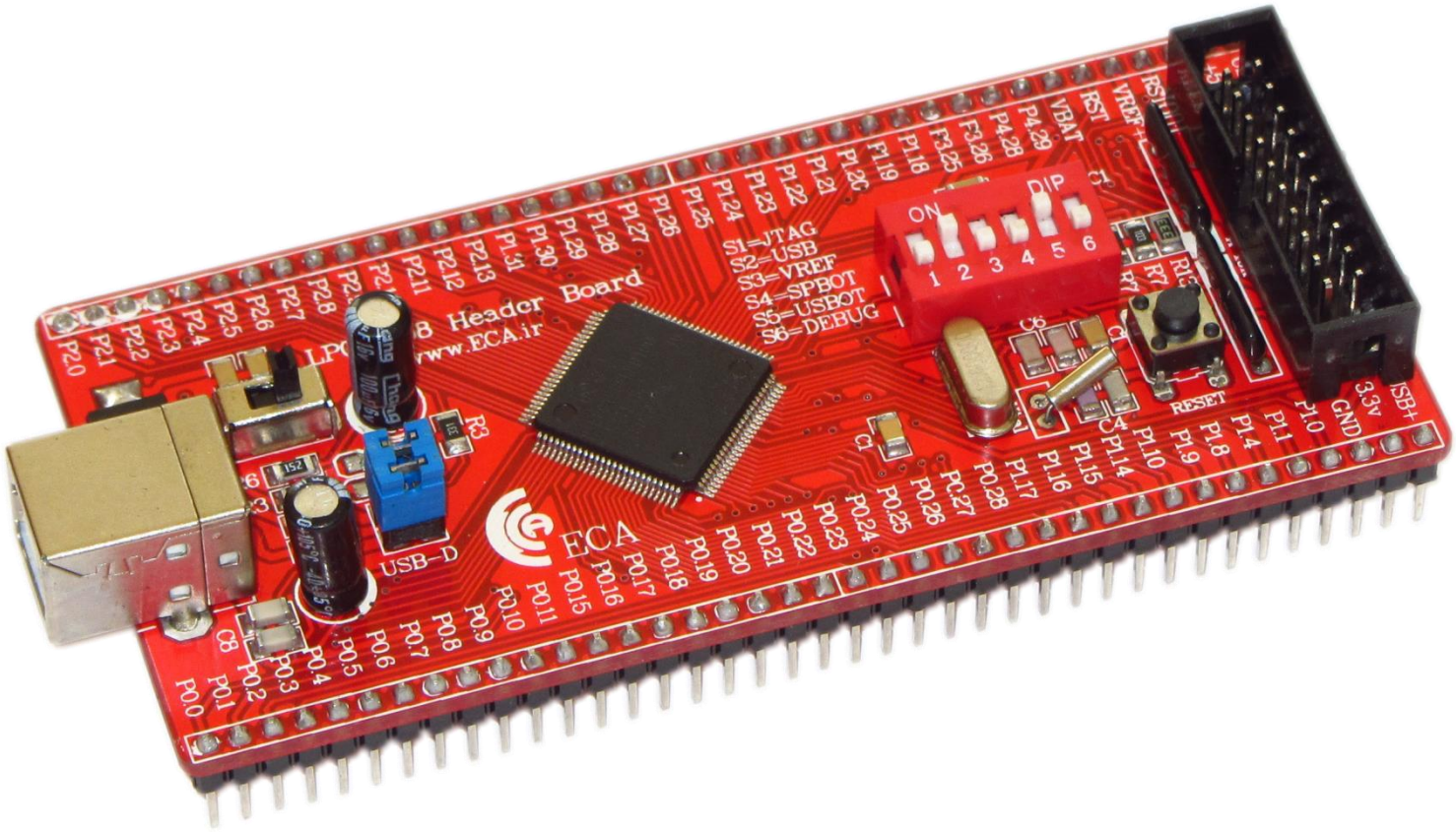
هدربرد – برد راه انداز LPC1768 Cortex-M3

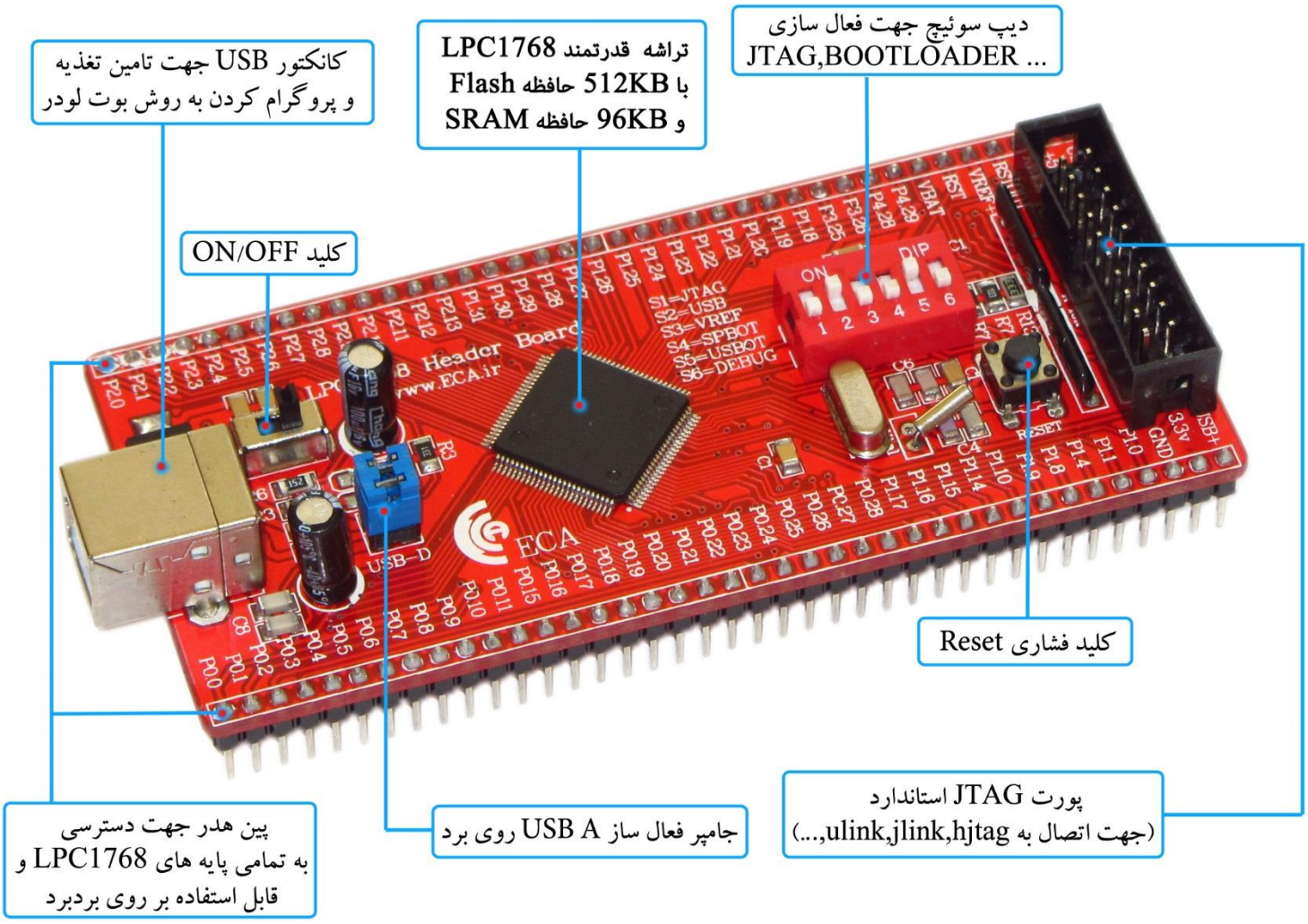


فهرست مطالب

هدربرد NXP LPC1768 ..... ۳
راهنمای جامپرهای موجود بر روی برد ۴
راهنمای دیپ سوئیچ موجود بر روی برد ۵
روش های مختلف پروگرام کردن برد، مزایا و معایب آنها ۶
نحوه ایجاد کد باینری توسط کامپایلر Keil ۶
نحوه‌ی پروگرام کردن میکرو با استفاده از بوت لودر USB ۸
نحوه پروگرام کردن میکرو با استفاده از پروگرامر J-Link ۱۶
پروگرام کردن برد توسط J-Link و کامپایلر Keil ۲۱
دیباگ کردن برد توسط کامپایلر Keil و پروگرامر J-Link ۲۵

هدربرد NXP LPC1768

تراشه LPC1768 ساخت شرکتNXP ، مبتنی بر هسته نسل دوم میکروکنترلرهای ARM Cortex-M3 طراحی گشته است. این هسته برای کاربردهای سیستم های نهفته (Embedded)در سرعت های بالا، توان مصرفی کم و پردازش ۳۲ بیتی طراحی گشته است. از جمله اهداف طراحی این نسل از پردازنده ها می توان به مصارف اندازه گیری، ارتباطات صنعتی، کنترل موتورهای صنعتی، سیستم های هوشمند و روباتیک اشاره نمود.



❊ تراشه ی قدرتمند LPC1768

فرکانس کاری : ۱۰۰ مگاهرتز

میزان حافظه فلش : ۵۱۲ کیلوبایت

میزان رم : ۶۴ کیلوبایت

آنالوگ به دیجیتال: ۸ کانال ۱۲ بیتی

دیجیتال به آنالوگ: ۲ کانال ۱۲ بیتی

پایه های قابل برنامه ریزی : ۷۰ عدد

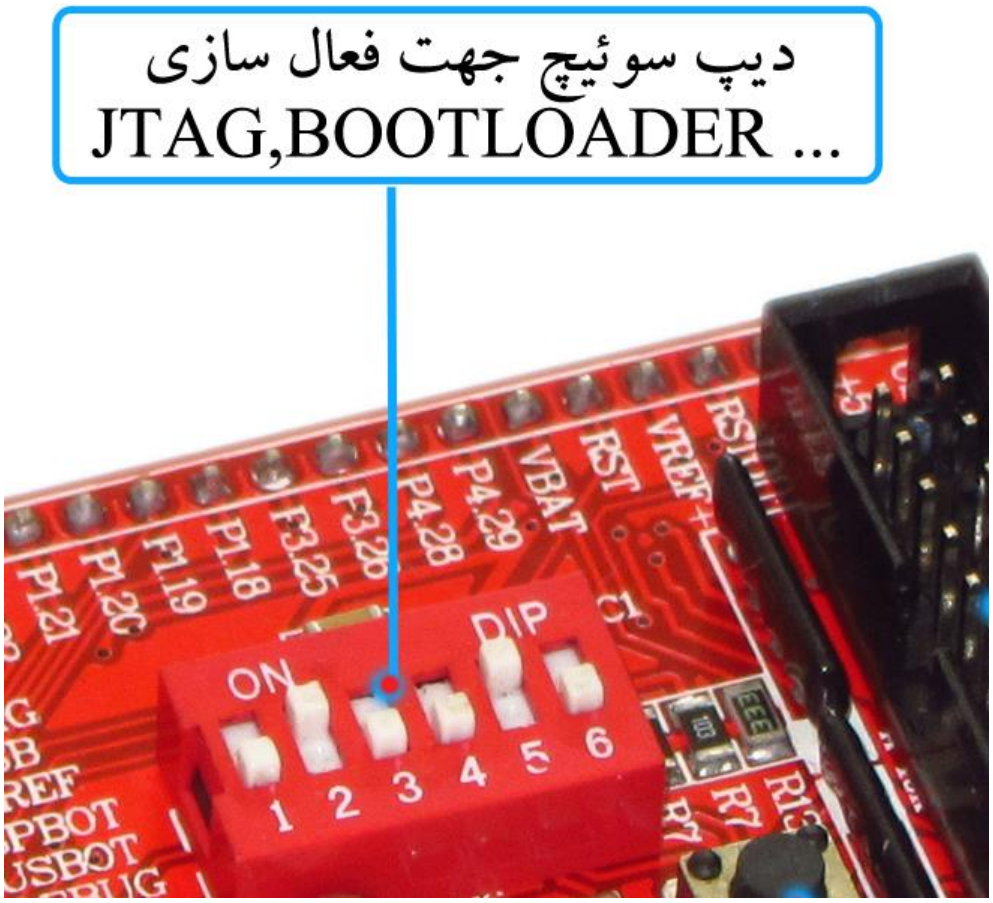
تایمر : ۴ عدد

دارای پروتکل های CAN , I2C , SPI , USART , Ethernet MAC, USB Device/Host/OTG

راهنمای جامپرهای موجود بر روی برد

جامپر	توضیحات
USB-D	جهت فعال نمودن USB A روی برد

راهنمای دیپ سوئیچ موجود بر روی برد



دیپ سوئیچ	کارکرد
JTAG	برای فعال کردن پورتJTAG
USB	برای فعال کردن پورتUSB
VREF	برای اتصال پایه VREF به VCC
SPBOOT	برای فعال کردن بوت لودر سریال توسط نرم افزارFlash Magic
USBBOOT	برای فعال کردن بوت لودر USB
DEBUG	برای فعال کردن حالت عیب یابی توسطJTAG

روش های مختلف پروگرام کردن برد، مزایا و معایب آنها

۲ روش برای پروگرام کردن برد آموزشی NXP LPC1768 وجود دارد:

۱- استفاده از بوت لودر ISP و بدون نیاز به پروگرامر خارجی
۲- استفاده از پروگرامر خارجی J-Link

که هر کدام مزایا و معایب خاص خود را دارند که به آنها اشاره خواهیم کرد.

۱- استفاده از بوت لودر ISP و بدون نیاز به پروگرامر خارجی

مزایا: در روش بوت لودر ISP احتیاجی به پروگرامر خارجی ندارد و میتوان مستقیما میکروکنترلر را با کابل USB به کامپیوتر متصل و پروگرام کرد.

معایب: امکان استفاده ی مستقیم از کامپایلر Keil جهت پروگرام نمودن وجود ندارد بلکه بایستی از نرم افزار اختصاصی جهت پروگرام استفاده کرد. همچنین امکان دیباگ میکروکنترلر در این روش وجود ندارد.مشکل دیگر این روش نیاز به تغییر وضعیت دیپ سوئیچ برای فعال کردن بوت لودر است.

۲- استفاده از پروگرامر خارجی J-Link

مزایا: در این روش، پروگرامر جیلینک مستقیماً به پورت JTAG متصل شده و میکروکنترلر بصورت مستقیم از طریق کامپایلر پروگرام می گردد. شما علاوه بر پروگرام نمودن، می توانید برنامه خود را دیباگ سخت افزاری کنید. یعنی اینکه برنامه خود را خط به خط اجرا نموده و فرایند اجرای برنامه را مشاهده نمایید. دیباگ سخت افزاری به منظور تسریع فریایند آموزش و در پروژه های حرفه ای جهت ایرادیابی و رفع باگ سریع برنامه مورد استفاده قرار می‌گیرد.

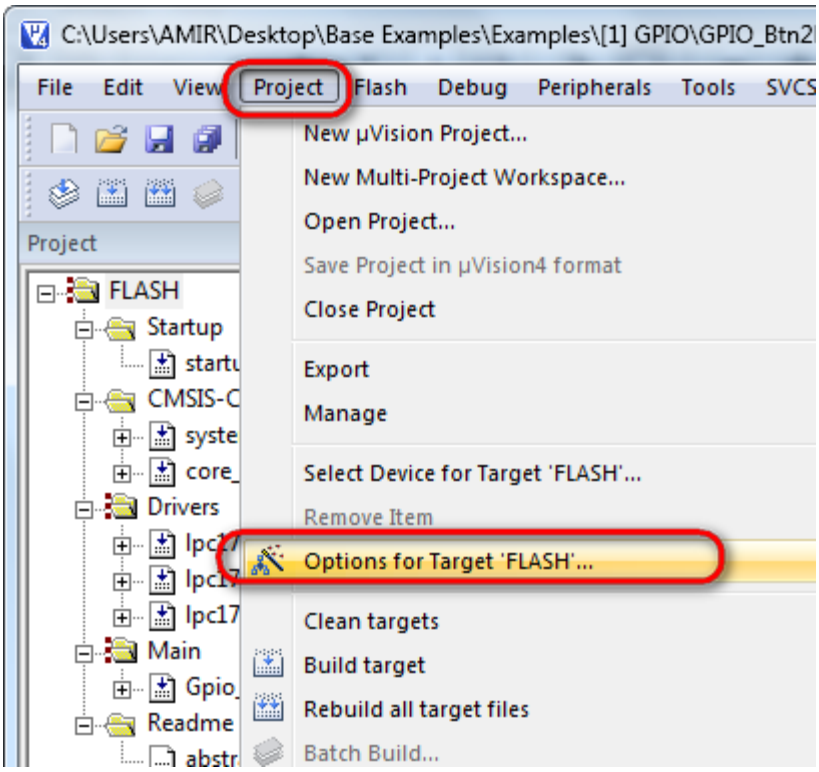
معایب: نیاز به تهیه ی سخت افزار J-Link دارد.

نحوه ایجاد کد باینری توسط کامپایلر Keil

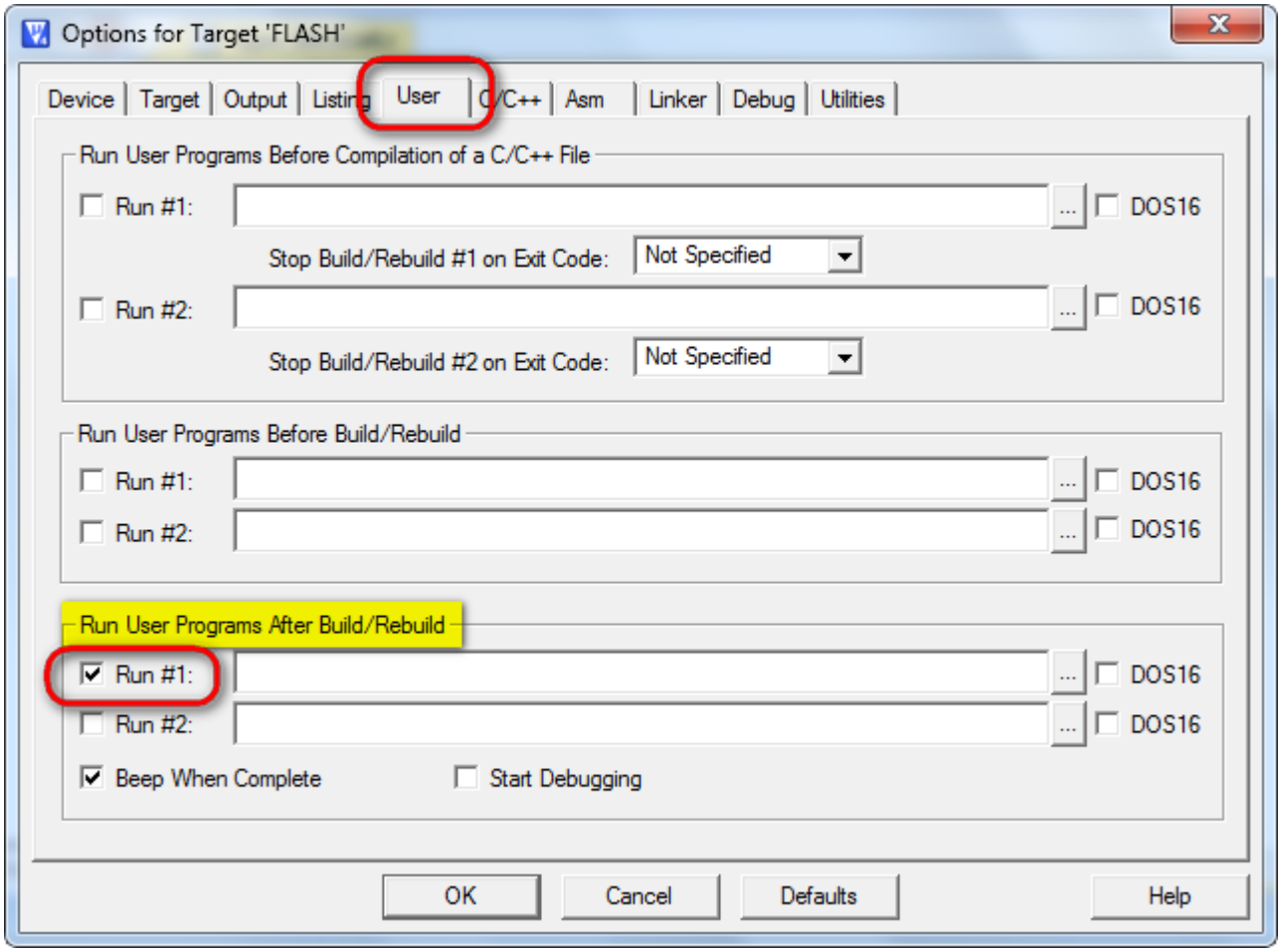
به صورت پیش‌فرض کد خروجی کامپایلر Keil به صورت هگز می‌باشد. درصورتیکه برای پروگرام کردن میکروکنترلر از طریق USB، بایستی از کد باینری استفاده شود. بنابراین به این نکته پرداخته خواهد شد که چگونه می‌توان با استفاده از Keil کد باینری ایجاد کرد.

به صورت پیش‌فرض Keil uVision نمی‌تواند کد باینری سازگار با بوت لودر USB ایجاد کند. به منظور ایجاد کد باینری سازگار با بوت لودر از یک خط دستور خارجی استفاده می‌شود.که به صورت زیر عمل می‌شود:

۱- پس از اجرای برنامه‌ی Keil از منوی Project گزینه‌ی 'Option for Target 'FLASH انتخاب شود.



۲- سپس به تب User رفته ور در قسمت "Run User Programs After Build/Rebuild" تیک مربوطه به "Run#1" زده شود.



۳- دستور fromel بصورت زیر در کادر مقابل "Run#1:" وارد می‌شود:

fromelf –bin ".\test.axf" –output ".\test.bin"

و در نهایت OK زده می‌شود.

در دستور فوق عبارت test مربوط به نام پروژه است و برای استفاده از این دستور در هر پروژه، نام آن پروژه به جای این عبارت قرار می‌گیرد. به طور مثال در شکل بالا این دستور برای برنامه نمونه lcd3.2 که در قسمت محصولات سایت قرار داده شده به کار برده شده است که در آن برنامه نام پروژه test است.

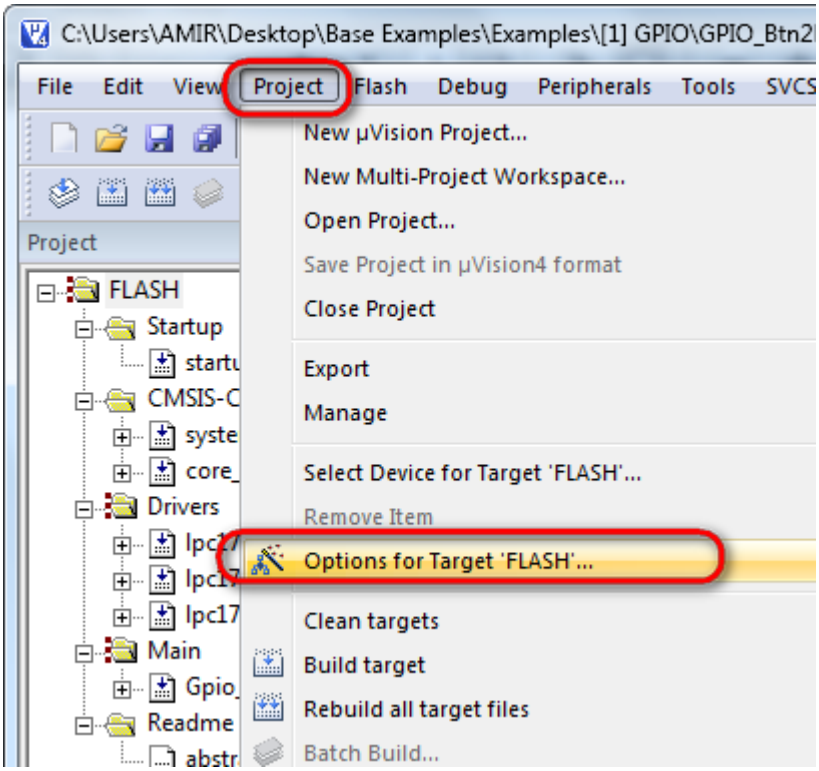
نحوه‌ی پروگرام کردن میکرو با استفاده از بوت لودر USB

برای استفاده از بوت لودر USB برای پروگرم کردن میکرو باید بوت لودر USB در حافظه میکروکنترلر قرار گرفته باشد. این بوت لودر در آدرس 0X00000000 تا 0X00001FFF از حافظه فلش میکروکنترلر قرار گرفته است. در صورتیکه برای اولین بار از هدر برد استفاده می‌شود، بوت لودر USB بر روی حافظه میکروکنترلر قرار داده شده است. ولی اگر قبلا میکروکنترلر توسط نرم افزار J-Link پروگرم شده باشد، بوت لودر از روی حافظه پاک شده و باید برای استفاده از آن دوباره کد مربوط به بوت لودر که در سایت قرار گرفته :

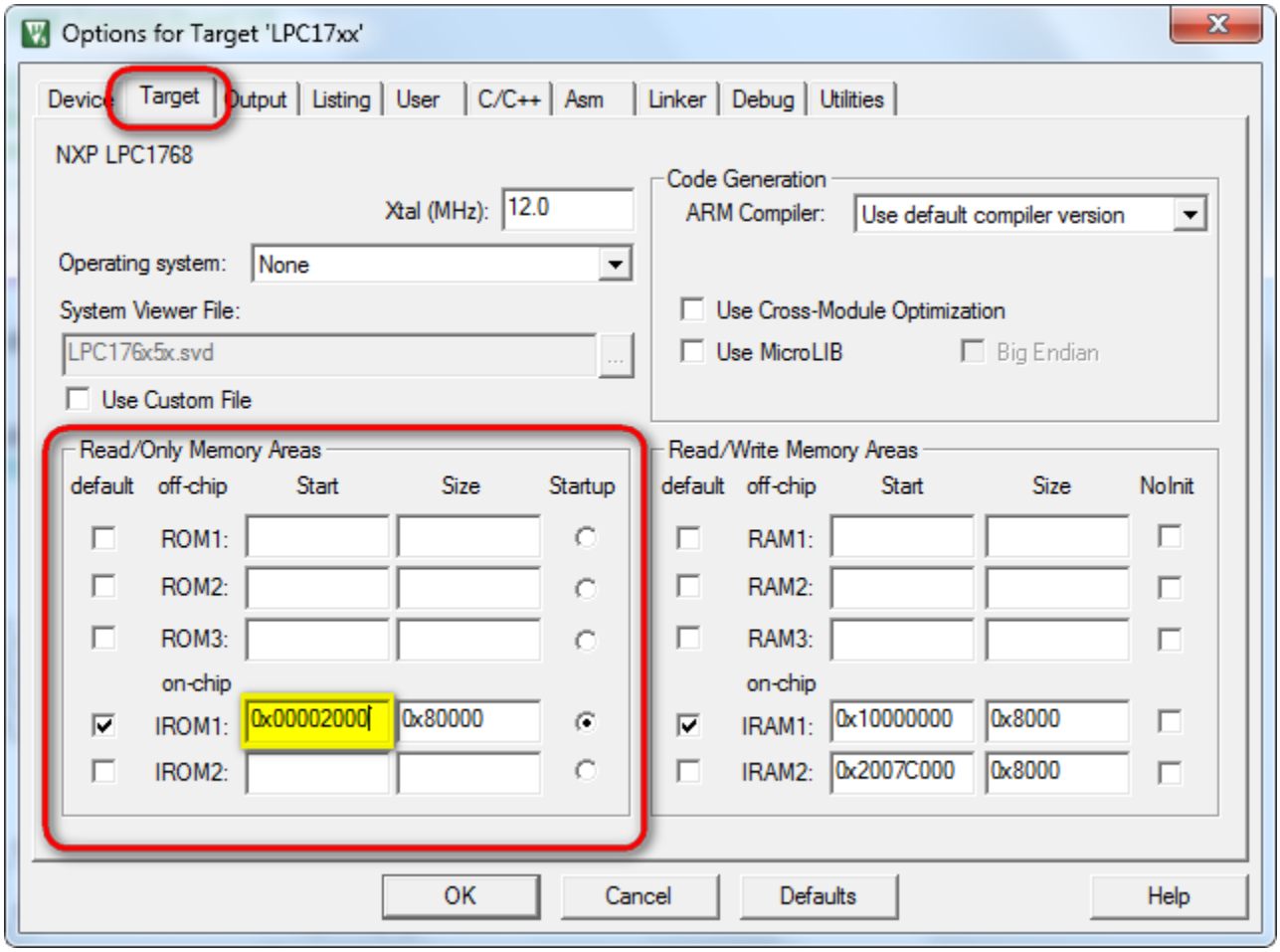
توسط پروگرام JTAG در حافظه میکروکنترلر پروگرم شود.

برای اینکه در استفاده های بعدی از بوت لودر برای پروگرم میکروکنترلر، کد بوت لودر توسط کد برنامه کاربر پاک نشود باید آدرس کد برنامه کاربر بعد از آدرس قرارگیری بوت لودر یعنی از آدرس 0x00002000 شروع شود. برای تنظیم آدرس شروع کد برنامه مطابق زیر عمل می‌شود:

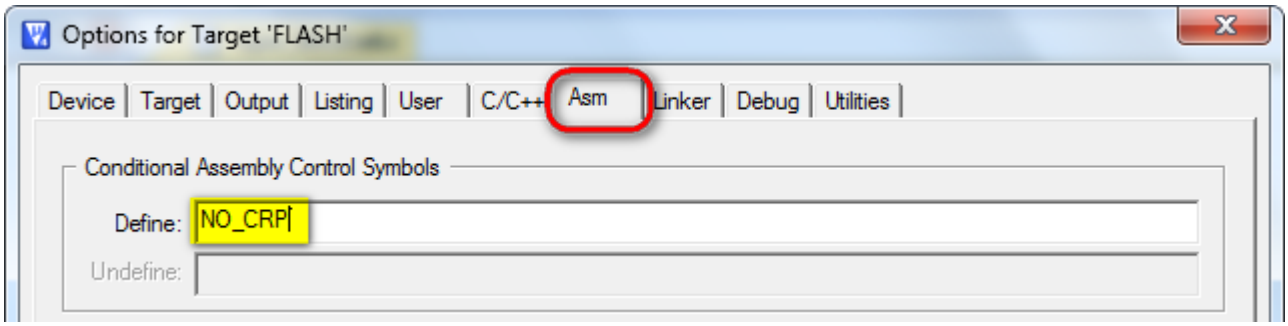
۱- از منوی Project گزینه‌ی 'Option for Target 'FLASH انتخاب شود



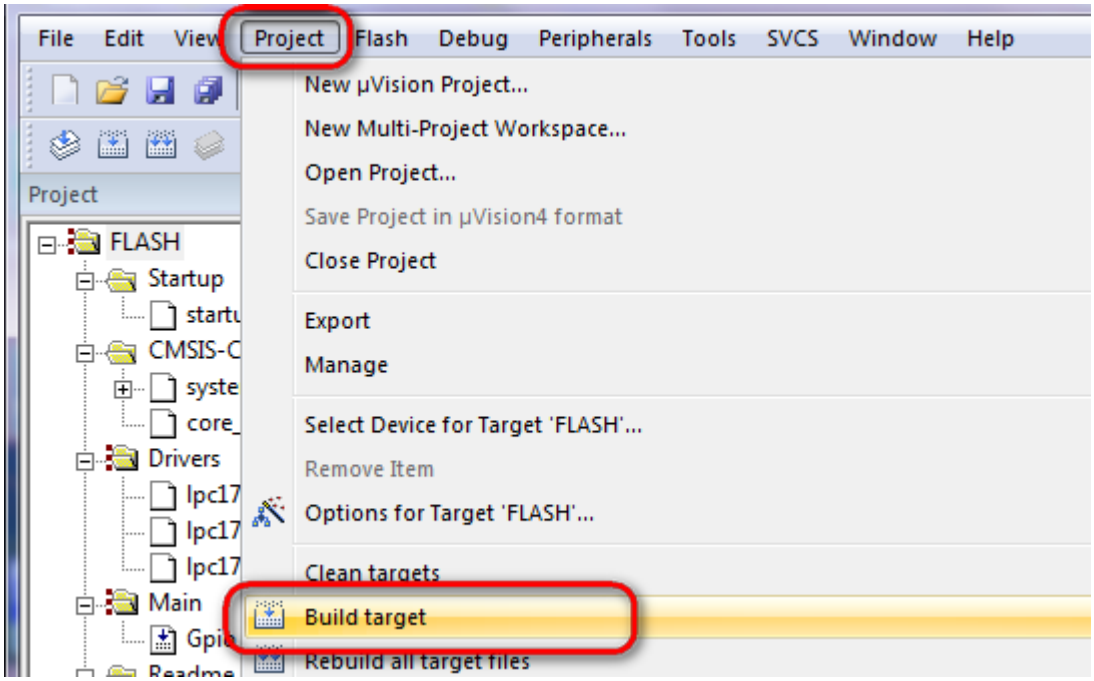
۲- به سربرگ Target رفته و از قسمت Read/Only Memory Areas بخش IROM1 مطابق زیر تنظیم می‌شود.



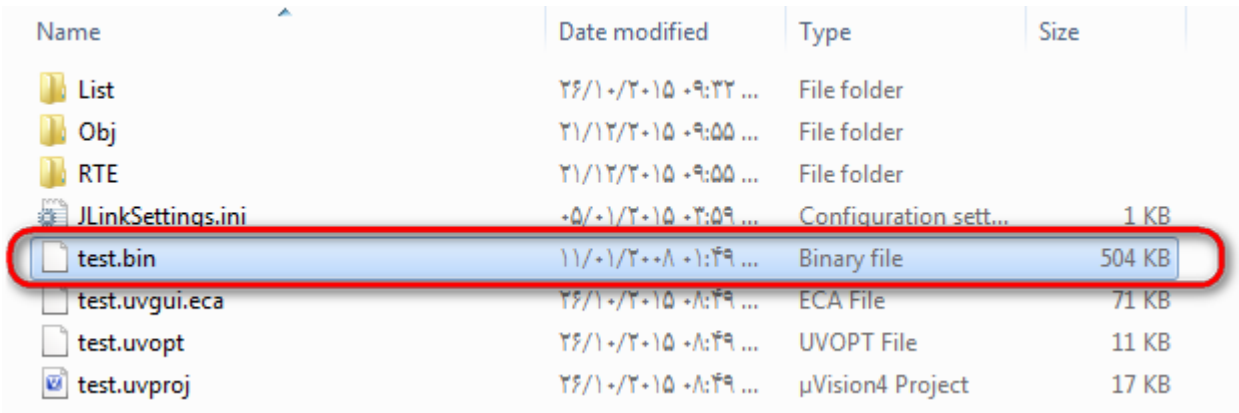
۳- سپس به سربرگ "Asm" رفته و مقابل کادر Define عبارت NO_CRP تایپ شود.



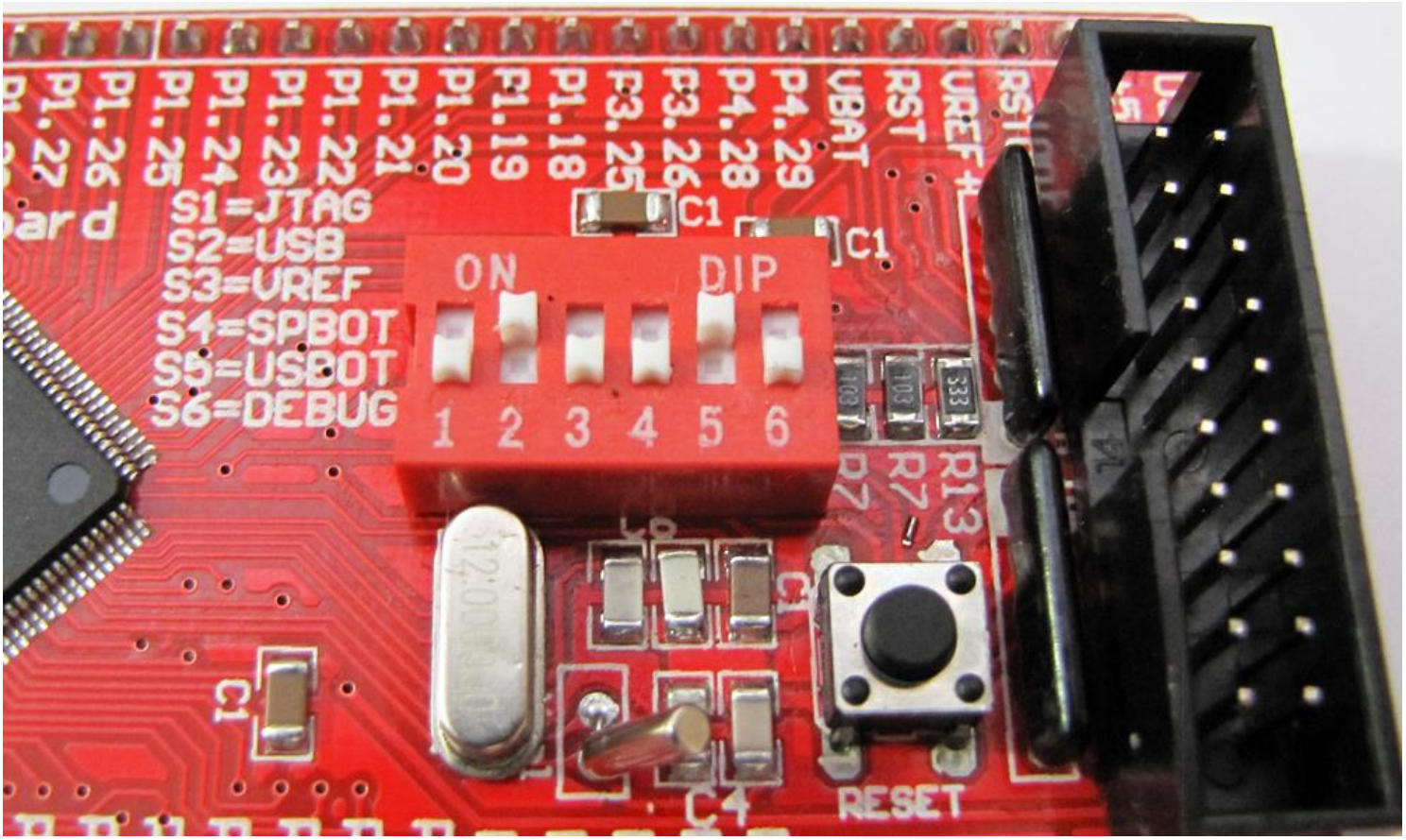
۴- بعد از اعمال تنظیمات فوق برنامه را کامپایل نموده



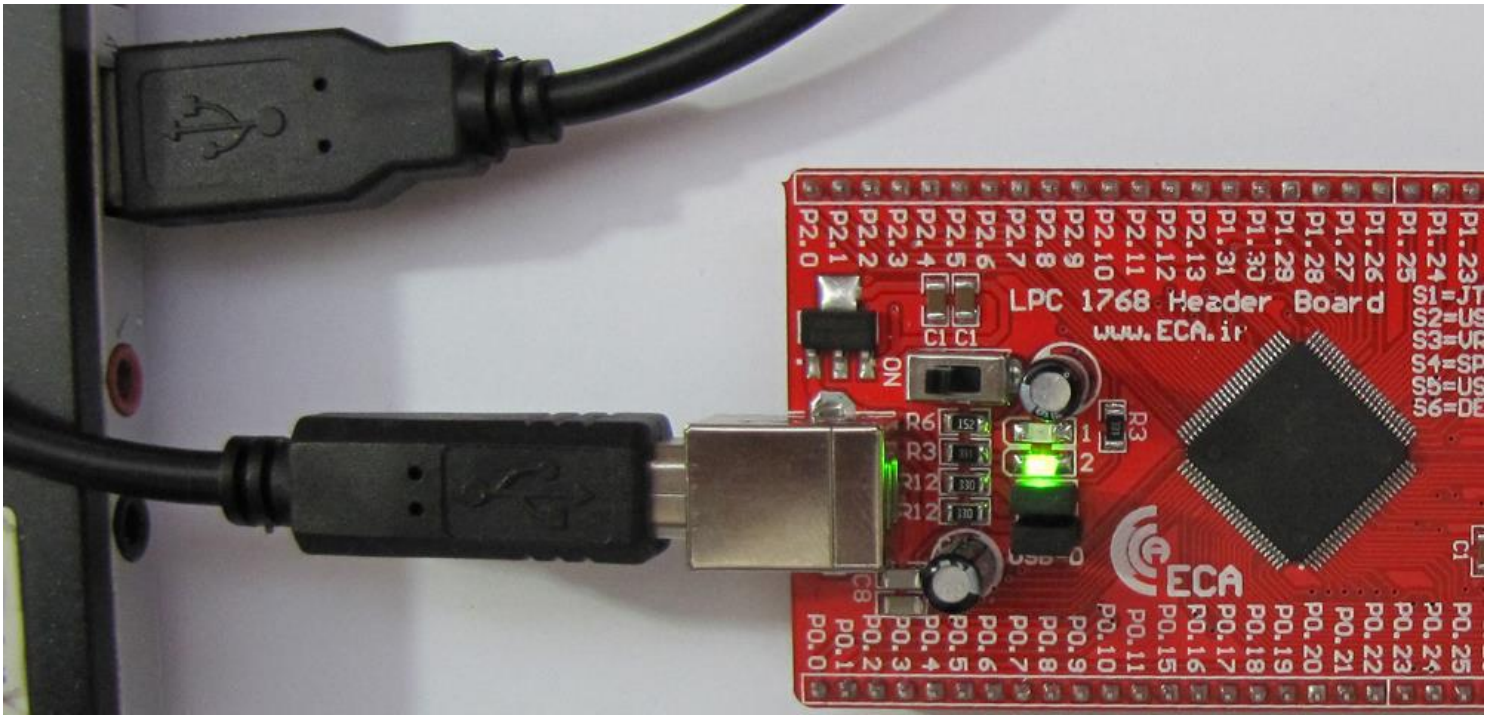
در صورتیکه برنامه بدون error کامپایل شود فایل باینری در مسیر پروژه ایجاد خواهد شد.



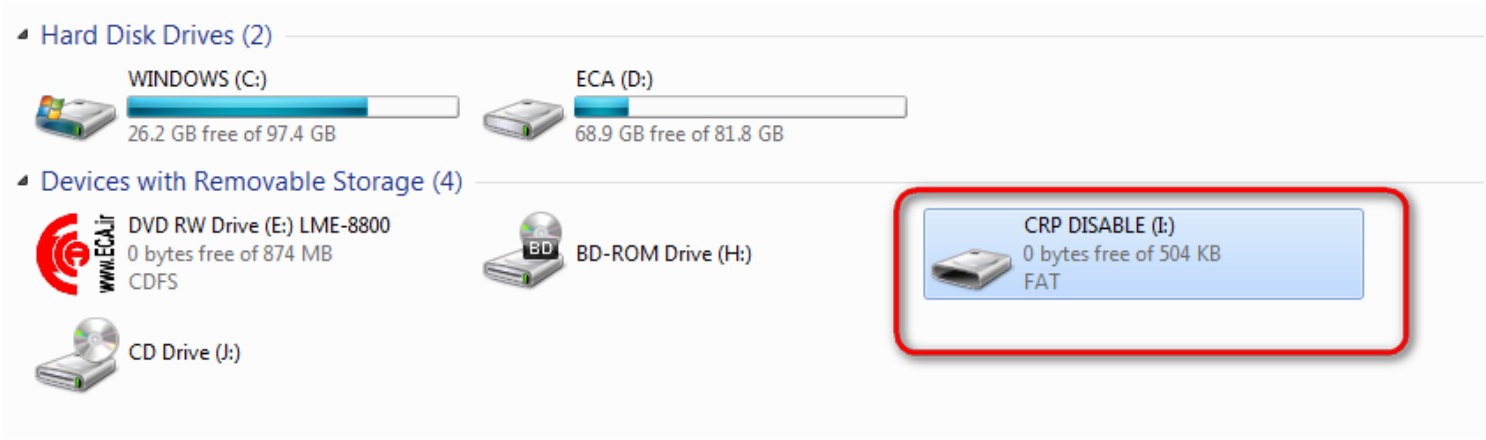
۵- حال دیپ سوئیچ های USB و USBOT قرار گرفته روی هدربرد را در حالت ON قرار داده شود.



۶- با کابل USB پورت usb میکروکنترلر را به پورت usb کامپیوتر وصل کرده سپس کلید تغذیه را ON نموده.



۷- با اتصال برد به کامپیوتر ؛ هدربرد به صورت یک درایو در My Computer ظاهر می‌شود:

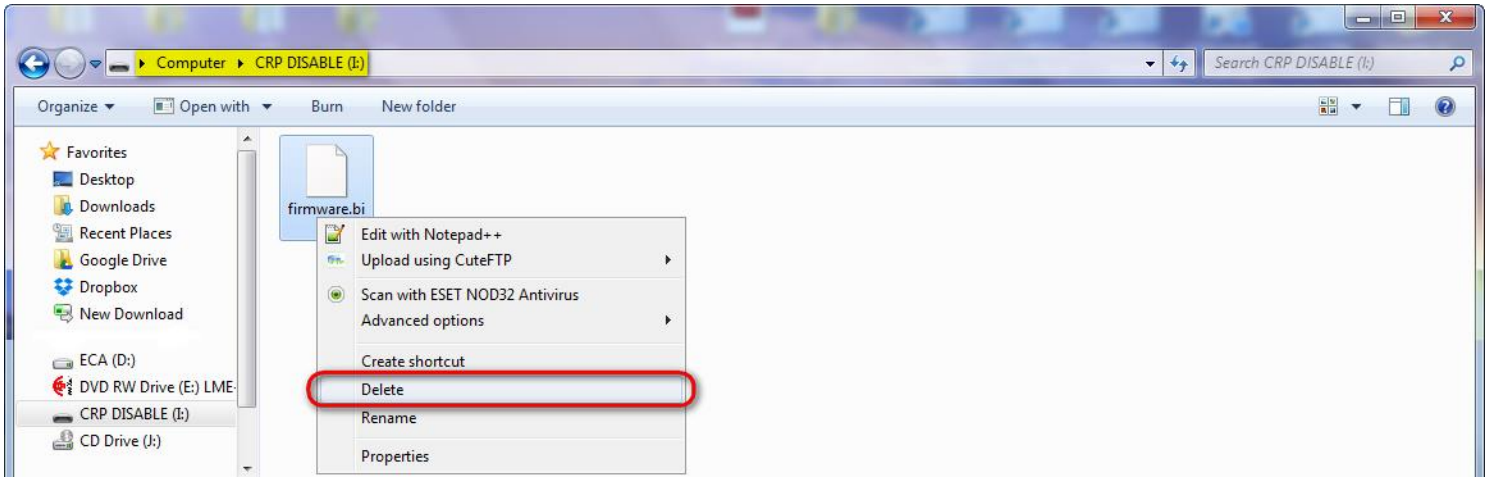


❊❊ اگر درایو شناخته نشد دو علت می‌تواند داشته باشد:

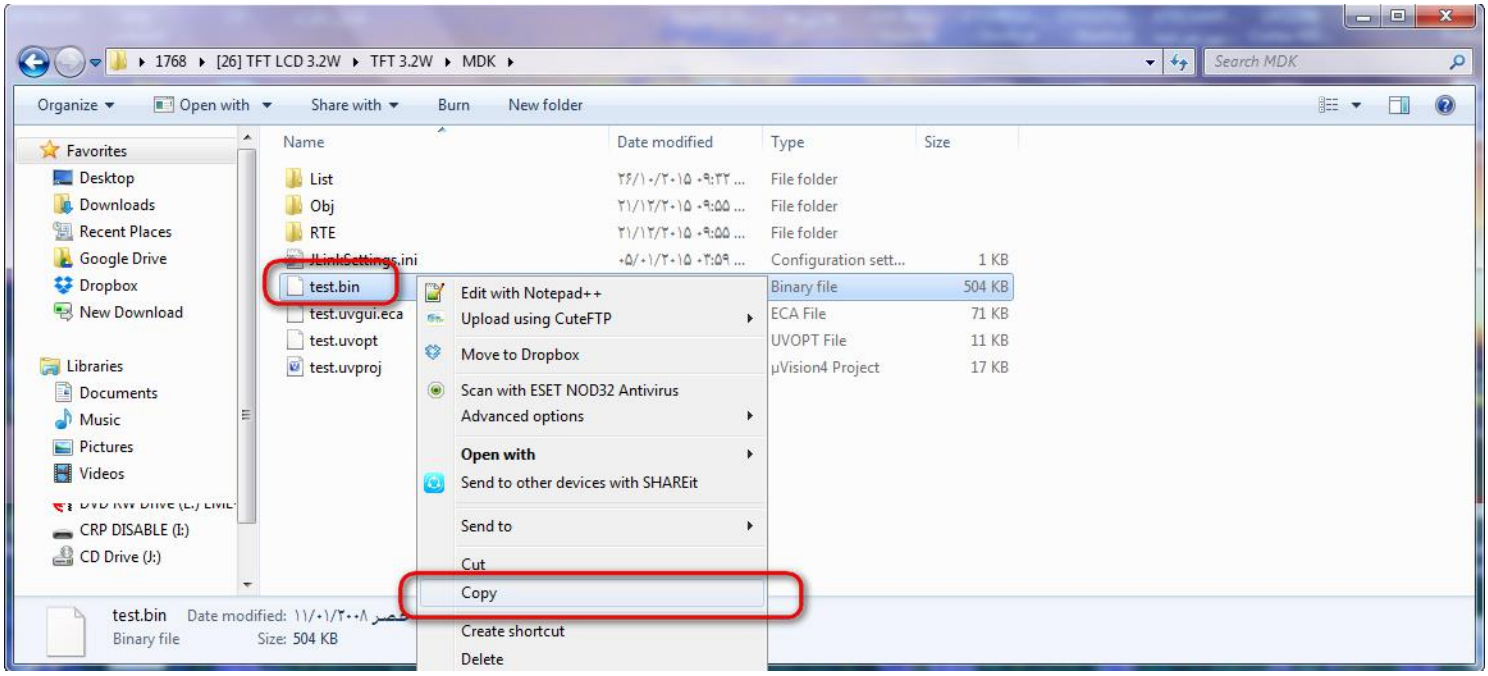
❊ کابل و یا پورت USB کامپیوتر خراب است.

❊ کد بوت لودر از حافظه میکروکنترلر پاک شده و دوباره باید توسط J-Link پروگرام شود.

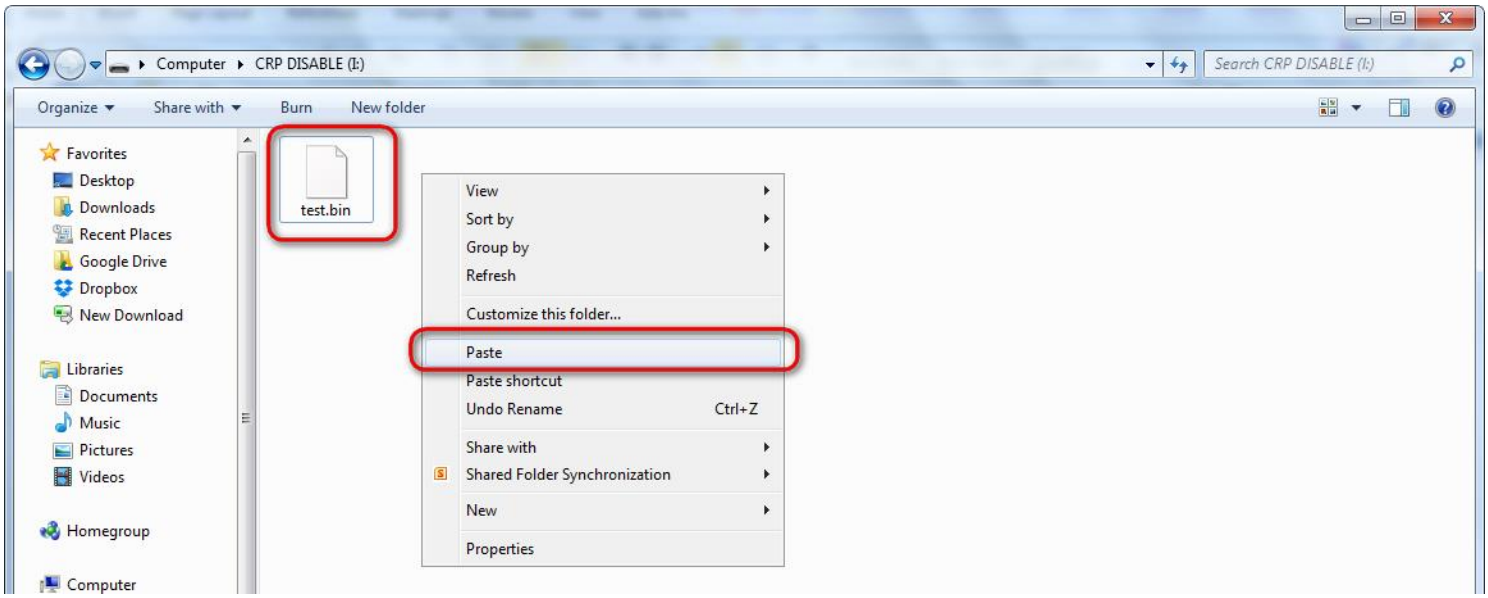
۸- درایو شناخته شده را باز کرده و فایل firmware.bin را از داخل آن حذف کرده



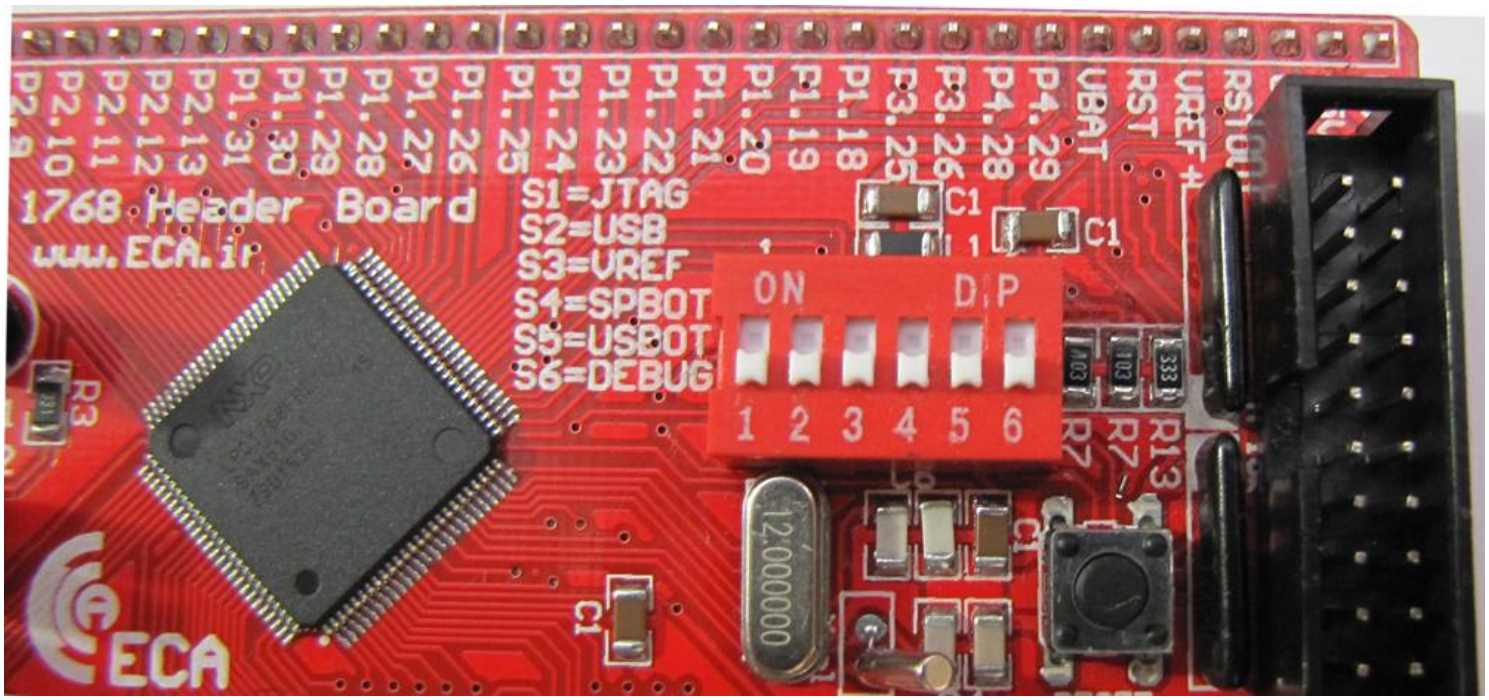
۹- فایل باینری تولیدی توسط کامپایلر Keil که در بالا توضیح داده شد از مسیر پروژه کپی کرده.



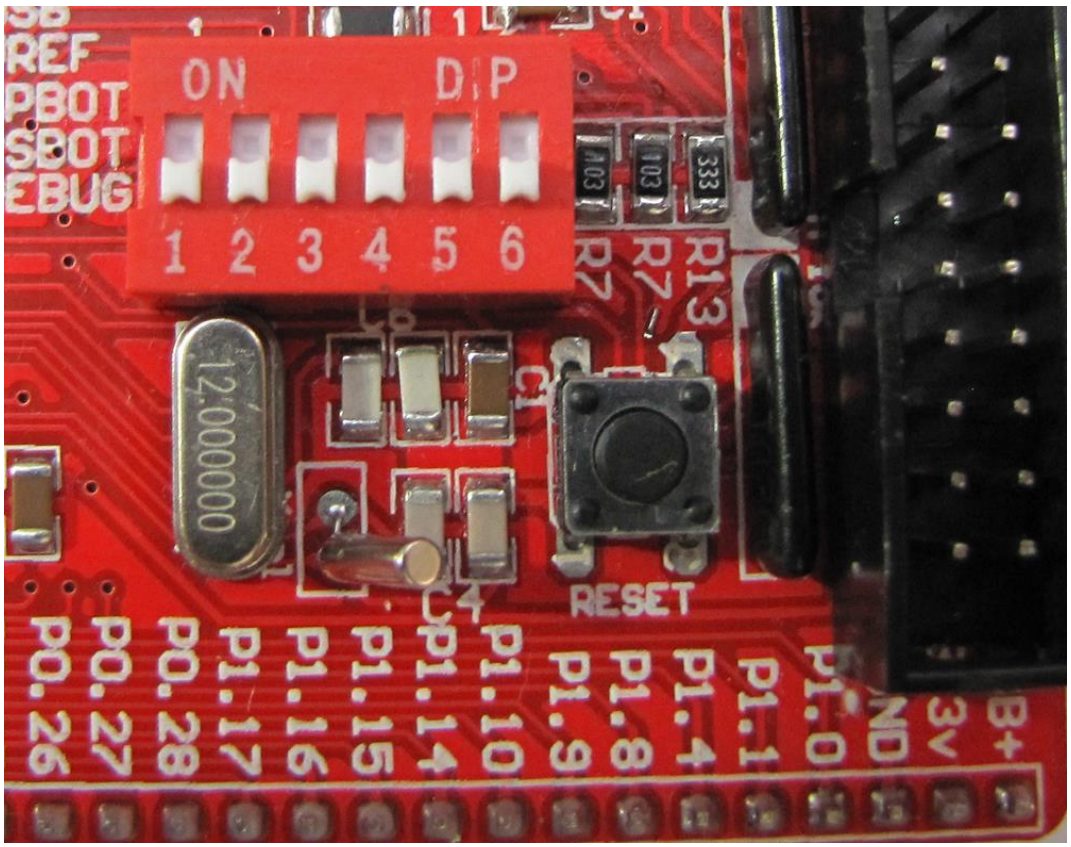
۱۰- و در مسیر درایو شناخته شده به Past می‌شود.



۱۱- اکنون پنجره را بسته و دیپ سوئیچ ها را به حالت اولیه برگردانده.



۱۲- بعد از ریست برد توسط دکمه‌ی فشاری Reset، برنامه به اجرا در می آید.

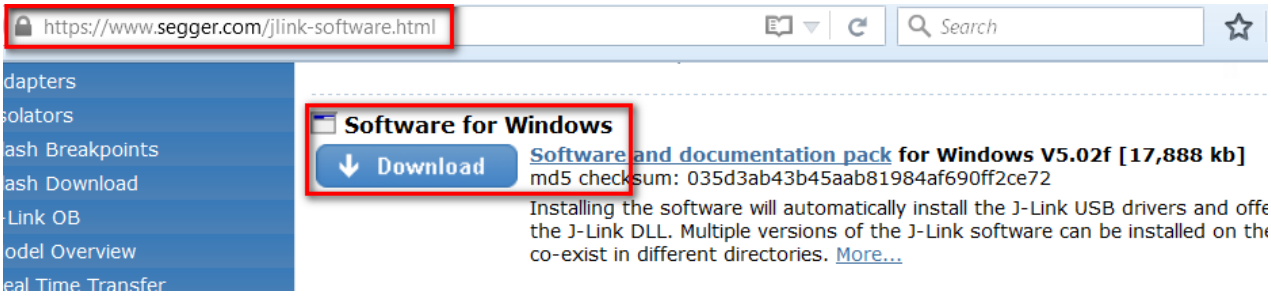


تذکر: لازم به ذکر است در صورتی مراحل بالا به درستی انجام نشود، برنامه به درستی کار نخواهد کرد.

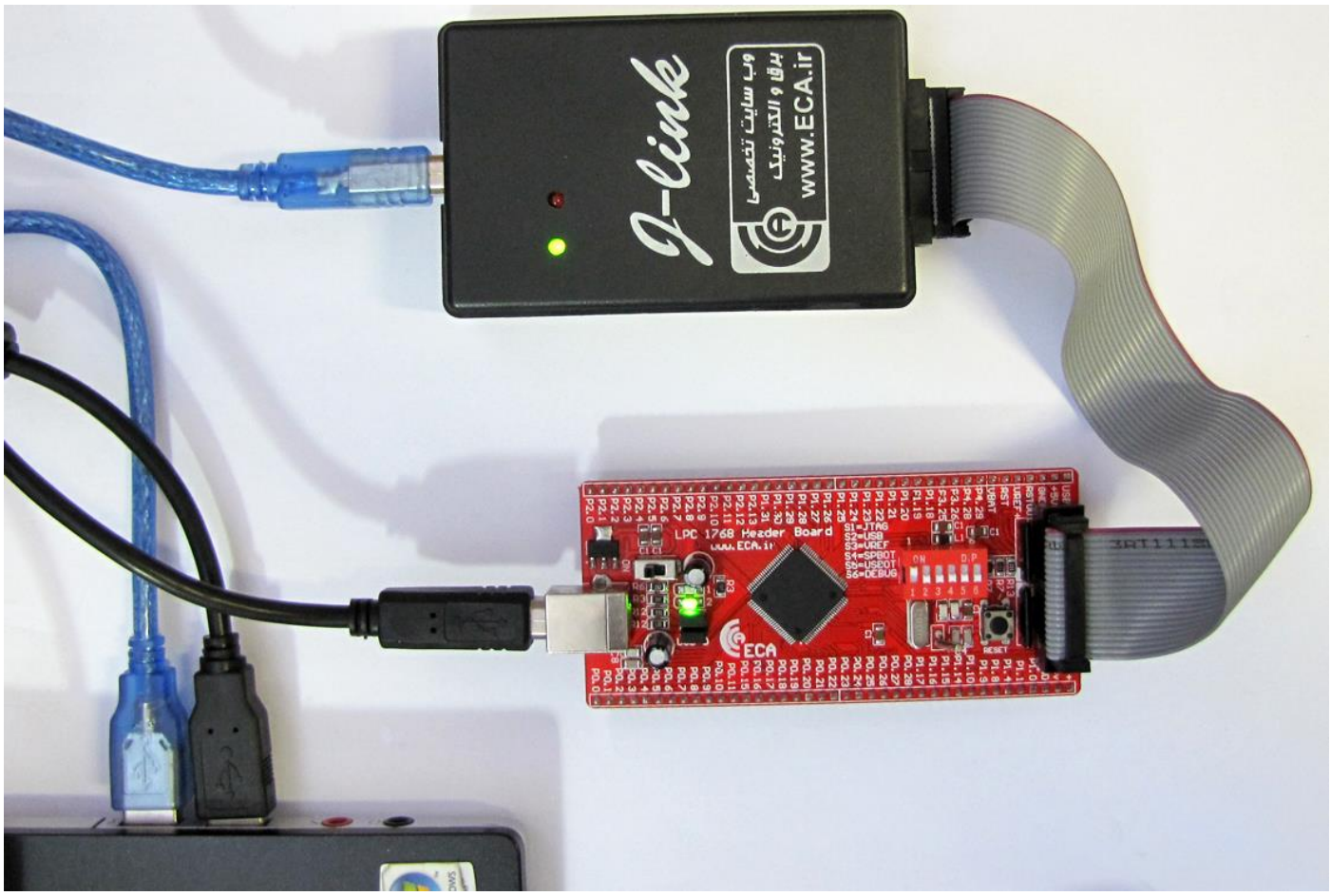
۳- آخرین نسخه نرم افزار J-Link را از دیسک همراه برد نصب نمائید.

DVD&CD:\Tools\ Setup_JLink_V494j.zip

یا آخرین نسخه ی آن را از سایت Segger دریافت و نصب نمایید.

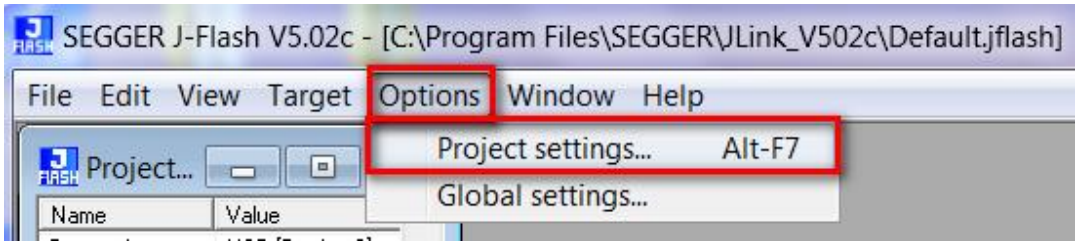


۴- هدربرد را به پروگرامر J-Link متصل نموده و پروگرامر J-Link هم به کامپیوتر(در اینجا کابل آبی رنگ)، لازم بذکر است بایستی منبع تغذیه‌ی هدربرد توسط USB تامین شود(در اینجا کابل مشکی رنگ).

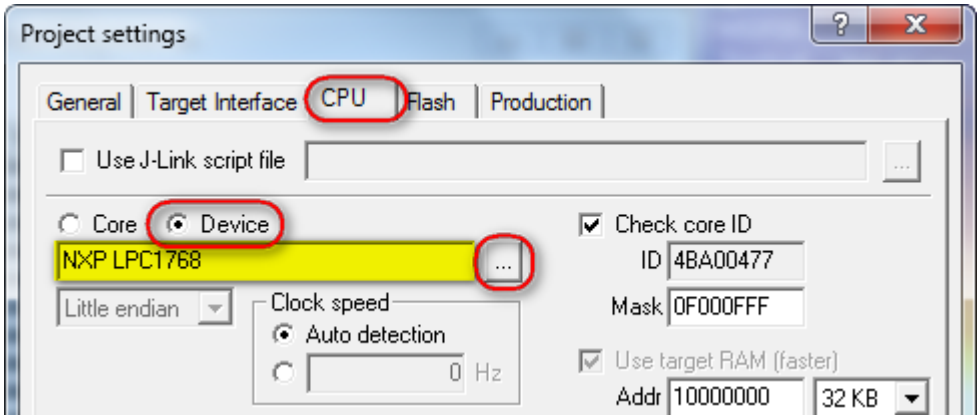


۵- نرم افزار J-Flash را اجرا نمائید.

۶- از بخش Options گزینه ی Project Settings را انتخاب کنید.

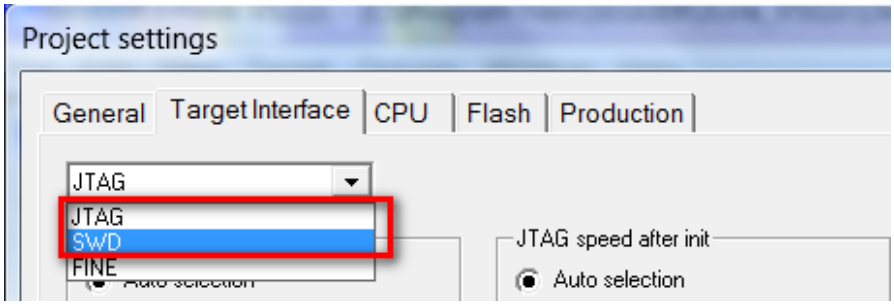


۷- از سربرگ CPU تیک گزینه ی Device را زده و میکروکنترلر NXP LPC1768 را انتخاب کنید.

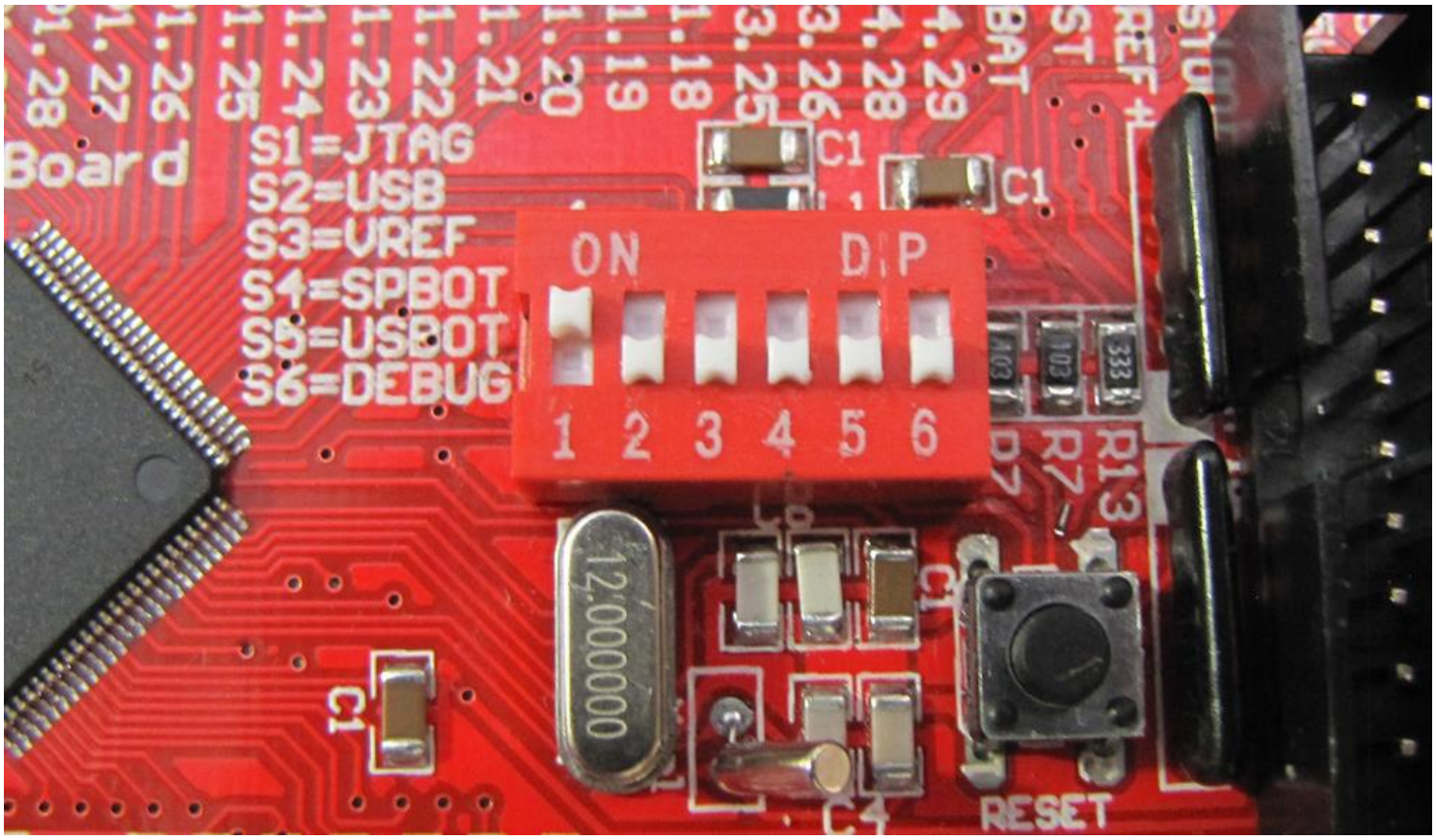


۸- حال از سربرگ Target Interface از لیست باز شو هم میتوان گزینه ی JTAG را انتخاب کرد و هم میتوان گزینه ی SWD را انتخاب کرد.

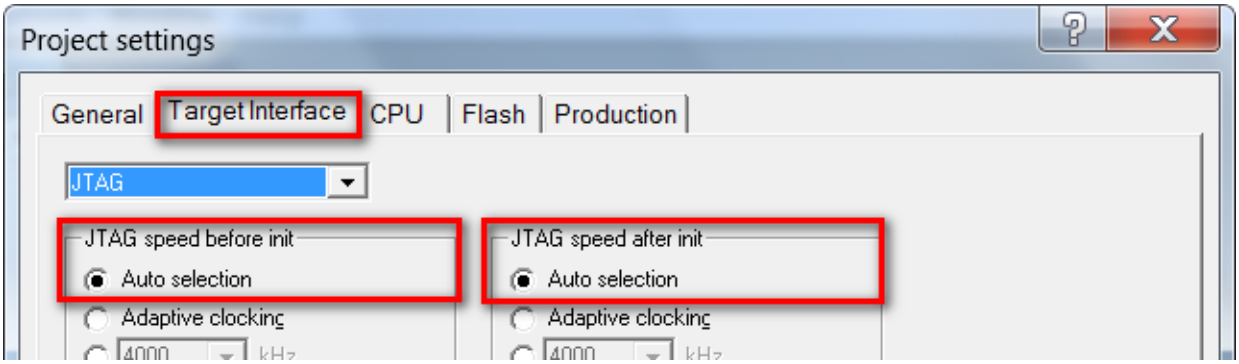
فرق پروتکل JTAG و SWD در این است که در روش SWD فقط از ۲ پایه و در روش JTAG از ۶ پایه جهت پروگرام کردن میکرو استفاده میشود ؛ مزیت روش SWD در اختیار گذاشتن پایه های آزاد بیشتری برای کاربر میباشد. که البته در این برد احتیاجی به رعایت این موضوع نمیباشد.



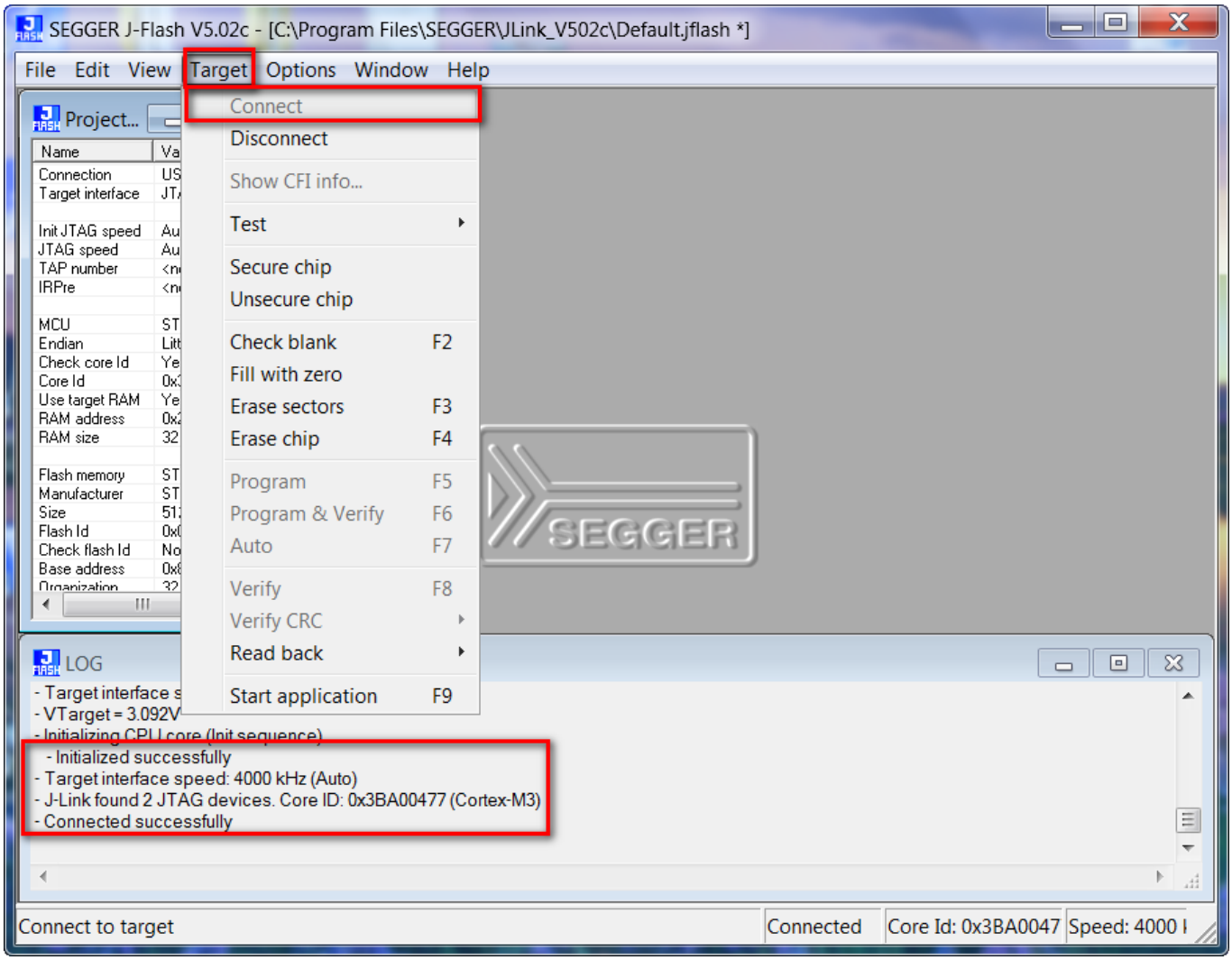
۹- میبایست دیپ سوئیچ JTAG جهت قابلیت پروگرام، دیباگ از طریق پروگرامر J-Llink مطابق شکل زیر فعال باشد.(مطابق جدول راهنمای دیپ سوئیچ ها)



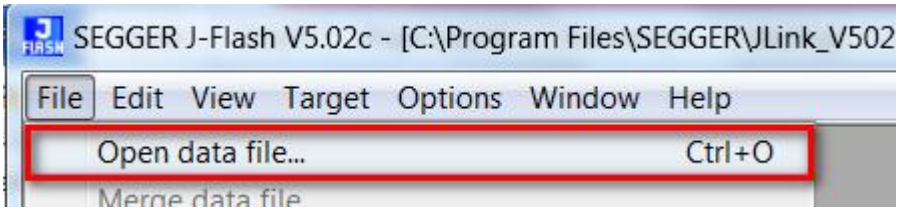
۱۰- کلاک را روی حالت Auto تنظیم نمائید.



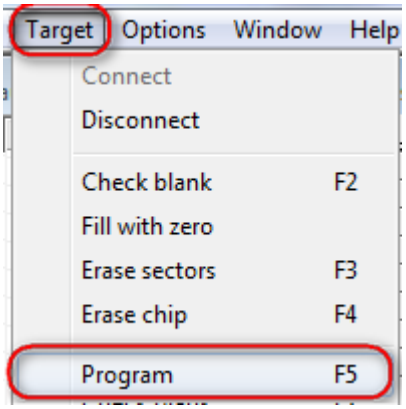
۱۱- در حالیکه تغذیه برد را وصل کرده اید از تب Target گزینه Connect را بزنید .در صورتیکه عملیات اتصال به درستی انجام شود پیغام Connected successfullyدر بخش LOG نمایش داده می شود.



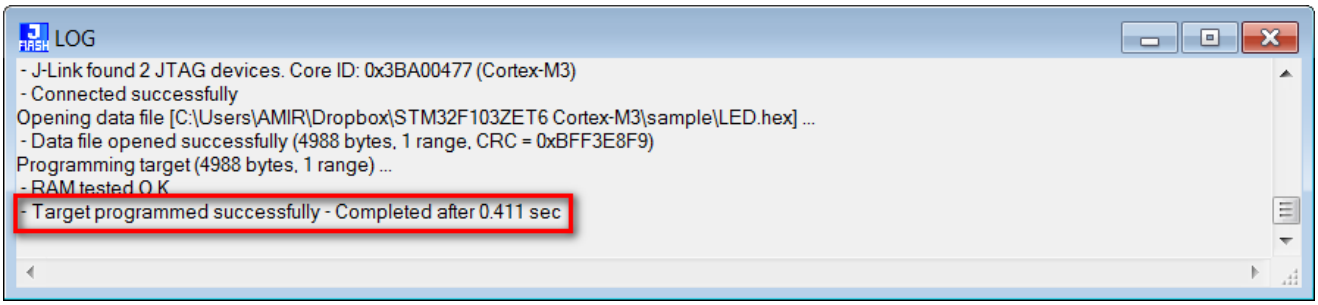
۱۲- حال می‌توانید از بخش File , فایل هگز یا Bin مورد نظر را از طریق گزینه ی Open data file وارد برنامه نمایید.



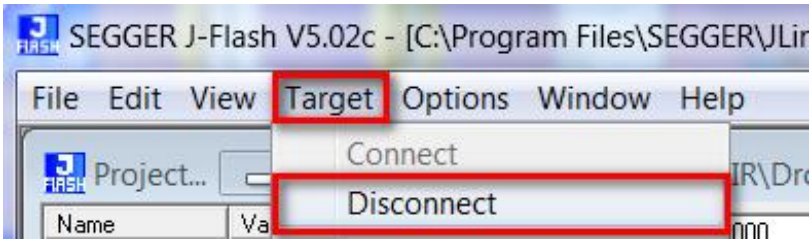
۱۳- از منوی Target با گزینه Program یا فشردن کلید F5 آن را بر روی میکروکنترلر پروگرام نمائید.



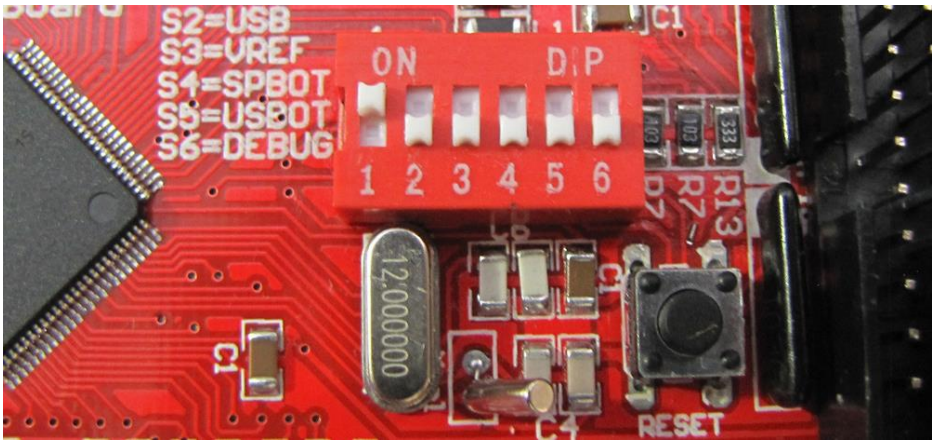
درصورت موفق آمیز بودن پیغام Target programmed successfully-Complated نمایش داده میشود.



۱۴- از تب Target گزینه Disconnect را بزنید.

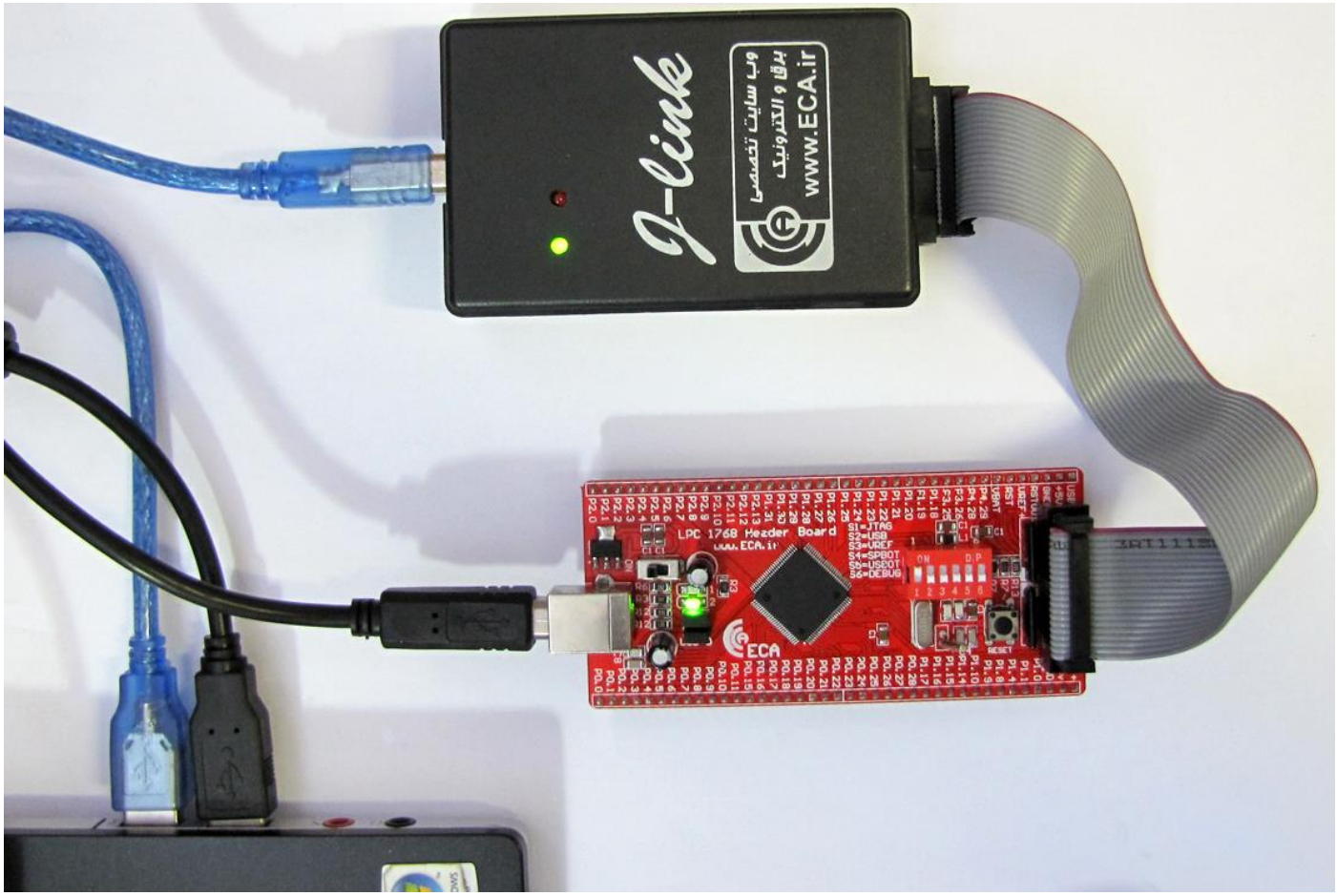


۱۵- بعد از ریست برد توسط دکمه‌ی فشاری Reset، برنامه به اجرا در می آید.

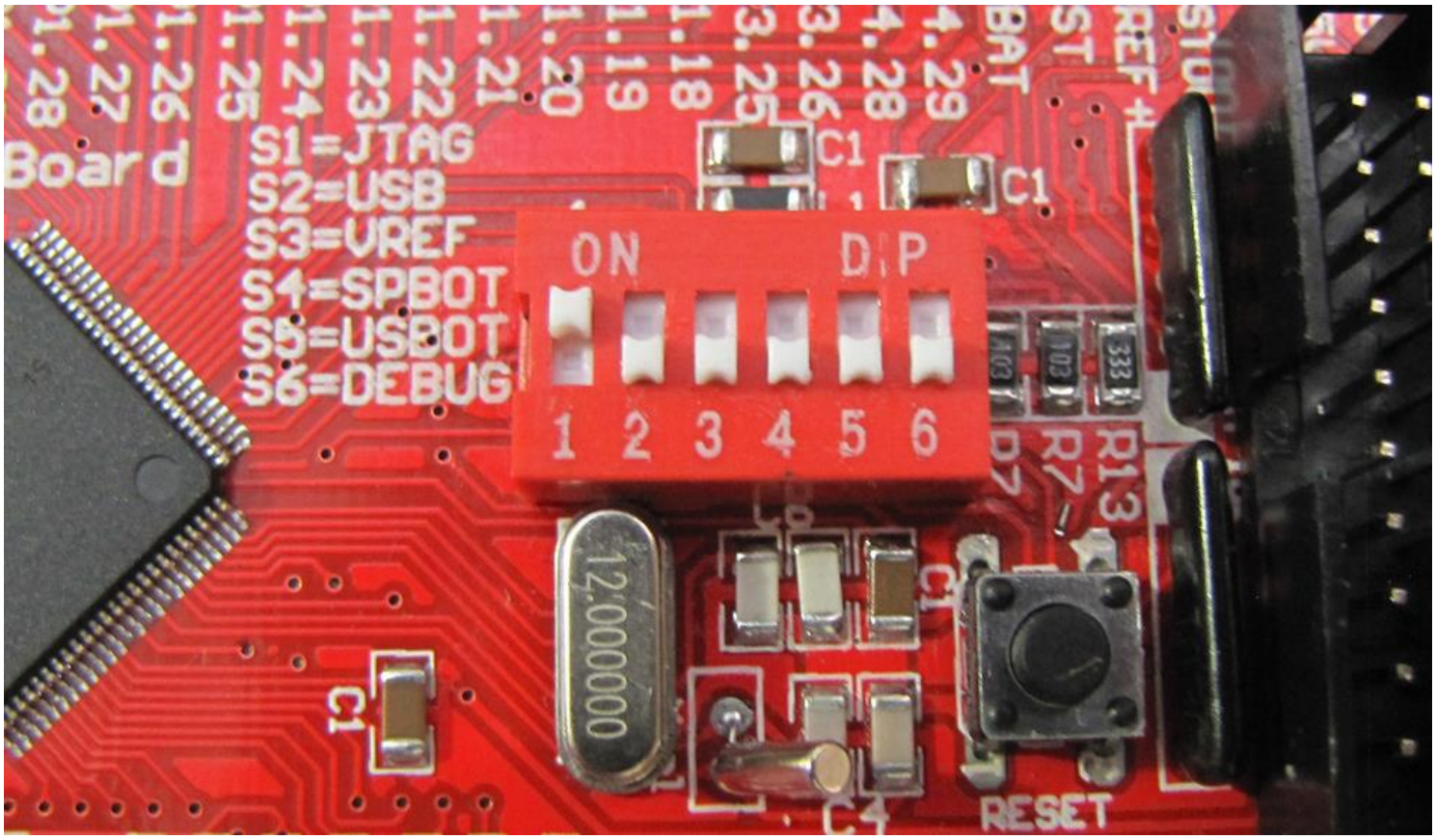


پروگرام کردن برد توسط J-Link و کامپایلر Keil

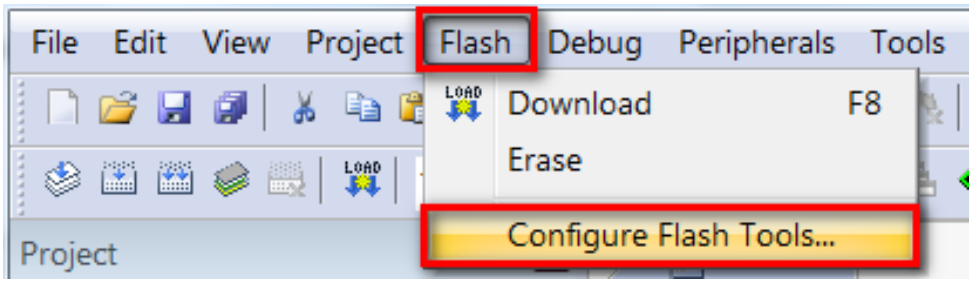
۱- هدربرد را به پروگرامر J-Link متصل نموده و پروگرامر J-Link هم به کامپیوتر(در اینجا کابل آبی رنگ)، لازم بذکر است بایستی منبع تغذیه‌ی هدربرد توسط USB تامین شود(در اینجا کابل مشکی رنگ).



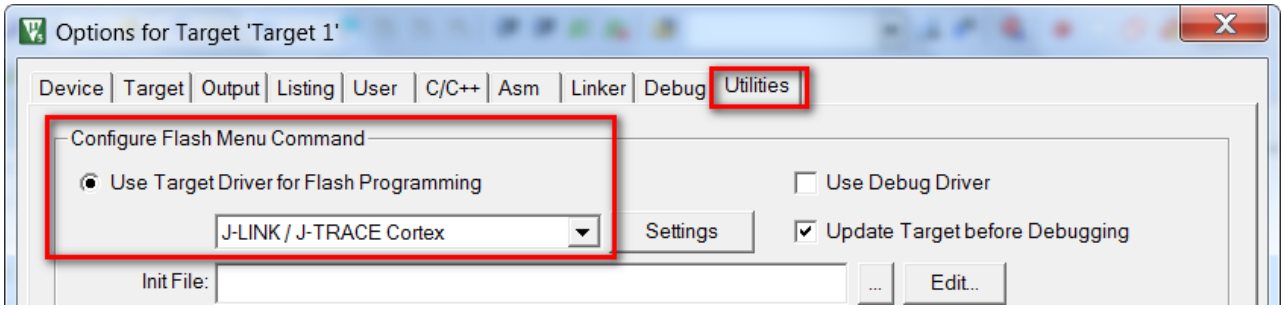
۲- میبایست دیپ سوئیچ JTAG جهت قابلیت پروگرام، دیباگ از طریق پروگرامر J-Llink مطابق شکل زیر فعال باشد.(مطابق جدول راهنمای دیپ سوئیچ ها)



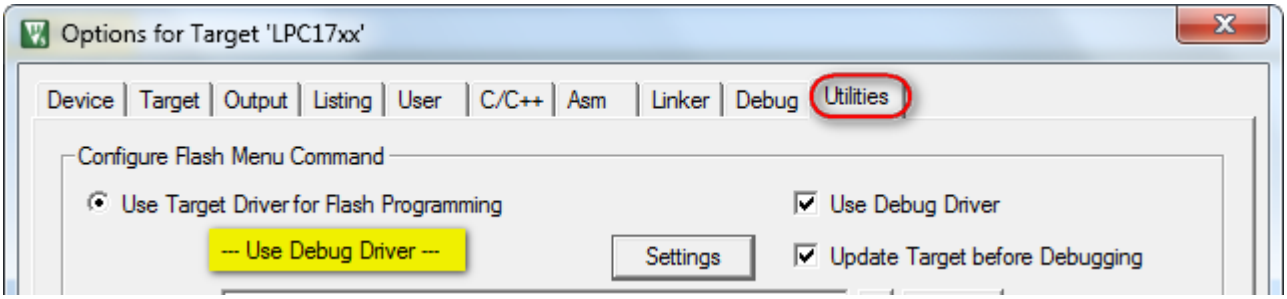
۳- از منو گزینه Flash را کلیک کرده و از آنجا گزینه ی ...Configure Flash Tools را انتخاب می‌کنیم:



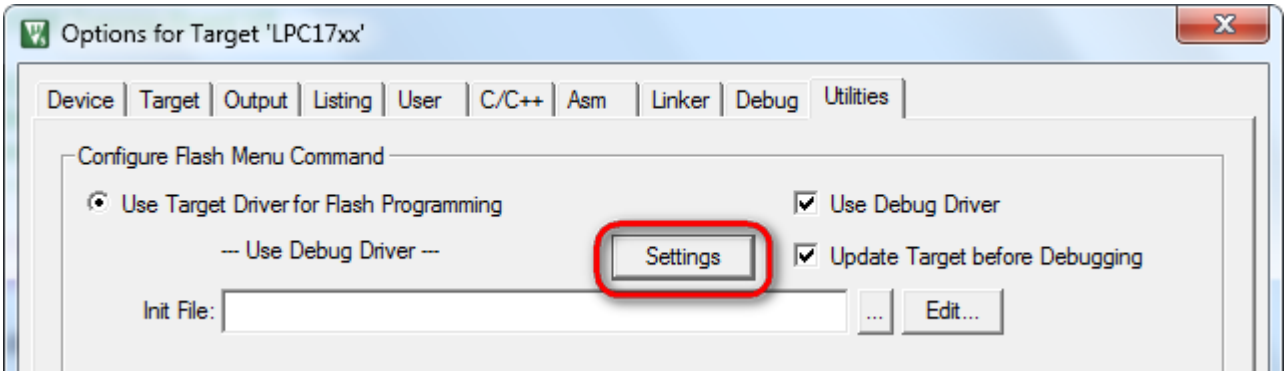
۴- سپس از سربرگ Utilities و از قسمت Configure Flash Menu Command گزینه ی Use Target Driver for Flash Programming را تیک زده و از آنجا نوع پروگرامر مورد استفاده را که J-Link میباشد را انتخاب می‌کنیم.



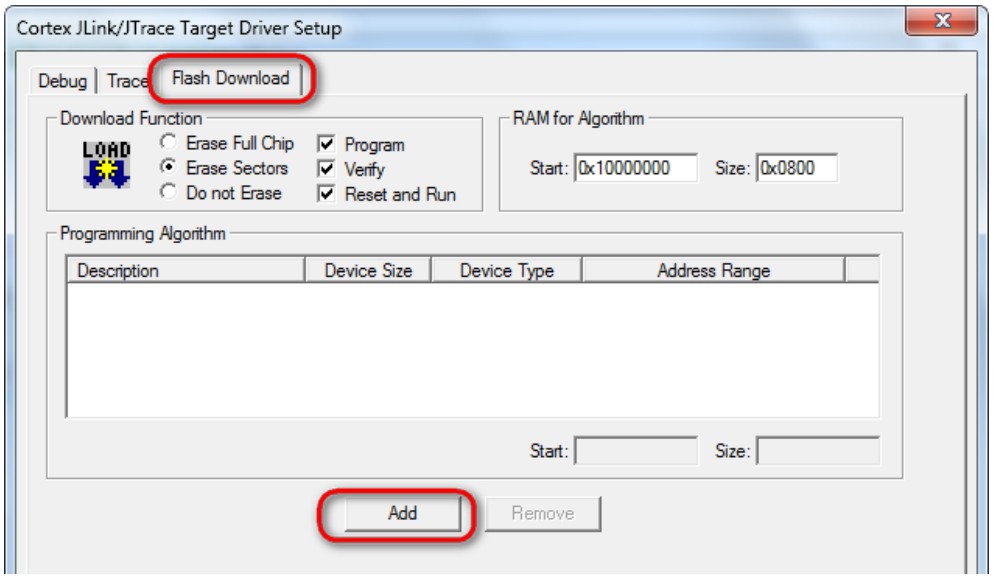
در بعضی مواقع Keil بصورت خودکار پروگرامر را شناسایی کرد و بصورت زیر نمایش میدهد



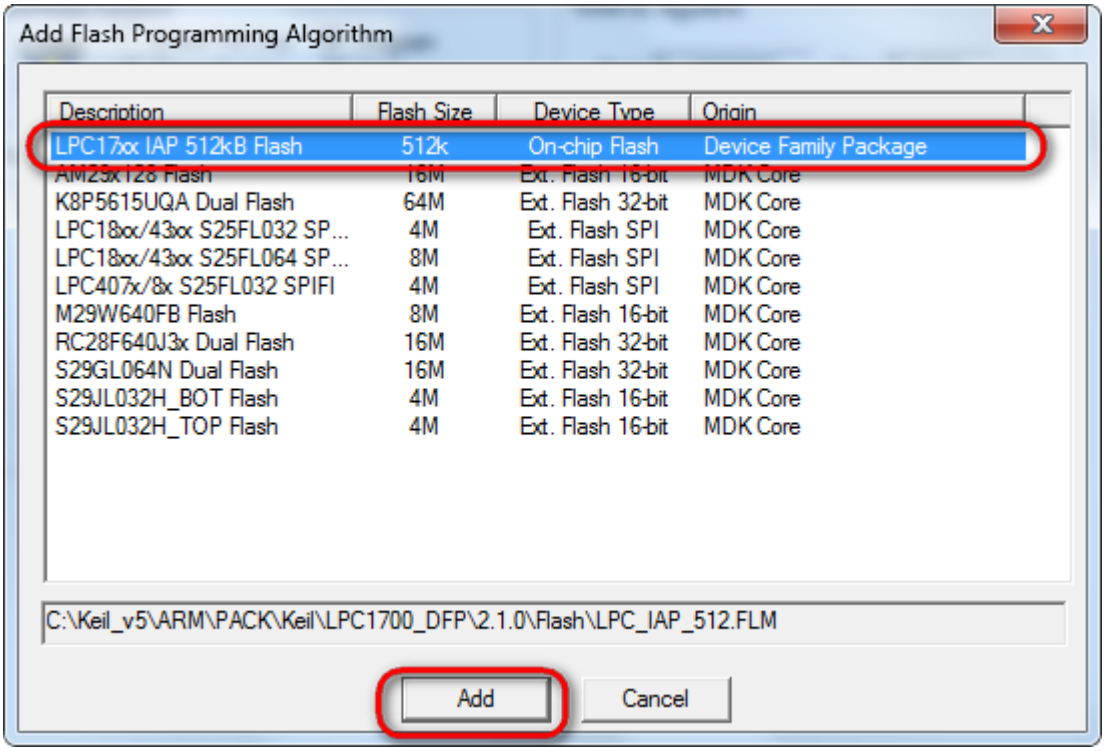
۵- حال بر روی Setting کلیک نموده



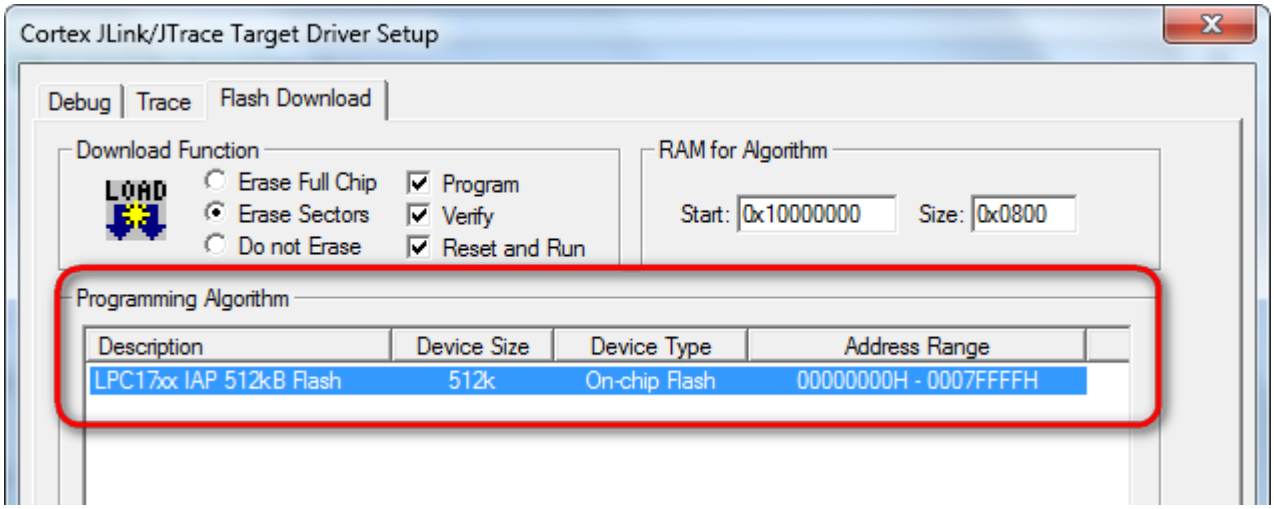
۶- سپس جهت اضافه کردن مقدار حافظه و دیگر مشخصات میکروی متصل از سربرگ Flash Download روی دکمه‌ی Add کلیک شود.



۷- بعد از انتخاب آیتم LPC 17xx روی Add کلیک شود

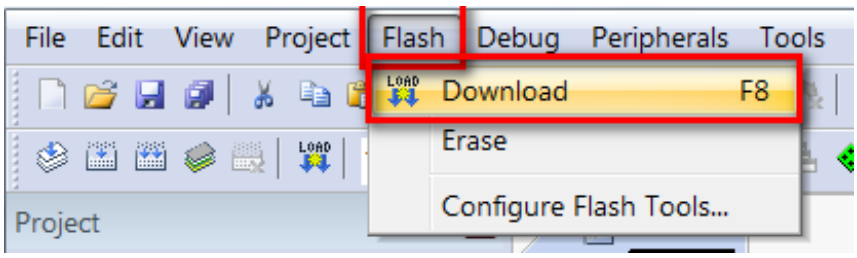


۸- مشاهده میشود این میکرو به لیست اضافه شد



و در نهایت بر روی Ok کلیک میکنیم.

۹- حال نوبت به قدم نهایی پروگرام کردن میکرو میرسیم برای اینکار از منوی Flash گزینه Download را میزنیم.

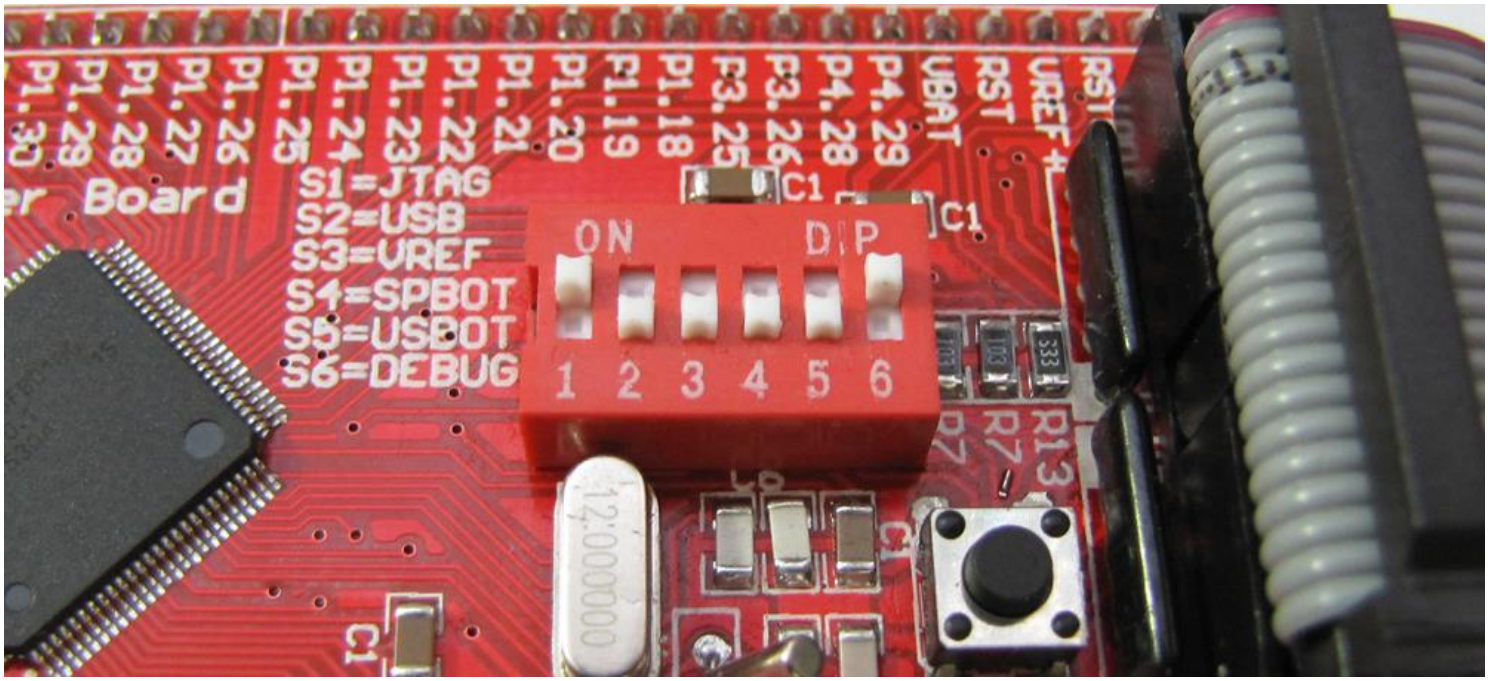


چند لحظه منتظر می مانیم تا عملیات انتقال انجام شود. پیامی مبنی بر موفق آمیز بودن عملیات در قسمت Log برنامه نمایش داده خواهد شد: که خط اول موفق آمیز بودن عملیات Erase کردن را نشان میدهد. خط دوم موفق آمیز بودن عملیات Program کردن و در نهایت خط سوم موفق آمیز بودن چک مجدد کدهای انتقال داده شده به میکرو را نمایش خواهد داد.

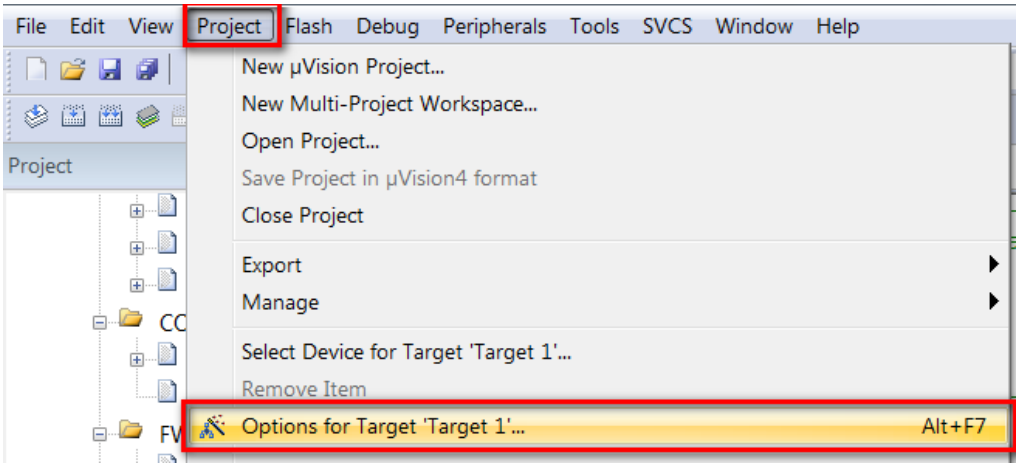
به همین سادگی توانستیم بدون اعمال تنظیمات سخت افزاری و نرم افزاری طولانی برد NXP LPC1768 را پروگرام کنیم. در قسمت های بعدی طریقه ی دیباگ کردن توسط J-Link و Keil را آموزش خواهیم داد.

دیباگ کردن برد توسط کامپایلر Keil و پروگرامر J-Link

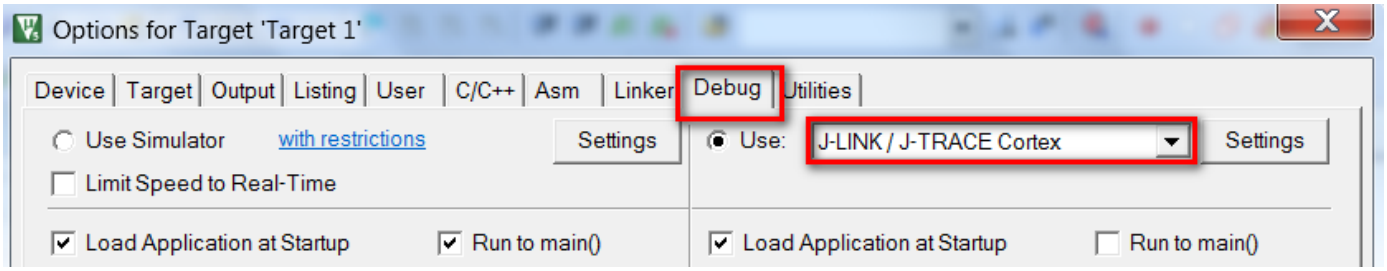
۱- ابتدا دیپ سوئیچ DEBUG فعال شود.



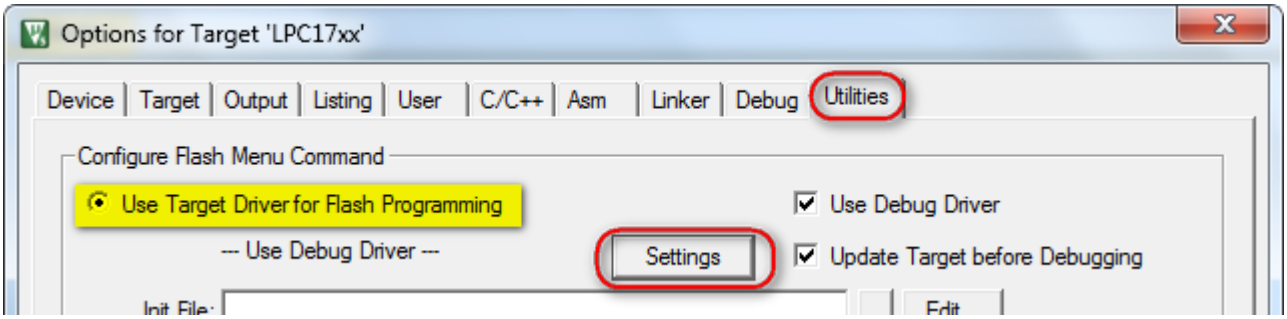
از منوی Project گزینه ی Options for Target را انتخاب نموده:



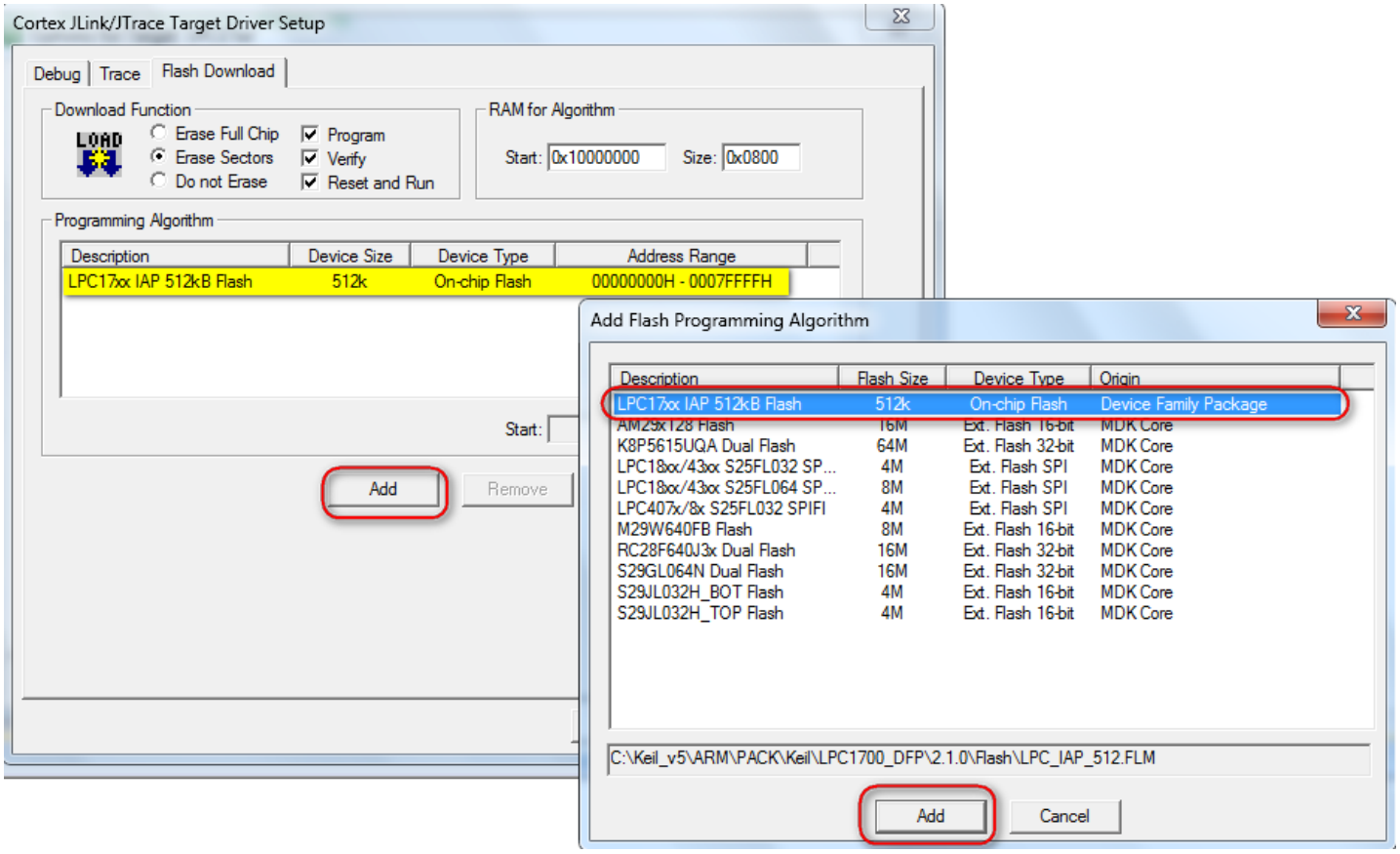
سپس از صفحه ی باز شده سربرگ Debug را انتخاب میکنیم سپس تیک Use را میزنیم و سپس نوع پروگرامر متصل به آن که J-Link است را انتخاب خواهیم کرد.



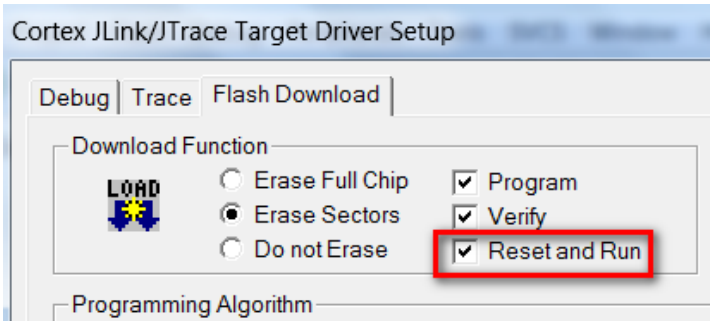
سپس به سربرگ Utilities رفته و بعد از زدن تیک گزینه ی Use Target Driver for Flash Programming نوع پروگرامر متصل به برد که در اینجا J-Link است را انتخاب میکنیم.سپس روی گزینه Settings کلیک میکنیم.



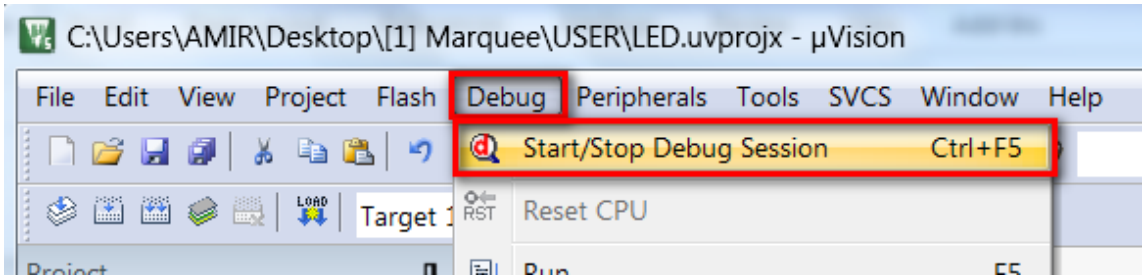
هم اکنون در صفحه‌ی جدید بر روی Add کلیک کرده و از پنجره ی باز شده میکرو مورد نظر خود (در اینجا LPC17xx) را انتخاب میکنیم و مجددا بر روی Add کلیک نمایید.



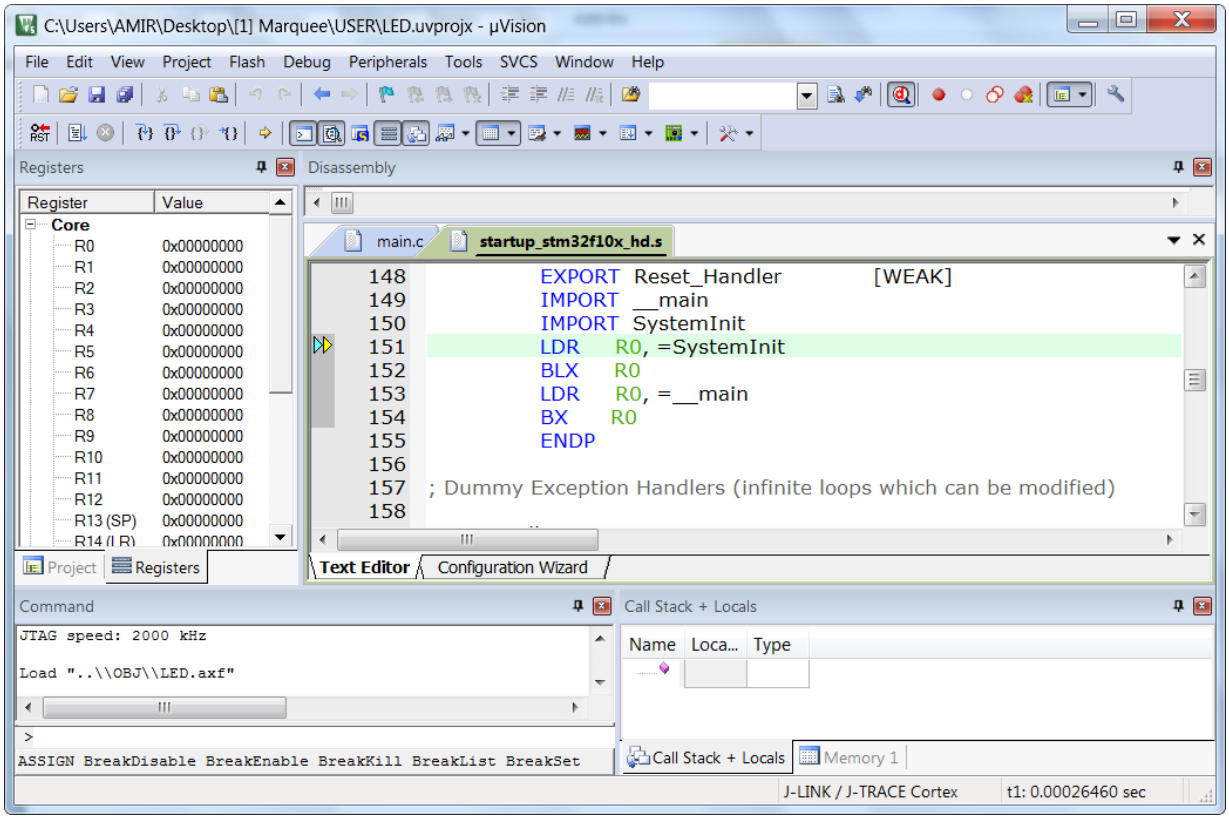
سپس برگه را بسته و همچنین تیک گزینه ی Reset and Run را میزنیم.سپس OK را میزنیم.



حال از منو Debug گزینه ی Start/Stop Debug Session را بزنید.

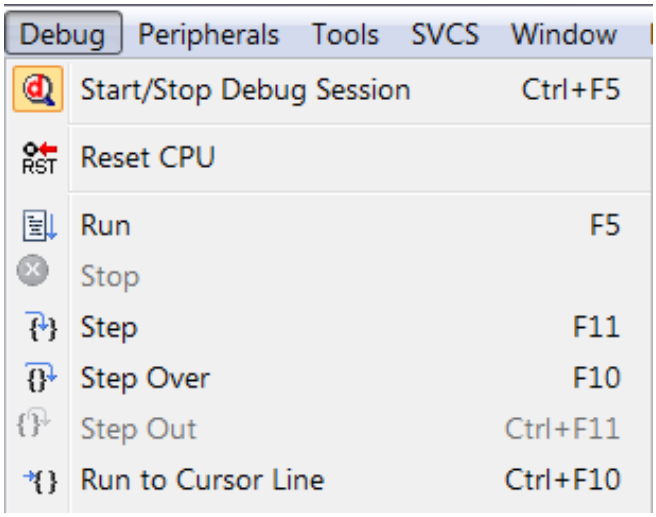


چند لحظه منتظر بمانید تا برنامه با میکرو از طریق پروگرامر J-Link ارتباط برقرار کرده و وارد حالت دیباگ شود.



هم اکنون برنامه جهت دیباگ کردن آماده است. حال برای انجام عملیات دیباگ کردن از منوی Debug استفاده میکنیم که بطور مختصر عملکرد هر یک از قسمت های اصلی را توضیح خواهیم داد.

Start/Stop Debug Session: برنامه و میکرو را جهت انجام عملیات دیباگ آماده میکنید



Reset CPU: بازنشانی میکرو و اجرای برنامه از اول

Run: اجرای کل برنامه

Stop: متوقف کردن برنامه

Step: خط مورد نظر را اجرا میکند

Step Over: اجرای یک دستور یا سطر بدون داخل شدن به توابع

Step Out: اجرای کامل تابع و برگشتن به تابع فراخوانی کننده

Run to Cursor Line: اجرای برنامه تا خطی که نشانگر موس در آن قرار دارد