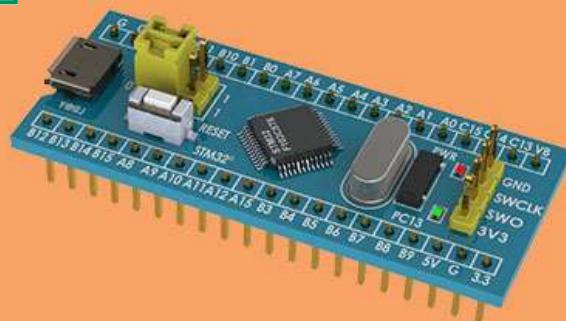


**LẬP TRÌNH STM32**

Bài 8: Lập trình STM32 đọc ADC một kênh

POSTED ON 13/07/2020 BY KHUÊ NGUYỄN

13
Th7**Khuê Nguyễn Creator**

Bài 8: ADC là gì, ADC 1 kênh chế độ single và cont

Lập trình STM32 đọc ADC một kênh sẽ giúp các bạn

- Hiểu ADC là gì
- Cách cấu hình ADC trên STM32 Cube MX
- Cách lập trình ADC chế độ Single Mode và Continuous Mode

Bài 8 trong serie **Học lập trình STM32 từ A tới Z**



Mục Lục

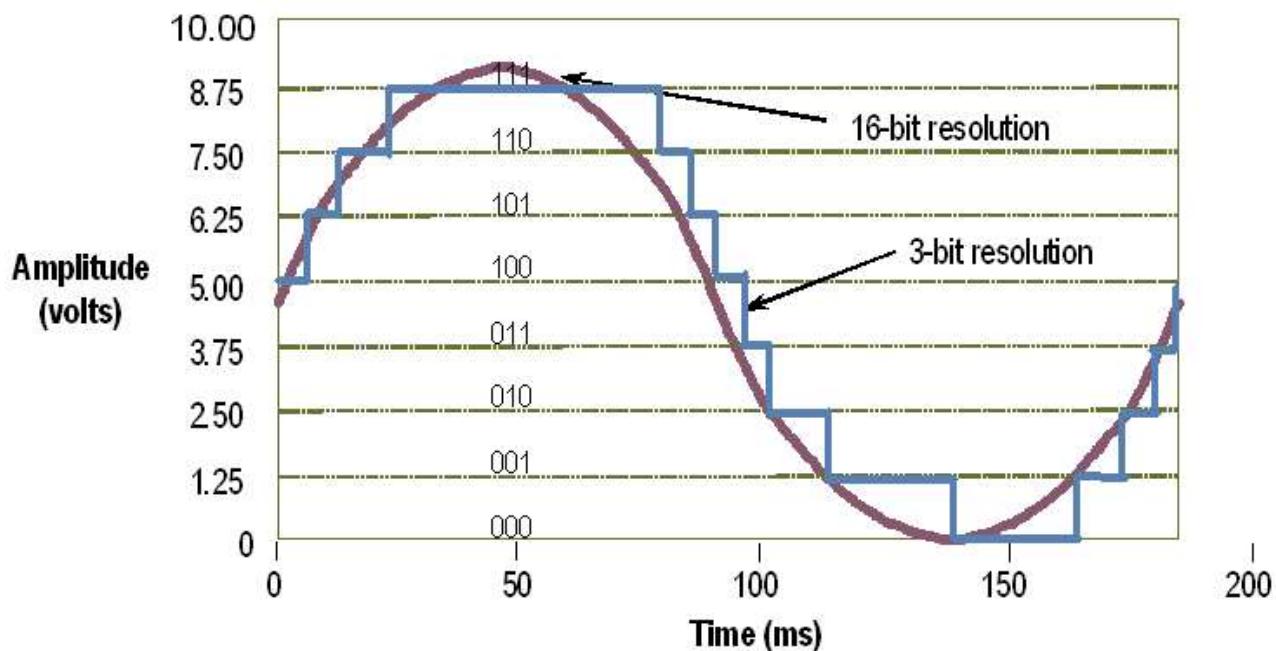
1. ADC là gì?
 2. Các chức năng chính của ADC trong STM32
 3. ADC một kênh chế độ Single và Continuous
 4. Cấu hình ADC một kênh trong STM32 CubeMX
 5. Lập trình ADC một kênh với STM32
 6. Kết
- 6.1. Related posts:

ADC là gì?

ADC Analog to Digital Convert là bộ chuyển đổi tương tự sang số. Đại lượng tương tự là Điện áp Vin được so sánh với điện áp mẫu Vref (giá trị lớn nhất), sau đó được chuyển đổi thành số lưu vào thanh ghi DATA của bộ chuyển đổi đó.

Có 2 tham số quan trọng của bộ ADC cần lưu ý:

- Tốc độ lấy mẫu (sampling) được tính theo số chu kỳ chuyển đổi
- Độ phân giải: Tính theo Bit bộ ADC có độ phân giải 10 Bit sẽ có $2^{10} = 1024$ giá trị



Bài 8: Lập trình STM32 đọc ADC một kênh 53

Các chức năng chính của ADC trong STM32

Độ phân giải 12Bit

Sinh ra ngắt tại các sự kiện End of convert, End of Injected, Analog Watchdog

Chế độ Single hoặc Continuous

Chế độ Scan tự động quét từ Kênh 0 đến Kênh n (mỗi bộ có 10 kênh tối đa)

Có cơ chế cân chỉnh tay

Data Alignment (Căn chỉnh Data) căn trái hoặc căn phải

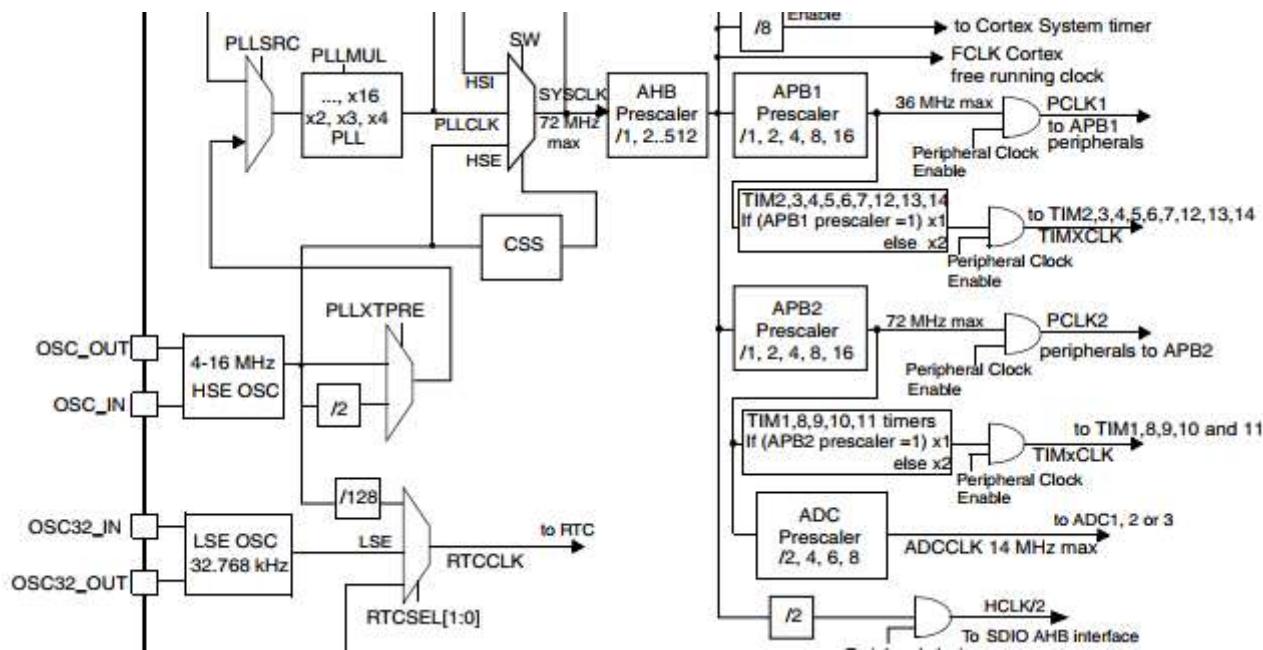
Cài đặt thời gian chuyển đổi đến từng Kênh

Có thể kích hoạt bằng xung bên ngoài

Chế độ Dual mode sử dụng cùng lúc 2 hoặc nhiều bộ ADC

Hỗ trợ DMA

Tần số chuyển đổi ADC được lấy từ bộ ABP2 thông qua ADC prescaler và phải nhỏ hơn 14mhz



Bài 8: Lập trình STM32 đọc ADC một kênh 54

Chi tiết tham khảo mục 11.2 trong reference manual

ACD một kênh chế độ Single và Continuous

Với chế độ Single bộ ADC chỉ chuyển đổi 1 lần rồi dừng, một sự kiện ngắt được sinh ra nếu bit EOCIR được set lên 1

Với chế độ Continuous bộ ADC sẽ chuyển đổi liên tục, một sự kiện ngắt được sinh ra nếu bit EOCIR được set lên 1

DMA sẽ sinh ra nếu bit DMA dc set lên 1(DMA chỉ có trên ADC1 và ADC3)

Kết quả convert được lưu vào thanh ghi DR

Bắt đầu convert bằng cách set bit ADON lên 1

Cơ chế để tạo ADC như sau:

1. Enable bộ clock cho ADC, ghi hệ số chia cho ADC prescaler
2. Chọn các kênh cần chuyển đổi
3. Chọn chế độ chuyển đổi Single, Continuous, Scan, Discontinuous

4. Chọn thời gian lấy mẫu (Sampling Time)
5. Chọn Ngắt hoặc DMA
6. Start bộ chuyển đổi
7. Kiểm tra cờ EOC hoặc trong ngắt đọc dữ liệu từ thanh ghi DR về

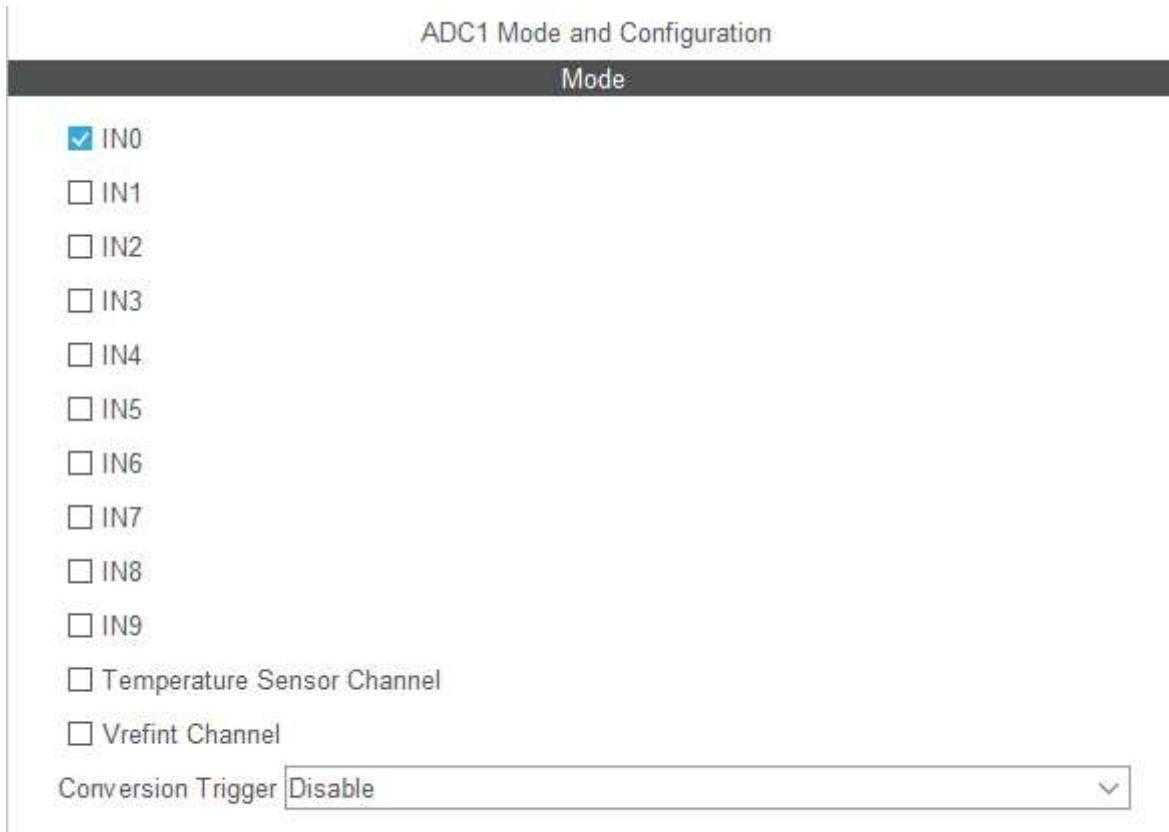
Cấu hình ADC một kênh trong STM32 CubeMX

Trong bài này chúng ta sẽ sử dụng ADC1 channel 0 chế độ Single hoặc Continuous để đọc giá trị từ biến trở về.

Đầu tiên các bạn mở phần mềm CubeMX, chọn chip STM32F103C8 và start project.

Trong Sys chọn debug: serial wire các bước này mình đã nói kĩ trong [Bài 3](#) rồi nhé

Tiếp tục trong ADC1 tick chọn Chanel 0



Bài 8: Lập trình STM32 đọc ADC một kênh 55

Trong configurations chọn như hình.

Giải thích:

- Data Alignment: Right Align Căn chỉnh data bên phải. Vì thanh ghi dữ liệu là 16 bit, chuyển đổi là 12bit, thanh ghi này sẽ xác định dữ liệu sẽ được căn lề bên trái hoặc phải
- Scan Mode: Disable. Tắt chế độ quét(Chỉ sử dụng được khi sử dụng nhiều kênh)
- Continuous mode: Disable . Tắt chế độ làm việc liên tục sẽ là chế độ Single
- Rank: 1.Mức ưu tiên của kênh (sử dụng khi chuyển đổi nhiều kênh)
- Sampling Time: 239.5 Cycles.Thời gian chuyển đổi tính theo chu kì ADC

Lưu ý: Sampling time càng cao thì chuyển đổi càng chính xác

Configuration

Reset Configuration

NVIC Settings	DMA Settings	GPIO Settings
Parameter Settings	User Constants	
<input type="button" value="Search (Ctrl+F)"/>	<input type="button" value=""/>	<input type="button" value=""/>
ADCs_Common_Settings Mode: Independent mode		
ADC_Settings Data Alignment: Right alignment Scan Conversion Mode: Disabled Continuous Conversion Mode: Disabled Discontinuous Conversion Mode: Disabled		
ADC-Regular_ConversionMode Enable Regular Conversions: <input checked="" type="checkbox"/> Number Of Conversion: 1 External Trigger Conversion Source: Regular Conversion launched by software		
Rank Channel: Channel 0 Sampling Time: 239.5 Cycles		
ADC_Injected_ConversionMode Enable Injected Conversions: Disable		
WatchDog Enable Analog WatchDog Mode: <input type="checkbox"/>		

Bài 8: Lập trình STM32 đọc ADC một kênh 56

Trong NVIC tick vào bật ngắt ADC1

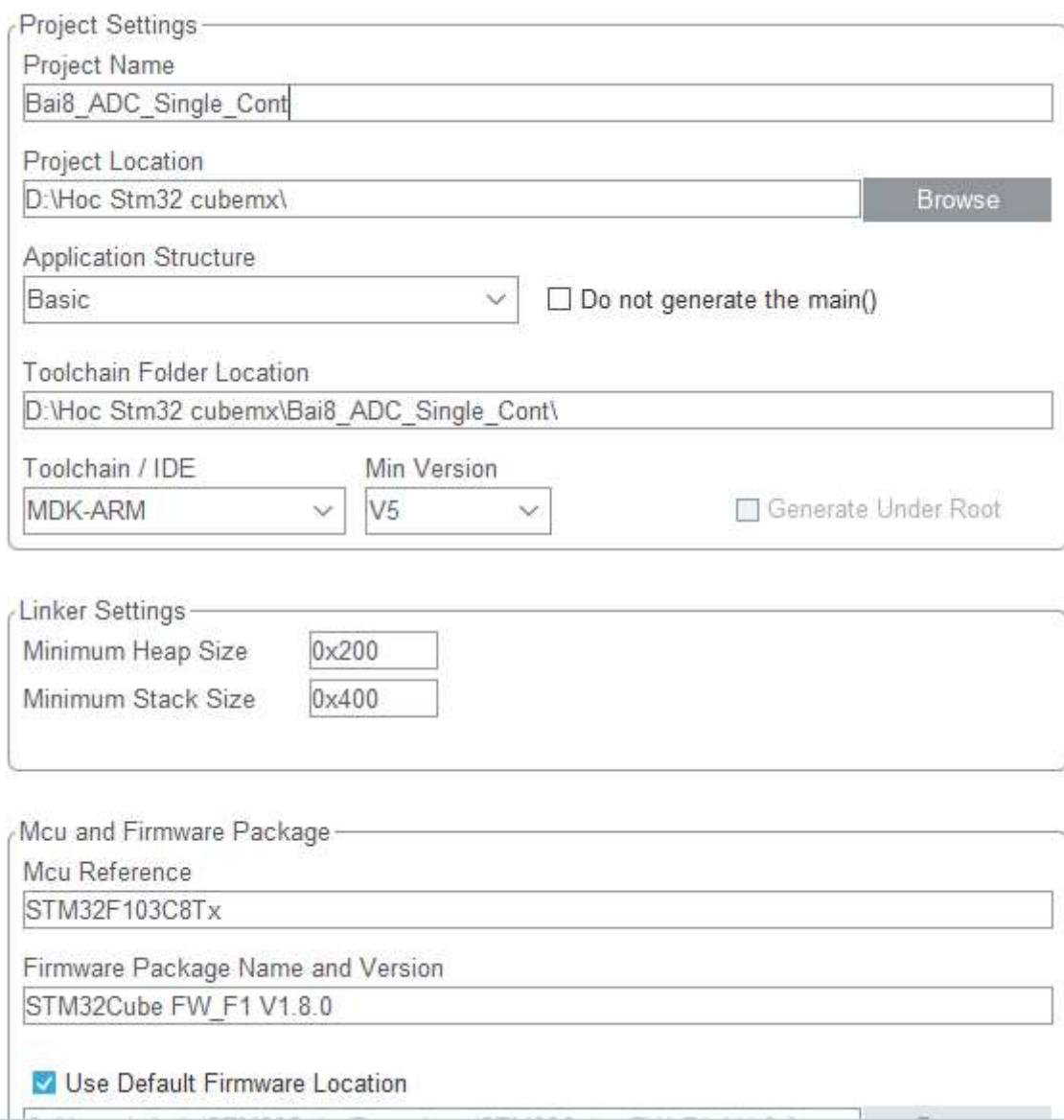
Configuration

Reset Configuration

NVIC Settings	DMA Settings	GPIO Settings
Parameter Settings	User Constants	
NVIC Interrupt Table ADC1 and ADC2 global interrupts: Enabled, Preemption Priority 0, Sub Priority 0		

Bài 8: Lập trình STM32 đọc ADC một kênh 57

Đặt tên project chọn toolchain và gen code



Bài 8: Lập trình STM32 đọc ADC một kênh 58

Lập trình ADC một kênh với STM32

Nhấn F7 để Build project. Trong stm32f1xx_it.c tìm đến hàm xử lý ngắt ADC.

Go to define nó. Hẳn bạn đã rất quen thuộc bước này.

```

203 void ADC1_2_IRQHandler(void)
204 {
205     /* USER CODE BEGIN ADC1_2_IRQHandler 0 */
206
207     /* USER CODE END ADC1_2_IRQHandler 0 */
208     HAL_ADC_IRQHandler();
209     /* USER CODE BEGIN */
210
211     /* USER CODE END */
212 }
213
214 /* USER CODE BEGIN */
215
216 /* USER CODE END 1 */
217 /*****END OF FILE*****/
218

```

-----^
ied identifier - function

Bài 8: Lập trình STM32 đọc ADC một kênh 59

Tìm tới hàm HAL_ADC_ConvCpltCallback(hadc), Go to define

```

1799     if(ADC_IS_SOFTWARE_START_REGULAR(hadc)          &&
1800         (hadc->Init.ContinuousConvMode == DISABLE)    )
1801     {
1802         /* Disable ADC end of conversion interrupt on group regular */
1803         __HAL_ADC_DISABLE_IT(hadc, ADC_IT_EOC);
1804
1805         /* Set ADC state */
1806         CLEAR_BIT(hadc->State, HAL_ADC_STATE_REG_BUSY);
1807
1808         if (HAL_IS_BIT_CLR(hadc->State, HAL_ADC_STATE_INJ_BUSY))
1809         {
1810             SET_BIT(hadc->State, HAL_ADC_STATE_READY);
1811         }
1812     }
1813
1814     /* Conversion complete callback */
1815 #if (USE_HAL_ADC_REGISTER_CALLBACKS == 1)
1816     hadc->ConvCpltCallback(hadc);
1817 #else
1818     HAL_ADC_ConvCpltCallback(hadc);
1819 #endif /* USE_HAL_ADC_REGISTER_CALLBACKS */
1820
1821     /* Clear regular group conversion flag */
1822     __HAL_ADC_CLEAR_FLAG(hadc, ADC_FLAG_STRT | ADC_FLAG_EOC);
1823

```

Bài 8: Lập trình STM32 đọc ADC một kênh 60

Copy và paste vào khu tiền xử lý trước main()

```

1895
1896 /**
1897 * @brief Conversion complete callback in non blocking mode
1898 * @param hadc: ADC handle
1899 * @retval None
1900 */
1901 weak void HAL_ADC_ConvCpltCallback(ADC_HandleTypeDef* hadc)
1902 {
1903     /* Prevent unused argument(s) compilation warning */
1904     UNUSED(hadc);
1905     /* NOTE : This function should not be modified. When the callback is needed,
1906            function HAL_ADC_ConvCpltCallback must be implemented in the user file.
1907 */
1908 }
1909
1910 /**
1911 * @brief Conversion DMA half-transfer complete callback
1912 * @param hadc: ADC handle
1913 * @retval None
1914 */
1915 weak void HAL_ADC_ConvHalfCpltCallback(ADC_HandleTypeDef* hadc)
1916 {
1917     /* Prevent unused argument(s) compilation warning */
1918     UNUSED(hadc);

```

Bài 8: Lập trình STM32 đọc ADC một kênh 61

Sau đó chúng ta khởi tạo một biến u16_ADCVal lưu giá trị ADC. Trong hàm xử lý ngắt ta điều hướng ngắt, nếu ngắt là ADC1, đọc giá trị ADC vào biến vừa khởi tạo.

Khởi tạo một biến Voltage lưu trữ giá trị điện áp tại chân CH0 để so sánh với thực tế. $\text{Voltage} = \text{ADC_Value}/4095*3.3$

Trong đó 4095 là giá trị cực đại của ADC.

3.3 là điện áp tham chiếu Vref

```

58  /* Private user code */
59  /* USER CODE BEGIN 0 */
60  uint16_t u16_ADCVal;
61  float Voltage;
62  void HAL_ADC_ConvCpltCallback(ADC_HandleTypeDef* hadc)
63 {
64     /* Prevent unused argument(s) compilation warning */
65     UNUSED(hadc);
66     if(hadc->Instance == ADC1) // neu ngat ADC1
67     {
68         u16_ADCVal = HAL_ADC_GetValue(&hadc);
69         Voltage = (float)u16_ADCVal/4095*3.3;
70     }
71     /* NOTE : This function should not be modified. When the callback is needed,
72            function HAL_ADC_ConvCpltCallback must be implemented in the user file.
73 */
74 }
75 /* USER CODE END 0 */
76 ...

```

Bài 8: Lập trình STM32 đọc ADC một kênh 62

Sau đó khởi động ADC1 bằng lệnh HAL_ADC_Start_IT(&hadc1); trong hàm while(1), để lây khoảng 50 ms mỗi lần đọc.

Vì chế độ single là chế độ đọc một lần vì vậy ta phải liên tục gọi hàm Start, chế độ này sẽ tiết kiệm năng lượng hơn chế độ Cont và phụ thuộc vào lập trình viên khi nào cần dùng sẽ bật.

```

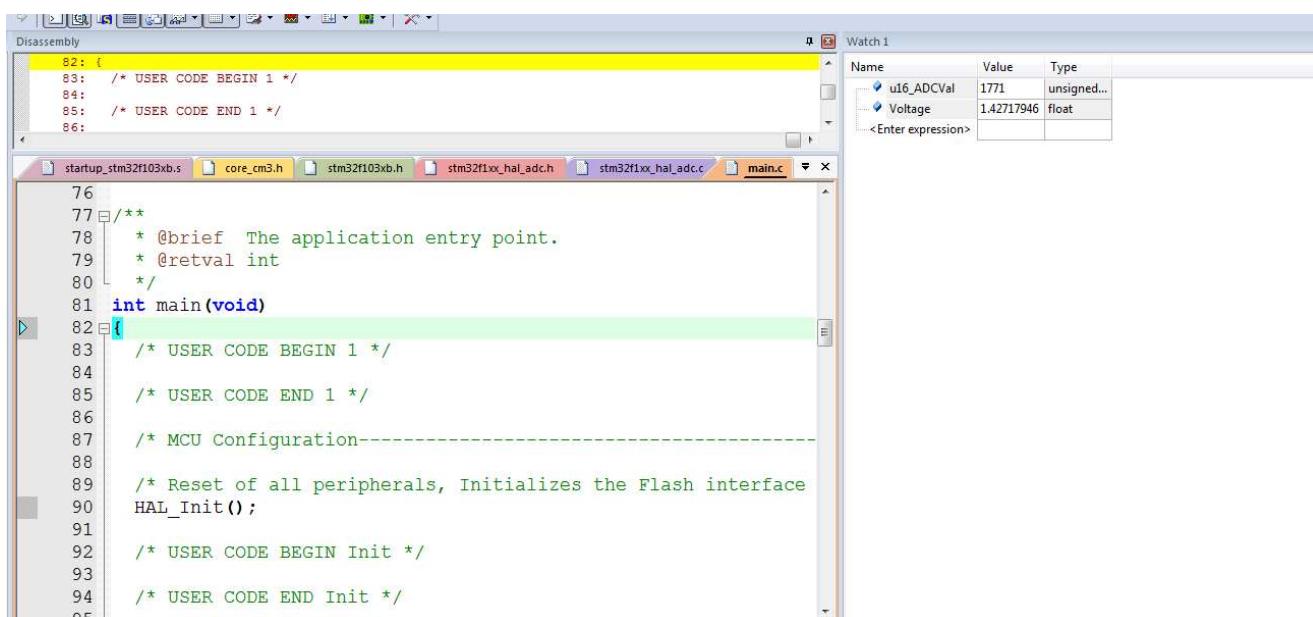
107     /* USER CODE END 2 */
108
109     /* Infinite loop */
110     /* USER CODE BEGIN WHILE */
111     while (1)
112     {
113         /* USER CODE END WHILE */
114
115         /* USER CODE BEGIN 3 */
116         HAL_ADC_Start_IT(&hadc1);
117         HAL_Delay(50);
118     }
119     /* USER CODE END 3 */
120 }
121
122
123 /**
124 +-----+
125

```

Bài 8: Lập trình STM32 đọc ADC một kênh 63

F7 để Build sau đó nạp chương trình, nhấn vào nút debug (Ctr + F5), Add 2 biến u16_ADCVal và Voltage vào.

Nhấn F5 để chạy debug, ta thấy sai số trong khoảng +-0.04V.



Bài 8: Lập trình STM32 đọc ADC một kênh 64



Bài 8: Lập trình STM32 đọc ADC một kênh 65

Với chế độ Continuous các bạn tìm đến hàm khởi tạo ADC : MX_ADC1_Init()

Chỉnh dòng hadc1.Init.ContinuousConvMode = DISABLE; Thành ENABLE

```

177
178     /* USER CODE BEGIN ADC1_Init_1 */
179
180     /* USER CODE END ADC1_Init_1 */
181     /** Common config
182     */
183     hadc1.Instance = ADC1;
184     hadc1.Init.ScanConvMode = ADC_SCAN_DISABLE;
185     hadc1.Init.ContinuousConvMode = ENABLE; // Đổi sang ENABLE
186     hadc1.Init.DiscontinuousConvMode = DISABLE;
187     hadc1.Init.ExternalTrigConv = ADC_SOFTWARE_START;
188     hadc1.Init.DataAlign = ADC_DATAALIGN_RIGHT;
189     hadc1.Init.NbrOfConversion = 1;
190     if (HAL_ADC_Init(&hadc1) != HAL_OK)
191     {
192         Error_Handler();
193     }
194     /** Configure Regular Channel
195     */
196     sConfig.Channel = ADC_CHANNEL_0;
197     sConfig.Rank = ADC_REGULAR_RANK_1;

```

Bài 8: Lập trình STM32 đọc ADC một kênh 66

Chuyển hàm HAL_ADC_Start_IT(&hadc1); lên trên While(1)

```

100
101     /* USER CODE END SysInit */
102
103     /* Initialize all configured peripherals */
104     MX_GPIO_Init();
105     MX_ADC1_Init();
106     /* USER CODE BEGIN 2 */
107     HAL_ADC_Start_IT(&hadc1);
108     /* USER CODE END 2 */
109
110     /* Infinite loop */
111     /* USER CODE BEGIN WHILE */
112     while (1)
113     {
114         /* USER CODE END WHILE */
115
116         /* USER CODE BEGIN 3 */
117     }

```

Bài 8: Lập trình STM32 đọc ADC một kênh 67

Nhấn F7 Build và nạp vào chip. Vào debug và xem kết quả.

Kết

Lập trình STM32 đọc ADC một kênh là một trong những bài bắt buộc phải nắm rõ trong lập trình STM32. Bởi vì các thiết bị Analog đều cần phải sử dụng ADC để vi điều khiển có thể hiểu được,

Hãy cố gắng làm chủ ADC một kênh nhé

4.5/5 - (4 bình chọn)

Related Posts:

1. [Lập trình STM32 từ A tới Z](#)
2. [Bài 12: Lập trình STM32 với giao thức SPI](#)
3. [Bài 11: Lập trình STM32 với Giao thức UART](#)
4. [Bài 10: Giao thức I2C, lập trình STM32 với module RTC DS3231](#)
5. [Bài 3: Lập trình STM32 GPIO điều khiển Led và nút nhấn](#)
6. [Bài 2: Tổng quan về KIT STM32F103C8T6 Blue Pill](#)



KHUÊ NGUYỄN

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

13 THOUGHTS ON “BÀI 8: LẬP TRÌNH STM32 ĐỌC ADC MỘT KÊNH”

Nhân Trần says:



Ad hướng dẫn dùm mình 1 tí, mình cần chuyển đổi 1 số thực (float) sang số uint8_t thì dùng hàm gì nhỉ? mình mới nhập môn nên còn gà lăm!

29/08/2021 AT 10:39 CHIỀU

TRẢ LỜI



Nhân Trần says:

Cụ thể là mình đang làm bài tập đọc ADC, đọc được giá trị điện áp Voltage (dạng số thực), muốn gửi lên Hercules Teminal để xem giá trị (qua Uart)
HAL_UART_Transmit(&huart1,Voltage,sizeof(u8_TxBuff),100);
Mong bạn hướng dẫn giúp ! Cảm ơn .

29/08/2021 AT 11:03 CHIỀU

TRẢ LỜI



Khuê Nguyễn says:

bạn dùng sprintf để chuyển số thực thành chuỗi nhé. Dùng %f để thêm số thực vào chuỗi

04/09/2021 AT 4:09 CHIỀU

TRẢ LỜI



Van Thanh says:

Giả sử mình có một nút nhấn, mà mình muốn lúc nút nhấn không bật thì mới dùng adc thì làm sao ạ ? Mong anh giải đáp, em mới học lập trình nên hơi non.

21/11/2021 AT 11:08 CHIỀU

TRẢ LỜI



Khuê Nguyễn says:

Khi bật nút nhấn thì e chạy hàm đọc ADC là dc

23/11/2021 AT 10:48 SÁNG

TRẢ LỜI



Tú Huỳnh says:

em phải xem video hay đọc bên web khác trước rồi vài cái đọc lại bài anh viết
em mới biết là anh đang viết cái gì ^^.^

04/12/2021 AT 12:40 CHIỀU

TRẢ LỜI



Khuê Nguyễn says:

E ko hiểu chỗ nào thế :v

04/12/2021 AT 3:05 CHIỀU

TRẢ LỜI



Tú Huỳnh says:

bên phần lý thuyết ấy ạ. Hình ảnh như lấy từ datasheet ra. Em người mới vào đọc không hình dung được gì cả.

04/12/2021 AT 5:33 CHIỀU

TRẢ LỜI



Tú Huỳnh says:

cơ mà sau khi đã có hình dung sơ sơ về kiến thức này từ youtube hay web khác rồi, vào đọc bài của đọc thấy anh viết chi tiết và sâu hơn.

04/12/2021 AT 7:45 CHIỀU

TRẢ LỜI



Tú Huỳnh says:

Nhưng mà có hình dung sơ sơ từ video hay web khác rồi vào bài anh đọc em thấy anh viết chi tiết với sâu hơn người khác.

04/12/2021 AT 7:42 CHIỀU

TRẢ LỜI



An says:

Thử kiểu này xem thử, trả về chuỗi str
sprintf(str, "%0.2f", Voltage); // lạm tròn số

04/04/2022 AT 11:33 SÁNG

TRẢ LỜI



Minh Chí says:

bạn bỏ cái số đăng sau mỗi đoạn đi, đọc mãi k hiểu kênh 17 là gì, kênh 18 là gì???? luận mãi mới ra nó chả có ý nghĩa gì ????????

02/07/2022 AT 8:44 CHIỀU

TRẢ LỜI

Khuê Nguyễn says:

ok

28/07/2022 AT 11:37 CHIỀU

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



Khuê Nguyễn Creator



PROTEUS

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



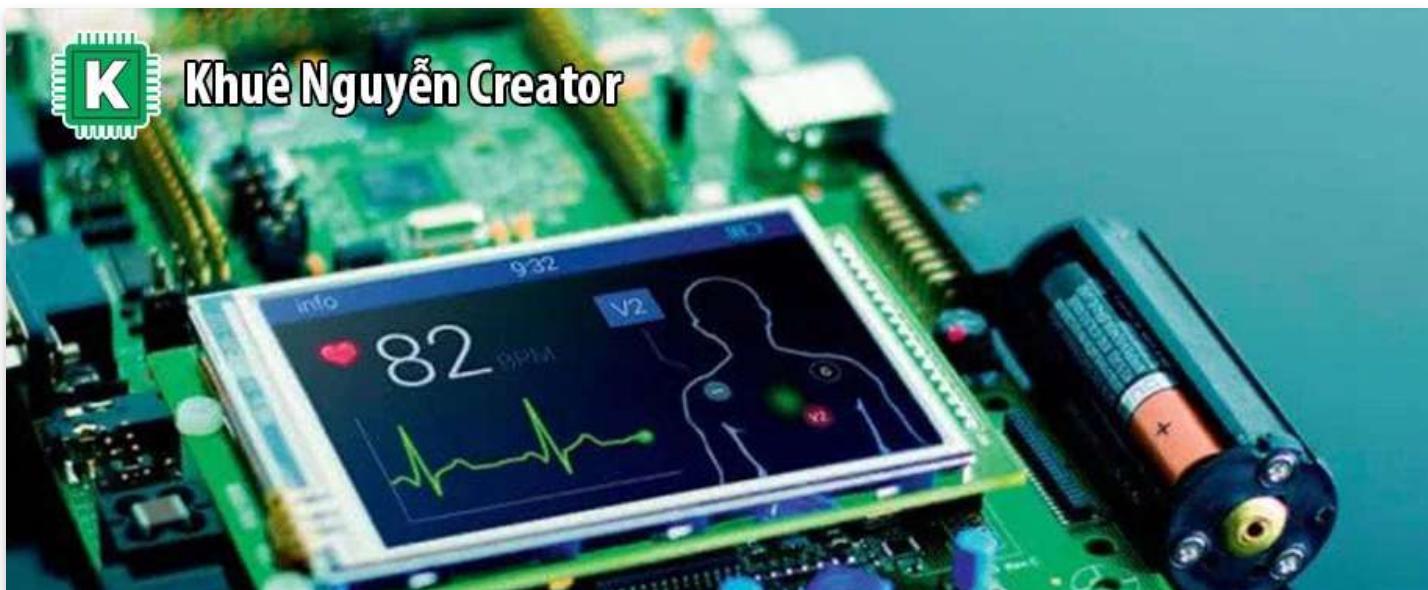
Khuê Nguyễn Creator

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

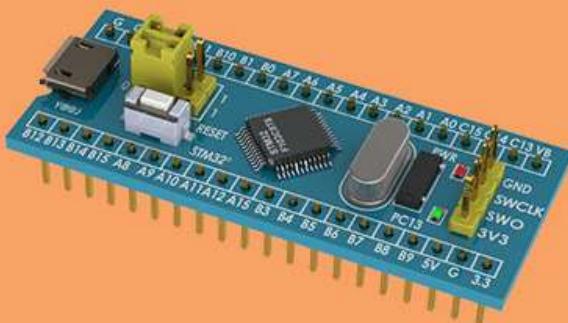
Trong bài này chúng ta sẽ cùng nhau tìm hiểu STM32 DAC với KIT STM32F407VE...

[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 Compiler 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



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.

Liên Kết

Nhóm: Nghịên Lập Trình

Fanpage: Khuê Nguyên Creator

My Shop

Thông Tin

Tác Giả

Chính Sách Bảo Mật



Copyright 2022 © Khuê Nguyễn