

میکروکنترلر های ARM

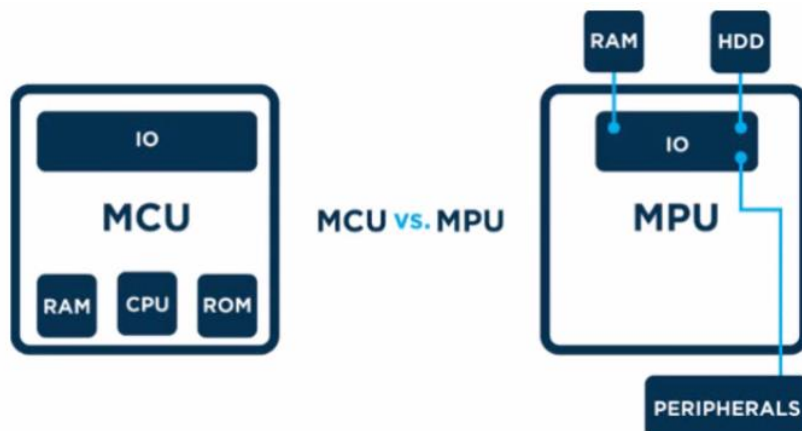
سری STM32F103



فرحان دائمی مژدهی

تعاریف اولیه:

۱. **میکروکنترلر**: یک مدار مجتمع یا چیپ الکترونیکی (IC) است که دارای RAM, ROM, CPU, GPIO, ... و قابل برنامه ریزی است. درواقع میکروکنترلر یک کامپیوتر کوچک است که برای مصارف خاصی برنامه ریزی می‌شوند.



۲. **ARM**: یک معماری است (نه میکروکنترلر) که برای ساخت پردازنده های ۳۲ بیتی و ۶۴ بیتی استفاده می‌شود و توسط کمپانی ARM Holding توسعه داده شده. نکته قابل ذکر این است که این شرکت تولید کننده هیچ گونه میکروکنترلی نیست و تنها معماری خود را به شرکت های تولید کننده میکروکنترلر نظیر ST, NXP, ... می‌فروشد.

۳. به میکروکنترلی که در آن از معماری ARM استفاده شده، میکروکنترلر ARM می‌گوییم.

۴. اکثر سیستم های نهفته مانند میکروکنترلر ها، موبایل، تبلت و به طور کل سیستم هایی با حجم کم و امکانات بالا از این پردازنده استفاده می‌کنند. زیرا این میکروکنترلر ها:

- قیمت بسیار ارزان و مناسبی دارند.
- سرعت بسیار بالایی دارند.
- توان مصرفی بسیار پایینی دارند.

۵. **Cortex**: مدل های مختلف معماری های ARM را با نام Cortex می‌شناسند. به طور کلی Cortex دارای ۳ پروفایل است:

- **A**: کاربرد های application مانند گوشی های موبایل
- **R**: کاربرد های Real-time
- **M**: کاربرد هایی با توان مصرفی پایین

۴. **Pipe line**: یک تکنیک طبیعی در زندگی است. به عنوان مثال در یک خط تولید که مرحله‌ای به صورت زیر دارد، استفاده از تکنیک pipe line سرعت تولید را افزایش می‌دهد.



$$3 \times (1 + 2 + 3) = 18$$

• مدت زمان تولید سه کالا بدون pipe line:

$$6 + 3 + 3 = 12$$

• مدت زمان تولید سه کالا با استفاده از pipe line:

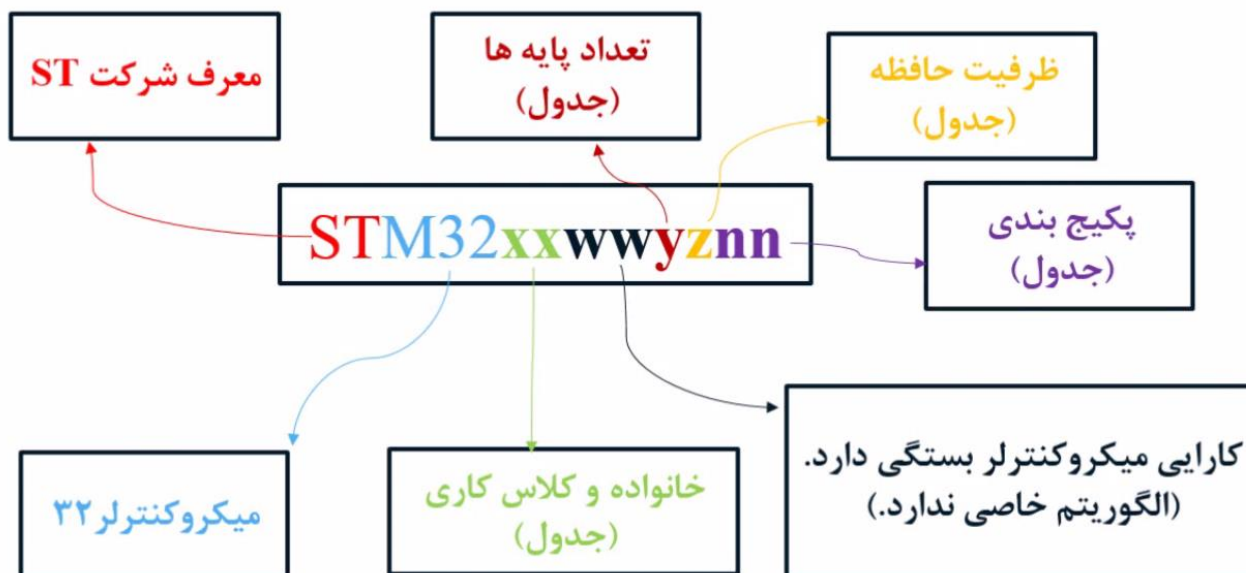
میکروکنترلری که قرار است از آن استفاده کنیم از معماری **ARM Cortex-M3** استفاده می‌کند. که در اینجا عدد ۳ نشان دهنده تعداد pipe line در ساختار این میکروکنترلر است.

آشنایی با شرکت ST:

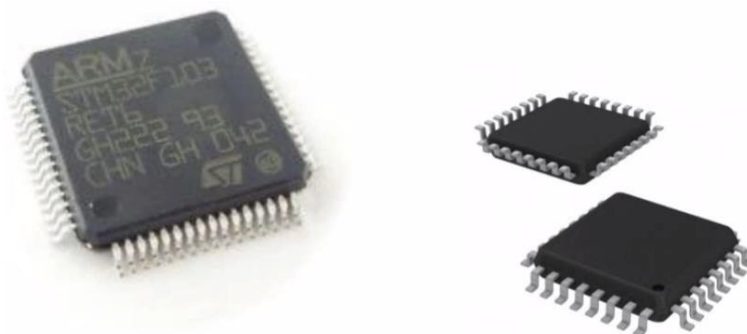
شرکت ST-Microelectronics که یک شرکت فرانسوی-ایتالیایی و چندملیتی تولیدکننده تجهیزات الکترونیکی و نیمه هادی‌ها می‌باشد، که دفتر مرکزی آن در شهر ژنو، سوئیس قرار دارد. این شرکت یکی از رقبای موفق در تولید میکروکنترلر در بین دیگر تولید کنندگان بوده است. محصولات این شرکت به دلایل زیادی از جمله تنوع مدل‌ها و سرعت بالا، توان مصرفی پایین، کتابخانه‌های ارائه شده توسط سازنده و ... در جهان مورد توجه قرار گرفته است.

میکروکنترلر های STM32:

این میکروکنترلر های ۳۲ بیتی که از میکروکنترلر های ARM هستند به چندین روش قابل برنامه ریزی و استفاده هستند. یکی از این روش‌ها استفاده از نرم افزار Keil MDK μ vision است که بسیار قدرتمند بوده و قابلیت های بسیار زیادی دارد. روش دیگر استفاده از نرم افزار های Arduino IDE است که بسیار ساده و قابل فهم می‌باشد. نام گذاری میکروکنترلر های STM32 به صورت زیر است:



مثال: می‌خواهیم مشخصات میکروکنترلر STM32F103RET6 را از روی نام آن استخراج کنیم:



Code	Core	Max freq [MHz]	Max FLASH [KB]	Max SRAM [KB]	target
L0	CortexM0+	32	192	20	Ultra low-power
F0	CortexM0	48	256	32	Mainstream
F3	CortexM4	72	512	80	Mainstream
L1	CortexM3	32	512	80	Ultra low-power
F1	CortexM3	72	1024	96	Mainstream
F2	CortexM3	120	1024	128	High performance
L4	CortexM4	80	1024	320	Ultra low-power
F4	CortexM4	180	2048	384	High performance
F7	CortexM7	216	2048	512	High performance
H7	CortexM7	400	2048	1024	High performance

خانواده و کلاس کاری:

Package pin count [y]				FLASH memory size [z]	
Code	Number of pins			Code	FLASH size [KB]
A	169			4	16
B	208			6	32
C	48			8	64
F	20			B	128
G	28			Z	192
H	40			C	256
I	176			D	384
J	72			E	512
K	32			F	768
M	81			G	1024
N	216			H	1536
Q	132			I	2048
R	64				
T	36				
U	63				
V	100				
Z	144				

تعداد پایه (جدول)

حافظه فلاش

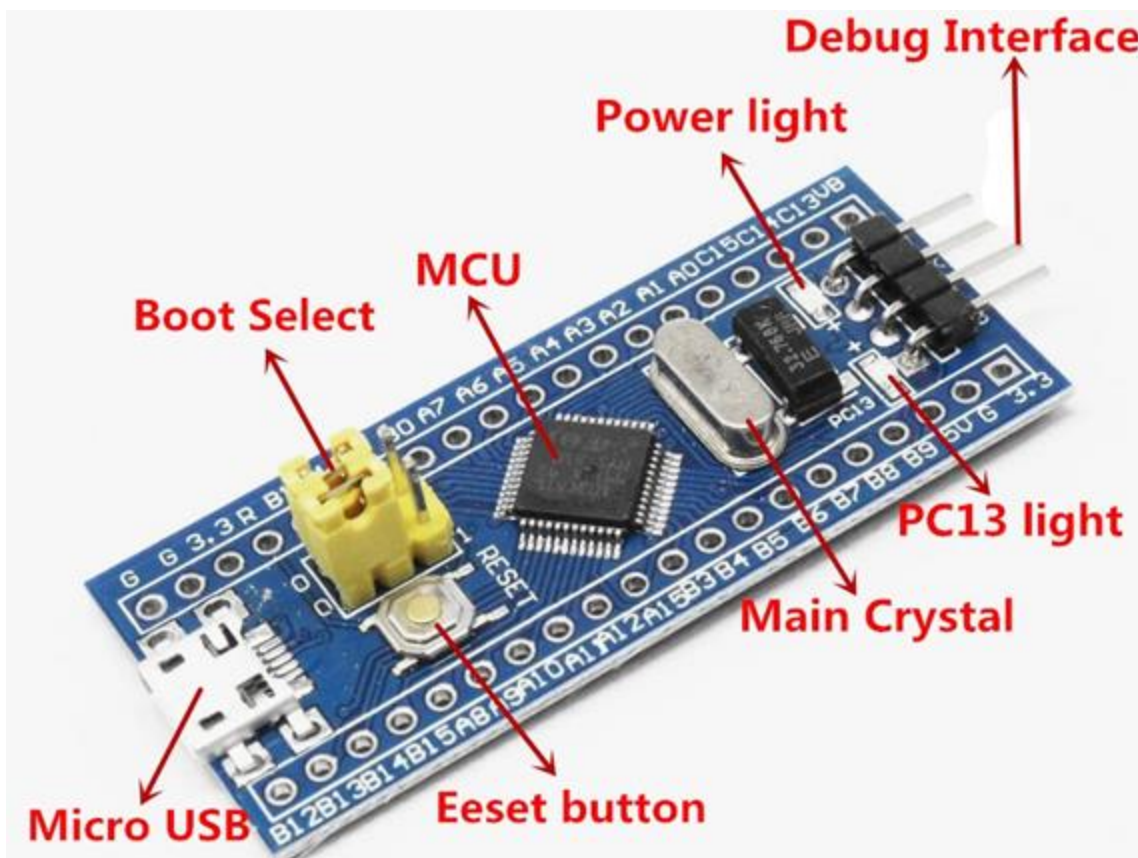
STM32F103RET6

تایمینگ

تایمینگ

Package name	
Code	Package
T	LQFP
P	TSSOP
H	TFBGA , LFBGA , UFBGA
E	EWLCSP
Y	WLCSP

بورد STM32F103C8T6:





مشخصات فنی برد دیسکاوری STM32F103:

STM32F103C8T6 Specifications

The **ARM Cortex M3 STM32F103C8 Microcontroller** is used in the Blue pill board. Unlike the name, "Blue Pill" the Microcontrollers name STM32F103C8T6 has a meaning behind it.

- STM » stands for the manufacturers name STMicroelectronics
- 32 » stands for 32-bit ARM architecture
- F103 » stands to indicate that the architecture ARM Cortex M3
- C » 48-pin
- 8 » 64KB Flash memory
- T » package type is LQFP
- 6 » operating temperature -40°C to +85°C

Now let us look into the specifications of this Microcontroller.

Architecture: 32-bit ARM Cortex M3

Operating Voltage: 2.7V to 3.6V

CPU Frequency: 72 MHz

Number of GPIO pins: 37

Number of PWM pins: 12

Analog input Pins: 10 (12-bit)

USART Peripherals: 3

I2C Peripherals: 2

SPI Peripherals: 2

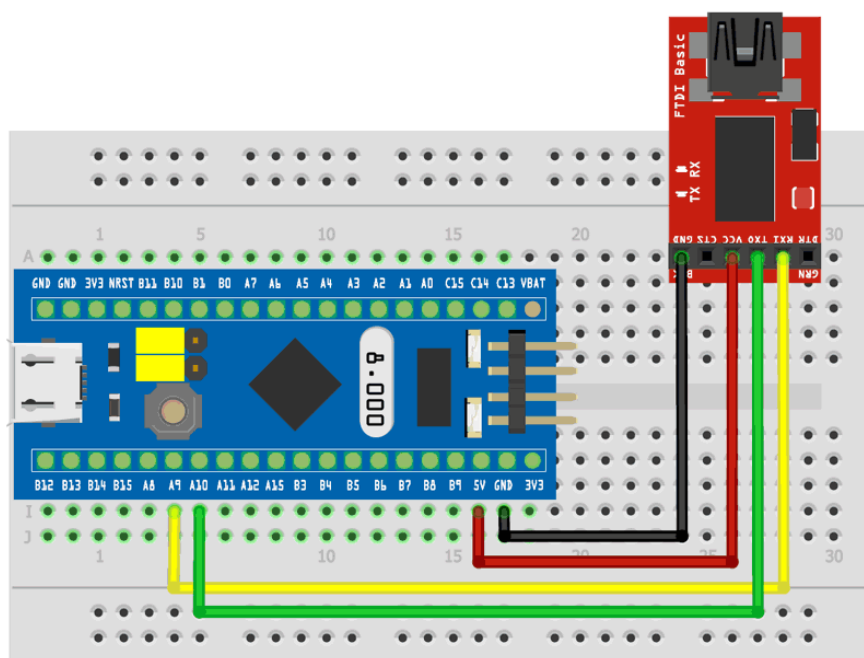
Can 2.0 Peripheral: 1

Timers: 3(16-bit), 1 (PWM)

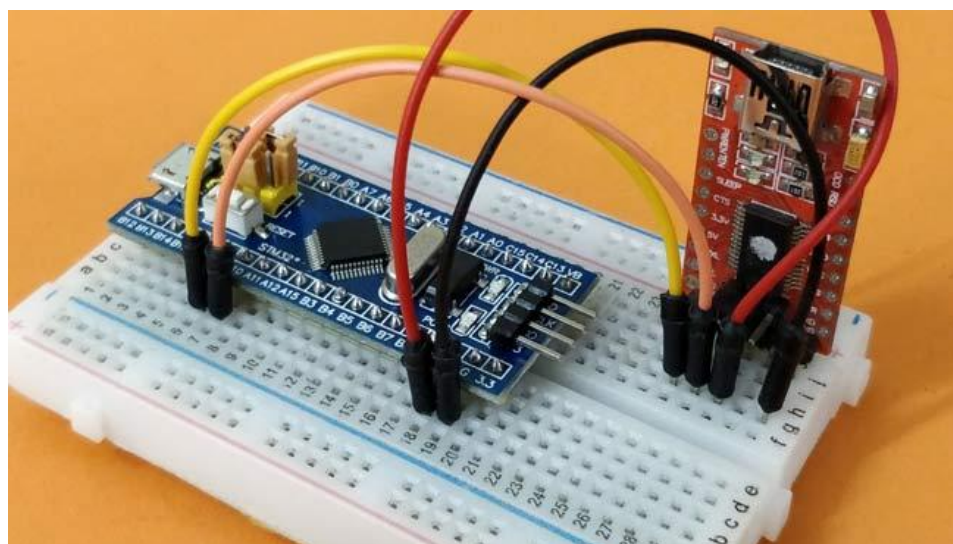
Flash Memory: 64KB

RAM: 20kB

بوردهایی مانند برد STM32-F103C8T6 را که شامل یک میکروکنترلر STM32، مدار پروگرامر، مدار تغذیه و ... می باشد را برد دیسکاواری آن میکروکنترلر می گویند. میکروکنترلرهای STM32 را می توان از چند طریق پروگرام کرد. یکی از این راه ها استفاده از پروگرامر های ST-Link و J-Link است که نرم افزار Keil به راحتی آنها را شناخته و با آنها سازگاری بالایی دارد. روش دیگر استفاده از یک مبدل USB به سریال FT232RL است که نرم افزار Arduino IDE به راحتی می تواند آنها را بشناسد. برای پروگرام کردن این برد توسط نرم افزار Arduino IDE ابتدا مبدل USB به Serial را به صورت زیر متصل کرده و سپس برد STM32F1 را روی Arduino IDE نصب می کنیم.



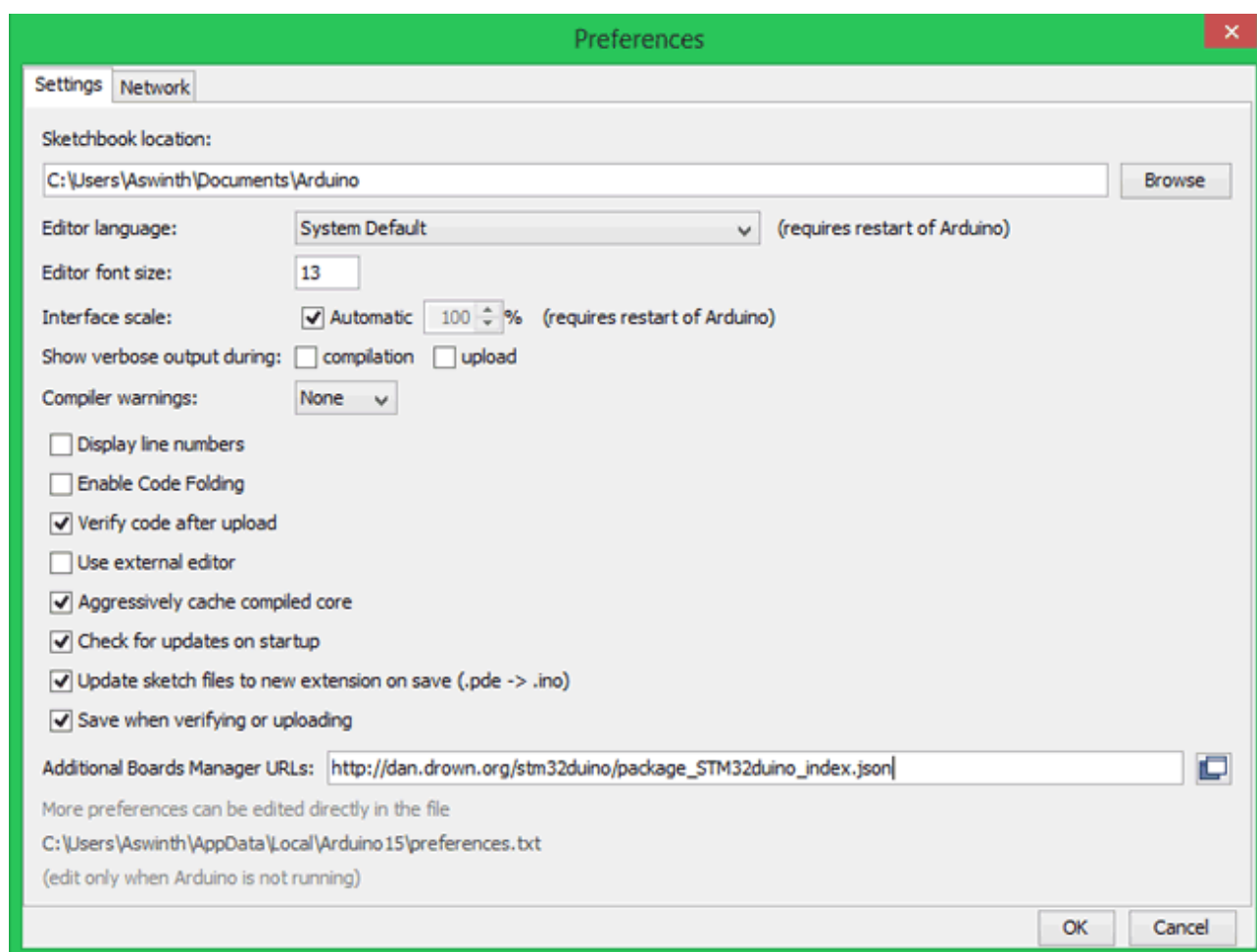
FTDI >> STM32
Gnd >> Gnd
Vcc >> 5V
Rx >> A9
Tx >> A10



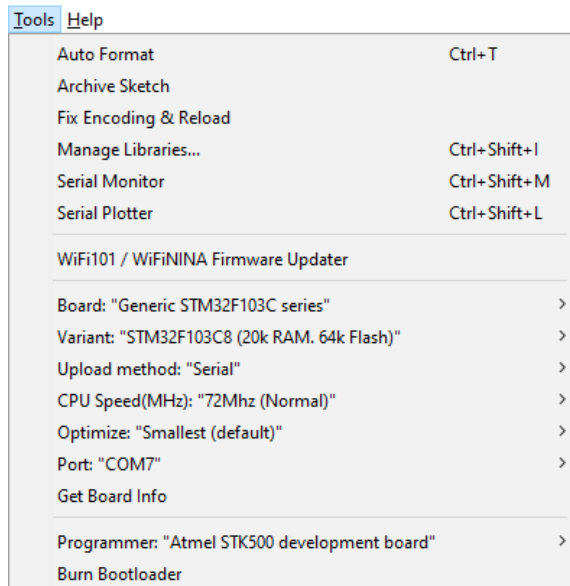
افزودن برد STM32F1 به Arduino IDE:

برای استفاده از برد دیسکاواری STM32F103 به عنوان یک برد آردوینو ابتدا باید آن را به لیست برد های نرم افزار آردوینو اضافه کرد. برای این کار از منوی File به قسمت Preferences رفته و در قسمت Additional Board Manager URLs لینک زیر را اضافه کنید.

http://dan.drown.org/stm32duino/package_STM32duino_index.json



سپس از منوی Tools در قسمت boards وارد Board Manager شده و عبارت STM32F1 را سرچ کنید و برد مورد نظر را install کنید.



برای پروگرام کردن میکروکنترلر با استفاده از مبدل USB به Serial از منوی tools برد خود را Generic STM32F103C Series انتخاب کرده و تنظیمات مربوط به آن را در منوی tools به صورت روبه رو تغییر می‌دهیم. دقت کنید که شیوه آپلود کردن به صورت سریال خواهد بود پس upload method باید روی Serial تنظیم شود.

برای آپلود برنامه روی برد STM32 لازم است میکروکنترلر را در حالت برنامه پذیری ببریم. برای اینکار باید پایه boot 0 آن را 1 و boot 1 آن را 0 قرار دهیم. پس از هر بار تغییر وضعیت بوت ها باید میکرو را ریست کنیم.



در قسمت Examples از منوی File نمونه برنامه های مخصوص برد STM32 را میتوانید با عنوان A_STM32 Examples بیابید. به عنوان مثال می‌توانید پروژه Blink آن را از قسمت Digital باز کرده و پس از تنظیم جامپرهای بوت روی برد و ریست کردن آن می‌توانید برنامه را روی برد پروگرام نمایید. توجه داشته باشید که LED روی برد بر روی پایه PC13 قرار دارد و هنگامی که این پایه 1 منطقی باشد LED روشن خواهد شد.

```
1 void setup() {
2     pinMode(PC13, OUTPUT);
3 }
4
5 void loop() {
6     digitalWrite(PC13, HIGH);
7     delay(1000);
8     digitalWrite(PC13, LOW);
9     delay(1000);
10 }
```

دستوراتی که برای برد STM32 استفاده می‌کنیم کاملاً مشابه دستورات آردوینو است. برای مثال دستورات `digitalWrite()`، `digitalRead()` و `analogRead()` به طور مشابه استفاده می‌شود.

تولید پالس PWM

تولید یک پالس مربعی و کنترل عرض آن کاربردهای زیادی دارد که کنترل سرعت یک موتور با استفاده از یک درایور پرکاربرد ترین نمونه آن است. در میکروکنترلر STM32F103 تایمر ها دارای ۱۶ بیت هستند و میتوانند مقداری از ۰ تا ۶۵۵۳۵ داشته باشند. این میکروکنترلر دارای ۱۵ پایه با قابلیت ایجاد پالس PWM است که می‌توانید این پایه ها را در نقشه صفحه ۵ مشاهده کنید. برای استفاده از این قابلیت ابتدا با دستور `pinMode()` پایه را روی حالت PWM تنظیم کرده و برای مقدار دهی به آن از دستور `pwmWrite()` استفاده می‌کنیم.

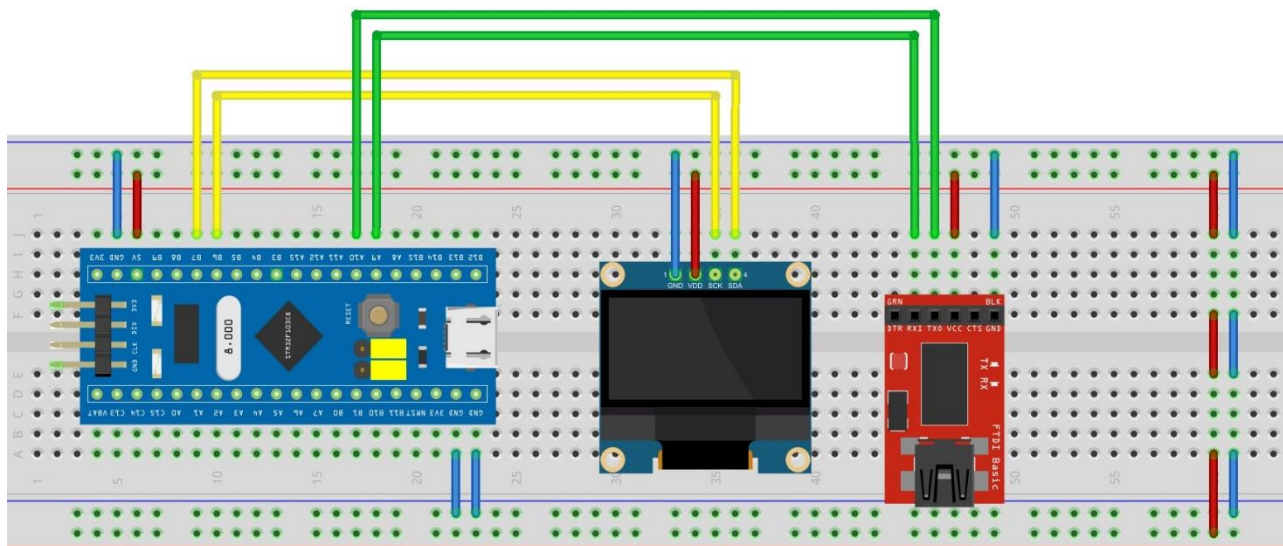
```
1 void setup() {
2     pinMode(PA1, PWM);
3 }
4 void loop() {
5     pwmWrite(PA1, 65535);
6 }
```

راه اندازی نمایشگر گرافیکی OLED (SH-1106)



این نمایشگر گرافیکی دارای ۶۴ * ۱۳۲ پیکسل است که هر پیکسل آن یک LED بسیار کوچک با رنگ سفید یا آبی می‌باشد. این نمایشگر توسط پروتوکل I2C می‌تواند از یک میکروکنترلر دستور بگیرد. تفاوت نمایشگر کاراکتری و گرافیکی در این است که در نمایشگر های کاراکتری تنها میتوان حروف و اعداد و ... را به صورت کاراکتر در مکان های مشخصی چاپ نمود. اما در نمایشگر های گرافیکی شما می‌توانید اشکال هندسی و همچنین عکس را نیز نمایش دهید.

برای راه اندازی این نمایشگر ابتدا مدار زیر را بر روی برد ببندید.



برای نصب کتابخانه مربوط به نمایشگر SH1106 در نرم افزار آردوینو از منوی sketch به قسمت include library رفته و Add .ZIP library را انتخاب کنید. فایل Adafruit_SH1106_STM32-master.zip را انتخاب کنید و صبر کنید تا کتابخانه نصب شود. سپس به همین ترتیب فایل Adafruit-GFX-Library-master.zip را نیز نصب کنید.

پس از نصب کتابخانه های مورد نیاز ابتدا کتابخانه Adafruit_SH1106_STM32.h را فراخوانی کرده و با استفاده از کلاس Adafruit_SH1106 یک صفحه با نام display() میسازیم.

```
1 #include <Adafruit_SH1106_STM32.h>
2 Adafruit_SH1106 display(-1);
```

برای شروع ارتباط با نمایشگر از تابع begin() استفاده میکنیم که ورودی آن آدرس های نمایشگر و میکروکنترلر است. توجه داشته باشید که این دستور لوگوی شرکت Adafruit که سازنده نمایشگرها و نویسنده کتابخانه های مربوط به آنهاست بر روی نمایشگر نمایش داده می شود. تابع display() صفحه نمایشگر را به روز رسانی می کند.

```
1 #include <Adafruit_SH1106_STM32.h>
2 Adafruit_SH1106 display(-1);
3 void setup() {
4     display.begin(0x2, 0x3C);
5     display.display();
6 }
```

دستورات پر کاربرد کتابخانه Adafruit_SH1106_STM32.h

نام دستور	کاربرد
<code>display.clearDisplay()</code>	پاک کردن صفحه نمایش
<code>display.fillCircle(x, y, r, color)</code>	رسم دایره توپر در مختصات (x,y) با شعاع r و رنگ color
<code>display.drawCircle(x, y, r, color)</code>	رسم دایره توخالی در مختصات (x,y) با شعاع r و رنگ color
<code>display.display()</code>	آپدیت کردن صفحه نمایش
<code>display.fillRect(x, y, w, h, color)</code>	رسم مستطیل توپر در مختصات (x,y) با طول و عرض w و h
<code>display.drawRect(x, y, w, h, color)</code>	رسم مستطیل توخالی در مختصات (x,y) با طول و عرض w و h
<code>display.setTextSize(s)</code>	تغییر دادن اندازه نوشته
<code>display.setTextColor(color)</code>	تغییر دادن رنگ نوشته
<code>display.setCursor(0 , 0)</code>	رفتن به مختصات (0,0)
<code>display.print("Hello")</code>	چاپ کردن کلمه Hello
<code>display.println("Hello")</code>	چاپ کردن کلمه Hello و رفتن به خط بعدی
<code>display.invertDisplay()</code>	اینورت (برعکس) کردن رنگ های نمایشگر
<code>display.drawPixel(x , y , color)</code>	تغییر رنگ یک پیکسل در مختصات (x,y)
<code>display.getPixel(x , y)</code>	گرفتن رنگ یک پیکسل در مختصات (x,y)
<code>display.width()</code>	طول نمایشگر
<code>display.height()</code>	عرض نمایشگر
<code>display.drawLine(x1, y1, x2, y2, color)</code>	رسم خط بین دو نقطه با مختصات های (x1,y1) و (x2,y2)
<code>display.drawTriangle(x1,y1,x2,y2,x3,y3,color)</code>	رسم مثلث توخالی بین سه نقطه
<code>display.fillTriangle(x1,y1,x2,y2,x3,y3,color)</code>	رسم مثلث توپر بین سه نقطه
<code>display.drawChar(x,y,c,color,bg,size)</code>	چاپ یک کاراکتر

نکته: در دستورات بالا color یا رنگ می تواند WHITE یا BLACK باشد.