hiệu lên đến 2 mét.

• DHT22 có giao diện nối tiếp một dây, giúp nó tích hợp với các bộ vi xử lý và vi điều khiển đơn giản và nhanh chóng.

- Cảm biến này cũng có khả năng chống nhiễu.

• Vì độ ẩm tương đối phụ thuộc vào nhiệt độ, cảm biến này được bù nhiệt độ và hiệu chuẩn trong các buồng hiệu chuẩn chính xác để có được các phép đo độ ẩm chính xác.

• Hệ số hiệu chuẩn này được lưu dưới dạng bộ nhớ OTP của chương trình. Các tín hiệu phát hiện cảm biến bên trong có thể gọi đây là hệ số hiệu chuẩn.

• DHT22 có độ chính xác cao với chất lượng tuyệt vời và phản hồi nhanh.

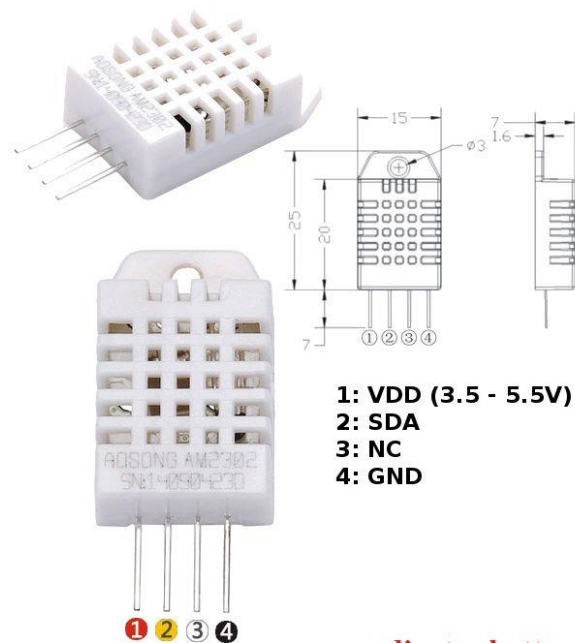
- DHT22 có tính ổn định lâu dài vượt trội.

- DHT22 cho đầu ra kỹ thuật số được hiệu chỉnh cao.

• Cảm biến này có sẵn dưới dạng gói 4 chân hoàn toàn có thể hoán đổi cho nhau.

3. Sơ đồ chi tiết

- Sơ đồ khối



Cảm biến DHT22 là sự kế thừa của Cảm biến DHT11. DHT22 có sẵn dưới dạng cảm biến cũng như mô-đun. Hoạt động của cả cảm biến và mô-đun của DHT22 là tương tự. Sự khác biệt nằm ở mạch điện bên trong. Mô-đun chứa bộ lọc tích hợp sẵn tụ điện và điện trở kéo lên. Trong khi trong cảm biến, chúng phải được kết nối bên ngoài. Cảm biến DHT22 và mô-đun đều có 8-bit vi điều khiển kết nối với nó để thực hiện các phép tính. Mô-đun DHT22 đi kèm dưới dạng gói 3 chân trong khi cảm biến có dạng gói 4 chân. Mô-đun có phạm vi đo cao, độ chính xác tốt hơn và đắt hơn một chút so với cảm biến.

Hiệu suất của cảm biến này có thể bị suy giảm khi tiếp xúc với tia cực tím và ánh sáng mạnh trong thời gian dài. Để liên lạc không bị lỗi và không bị nhiễu, nên sử dụng dây bảo vệ chất lượng cao cho các kết nối.

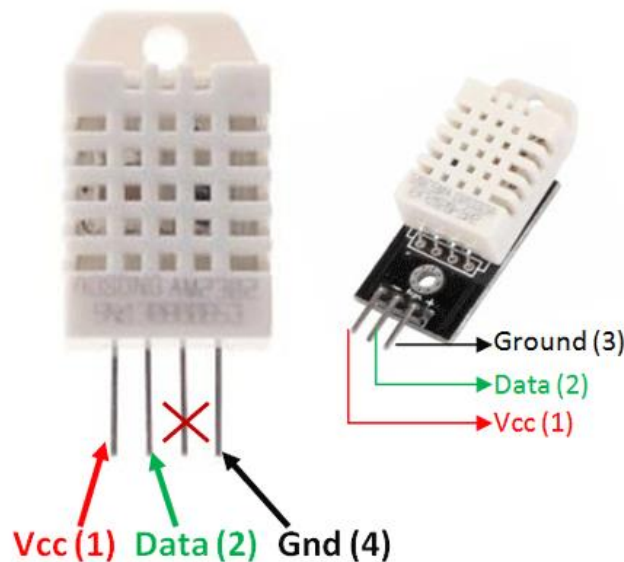
- *Sơ đồ mạch*

DHT22 sử dụng một giao diện nối tiếp dây đơn để truyền dữ liệu đến bộ vi xử lý. Chân DATA có sẵn trên cảm biến được sử dụng để truyền dữ liệu và được kết nối với bộ vi xử lý.

Khi nguồn được cấp vào, ban đầu cảm biến vẫn ở trạng thái không ổn định trong một giây, trong thời gian này, không có hướng dẫn nào được gửi đến cảm biến. Đối với giao tiếp thời gian duy nhất giữa MCU và cảm biến, mất 5m. Chỉ bắt đầu gửi dữ liệu sau khi nhận được tín hiệu bắt đầu từ MCU.

Khi cảm biến này được sử dụng cho dải RH cao hơn mức khuyến nghị, nó có thể làm tăng sự lão hóa của thiết bị. Độ nhạy của cảm biến DHT22 có thể thay đổi khi nó tiếp xúc với hơi hóa chất.

- *Sơ đồ chân*



DHT22 có sẵn dưới dạng gói hàng đơn 4 chân. Mô-đun cảm biến này tính toán các giá trị nhiệt độ theo độ C. Các giá trị này sau đó được chuyển đổi thành Fahrenheit và kelvin bằng công thức chuyển đổi. Cấu hình chân của cảm biến này được đưa ra bên dưới:

- Chân 1- VDD: là chân cấp nguồn. Điện áp một chiều được áp dụng cho chân này.
- Chân 2- DATA / SDA: là chân dữ liệu nối tiếp. Chân này được kết nối với vi điều khiển để truyền dữ liệu từ cảm biến đến bộ điều khiển.
- Chân 3- NC: còn được gọi là chân không có kết nối mà không được kết nối.
- Chân 4-GND: là chân nối đất. Chân này được nối với đất.

4. Ứng dụng của DHT22

Để đảm bảo độ tin cậy và ổn định, DHT22 sử dụng kỹ thuật thu thập tín hiệu kỹ thuật số và công nghệ cảm biến độ ẩm. Kích thước nhỏ và tiêu thụ năng lượng thấp làm cho nó phù hợp cho các ứng dụng khắc nghiệt. Một số ứng dụng của cảm biến này được liệt kê bên dưới:

- DHT22 được áp dụng trong HVAC.
- Thiết bị kiểm tra và kiểm tra sử dụng DHT22 để đo các giá trị nhiệt độ và độ ẩm.
- Cảm biến này được sử dụng trong các thiết bị gia dụng và sản phẩm tiêu dùng để đo các giá trị nhiệt độ và độ ẩm.
- Trong các đơn vị y tế để phát hiện các giá trị độ ẩm trong các đơn vị cách ly bệnh nhân, DHT22 được sử dụng.
- Đối với các thiết bị y tế được sử dụng bởi bệnh nhân nhạy cảm với nhiệt độ cao và mắc một số bệnh về phổi, DHT22 được sử dụng để kiểm tra độ ẩm không khí.
- Sensor này được sử dụng trong các trạm thời tiết, tủ lạnh, máy Fax, thiết bị chế biến thực phẩm, v.v.

Phần B: Tìm hiểu về Firebase Realtime Database

I. Firebase Realtime Database là gì?



Realtime Database một service của Firebase. Theo định nghĩa trong tài liệu của firebase thì Realtime Database là:

Store and sync data with our NoSQL cloud database. Data is synced across all clients in realtime, and remains available when your app goes offline.

The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronized in realtime to every connected client. When you build cross-platform apps with our iOS, Android, and JavaScript SDKs, all of your clients share one Realtime Database instance and automatically receive updates with the newest data.

Dịch nôm na thì nó có nghĩa là một cơ sở dữ liệu NoSQL lưu và đồng bộ dữ liệu trên mây. Dữ liệu được đồng bộ trên tất cả clients trong thời gian thực, và vẫn khả dụng khi ứng dụng offline.

Firebase Realtime Database là cơ sở dữ liệu lưu trữ trên mây. Dữ liệu được lưu trữ và đồng bộ hóa theo thời gian thực với mỗi client được kết nối. Khi bạn xây dựng ứng dụng đa nền tảng với iOS, Android, và javascript SDK, tất cả các client của bạn chia sẻ một thể hiện Realtime Database và tự động tiếp nhận các thay đổi với dữ liệu mới nhất.

II. Các khả năng chính của Realtime Database

- *Realtime:*

Firebase Realtime Database sử dụng đồng bộ dữ liệu mỗi khi dữ liệu có thay đổi, mọi thiết bị được kết nối sẽ nhận được thay đổi trong vài mili giây.

- *Offline:*

Khi người dùng ngoại tuyến, dữ liệu sẽ được lưu trên bộ nhớ cache của thiết bị và tự động đồng bộ khi bạn trực tuyến. Tất cả là tự động.

- *Accessible from Client Devices*

Firebase Realtime Database có thể truy cập từ một thiết bị mobile hoặc trình duyệt web. Nó không cần một ứng dụng server nào cả. Bảo mật và xác thực dữ liệu

có thể thông qua các Rule bảo mật của Firebase Realtime Database, các rule được thực thi khi dữ liệu được đọc hoặc ghi.

- *Firebase Cloud Storage*

Firebase Cloud Storage là một không gian lưu trữ dữ liệu, nó giống như một chiếc ổ cứng. Bạn có thể upload và download các loại file bạn muốn. Đó có thể là một file ảnh, hay file văn bản, .zip, ...

- *Firebase Cloud Function*

Cloud Functions Firebase cho phép ta viết những câu truy vấn database lưu trữ trên cloud. Code của bạn được lưu trữ trong cloud của Google và chạy trong một môi trường bảo mật, được quản lý. Ta không cần quan tâm đến vấn đề mở rộng các máy chủ. Với firebase, khi bạn muốn lấy dữ liệu bạn cần phải viết các câu truy vấn trực tiếp từ client.

Điều này có thể vô tình để lộ một số thông tin nhạy cảm. Để khắc phục vấn đề đó, Cloud Function đã ra đời. Nhiều lúc, các developers muốn kiểm soát logic trên server để tránh giả mạo phía client. Ngoài ra, đôi khi không muốn mã của mình khi bị decode sẽ gây ra các vấn đề về bảo mật. Cloud Functions được tách biệt hoàn toàn với client, vì vậy ta có thể yên tâm nó bảo mật và luôn thực hiện chính xác những gì ta muốn.

- *Firebase Analytics*

Firebase Analytics là tính năng giúp ta phân tích hành vi của người sử dụng trên ứng dụng. Cuối cùng nó sẽ đưa ra lời khuyên về lộ trình xây dựng ứng dụng. Để làm việc này cần cài đặt SDK (Software Development Kit, cụ thể hơn là FirebaseAnalytics.uninstallpackage), chức năng phân tích sẽ trở nên khả dụng.

Khi đó, ta không chỉ xem được hành vi của người dùng mà còn có thể biết được thông tin về như hiệu quả quảng cáo, tình trạng trả phí, v.v. Với tính năng này, chúng ta có thể biết người dùng thường xuyên truy cập tính năng nào, từ đó có thể đưa ra chiến lược phát triển sản phẩm của mình.

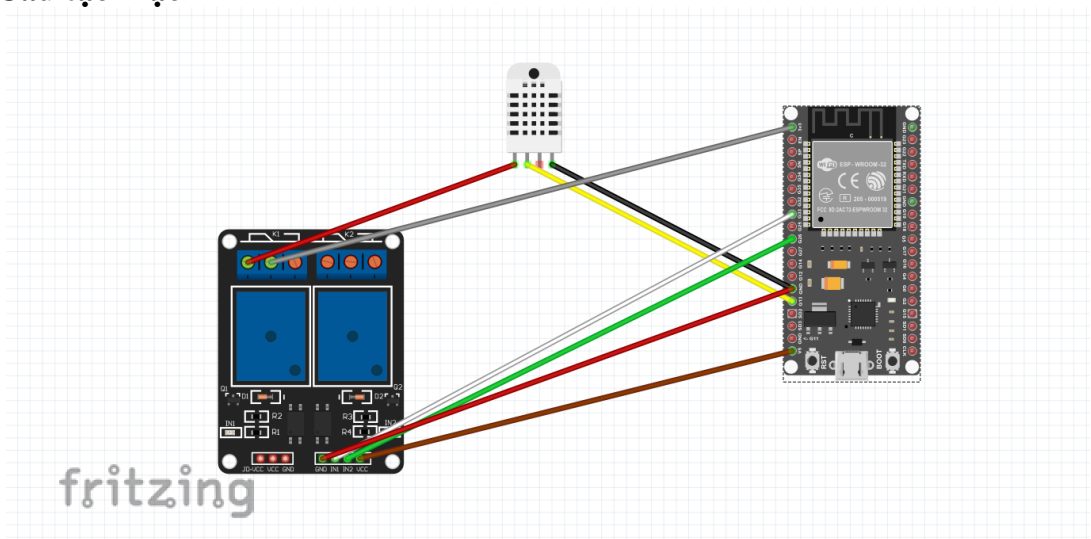
- *Machine Learning Kit*

Với Machine Learning Kit bạn có thể làm một số việc:

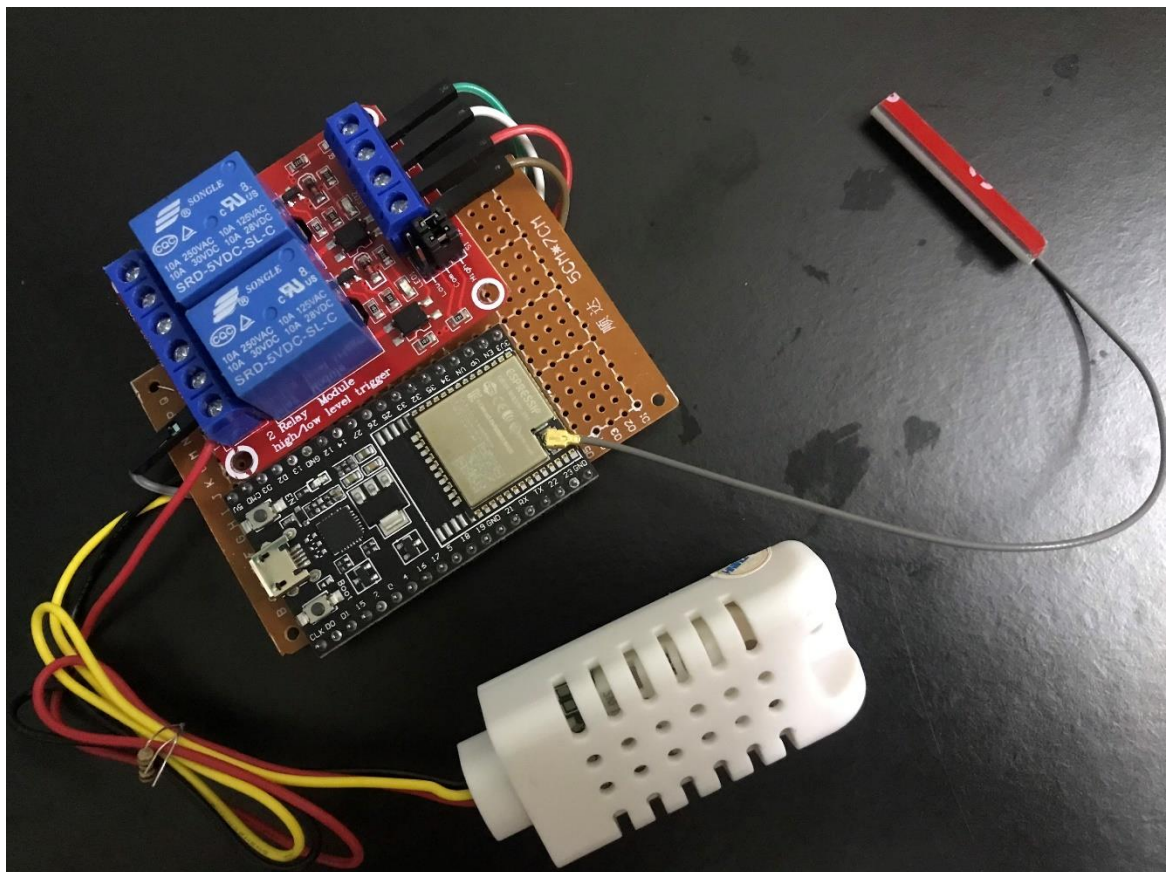
- Text recognition (nhận dạng văn bản viết tay/ máy)
- Barcode scanning (quét mã vạch)
- Landmark recognition (nhận diện mốc)
- Image labeling (ghi nhãn hình ảnh)
- Face detection (nhận diện khuôn mặt)

Phần C: Nguyên lý hoạt động của mạch

I. Cấu tạo mạch



Hình 2.1. Mô phỏng trên frizing



Hình 2.2. Mạch thật

II. Nguyên lý hoạt động

Mạch hoạt động dựa trên KIT Arduino ESP32 WROOM 32U kết nối với Internet để điều khiển các thiết bị trong phòng học từ xa. Ta có thể điều khiển bật tắt quạt, đèn thông qua điện thoại, máy tính, ...có kết nối với Internet. Ngoài ra, trong mạch còn có cảm biến nhiệt độ, độ ẩm; nó có thể đo được nhiệt độ và độ ẩm trong phòng học và hiển thị lên Firebase.

III. Các bước giao tiếp với Realtime Database Firebase sử dụng ESP32

1. Cách sử dụng Firebase
 - Tạo project trên firebase
 - Khởi tạo và cài đặt Firebase Realtime Database
 - Khởi tạo cấu trúc dữ liệu cho Realtime Database Firebase
2. Kết nối ESP32 với Firebase truyền nhận dữ liệu
 - Thêm thư viện và kết nối với Realtime Database Firebase
 - Lập trình ESP32 truyền dữ liệu lên Firebase
 - Get dữ liệu từ Firebase về ESP32
 - Push dữ liệu từ ESP32 lên Firebase

IV. Code điều khiển

```
#include <WiFi.h>
#include <FirebaseESP32.h>
#include <DHT.h>
#include <math.h>

FirebaseData dulioufirebase;

#define FIREBASE_HOST "https://smarthome2023-60f1a-default-rtdb.asia-southeast1.firebaseio.com/"
#define FIREBASE_AUTH "GLjY2qHvoXBEIbDWX6YCNAqeV3EFSUyV0g8YAf0s"
#define WIFI_SSID "NhaTro66-T6"
#define WIFI_PASS "12345678"

const int DHTPIN = 32;
const int DHTTYPE = DHT22;
DHT dht(DHTPIN, DHTTYPE);

const int relay1 = 26;
const int relay2 = 33;
//int dem = 0;

String firestatus1 = "";
```

```
String firestatus2 = "";

void ketnoiwifi() {
  Serial.println("Connecting to Wifi...");
  WiFi.begin("NhaTro66-T6", "12345678");
  while(!(WiFi.status() == WL_CONNECTED)){
    delay(1000);
    Serial.print("...");
  }
  Serial.println("");
  Serial.println("Wifi connected");
}

void setup() {
  Serial.begin(115200);
  ketnoiwifi();
  dht.begin();
  pinMode(relay1, OUTPUT);
  pinMode(relay2, OUTPUT);

  digitalWrite(relay1, LOW);
  digitalWrite(relay2, LOW);

  Firebase.begin(FIREBASE_HOST,FIREBASE_AUTH);
  Firebase.setString(dulieufirebase,"THCS Nhom4 ", "SmartHome2023");

  Firebase.setString(dulieufirebase,"RELAY1 : ", "OFF");
  Firebase.setString(dulieufirebase,"RELAY2 : ", "OFF");

}

void loop() {
  float t = dht.readTemperature();
  float h = dht.readHumidity();

  Firebase.setFloat(dulieufirebase,"Tempurature = ",t);
  Firebase.setFloat(dulieufirebase,"Humidity = ",h);

  Serial.print(" Tempurature = ");
```



```
Serial.println(t);
Serial.println("°C");
Serial.print(" Humidity = ");
Serial.println(h);
Serial.println("%");
delay(2000);

Firebase.getString(dulieufirebase,"RELAY1 : ");
firestatus1 = dulieufirebase.stringData();
if (firestatus1 == "ON" || firestatus1 == "on") {
    Serial.println(" RELAY 1 ON!");
    digitalWrite(relay1, HIGH);
}
else if (firestatus1 == "OFF" || firestatus1 == "off") {
    Serial.println("RELAY 1 OFF!");
    digitalWrite(relay1, LOW);
}
else {
    Serial.println("Error!! Try again with ON/OFF");
}

Firebase.getString(dulieufirebase,"RELAY2 : ");
firestatus2 = dulieufirebase.stringData();
if (firestatus2 == "ON" || firestatus2 == "on") {
    Serial.println("RELAY 2 ON!");
    digitalWrite(relay2, HIGH);
}
else if (firestatus2 == "OFF" || firestatus2 == "off") {
    Serial.println("RELAY 2 OFF!");
    digitalWrite(relay2, LOW);
}
else {
    Serial.println("Error!! Try again with ON/OFF");
}
}
```

Phần D: Ưu điểm, nhược điểm và hướng phát triển

I. ƯU ĐIỂM

- Dễ dàng cài đặt
- Thiết kế đơn giản
- Tiết kiệm chi phí
- Dễ dàng sửa chữa và tìm lỗi khi xảy ra sự cố

II. NHƯỢC ĐIỂM

- Kit ESP32 khan hiếm tại Việt Nam
- ESP32 nhận wifi chậm chờn, dễ xảy ra hỏng hóc

III. HƯỚNG PHÁT TRIỂN

Có thể ứng dụng trong hệ thống nhà thông minh, đô thị thông minh và các hệ thống IoT khác.

--HẾT--