

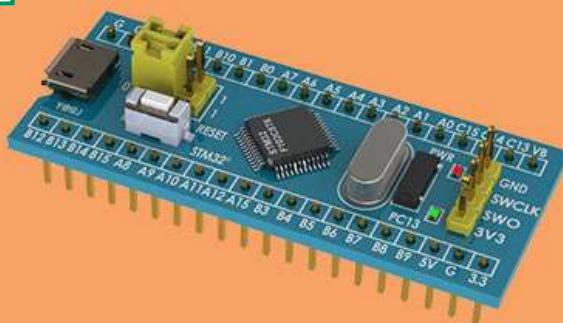
LẬP TRÌNH STM32

Bài 15: Lập trình STM32 WWDG Window Watchdog Timer kiểm lỗi phần mềm

POSTED ON 02/03/2021 BY KHUÊ NGUYỄN

02
Th3

Lập trình STM32 và CubeMX

**Khuê Nguyễn Creator**

Bài 15: Sử dụng STM32 WWDG kiểm lỗi phần mềm

STM32 WWDG hay Window Watchdog timer là một timer định thời sử dụng trong việc kiểm lỗi phần mềm, bằng cách kiểm tra các tác vụ phần mềm có chạy đúng thời gian quy định hay không.

Bài 15 trong serie **Học lập trình STM32 từ A tới Z**

Mục Lục

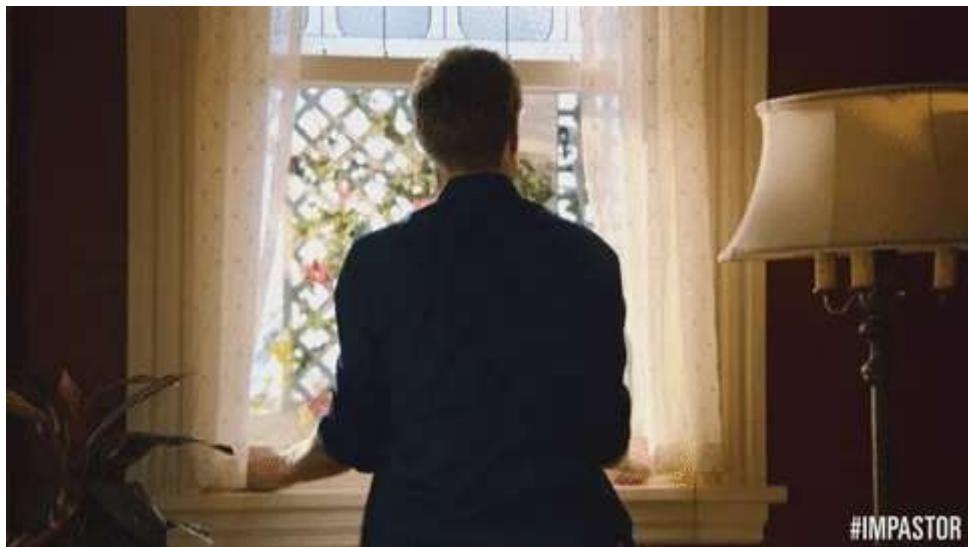
1. WWDG là gì?
2. Sơ đồ và nguyên lý hoạt động của STM32 WWDG
 - 2.1. Sơ đồ khối của STM32 WDG
 - 2.2. Nguyên lý làm việc của WWDG Timer
 - 2.3. Công thức tính thời gian
3. Lập trình STM32 WWDG với CubeMx và Keil C
 - 3.1. Cấu hình trong CubeMx
 - 3.2. Lập trình STM32 WDG trên Keil C
4. Kết
 - 4.1. Related posts:



WWDG là gì?

WWDG (Window Watchdog timer) là một bộ định thời sử dụng trong việc phát hiện lỗi phần mềm bởi các tác nhân như code không chặt chẽ, nhiều bug, các tác nhân bên ngoài làm ảnh hưởng tới sự đúng đắn của chương trình.

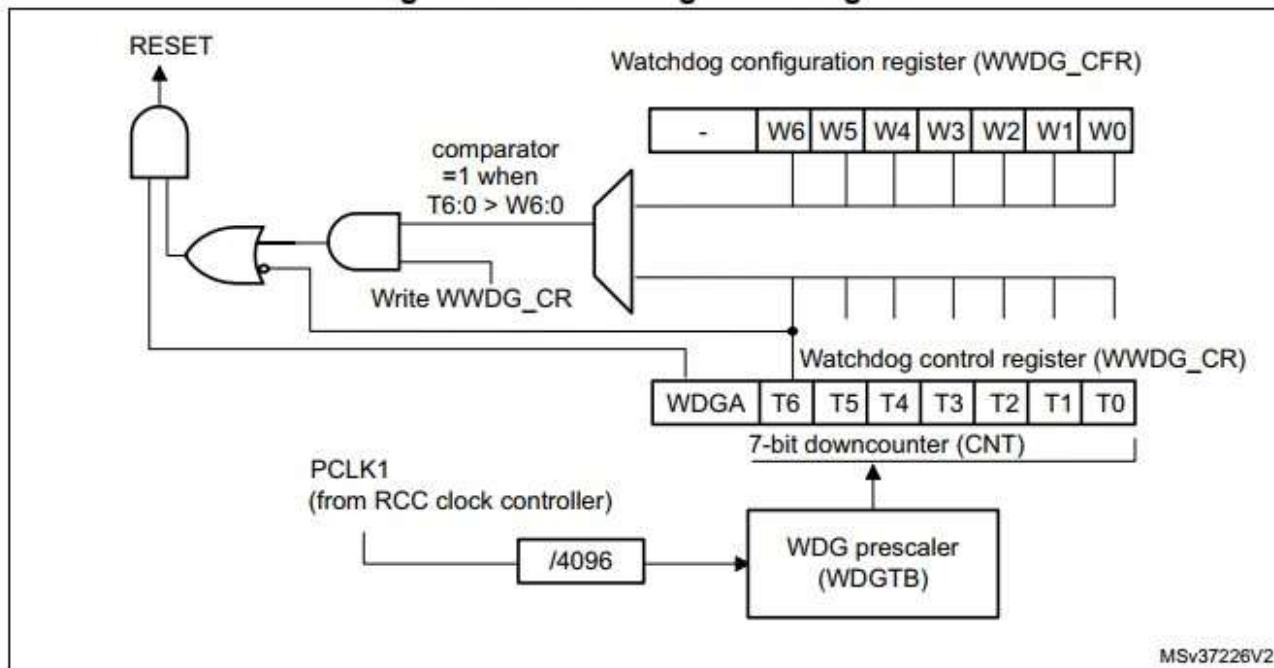
Cũng giống như IWDG, WWDG sẽ tạo ra 1 tín hiệu reset hệ thống khi phát hiện sự cố, chỉ khác là thời điểm để làm mới bộ định thời phải trong khoảng từ: Tmin – Tmax, nếu sớm hơn hoặc muộn hơn đều không được.



Sơ đồ và nguyên lý hoạt động của STM32 WWDG

Sơ đồ khối của STM32 WDG

Figure 183. Watchdog block diagram



Bộ STM32 WWDG bao gồm:

- WWDG_CRF: Thanh ghi chứa giá trị so sánh (Window)
- WWDG_CR: Thanh ghi điều khiển, gồm 1 Bit điều khiển WDGA và 7 bit giá trị nạp lại cho counter
- WWDGTTB: Bit xác định giá trị chia tần

Bộ STM32 WWDG sử dụng xung clock từ bộ dao động của chương trình vậy nên nếu clock bị treo thì nó cũng sẽ không hoạt động.

Nguyên lý làm việc của WWDG Timer

Khi bit WDGA được Active, Timer sẽ bắt đầu đếm xuống từ giá trị của 7-bit counter tới 0

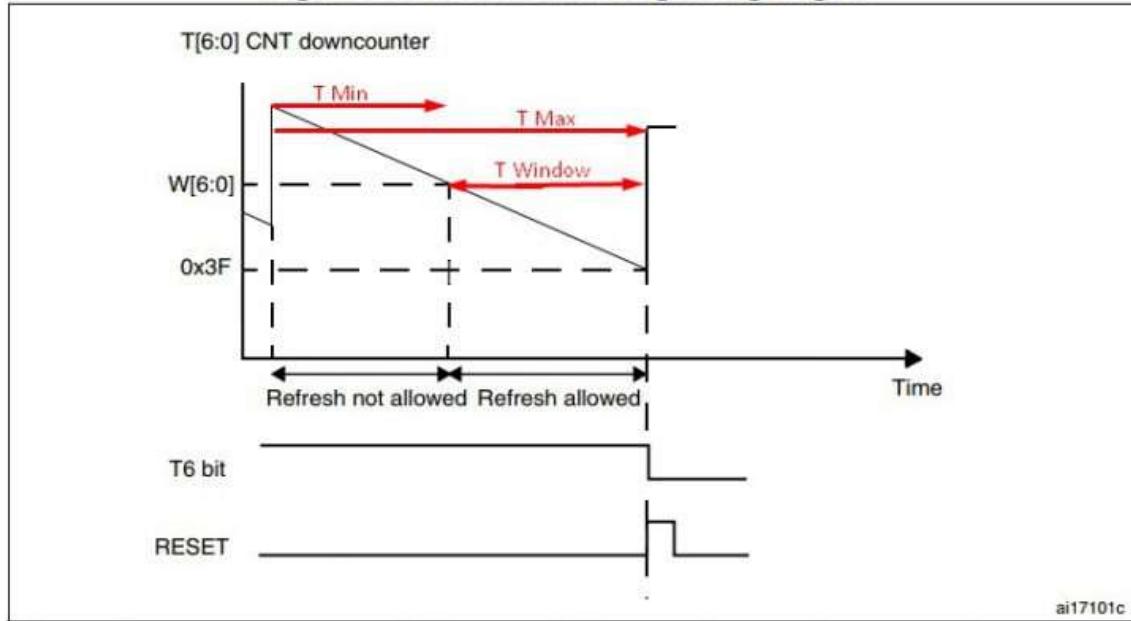
Trong khi counter đếm, nếu có 1 lệnh refresh (Write WWDG_CR) sẽ xảy ra 3 tình huống.

1. Nếu giá trị của counter > Giá trị thanh ghi WWDG_CRF (Window) => xảy ra sự kiện reset sớm EWI, có thể được bật để nhảy vào ngắt reset sớm
2. Nếu giá trị của Counter < WWDG_CRF < 0x40 => không xảy ra reset và Counter sẽ được nạp lại
3. Nếu giá trị Counter < 0x40 => Sảy ra sự kiện ngắt

Nghĩa là Nếu nạp lại giá trị cho Counter trong khoảng T Window => chương trình chạy bình thường, ngoài khoảng đó sẽ xảy ra ngắt.

Giá trị Twindow = Tmax – Tmin

Figure 184. Window watchdog timing diagram



The formula to calculate the WWDG timeout value is given by:

$$t_{\text{WWDG}} = t_{\text{PCLK1}} \times 4096 \times 2^{\text{WDGTB}[1:0]} \times (\text{T}[5:0] + 1) \quad (\text{ms})$$

Công thức tính thời gian

Công thức tính thời gian time out có thể tham khảo công thức và bảng tham khảo thời gian Time out cho f = 36Mhz

$$T_{WWDG} = \frac{(4096 \times 2^{WDGTB} \times (t[5:0]+1))}{f_{PCLK1}}$$

Table 98. Minimum and maximum timeout values @36 MHz (f_{PCLK1})

| Prescaler | WDGTB | Min timeout value | Max timeout value |
|-----------|-------|-------------------|-------------------|
| 1 | 0 | 113 µs | 7.28 ms |
| 2 | 1 | 227 µs | 14.56 ms |
| 4 | 2 | 455 µs | 29.12 ms |
| 8 | 3 | 910 µs | 58.25 ms |

Công thức tính toán khi lập trình

Khi lập trình ta sẽ ước tính sẵn thời gian Twindow chương trình hoạt động. Vậy nên ta cần tính 2 giá trị để nạp vào Counter và Window theo công thức

$$\text{COUNTER} = \frac{\text{MAX Time}_{(\text{ms})} \times \text{APBI CLK}}{4096 \times 2^{(\text{PR})} \times 1000} + 64$$

$$\text{WINDOW} = \text{COUNTER} - \frac{\text{MIN Time}_{(\text{ms})} \times \text{APBI CLK}}{4096 \times 2^{(\text{PR})} \times 1000}$$

WWGD FORMULA

Lập trình STM32 WWDG với CubeMx và Keil C

Mô tả phương pháp: Chúng ta sử dụng led PC13 khi vào chương trình sẽ nháy 1 lần, trong while(1) chúng ta sẽ delay(ms) sau đó refresh lại STM32 WWDG, nếu

thời gian refresh nằm trong khoảng Window, chương trình sẽ chạy ở trong đó và Led PC13 sẽ sáng.

Nếu refresh ngoài window, chip sẽ bị reset và PC13 sẽ nháy liên tục.

Cấu hình trong CubeMx

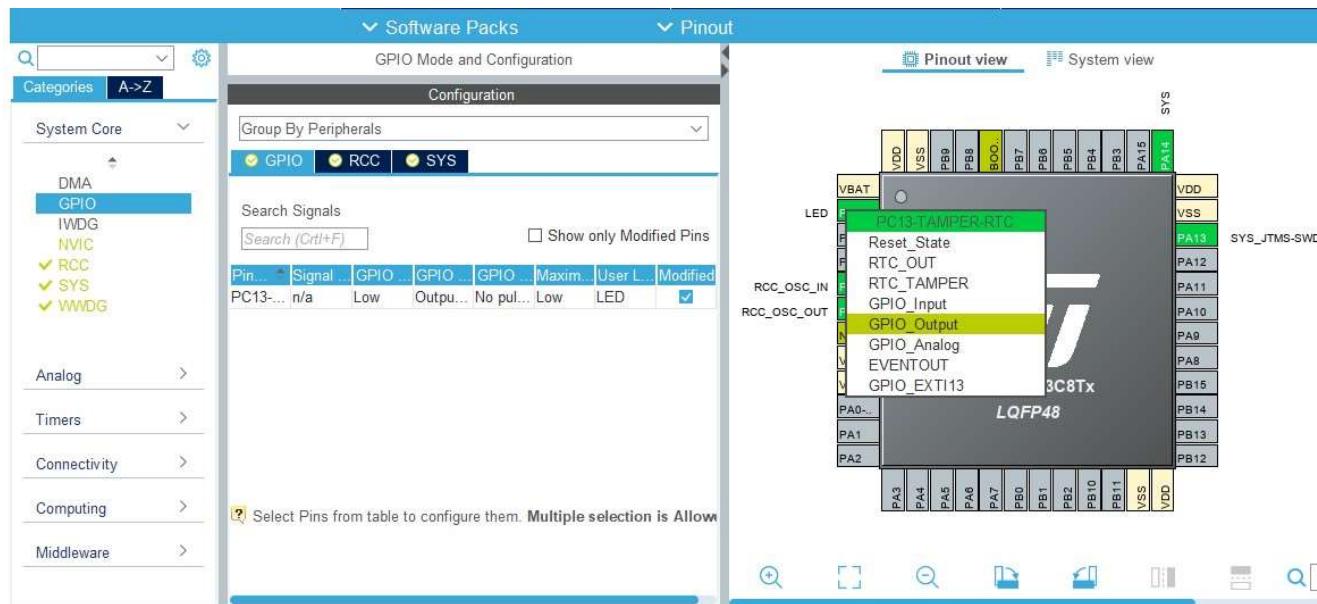
Trong SYS để Debug là Serial Wire

Trong RCC để chip chạy thạch anh ngoài HSE

The screenshot shows the CubeMX software interface with the 'Clock Configuration' tab selected. In the left sidebar, under 'System Core', the 'RCC' option is highlighted. The main configuration area shows the following settings:

- Mode:**
 - High Speed Clock (HSE): Crystal/Ceramic Resonator
 - Low Speed Clock (LSE): Disable
 - Master Clock Output
- Configuration:**
 - Reset Configuration
 - Parameter Settings** (highlighted)
 - GPIO Settings
 - User Constants
- System Parameters:**
 - VDD voltage (V): 3.3 V
 - Prefetch Buffer: Enabled

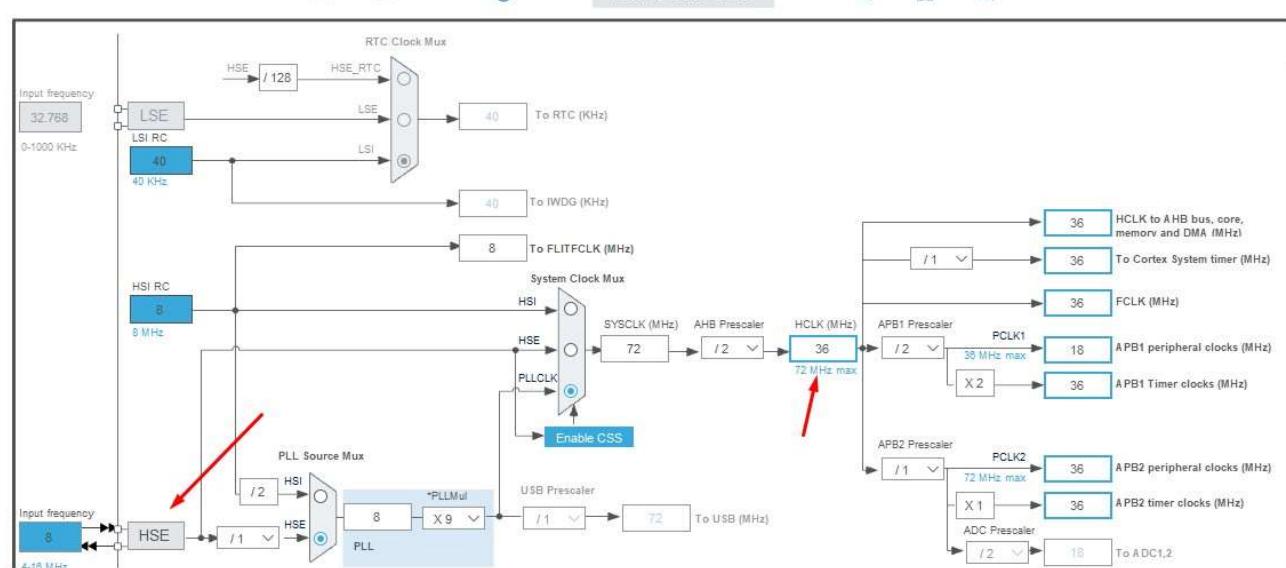
Trong GPIO chọn PCB 13 là Output và đổi tên thành LED



Trong WWDG chọn Active và setup thông số như hình



Trong Clock chọn tần số dao động là 36 Mhz



Giải thích giá trị nạp vào:

Ta sẽ chọn thời gian Window từ 30ms – 50ms $\Rightarrow T_{min} = 30$, $T_{max} = 50$. Áp dụng công thức với AHBI CLK = 36 000 000. Ta sẽ tính ra được Counter và Window là 119 và 86.

Nạp giá trị Window vào WWDG window value, Counter vào WWDG free-running

Lập trình STM32 WDG trên Keil C

Trước while(1) chúng ta bật tắt led PC13, sau đó khởi động WWDG với lệnh Init.

Trong While(1) delay 1 khoảng thời gian sau đó refresh lại counter

```

88  /* Initialize all configured peripherals */
89  MX_GPIO_Init();
90  HAL_GPIO_WritePin(LED_GPIO_Port, LED_Pin, GPIO_PIN_RESET);
91  HAL_Delay(500);
92  HAL_GPIO_WritePin(LED_GPIO_Port, LED_Pin, GPIO_PIN_SET);
93  HAL_Delay(500);
94  MX_WWDG_Init();
95  /* USER CODE BEGIN 2 */
96
97  /* USER CODE END 2 */
98
99  /* Infinite loop */
100 /* USER CODE BEGIN WHILE */
101 while (1)
102 {
103     /* USER CODE END WHILE */
104
105     /* USER CODE BEGIN 3 */
106
107     HAL_Delay(1);
108     HAL_WWDG_Refresh(&hwdg);
109 }
110 /* USER CODE END 3 */
111 }
112

```

Nếu thời gian delay từ 30 – 50ms led sẽ luôn sáng, nếu vượt ra khoảng này led sẽ nhấp nháy liên tục

Kết

STM32 WWDG là một timer nâng cao trong việc tìm kiếm và sửa lỗi phần mềm. Vậy nên Timer này dường như ít được mọi người sử dụng. Thế nhưng hiệu quả nó đem lại thì rất ấn tượng nếu chương trình cần tính chính xác cao.

Nếu thấy bài viết này hữu ích hãy chia sẻ cho người khác nhé, chúc các bạn học tập thật tốt !!!

5/5 - (1 bình chọn)

Related Posts:

1. [Lập trình STM32 điều khiển LCD1602 chế độ 8bit và 4bit](#)
2. [Bản đồ bộ nhớ \(Memory map\) vi điều khiển STM32F103](#)
3. [Bài 16: Lập trình STM32 USB CDC truyền nhận dữ liệu qua cổng COM ảo](#)
4. [Bài 12: Lập trình STM32 với giao thức SPI](#)
5. [Bài 11: Lập trình STM32 với Giao thức UART](#)
6. [Bài 8: Lập trình STM32 đọc ADC một kênh](#)



KHUÊ NGUYỄN

Chỉ là người đam mê điện tử và lập trình. Làm được gì thì viết cho anh em xem thôi. :D

Trả lời

Email của bạn sẽ không được hiển thị công khai. Các trường bắt buộc được đánh dấu *

Bình luận *

Tên *

Email *

Trang web

PHẢN HỒI

Fanpage

Khuê Nguyễn Creator - Họ...
2.754 lượt thích

Đã thích Chia sẻ

**Khuê Nguyễn Creator - Học
Lập Trình Vi Điều Khiển**
khoảng một tháng trước

Lý do thời gian gần đây mình không viết bài
và làm thêm gì cả là đây 😊)
Chính thức ra mắt sản phẩm định vị thông
minh vTag.

Đây là một sản phẩm định vị đa năng với
3 công nghệ định vị WIFI, GPS, LBS kết
hợp với sóng NB-IOT dành riêng cho các
sản phẩm IOT.

Chỉ với 990.000đ chúng ta đã có thể có
sản phẩm để:

- Định vị trẻ em, con cái... [Xem thêm](#)

Bài viết khác

Lập trình 8051 - AT89S52



Bài 1: Tổng quan về 8051 và chip AT89S51 - 52

Tổng quan về 8051

8051 là một dòng chip nhập môn cho lập trình viên nhúng, chúng được sử...

[ĐỌC THÊM](#)

Lập trình STM32 và CubeMX



Lập trình STM32 HID Host

giao tiếp với chuột và bàn phím

Lập trình STM32 USB HID Host giao tiếp với chuột và bàn phím máy tính

Trong bài này chúng ta sẽ cùng học STM32 HID Host, biến STM32 giống như...

[ĐỌC THÊM](#)



Lộ trình học lập trình nhúng từ A tới Z

Lập trình nhúng là một ngành có cơ hội nhưng cũng đòi hỏi nhiều kiến...

3 COMMENTS

[ĐỌC THÊM](#)

Lập trình STM32 và CubeMX



Khuê Nguyễn Creator



Lập trình STM32F407 SDIO đọc dữ liệu thẻ nhớ

Lập trình STM32 SDIO đọc ghi dữ liệu vào thẻ nhớ SD card

Trong bài này chúng ta cùng học cách lập trình STM32 SDIO, một chuẩn giao...

[ĐỌC THÊM](#)

Lập trình STM32 và CubeMX



Khuê Nguyễn Creator



Lập trình STM32F407 DAC chuyển đổi số sang tương tự

Lập trình STM32 DAC tạo sóng hình Sin trên KIT STM32F407 Discovery

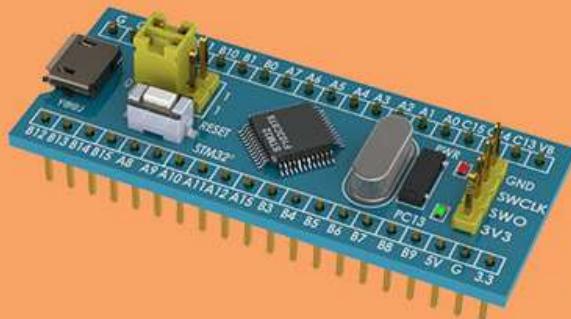
Trong bài này chúng ta sẽ cùng nhau tìm hiểu STM32 DAC với KIT STM32F407VE...

[ĐỌC THÊM](#)

Lập trình STM32 và CubeMX



Khuê Nguyễn Creator



Sử dụng hàm printf để in Log khi Debug trên STM32

Hướng dẫn sử dụng printf với STM32 Uart để in Log trên Keil C

Trong bài này chúng ta sẽ học cách retarget hàm printf của thư viện stdio...

3 COMMENTS

[ĐỌC THÊM](#)

ESP32 và Platform IO



Khuê Nguyễn Creator



Bài 9 WIFI: Lập trình ESP32 OTA nạp firmware trên Internet

Lập trình ESP32 FOTA nạp firmware qua mạng Internet với OTA Drive

Trong bài này chúng ta sẽ học cách sử dụng ESP32 FOTA (Firmware Over The...

4 COMMENTS

[ĐỌC THÊM](#)

Lập trình Nuvoton



Khuê Nguyễn Creator



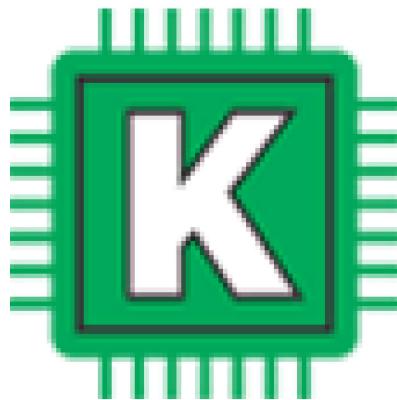
Cài đặt SDC Complier và Code:Blocks IDE

Hướng dẫn cài đặt SDCC và Code:Blocks lập trình Nuvoton

Ở bài này chúng ta sẽ cài đặt các công cụ cần thiết cho việc...

[ĐỌC THÊM](#)





KHUÊ NGUYỄN CREATOR

Chia sẻ đam mê

Blog này làm ra để lưu trữ tất cả những kiến thức, những câu chuyện của mình. Đôi khi là những ý tưởng nhất thời, đôi khi là các dự án tự mình làm. Chia sẻ cho người khác cũng là niềm vui của mình, kiến thức mỗi người là khác nhau, không hẳn quá cao siêu nhưng sẽ có lúc hữu dụng.

DMCA PROTECTED

Liên Kết

Nhóm: Nghịên Lập Trình

Fanpage: Khuê Nguyên Creator

My Shop

Thông Tin

Tác Giả

Chính Sách Bảo Mật



Copyright 2022 © Khuê Nguyễn