

## نحوه پروگرام کردن میکروکنترلر های AVR با استفاده از Arduino ISP

سلام بر اردوینو کاران جوان.

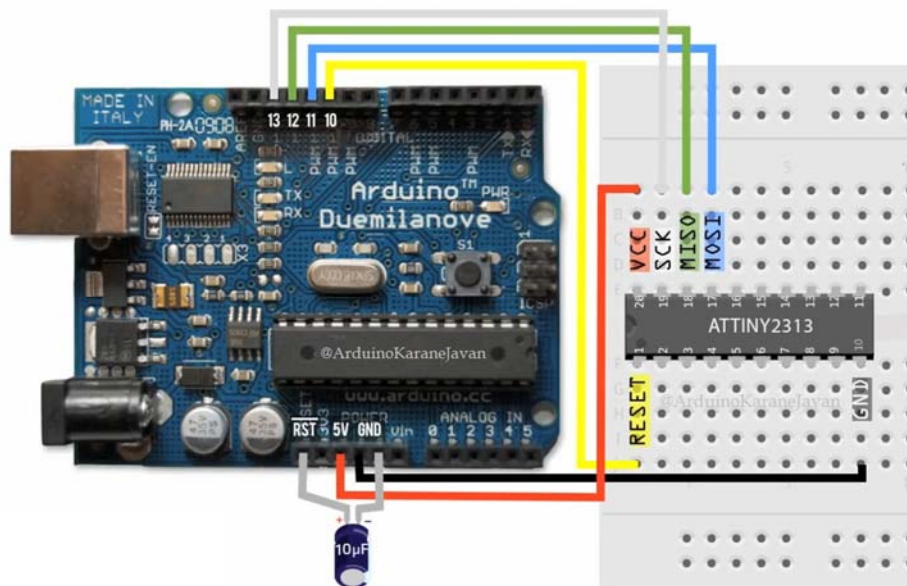
من رضا اژدرکش هستم و امروز میخوام براتون نحوه پروگرام کردن فایل های HEX توسط اردوینو و با استفاده از پروگرامر Arduino ISP رو توضیح بدم.

دوستان گلم همانطور که در فایل قبلی که داخل کانال تلگرام اردوینو کاران جوان قرار دادم و نحوه ساخت یک پروگرامر Arduino ISP رو توضیح دادم، یک پروگرامر با برد اردوینو نانو یا UNO می سازیم و ازش توی این آموزش استفاده میکنیم.

برای عزیزانی که به این فایل دسترسی ندارند، مجدداً نحوه ساخت پروگرامر رو توضیح میدم:

- 1- برد اردوینو UNO رو به کامپیوتر متصل کنید.
- 2- در IDE اردوینو فایل 11-ArduinoISP>>Example>>File را باز کنید.
- 3- از منوی Tools پورت مناسب رو انتخاب کنید و برنامه را داخل اردوینو UNO آپلود کنید.
- 4- بین پین های GND و RST اردوینو یک خازن الکترولیتی 10 الی 100 میکروفاراد متصل کنید.
- 5- حالا پینهای SCK, MOSI, MISO و پین شماره 10 اردوینو UNO که بجای ریست استفاده میشود، را نظیر به نظیر، به پینهای SCK, MOSI, MISO, Reset میکروکنترلر متصل نمایید.

تا اینجا پروگرامر Arduino ISP ما آماده است و اگر بخواهیم با این پروگرامر مثلاً برای ATTINY2313 در IDE اردوینو برنامه بنویسیم. و آپلود کنیم، براحتی میتونیم کدهای مورد نظر را نوشته و از منوی Board برد اردوینو ATTINY2313 را انتخاب کرده و از منوی sketch گزینه UploadUsingProgrammer رو بزیم تا برنامه داخل میکروکنترلر آپلود شود.



نحوه نصب برد های AVR از جمله ATMEGA8 در IDE آردوینو را در همان فایل نحوه ساخت برد آردوینوی دلخواه در [کانال تلگرام آردوینو کاران جوان](#) به صورت مفصل توضیح داده ام.

اما حالا ما یک فایل هگز داریم که از اینترنت دانلود کرده ایم و اصلا نمیدانیم به چه زبانی نوشته شده است. و فقط میدانیم که برای چه میکروکنترلی هست.

بعنوان مثال، ما نقشه ساخت یک تستر قطعات الکترونیک را از اینترنت گرفته ایم که با میکروکنترلر ATMEGA8 ساخته شده است. ولی برنامه آن را نداریم و فقط فایل هگز برنامه را دانلود کرده ایم.

حالا میخواهیم با استفاده از همین پروگرامری که ساخته ایم و با استفاده از IDE آردوینو این فایل هگز را داخل میکروکنترلر ATMEGA8 آپلود کنیم.

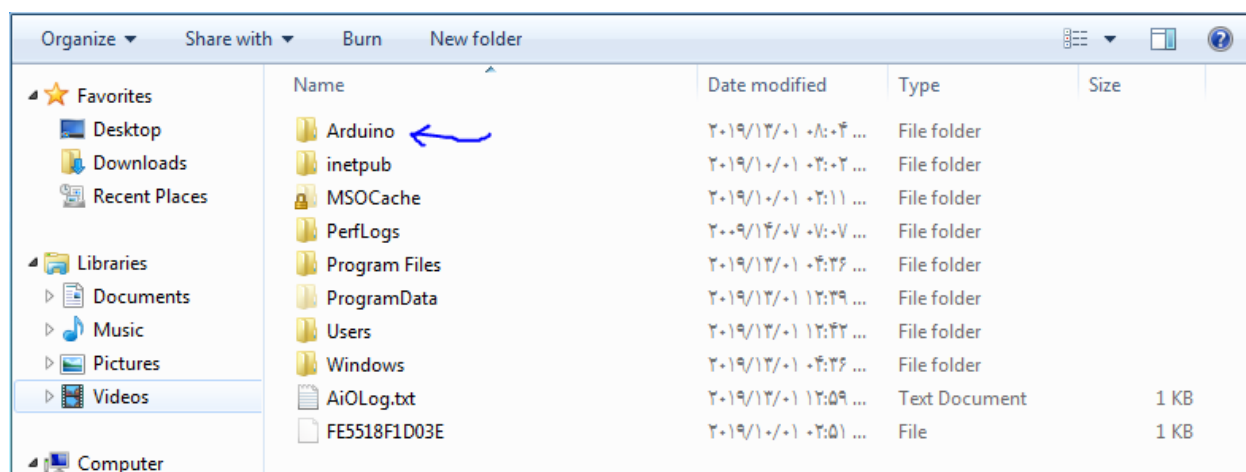
سوال: چطور این کار را انجام دهیم؟؟؟

جواب: به سختی. ☺

چون IDE آردوینو همچین امکانی را برای ما فراهم نکرده است و ما باید با استفاده از شگردی این کار را انجام دهیم.

همانطور که میدانید، پروگرامر IDE آردوینو avrdude.exe هست. پس ما میتوانیم با نوشتن یک بچ فایل (batch File) مسیر فایل هگز را به avrdude بدهیم تا آنرا برای ما پروگرام نماید.

قبل از اینکار باید پوشه Arduino را از داخل Program Files به داخل درایو C کپی کنیم. چون مسیر هایی که داخل بچ فایل نوشته میشوند نباید دارای کاراکتر فاصله باشند و چون نام Program Files دارای فاصله است، نمیتوانیم این کار را در همان محل نصب آردوینو انجام دهیم. لذا پوشه Arduino را از داخل Program Files به داخل درایو C کپی می کنیم.



حالا همان پوشه آردوینو در درایو C را باز کرده و داخل آن یک پوشه جدید بنام hexFiles درست میکنیم. و فایل هگزی که میخواهیم داخل میکروکنترلر آپلود کنیم را درون این پوشه قرار میدهیم.

Name	Date modified	Type	Size
drivers	۲۰۱۹/۱۲/۰۱ ۰۸:۰۲ ...	File folder	
examples	۲۰۱۹/۱۲/۰۱ ۰۸:۰۲ ...	File folder	
hardware	۲۰۱۹/۱۲/۰۱ ۰۸:۰۲ ...	File folder	
hexFiles	۲۰۱۹/۱۲/۰۱ ۰۸:۲۰ ...	File folder	
java	۲۰۱۹/۱۲/۰۱ ۰۸:۰۴ ...	File folder	
lib	۲۰۱۹/۱۲/۰۱ ۰۸:۰۴ ...	File folder	
libraries	۲۰۱۹/۱۲/۰۱ ۰۸:۰۴ ...	File folder	
reference	۲۰۱۹/۱۲/۰۱ ۰۸:۰۴ ...	File folder	
tools	۲۰۱۹/۱۲/۰۱ ۰۸:۰۴ ...	File folder	
tools-builder	۲۰۱۹/۱۲/۰۱ ۰۸:۰۴ ...	File folder	
arduino.exe	۲۰۱۷/۰۶/۱۲ ۰۱:۵۵ ...	Application	395 KB

دقت نمایید که نام فایل های هگزی که داخل این پوشه قرار می دهید نیز نباید دارای فاصله بین حروف باشند. چون در این صورت، آپلود انجام نخواهد شد.

خب حالا می رسیم به نحوه پروگرام کردن فایل هگز فوق با avrdude ، برای اینکار ابتدا یک فایل تکست روی دسکتاپ ایجاد نمایید و متن زیر را در داخل آن کپی کنید:

```
C:\Arduino\hardware\tools\avr\bin\avrdude -C
C:\Arduino\hardware\tools\avr\etc\avrdude.conf -v -v -v -v -p atmega8 -c stk500v1
-P COM9 -b 19200 -U flash:w:C:\Arduino\hexFiles\tester.hex:i
```

pause

توضیحات کد های فوق بر حسب رنگ بندی قسمتها:

- 1- مسیر فایل avrdude.exe
- 2- مسیر فایل avrdude.conf که بعد از C- با حرف بزرگ نوشته می شود.
- 3- نام میکروکنترلی که میخواهیم فایل هگز را داخل آن آپلود کنیم. بعد از p- با حرف کوچک.
- 4- نام پروگرامر پیش فرض آردوینو. بعد از c- با حرف کوچک نوشته می شود.
- 5- شماره پورت com ای که وقتی پروگرامر مان را میزنیم شناسایی می شود. بعد از P- بزرگ.
- 6- بادریت پروگرام کردن که بعد از b- با حرف کوچک نوشته می شود.
- 7- محل آپلود کردن برنامه. بعد از U- نوشته می شود.
- 8- مسیر فایل هگز که خودمان ایجاد کردیم. و بعد از w: نوشته می شود که رایت شود.
- 9- نام فایلی که می خواهیم آپلود کنیم.

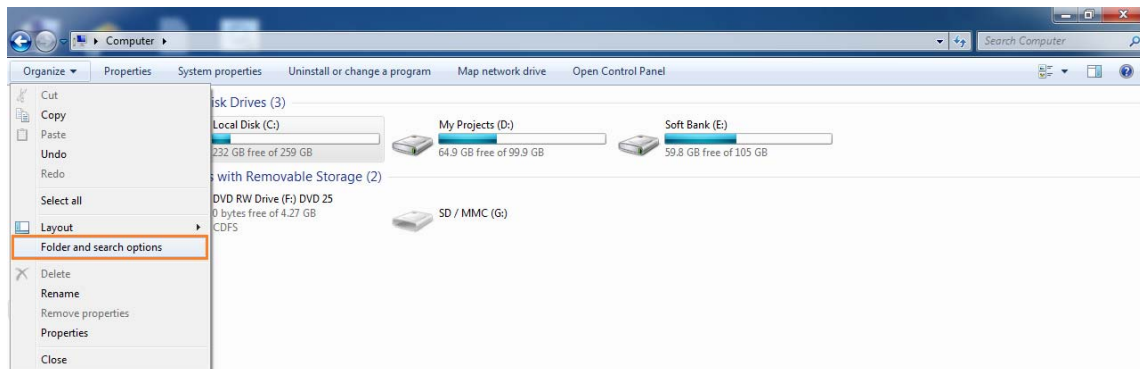
همه این عبارات در یک خط نوشته می شوند و در انتهای مسیر هم یک علامت i: قرار می دهیم.

تنها چیزی که ما باید تغییر دهیم، نام میکروکنترلر و شماره پورت com و نام فایل هگز است. (3-5-9). دستور بعدی که در خط بعد نوشته می شود، pause باعث میشود تا پنجره آپلود کردن فایل پس از آپلود کردن بسته نشود و ما بتوانیم موفقیت آمیز بودن آپلود را ببینیم.

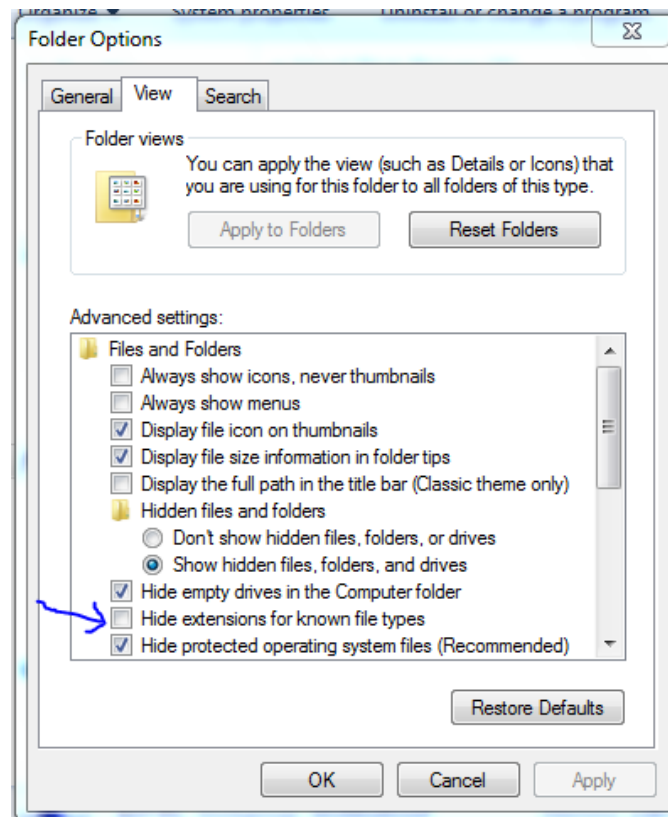
خیلی هم عالی...

حالا که این دستورات را نوشتیم. فایل تکست فوق را save کرده و می بندیم. و سپس باید این فایل تکست را به بچ فایل تبدیل کنیم.

وارد یکی از پوشه های موجود در دسکتاپ کامپیوتر شده و یا My Computer را باز کنید و از منوی Organize گزینه Folder and search options را انتخاب کنید.

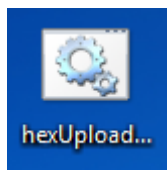


حالا در تب View تیک گزینه Hide extensions for known file types را بردارید. تا اکستنشن فایلها نمایش داده شوند.



و در مرحله آخر، فایل تکستی را که ساخته بودیم و با نام New Text Document.txt ذخیره شده بود، نام و اکستنشن آنرا بصورت زیر تغییر می دهیم و ذخیره می کنیم:

hexUpload.bat

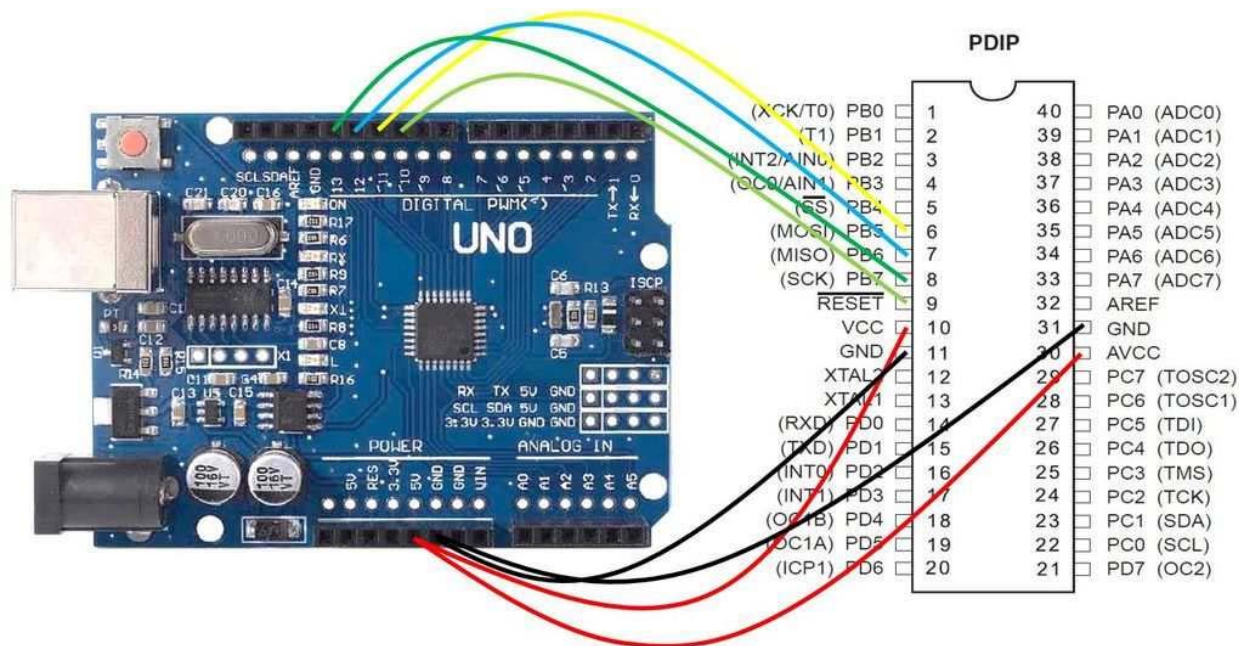


به این صورت فایل مورد نظر به یک فایل اجرایی یا بچ فایل تبدیل می شود. و با دابل کلیک بر روی این بچ فایل، فایل هگز برنامه در میکروکنترلر انتخاب شده آپلود می شود. دقت کنید که هنگام پروگرام شدن میکروکنترلر، باید LED های مربوط به TX, RX برد آردوینو UNO در حال چشمک زدن باشند و دلربایی کنند. ☺ ☺

هر خطایی هنگام آپلود فایل هگز، پیش بیاید به علت ایجاد فاصله در نامهای مسیر ها و یا رعایت نکردن حروف بزرگ و کوچک و یا جهت علامتهای / و \ بکار رفته در دستورات می باشد.

در ضمن هنگام بستن مدار، همه GND های میکروکنترلر باید به GND آردوینو UNO و همچنین همه VCC ها و AVCC میکروکنترلر هم باید به +5V آردوینو UNO متصل شوند.

بعنوان مثال: در پروگرام کردن ATMEGA32 بصورت زیر عمل میکنیم:



در صورت اجرای موفقیت آمیز آپلود فایل، avrdude از شما تشکر میکند. و با زدن یکی از کلید های کیبرد، پنجره فوق بسته می شود.

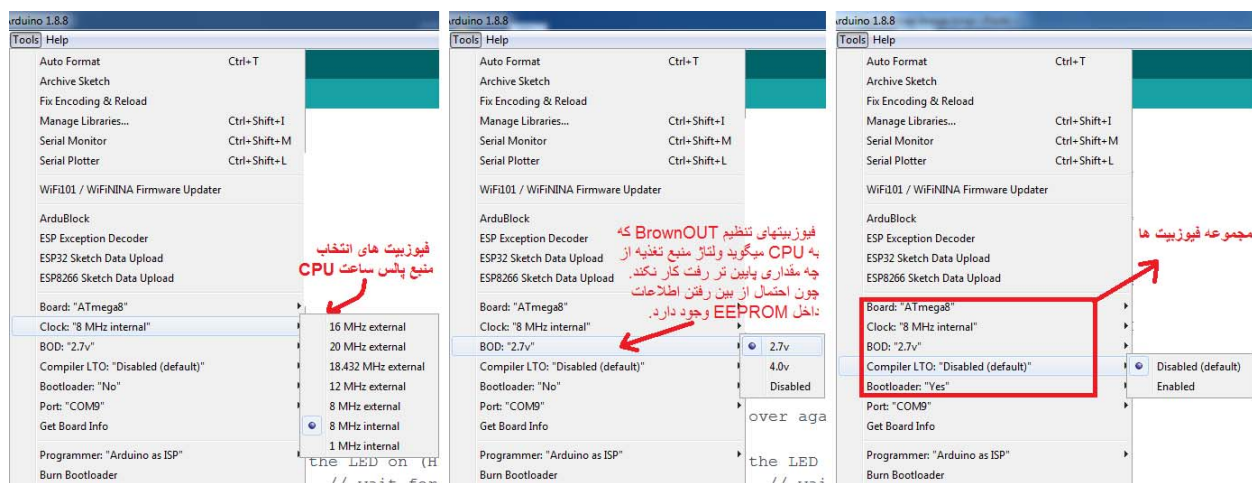


فیوز بیتها در واقع به میکروکنترلر میگویند که مثلا با چه فرکانسی کار کند، یا منبع پالس ساعت آن چه چیزی باشد. (1MHz داخلی، 8MHz داخلی، 16MHz خارجی یا ...) و همچنین تنظیمات دیگری مانند تنظیمات BrownOUT و غیره...

سوال: خب پس چرا ما در IDE آردوینو، فیوزبیت ها را تنظیم نمیکنیم؟

جواب: چرا ما در آردوینو هم همیشه فیوزبیت ها را تنظیم میکنیم. ولی با روشی بسیار ساده. و متفاوت با همه پروگرامرها. و آنقدر اینکار برایمان راحت است که یادمان میرود اینها فیوزبیت هستند.

بعنوان مثال به شکل زیر توجه نمایید. در این شکل تنظیمات فیوزبیت های میکروکنترلر ATMEGA8 را می بینید که بصورت کاملا مفهومی و قابل درک نشان داده شده اند و براحتی قابل تنظیم هستند:



در قسمت بعدی آموزش، نحوه تنظیم فیوزبیتها توسط avrdude آموزش داده خواهد شد. امیدوارم که این آموزش نیز مانند سایر آموزشهای کانال آردوینو کاران جوان، مورد استفاده شما عزیزان قرار بگیرد.

**در نهایت، خواهشی از شما عزیزان دارم:**

با خرید کردن از فروشگاه آردوینو کاران جوان، از این کانال حمایت نمایید. تا لااقل گوشه ای از هزینه های تحقیق و بررسی و زمانی که برای ارائه هر چه بهتر مطالب صرف می شود، را تامین نماید.

با سپاس فراوان

<http://arduinokaranejavan.sellfile.ir>

آدرس فروشگاه آردوینو کاران جوان:

**آردوینو کاران جوان همیشه بهترین ها را برای شما آرزو می کند.**

**پس با ما همراه باشید.**

رضا اژدرکش