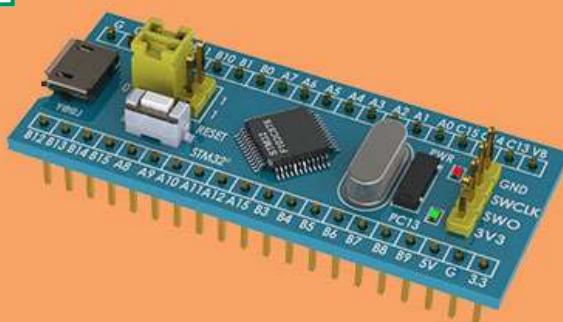


**LẬP TRÌNH STM32**

Bài 13: Lập trình STM32 RTC – Real Time Clock

POSTED ON 16/12/2020 BY KHUÊ NGUYỄN

16
Th12**Khuê Nguyễn Creator**

Bài 13: Lập trình STM32 RTC đồng hồ thời gian thực

STM32 RTC là một ngoại vi giúp cho vi điều khiển có thể chạy được các tác vụ thời gian thực như tính toán ngày, tháng, năm, phút giây mà không cần đến các chip thời gian thực khác như DS3231, DS1307...

Bài 13 trong serie **Học lập trình STM32 từ A tới Z**



Mục Lục

1. RTC là gì?
 - 1.1. Ưu nhược điểm của RTC on chip so với các IC RTC khác như DS3231, DS1307
 - 1.2. Các thông số chính của STM32 RTC
2. Cấu hình STM32 RTC tạo đồng hồ thời gian thực
3. Lập trình STM32 RTC trong Keil C
4. Kết nối phần cứng cho STM32 RTC
5. Kết
 - 5.1. Related posts:

RTC là gì?

RTC hay Real Time Clock là ngoại vi sử dụng như một bộ đồng hồ – lịch thời gian thực. Thuật ngữ thời gian thực là chỉ thời gian trong cuộc sống của con người hằng ngày, chứ không phải thời gian trong tính toán như các Timer khác.

Để bộ RTC có thể hoạt động chúng ta cần nối chân Vbat của vi điều khiển với một nguồn 3V, thường sử dụng Pin Cmos, nguồn điện này giúp duy trì xung RTC hoạt động bình thường.



real time clock được ứng dụng rất nhiều trong nhúng

Ưu nhược điểm của RTC on chip so với các IC RTC khác như DS3231, DS1307

Ưu điểm:

- Tiết kiệm chi phí linh kiện, góp phần giảm giá thành sản phẩm
- Có thể sử dụng thanh ghi Backup 16bit như 1 Eprom điều mà STM32 thiếu

Nhược điểm

- Sai số từ bộ dao động lớn (1%), khi chạy lâu dài sẽ dẫn tới sai thời gian

Các thông số chính của STM32 RTC

- Có thể cấu hình Clock đầu vào theo 3 nguồn đó là: LSE, HSE, LSI
- Bộ chia clock lên đến 20 bit, giúp bộ RTC hoạt động chính xác.
- Độ phân giải của timer RTC lên đến 32 bit – tức là 2^{32} giây mới tràn và cần reset lại.

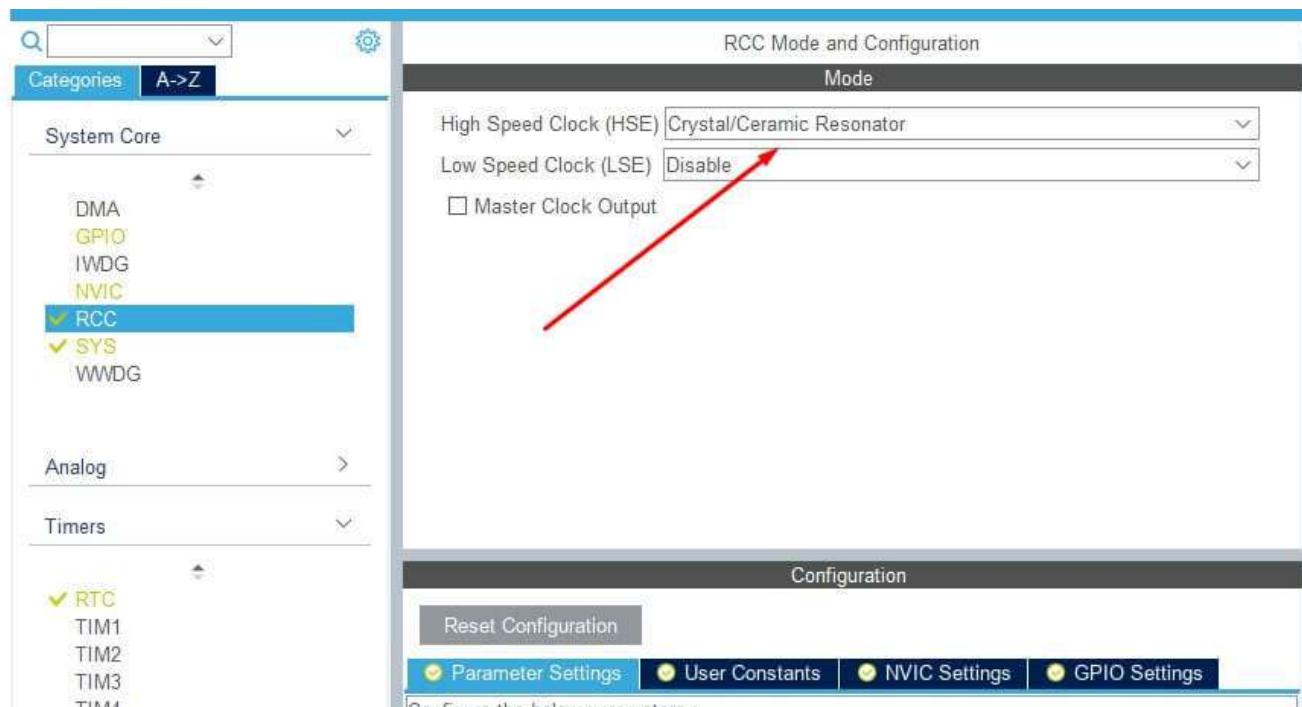
- 3 nguồn clock source có thể được sử dụng.
- 2 loại Reset RTC riêng biệt.
- Có các ngắt hỗ trợ là : ngắt Alarm, ngắt mỗi giây, ngắt tràn bộ đếm.

Trong bài hướng dẫn này, mình sẽ hướng dẫn các bạn tạo ra một chiếc đồng hồ chạy thời gian thực, nó sẽ vẫn chạy dù cho nguồn vi điều khiển có bị mất hay không. Bắt đầu nhé

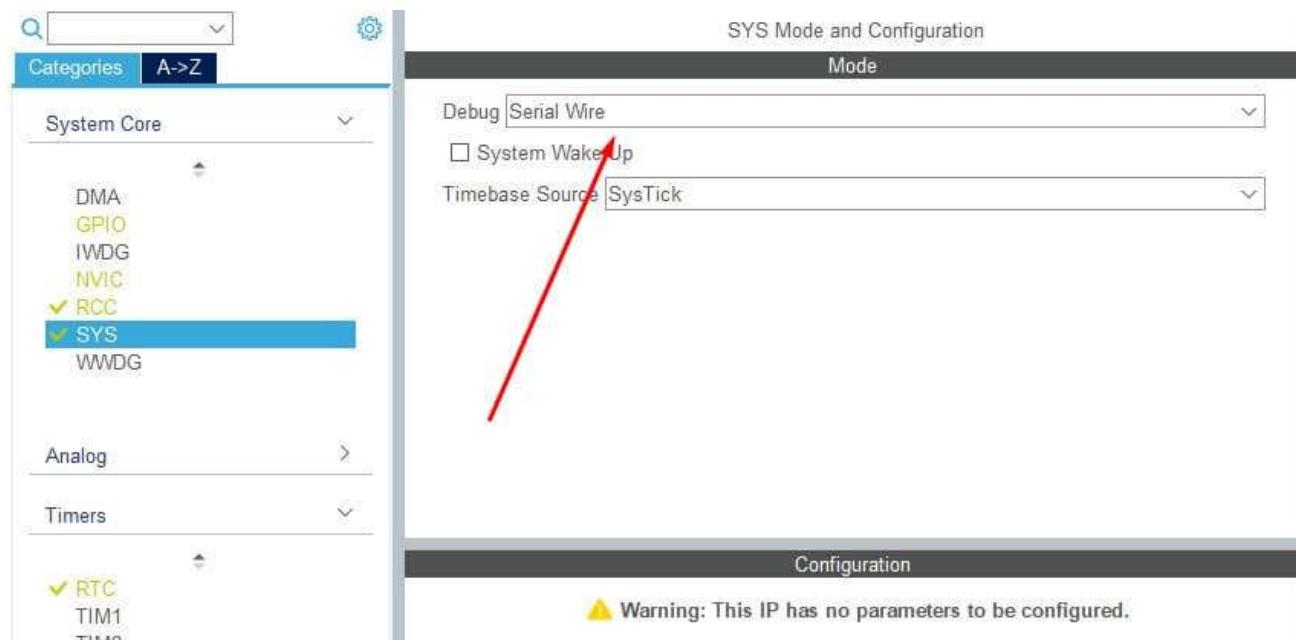
Cấu hình STM32 RTC tạo đồng hồ thời gian thực

Để bắt đầu ta sẽ cấu hình STM32 RTC trong Cube MX

Trong Tab System Core, chọn RCC là HSE



Tab SYS, chọn Debug là Serial Wire



Tab RTC tích chọn Active Source Clock để bật bộ RTC, chọn Active Calendar để cấu hình ngày giờ bắt đầu cho RTC, chọn Tamper nếu sử dụng chân Tamper làm chân xuất tín hiệu RTC Out

Trong tab cấu hình

- Data Format chọn là Binary data format: Định dạng cho thời gian là kiểu số nhị phân
- Time, minutes, sec: thời gian ban đầu của hệ thống
- Day, date, week, ...: Lịch ban đầu của hệ thống
- Các tab còn lại là cấu hình xung đầu ra cho RTC

RTC Mode and Configuration

Mode

- Activate Clock Source
- Activate Calendar

RTC OUT: Disable

Tamper

Configuration

Reset Configuration

Parameter Settings User Constants NVIC Settings

Configure the below parameters :

Search (Ctrl+F)

Calendar Time

Data Format	BCD data format
Hours	0
Minutes	0
Seconds	0

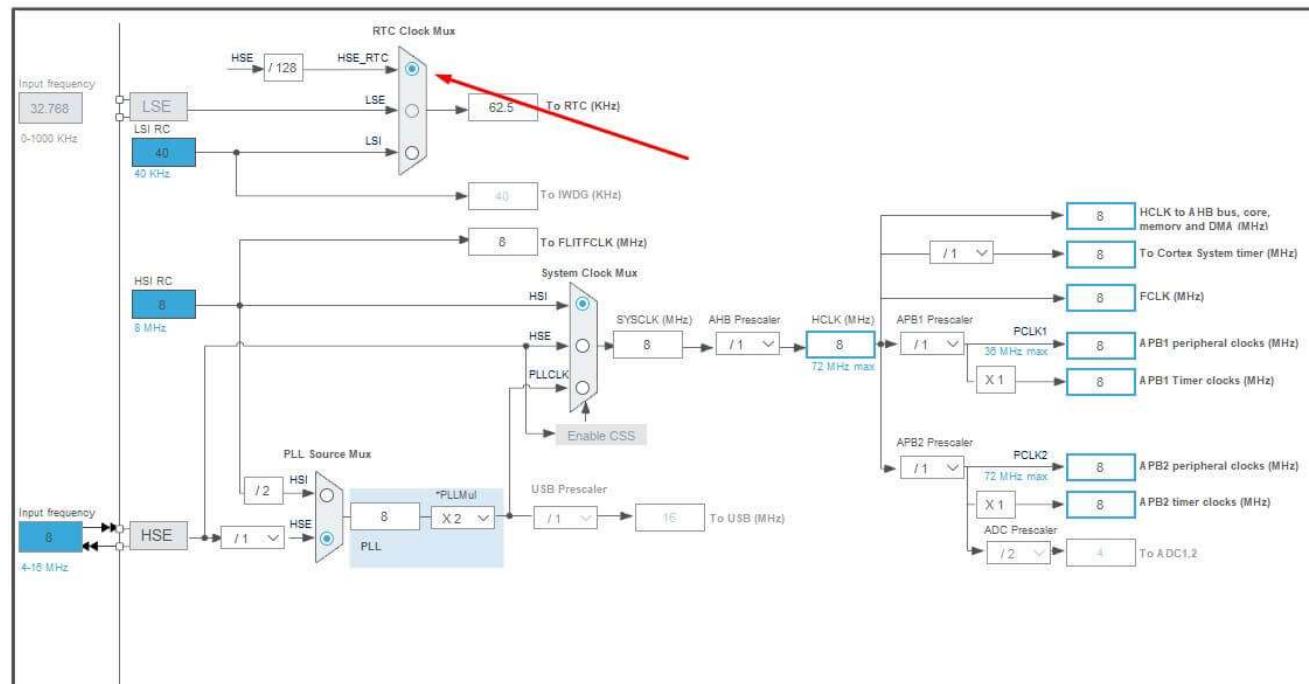
General

Auto Predivider Calculation	Enabled
Asynchronous Predivider value	Automatic Predivider Calculation Enabled
Output	Alarm pulse signal on the TAMPER pin

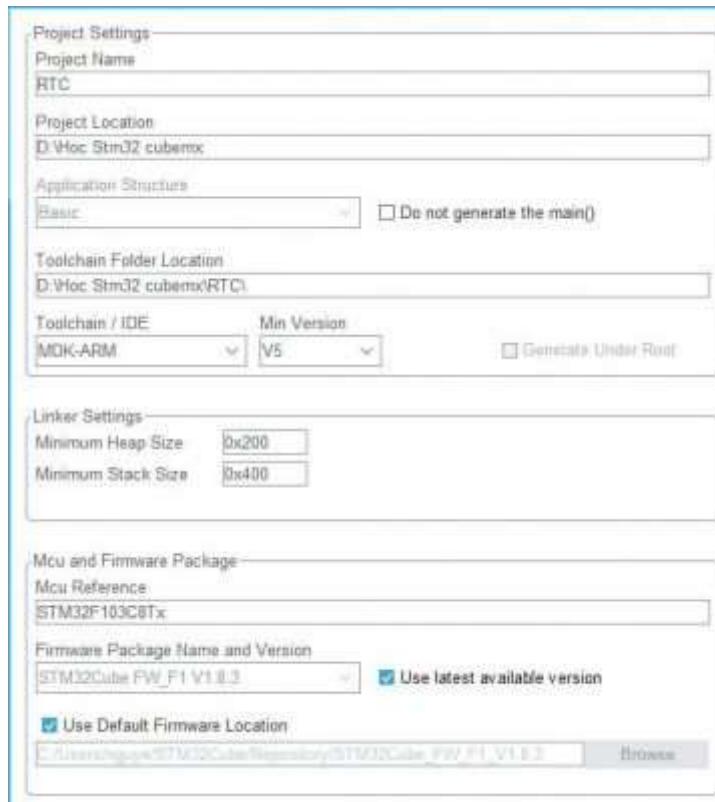
Calendar Date

Week Day	Monday
Month	January
Date	1
Year	0

Mở phần mềm lên và chọn chip STM32F103C8T6, cấu hình Clock và HSE, Cấu hình Clock cho RTC là HSE



Chuyển sang Project đặt tên và Gen Code



Lập trình STM32 RTC trong Keil C

Mở project đã tạo lên.

Trong phần cấu hình MX_RTC_Init(), ta sử dụng hai hàm Read và Write Backup Register. Công dụng của đoạn code này chính là, nếu Time và Date đã được ghi, thanh ghi RTC_BKP_DR1 sẽ có dữ liệu = 0x2608 (Số này tùy chọn), thanh ghi này sẽ không mất dữ liệu nếu chân Vbat được cấp 3V. Vì vậy ta chỉ có thể ghi dữ liệu vào RTC 1 lần mà thôi, khi reset MCU, dữ liệu thời gian, ngày tháng sẽ không bị ghi đè.

Copy đoạn code ghi dữ liệu vào RTC

```

183     /* USER CODE END Check_RTC_BKUP */
184
185
186     /** Initialize RTC and set the Time and Date*/
187     */
188     sTime.Hours = 12;
189     sTime.Minutes = 12;
190     sTime.Seconds = 12;
191
192     if (HAL_RTC_SetTime(&hrtc, &sTime, RTC_FORMAT_BIN) != HAL_OK)
193     {
194         Error_Handler();
195     }
196     DateToUpdate.WeekDay = RTC_WEEKDAY_MONDAY;
197     DateToUpdate.Month = RTC_MONTH_JANUARY;
198     DateToUpdate.Date = 1;
199     DateToUpdate.Year = 0;
200
201     if (HAL_RTC_SetDate(&hrtc, &DateToUpdate, RTC_FORMAT_BIN) != HAL_OK)
202     {
203         Error_Handler();
204     }
205     /* USER CODE BEGIN RTC_Init_2 */
206

```

Paste vào trong if(HAL_RTCEx_BKUPRead(&hrtc, RTC_BKP_DR1) == 0x2608) , cuối cùng sử dụng HAL_RTCEx_BKUPWrite(&hrtc, RTC_BKP_DR1, 0x2608); để ghi vào BKP_DR1

```

178     /* USER CODE BEGIN Check_RTC_BKUP */
179     if(HAL_RTCEX_BKUPRead(&hrtc, RTC_BKP_DR1) == 0x2608) // kiem tra thanh ghi backup
180     {
181         /** Initialize RTC and set the Time and Date*/
182         sTime.Hours = 12;
183         sTime.Minutes = 12;
184         sTime.Seconds = 12;
185
186         if (HAL_RTC_SetTime(&hrtc, &sTime, RTC_FORMAT_BIN) != HAL_OK)
187         {
188             Error_Handler();
189         }
190         DateToUpdate.WeekDay = RTC_WEEKDAY_MONDAY;
191         DateToUpdate.Month = RTC_MONTH_JANUARY;
192         DateToUpdate.Date = 1;
193         DateToUpdate.Year = 0;
194
195         if (HAL_RTC_SetDate(&hrtc, &DateToUpdate, RTC_FORMAT_BIN) != HAL_OK)
196         {
197             Error_Handler();
198         }
199         HAL_RTCEX_BKUPWrite(&hrtc, RTC_BKP_DR1, 0x2608); // sau khi ghi time, date, ghi vao thanh ghi bkp
200     }
201
202     /* USER CODE END Check_RTC_BKUP */
203

```

Phía trên main() tạo 2 biến toàn cục dạng struct lưu thời gian và ngày tháng

```

39  /* USER CODE END PM */
40
41  /* Private variables ----- */
42  RTC_HandleTypeDef hrtc;
43
44  /* USER CODE BEGIN PV */
45  RTC_TimeTypeDef sTime; // struct luu thoi gian
46  RTC_DateTypeDef sDate; // struct luu ngay thang nam
47  /* USER CODE END PV */
48
49  /* Private function prototypes ----- */
50  void SystemClock_Config(void);
51  static void MX_GPIO_Init(void);
52  static void MX_RTC_Init(void);
53

```

Trong While(1) chúng ta sẽ đọc giá trị của thanh ghi RTC vào 2 biến đó

```

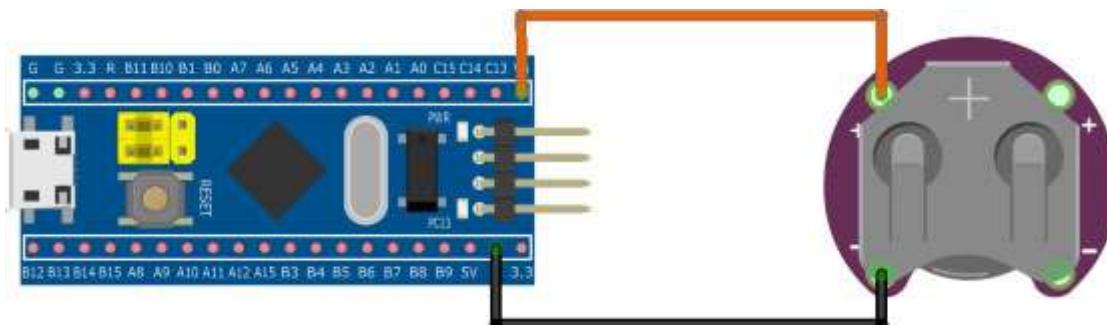
94  /* Infinite loop */
95  /* USER CODE BEGIN WHILE */
96  while (1)
97  {
98  /* USER CODE END WHILE */
99  HAL_RTC_GetTime(&hrtc, &sTime, RTC_FORMAT_BIN);
100  HAL_RTC_GetDate(&hrtc, &sDate, RTC_FORMAT_BIN);
101  /* USER CODE BEGIN 3 */
102  }
103  /* USER CODE END 3 */
104 }
105
106

```

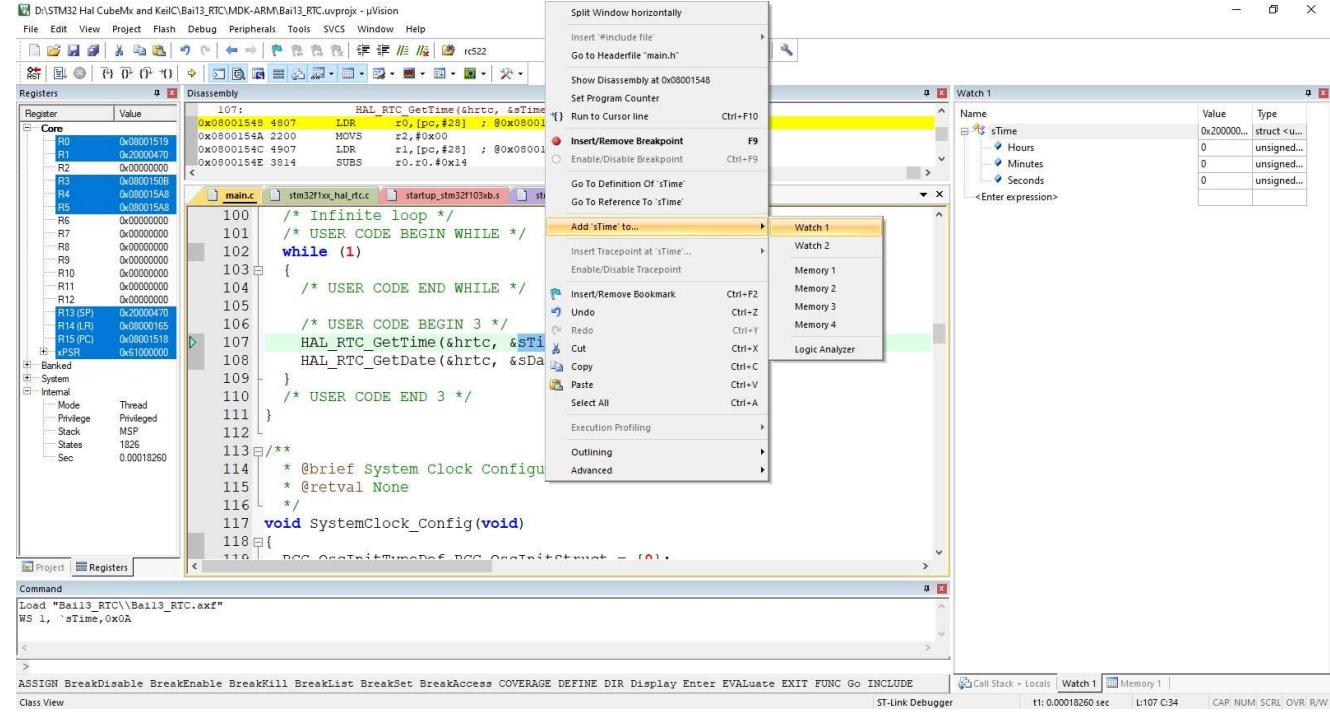
Nhấn F7 để build chương trình, F8 để nạp code vào Kit

Kết nối phần cứng cho STM32 RTC

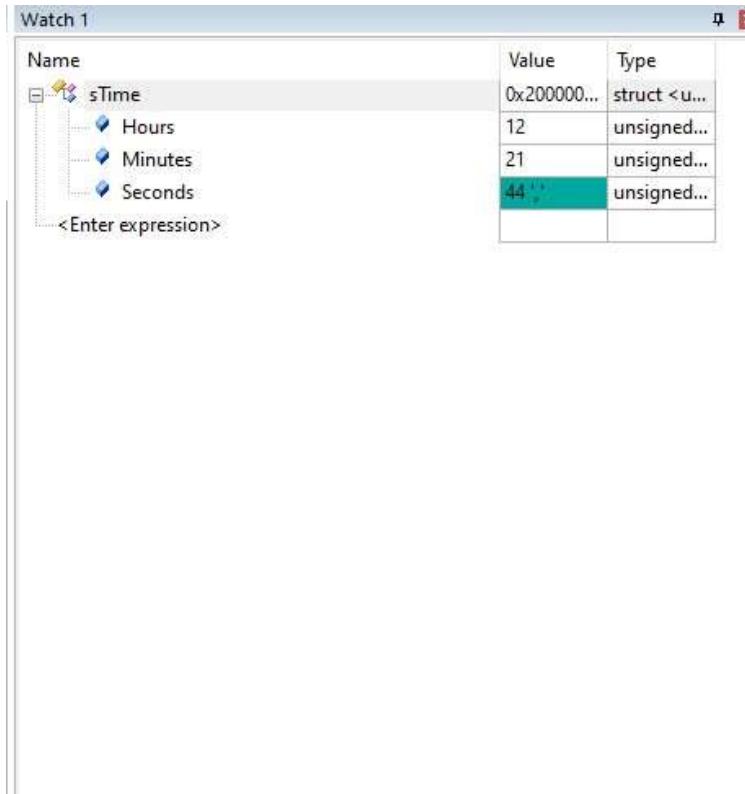
Để RTC có thể hoạt động được chúng ta cần nối Vbat với Pin 3V, đảm bảo rằng chân đó luôn có điện



Sau khi nạo chương trình vào chip nhấn vào Debug, Chọn các biến sTime, sDate cho vào Watch 1.



Run trong debug mode để xem sự thay đổi của các biến



Ta thấy rằng đồng hồ đã bắt đầu đếm số giây. Nhấn reset mạch, ta thấy rằng thời gian không bị reset về giá trị set ban đầu => Thanh ghi BKP không bị mất

dữ liệu.

Kết

STM32 RTC rất dễ học, được ứng dụng rất nhiều, các bạn có thể sử dụng nó làm một chiếc đồng hồ báo thức thông minh, một lịch vạn niên hiển thị cả ngày, tháng, năm.

Hi vọng kiến thức vừa rồi có thể giúp ích được cho các bạn, hẹn gặp lại vào các bài tiếp theo.

4.7/5 - (4 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. [Lập trình STM32 từ A tới Z](#)
5. [Bài 12: Lập trình STM32 với giao thức SPI](#)
6. [Bài 11: Lập trình STM32 với Giao thức UART](#)



KHUÊ NGUYỄN



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

3 THOUGHTS ON “BÀI 13: LẬP TRÌNH STM32 RTC – REAL TIME CLOCK”



Gerardo Viscarra says:

i like this awesome post

19/12/2020 AT 11:15 SÁNG

TRẢ LỜI



Khuê Nguyễn says:

Thank for reading!

19/12/2020 AT 5:02 CHIỀU

TRẢ LỜI



Tranliem says:

Khi reset kit, sTime không set lại giá trị ban đầu, nhưng sDate thì có, vì sao vậy, thk!

03/12/2021 AT 12:32 SÁNG

TRẢ LỜI

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.

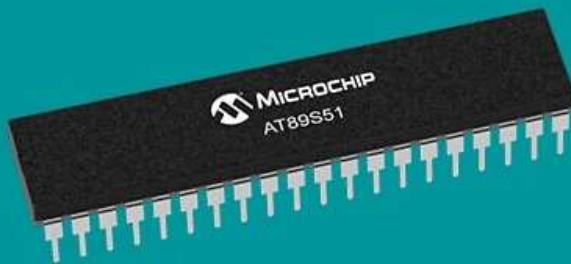
Đây là một sản phẩm định vị đa năng với
3 công nghệ định vị WIFI, GPS, LBS kết
hợp với sóng NB-IOT dành riêng cho các
sản phẩm IOT.

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

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

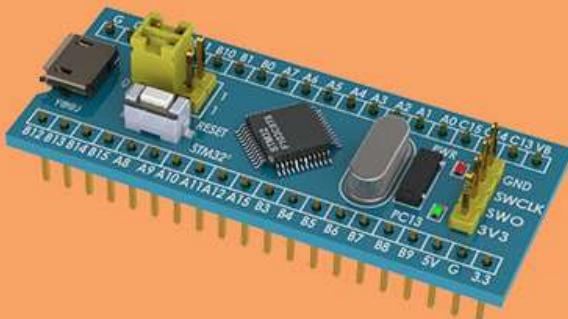
Trong bài này chúng ta sẽ cùng nhau tìm hiểu STM32 DAC với KIT STM32F407VE...

[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



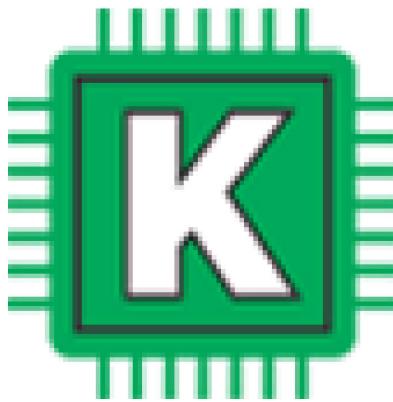
Cài đặt SDC Complier và Code:Blocks IDE

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