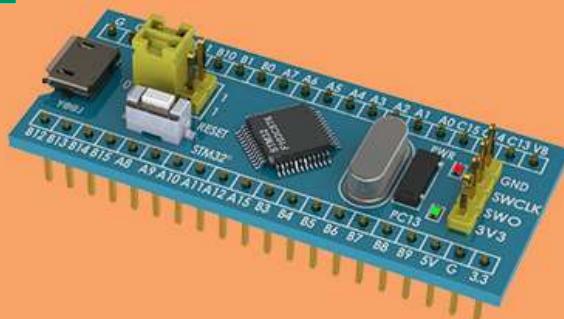
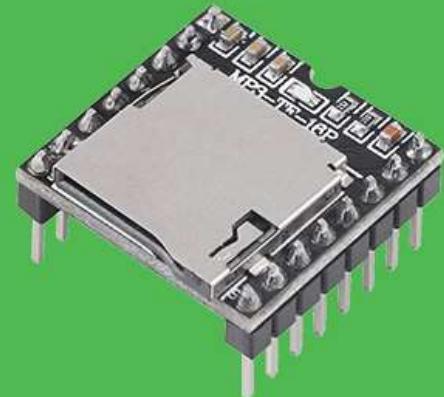
**LẬP TRÌNH STM32**

Lập trình STM32 DFPlayer phát nhạc từ thẻ nhớ

POSTED ON 14/08/2021 BY KHUÊ NGUYỄN

14
Th8**Khuê Nguyễn Creator**

Lập trình STM32 phát nhạc với DFPlayer Mini

Trong bài này chúng ta sẽ học cách **lập trình STM32 DFPlayer**, phát nhạc, âm thanh,... cực kì đơn giản sử dụng UART. DFPlayer là một module giải mã âm thanh rẻ, dễ điều khiển, có khuếch đại âm thanh nội nên rất được ưa chuộng

Bài này nằm trong Serie **Học STM32 từ A tới Z**

Mục Lục



1. Giới thiệu module DFPlayer
 - 1.1. Thông số kỹ thuật
 - 1.2. Sơ đồ chân
2. Cách điều khiển DFPlayer
 - 2.1. Định dạng khung truyền
 - 2.2. Các lệnh trong DFPlayer
3. Lập trình STM32 DFPlayer phát nhạc như 1 chiếc MP3
 - 3.1. Cấu hình CubeMX
 - 3.2. Lập trình với Keil C
 - 3.3. Kết quả phát nhạc bằng DFPlayer với STM32
4. Kết
 - 4.1. Related posts:

Giới thiệu module DFPlayer

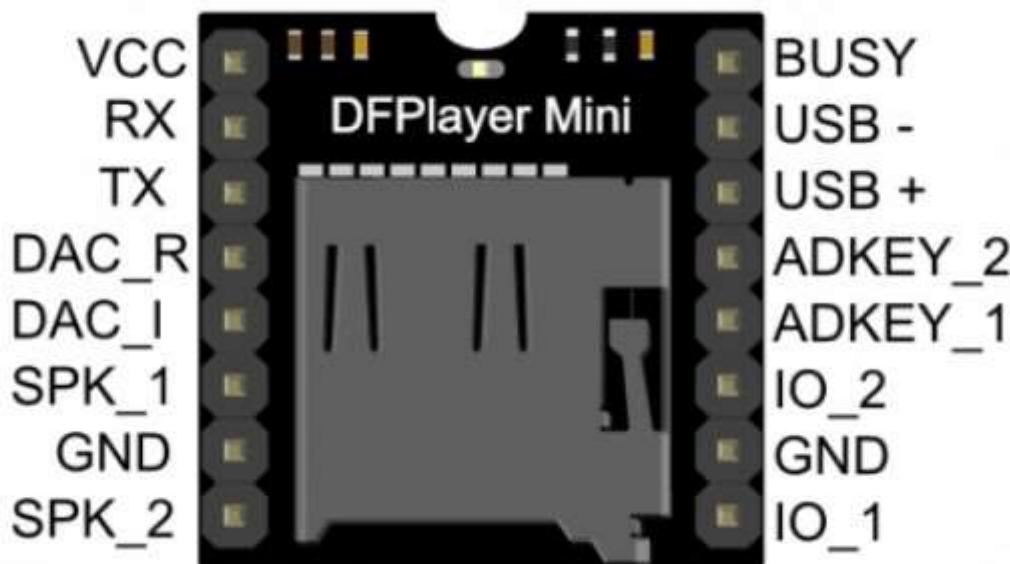
Mạch Phát Âm Thanh MP3 là **mạch phát âm thanh MP3** từ thẻ microSD cực kì nhỏ gọn mà còn tích hợp sẵn ampli để có thể kết nối với loa. Ngoài ra, ta còn có thể điều khiển chơi MP3 bằng Arduino. Module hỗ trợ giải mã MP3, WAV, WMA.

DFPlayer Mini MP3 là mạch phát tập tin âm thanh kiểu máy chơi nhạc MP3. Mạch có thể được sử dụng riêng lẻ chỉ cần pin, loa và nút nhấp hoặc kết hợp với **vi điều khiển** có giao tiếp chuẩn USART.

Thông số kỹ thuật

- Tốc độ lấy mẫu (Khz) ; 8 / 11.025 / 12 / 16 / 22.05 / 24 / 32 / 44.1 / 48 với ngõ ra 24bit
- Hỗ trợ đầy đủ FAT16, FAT32, thẻ TF hỗ trợ tối đa 32Gb.
- Có thể điều khiển qua các chân IO hay chuẩn nối tiếp.
- Các file âm thanh có thể sắp xếp theo thư mục (tối đa 100 mục) , mỗi mục chứa tối đa 255 bài hát.
- Có nhiều lệnh điều khiển, có thể sử dụng như một máy nghe nhạc MP3

Sơ đồ chân



Sơ đồ chân

No	Pin	Description	Note
1	VCC	Input Voltage	DC3.2~5.0V; Type: DC4.2V
2	RX	UART serial input	
3	TX	UART serial output	
4	DAC_R	Audio output right channel	Drive earphone and amplifier
5	DAC_L	Audio output left channel	Drive earphone and amplifier
6	SPK2	Speaker-	Drive speaker less than 3W
7	GND	Ground	Power GND
8	SPK1	Speaker+	Drive speaker less than 3W
9	IO1	Trigger port 1	Short press to play previous (long press to decrease volume)
10	GND	Ground	Power GND
11	IO2	Trigger port 2	Short press to play next (long press to increase volume)
12	ADKEY1	AD Port 1	Trigger play first segment
13	ADKEY2	AD Port 2	Trigger play fifth segment
14	USB+	USB+ DP	USB Port
15	USB-	USB- DM	USB Port
16	BUSY	Playing Status	Low means playing \High means no

Mô tả các chân

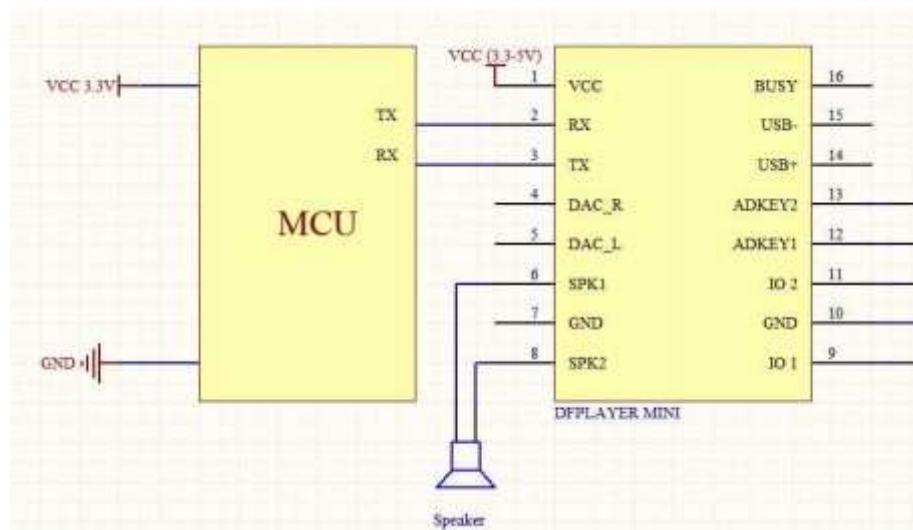
Trong bài này chúng ta sẽ sử dụng giao tiếp UART để điều khiển DFPlayer. vậy nên chỉ sử dụng 4 chân VCC, GND, TX, RX.

Nối loa vào 2 chân SPK_1 và SPK_2 là chạy được luôn vì DFPlayer có sẵn mạch khuếch đại âm thanh.

Nếu muốn sử dụng amply ngoài ta nối vào các chân DAC_L, DAC_R, GND tương ứng với 3 đường vào của audio.

Sơ đồ mạch STM32 DFPlayer

Như sau:



Ngoài ra các bạn có thể sử dụng các nút nhấn được mô tả trong datasheet

Cách điều khiển DFPlayer

Download: [datasheet DFPlayer mini](#)

Định dạng khung truyền

Đầu tiên chúng ta cần Setting UART như sau:

- Baudrate: 9600
- Checksum parity none: không kiểm tra chẵn/lẻ

- Data stop bits: 1

Để giao tiếp với DFPlayer, mỗi lần truyền lệnh chúng ta phải tuân theo định dạng sau:

1. Byte Start luôn luôn là 0x7E
2. Byte VER: Mặc định là 0xFF
3. Byte Length: Số lượng các byte sau byte này. Mặc định là 0x06
4. Byte CMD: Ghi mã lệnh sẽ được định nghĩa ngay sau đây
5. Byte Feedback: phản hồi, để 1 nếu cần phản hồi, 0 nếu không cần. Mặc định là 0
6. Byte para1 và para2 là 2 byte tham số truyền vào
7. 2 Byte Checksum: mã checksum để kiểm tra việc truyền có bị lỗi hay không
8. Byte End: Kết thúc khung truyền luôn luôn là 0xEF

Support for asynchronous serial communication mode via PC serial sending commands
Communication Standard:9600 bps

Data bits :1

Checkout :none

Flow Control :none

Format:	SS	VER	Len	CMD	Feedback	para1	para2	checksum	\$O
\$S								Each command feedback begin with \$, that is 0x7E	
VER								Version Information	
Len								Checksums are not counted	
CMD								Indicate the specific operations, such as play / pause, etc.	
Feedback								If need for feedback, 1: feedback, 0: no feedback	
para1								Query high data byte	
para2								Query low data byte	
checksum								Accumulation and verification [not include start bit \$]	
\$O								End bit 0xEF	

Trong thư viện mình sẽ tạo 1 mảng khung truyền có 10byte, vì checksum có 2 byte nhé.

```

14 L ****
15 #include "DFPLAYER.h"
16 uint8_t SendFrame[10] = {0x7E, 0xFF, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0xEF};
17
18 //***** LOW Level Functions *****/
19 static void DFPLAYER_SendUART(DFPLAYER_Name* MP3)
20 {
21     HAL_UART_Transmit(MP3->DFP_UART, (uint8_t *)MP3->SendBuff, sizeof(MP3->SendBuff), 1000);
22 }
23 static void DFPLAYER_Delay(uint16_t Time)
24 {
25     HAL_Delay(Time);
26 }
27

```

Để gửi một lệnh mình chỉ cần fill vào các vị trí tương ứng đó là CMD, para1, para 2 và checksum. Còn tất cả các vị trí còn lại để mặc định. Sau đó truyền qua UART frame này là DFPlayer sẽ nhận được.

```

48 static void DFPLAYER_SendCmd(DFPLAYER_Name* MP3, uint8_t cmd, uint16_t high_arg, uint16_t low_arg)
49 {
50     DFPLAYER_Delay(100);
51     MP3->SendBuff[3] = cmd;
52     MP3->SendBuff[5] = high_arg;
53     MP3->SendBuff[6] = low_arg;
54     DFPLAYER_FillChecksum(MP3);
55     DFPLAYER_SendUART(MP3);
56 }
57

```

Các lệnh trong DFPlayer

Các lệnh tương ứng là giá trị điền vào Byte CMD của frame truyền. Mình chỉ sử dụng các lệnh điều khiển (control) mà không sử dụng lệnh truy vấn (query).

CMD	Function Description	Parameters(16 bit)
0x01	Next	
0x02	Previous	
0x03	Specify tracking(NUM)	0-2999
0x04	Increase volume	
0x05	Decrease volume	
0x06	Specify volume	0-30
0x07	Specify EQ(0/1/2/3/4/5)	Normal/Pop/Rock/Jazz/Classic/Base
0x08	Specify playback mode (0/1/2/3)	Repeat/folder repeat/single repeat/ random
0x09	Specify playback source(0/1/2/3/4)	U/TF/AUX/SLEEP/FLASH
0x0A	Enter into standby – low power loss	
0x0B	Normal working	
0x0C	Reset module	
0x0D	Playback	
0x0E	Pause	
0x0F	Specify folder to playback	1~10(need to set by user)
0x10	Volume adjust set	{DH= 1:Open volume adjust } {DL: set volume gain 0~31}
0x11	Repeat play	{ 1:start repeat play} { 0:stop play}

Lập trình STM32 DFPLayer phát nhạc như 1 chiếc MP3

Cấu hình CubeMX

Cấu hình như bình thường các tab debug, clock,...

Chúng ta sẽ dùng UART1 để giao tiếp với DFPlayer, tham số cấu hình như sau:

Lập trình với Keil C

Download thư viện theo hướng dẫn: [Download tài liệu lập trình STM32](#)

Thay đổi đường dẫn thư viện và add file DFPlayer vào Project. Phần này mình đã hướng dẫn ở các bài trước

Khởi tạo biến MP3 lưu trữ setting của DFPlayer.

```

21  /* Includes --
22  #include "main.h"
23
24  /* Private includes --
25  /* USER CODE BEGIN Includes */
26  #include "DFPlayer.h"
27  /* USER CODE END Includes */
28
29  /* Private typedef --
30  /* USER CODE BEGIN PTD */
31  DFPLAYER_Name MP3;
32  /* USER CODE END PTD */
33

```

Trong main ta khởi tạo DFPlayer với huart1.

Chỉnh âm lượng (0 – 32) sao cho vừa nghe.

Sau đó bắt đầu chạy.

Trong while(1) chúng ta sử dụng câu lệnh next và prev để chuyển bài.

```

90     MX_GPIO_Init();
91     MX_USART1_UART_Init();
92     /* USER CODE BEGIN 2 */
93     DFPLAYER_Init(&MP3, &huart1);
94     DFPLAYER_SetVolume(&MP3, 20);
95     DFPLAYER_Play(&MP3);
96     /* USER CODE END 2 */
97 //
98     /* Infinite loop */
99     /* USER CODE BEGIN WHILE */
100    while (1)
101    {
102        /* USER CODE END WHILE */
103
104        /* USER CODE BEGIN 3 */
105        HAL_Delay(20000);
106        DFPLAYER_Next(&MP3);
107        HAL_Delay(20000);
108        DFPLAYER_Prev(&MP3);
109        HAL_Delay(20000);
110    }
111    /* USER CODE END 3 */
112 }
113

```

Trong thẻ nhớ chúng ta tạo một thư mục tên **mp3** sau đó copy các file .mp3 vào trong nhé.

Để phát nhạc theo từng thư mục, chúng ta tạo ra các thư mục có format 01, 02, 03... và bên trong file cũng phải đổi tên là 001, 002, 003

Khi đó dùng lệnh `DFPLAYER_PlayFileInFolder(DFPLAYER_Name* MP3, uint8_t folder, uint32_t num)` để chạy file đó.

Kết quả phát nhạc bằng DFPlayer với STM32



Facebook Watch

Kết

DFPlayer có thể sử dụng để tạo một chiếc máy nghe nhạc đơn giản, hoặc sử dụng với vi điều khiển chúng ta dễ dàng phát âm thanh sử dụng định dạng mp3. Giúp cho các ứng dụng phong phú hơn.

Bạn có thể dùng nó để phát thông tin chuyến bay hoặc số thứ tự như ở bệnh viện chẳng hạn. Như sản phẩm Led pháo hoa của mình cũng sử dụng DFPlayer để tạo âm thanh tiếng pháo nổ.

Nếu cảm thấy bài viết này có ích hãy chia sẻ với bạn bè nhé. Đừng quên ra nhập hội những anh em **Nghịen lập trình** nhé!!

5/5 - (2 bình chọn)

Related Posts:

1. **[Bài 21: Lập trình STM32 Bit Band điều khiển GPIO](#)**
2. **[Lập trình STM32 đọc nhiệt độ với DS18b20 giao tiếp I2C](#)**
3. **[Hướng dẫn download và sử dụng tài liệu Lập trình STM32](#)**

4. **Bài 14: Sử dụng STM32 IWDG Independent Watchdog Timer chống treo vi điều khiển**
5. **Bài 10: Giao thức I2C, lập trình STM32 với module RTC DS3231**
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

3 THOUGHTS ON “LẬP TRÌNH STM32 DFPLAYER PHÁT NHẠC TỪ THẺ NHỚ”



Tài says:

Setting Uart kiểu gì ạ, có bài nào hướng dẫn setting uart không ạ

29/07/2022 AT 5:15 CHIỀU

TRẢ LỜI



Khuê Nguyễn says:

Bài UART trong serie học stm32 từ A tới Z nhé

01/08/2022 AT 9:58 CHIỀU

TRẢ LỜI



Đạt says:

Build started: Project: kocony

*** Using Compiler ‘V5.06 update 6 (build 750)’, folder:
‘C:\Keil_v5\ARM\ARMCC\Bin’

Build target 'kocony'

linking...

kocony\kocony.axf: Error: L6218E: Undefined symbol DFPLAYER_Init
(referred from main.o).

kocony\kocony.axf: Error: L6218E: Undefined symbol DFPLAYER_Next
(referred from main.o).

kocony\kocony.axf: Error: L6218E: Undefined symbol DFPLAYER_Play
(referred from main.o).

kocony\kocony.axf: Error: L6218E: Undefined symbol DFPLAYER_Prev
(referred from main.o).

kocony\kocony.axf: Error: L6218E: Undefined symbol DFPLAYER_SetVolume
(referred from main.o).

em chạy 1 lần xong lần 2 thì nó bị lỗi này ạ. Anh giúp em fix với 😞

09/08/2022 AT 12:47 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 đẽ:

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

Bài viết khác

Lập trình 8051 - AT89S52

Khuê Nguyễn Creator



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



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

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

3 COMMENTS

[ĐỌC THÊM](#)

Lập trình STM32 và CubeMX

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

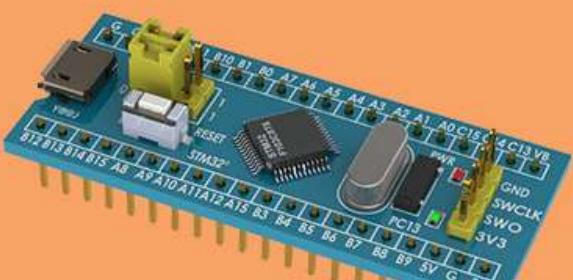
Trong bài này chúng ta sẽ cùng nhau tìm hiểu STM32 DAC với KIT STM32F407VE...

[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...

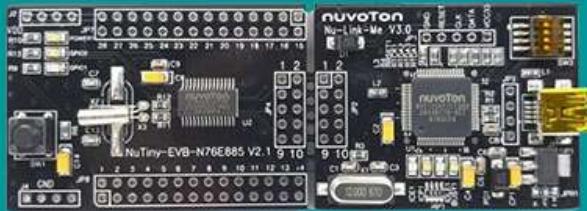
4 COMMENTS

[ĐỌC THÊM](#)

Lập trình Nuvoton



Khuê Nguyễn Creator

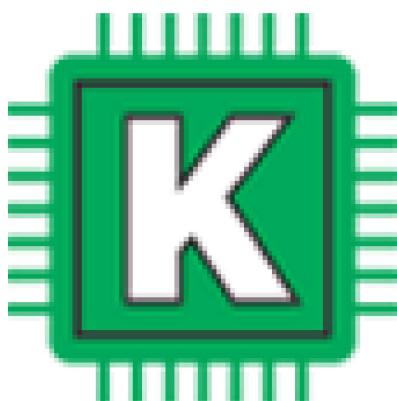


Cài đặt SDC Complier và Code:Blocks IDE

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: Nghịen Lập Trình

Fanpage: Khuê Nguyên Creator

My Shop

Thông Tin

Tác Giả

Chính Sách Bảo Mật



Copyright 2022 © Khuê Nguyễn