

mbed Tutorial 2: Lập trình ứng dụng cho mạch VBLUno với GNU ARM Eclipse

Nội dung trình bày:

- Cài đặt môi trường làm việc.
- Xây dựng một project cho VBLUno với GNU ARM Eclipse.

1. Cài đặt môi trường làm việc

Nhằm nâng cao hiểu biết làm việc của developers, việc sử dụng các môi trường phát triển (IDE) là điều cần thiết. Để phát triển các ứng dụng nhúng cho IoT, cần thiết là phát triển ứng dụng với bộ mã VBLUno của VNGIoTLab, chúng tôi xin đề xuất các môi trường sau:

- Phần mềm (hoàn toàn miễn phí)
 - o Compiler: GCC ARM Embedded
 - o IDE: Eclipse C/C++ với GNU ARM Eclipse plugin
 - o Build Tools: Cần cài đặt sử dụng Windows OS
 - o Flash và Debug: pyOCD hoặc OpenOCD
- Phần cứng:
 - o Bộ mã VBLUno của VNGIoTLab với BLE
 - o Module pin/giao tiếp CMSIS-DAP (DAPLink) của VNGIoTLab

Các bước cài đặt khá đơn giản, bạn chỉ cần theo hướng dẫn dưới đây:

- **Bước 1: Cài đặt GNU ARM Toolchain**

<http://gnuarmecclipse.github.io/toolchain/install/>

TUTORIAL 2: Lập trình ứng dụng cho mạch VBLUno với GNU ARM Eclipse

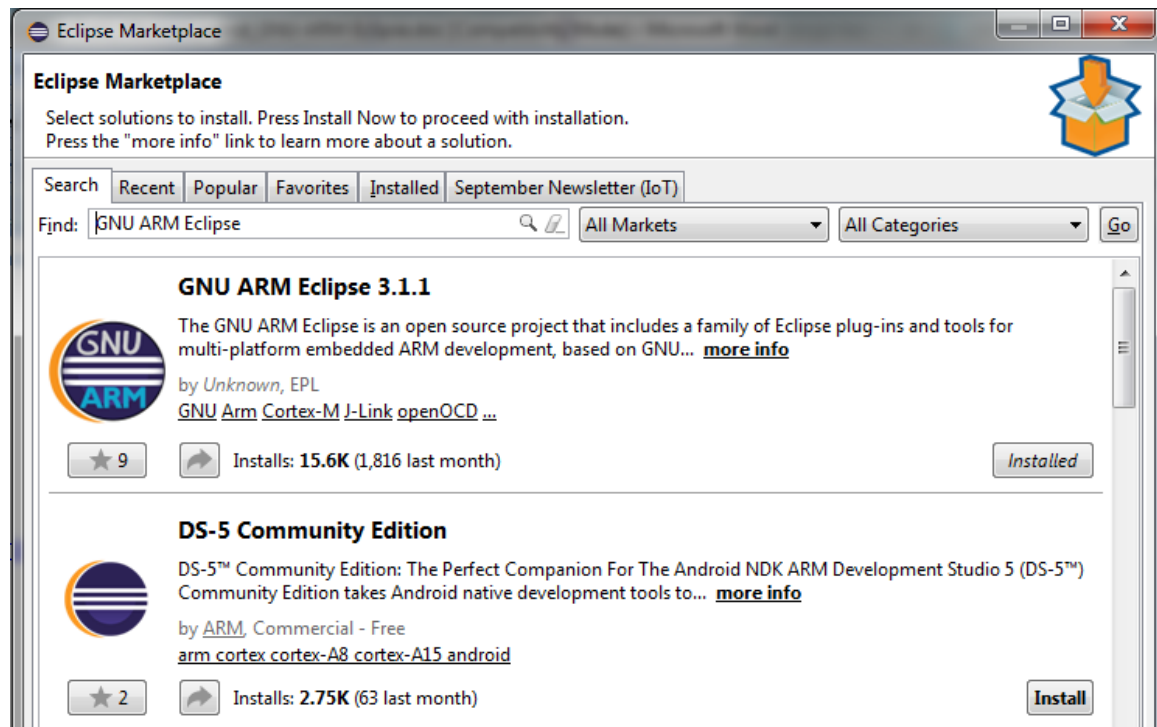
- **Bước 2: Cài đặt Eclipse C/C++**

<http://www.eclipse.org/downloads/>

- **Bước 3: GNU ARM Eclipse plugin**

+ Đây là plugin cho phép sử dụng Eclipse IDE phát triển ứng dụng cho các dòng vi xử lý khi cần ARM và trình biên dịch GCC_ARM. Sau khi cài đặt plugin, sử dụng Eclipse cho phép bạn có thể lập trình, biên dịch và tải lên môi trường GUI trực tuyến cho các developers.

+ Sau khi cài đặt Eclipse C/C++, bạn có thể vào Eclipse Marketplace, tìm và tải khóa "GNU ARM Eclipse" và install GNU ARM Eclipse plugin



+ Tham khảo thêm tại <http://gnuarmeclipse.github.io/>

- **Bước 4: Cài đặt GNU ARM Eclipse Windows Build Tool (Tùy chọn dành cho Windows OS)**

+ <http://gnuarmeclipse.github.io/windows-build-tools/>

TUTORIAL 2: Lập trình ứng dụng cho mạch VBLUno với GNU ARM Eclipse

+ Sau khi cài đặt, không quên set path hệ thống cho thư mục vừa cài đặt. Ví dụ:

"C:\Program Files (x86)\GNU ARM Eclipse\Build Tools\2.6-201507152002\bin"

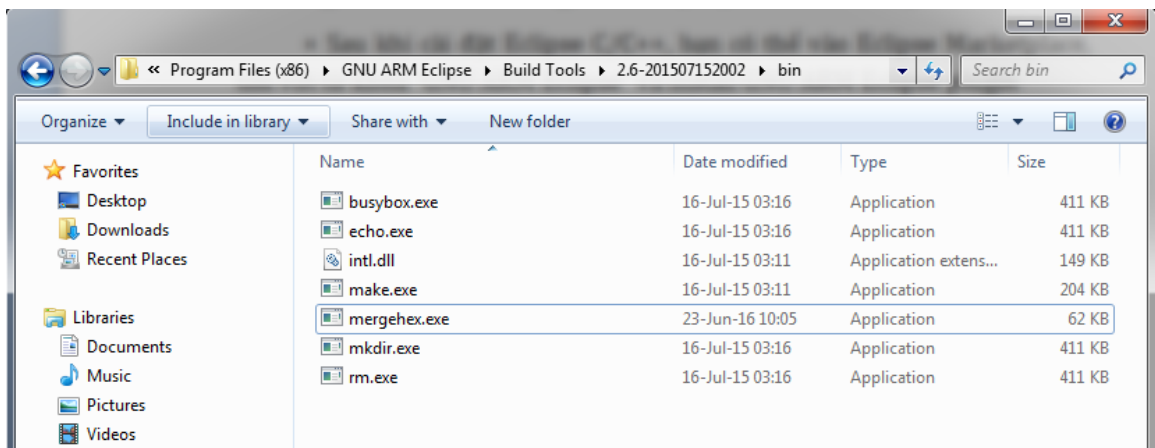
+ Vào thư mục trên, copy "busybox.exe" và đổi tên thành "mkdir.exe"

- **Bước 5: Cài đặt công cụ để merge các file hex cùng nhau**

+ Linux: cài đặt srec_cat

+ Windows:

- Sử dụng mergehex.exe. Bạn có thể download tại đây https://github.com/VNGIoTLab/mbed_for_VBLUno-CMSIS-DAP/tree/master/tools
- Copy mergehex.exe vào cùng thư mục "...GNU ARM Eclipse\Build Tools\2.6-201507152002\bin"



TUTORIAL 2: Lập trình ứng dụng cho mạch VBLUno với GNU ARM Eclipse

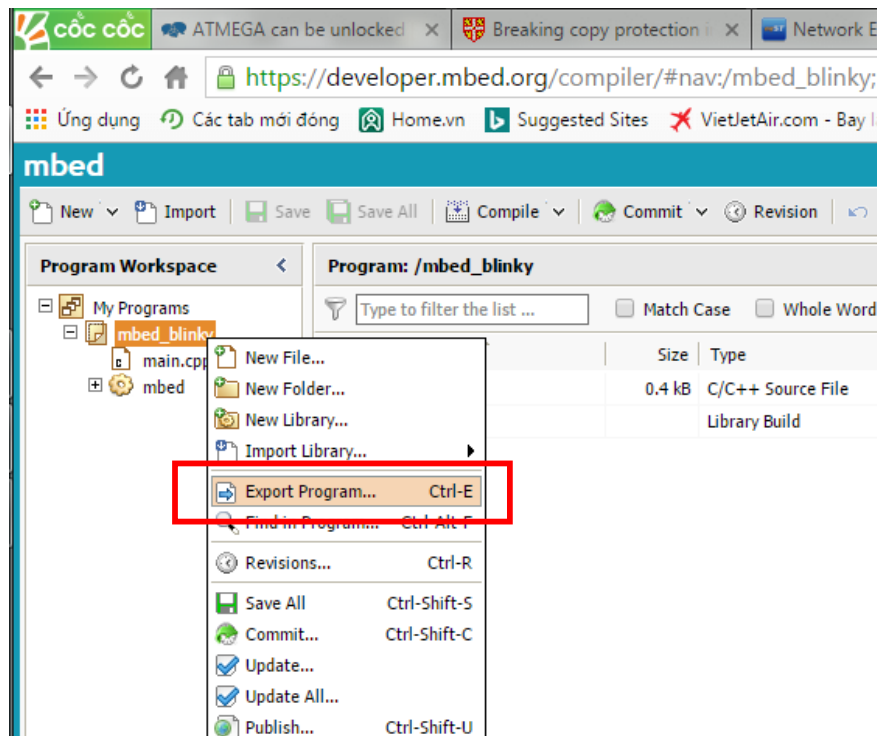
2. Xây dựng một project

- Project sau đây sử dụng:
 - o Bộ mạch VBLUno của VNGIoTLab.
 - o MCU: Nordic nRF51822 (Cortex-M0).
 - o Toolchain: GCC ARM Embedded.
 - o IDE: Eclipse với GNU ARM Eclipse plugin.
 - o Project cần ghi: Nháy led và truyền thông UART, sử dụng mbed-OS

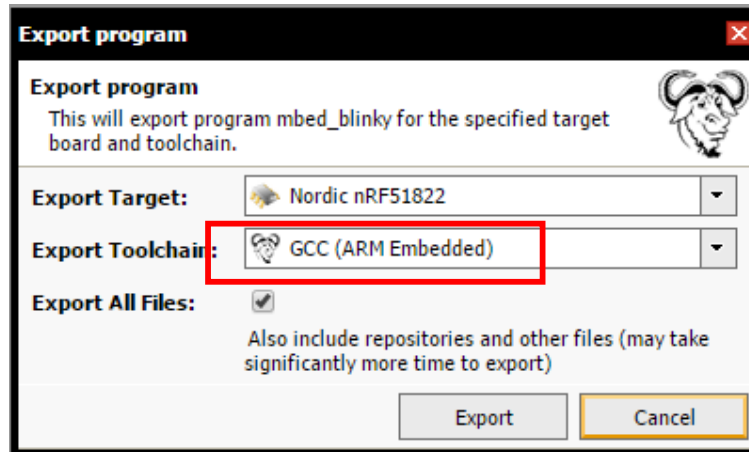
- Bước 1: Export project về máy tính

+ Có nhiều cách tải về mã nguồn project sử dụng mbed-OS cho VBLUno, bài hướng dẫn này chỉ ra cách tải về mã nguồn là export mã nguồn project về máy tính. Bạn chỉ cần tải về mã nguồn từ mbed Online Compiler về máy tính. Bước tiếp theo là tải về mã nguồn từ mbed Online Compiler về máy tính để tải về mã nguồn Makefile có sẵn, phù hợp với nRF51822.

+ Truy cập vào mbed Online Compiler, export project về máy tính làm trong Tutorial 1 về máy tính, lựa chọn toolchain là GCC (ARM Embedded)



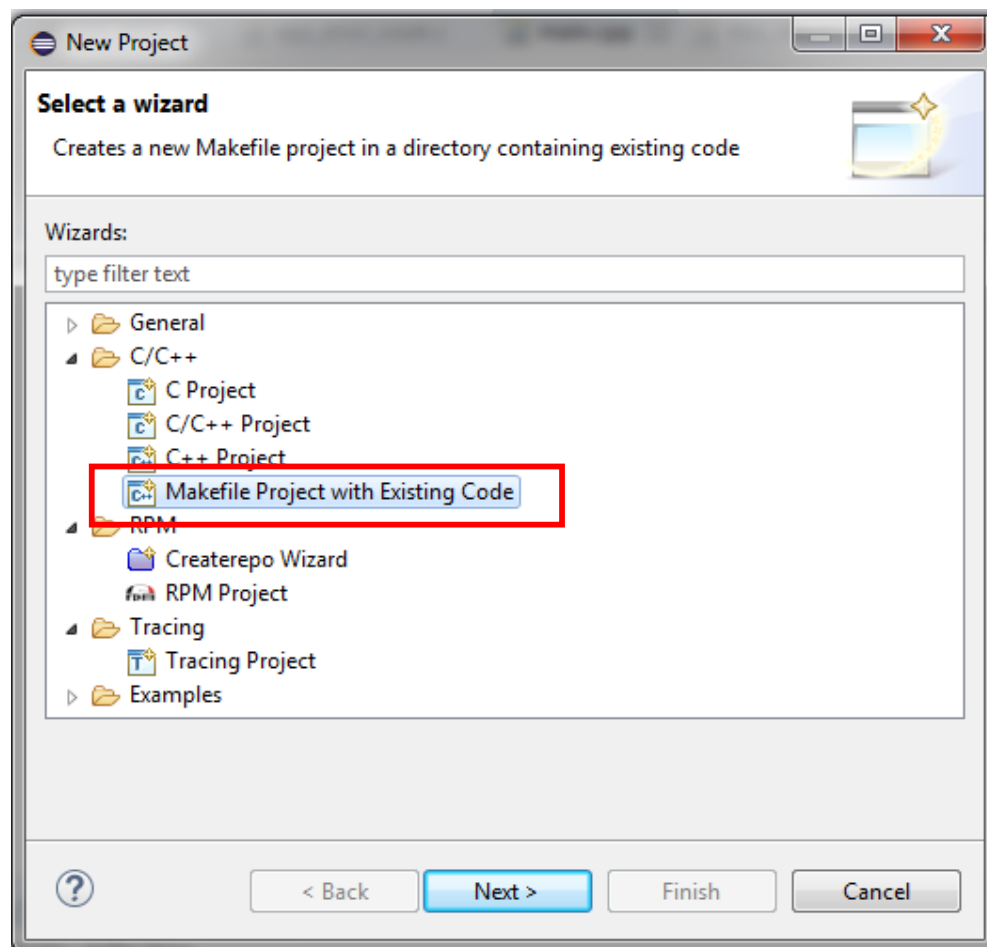
TUTORIAL 2: Lập trình ứng dụng cho mạch VBLUno với GNU ARM Eclipse



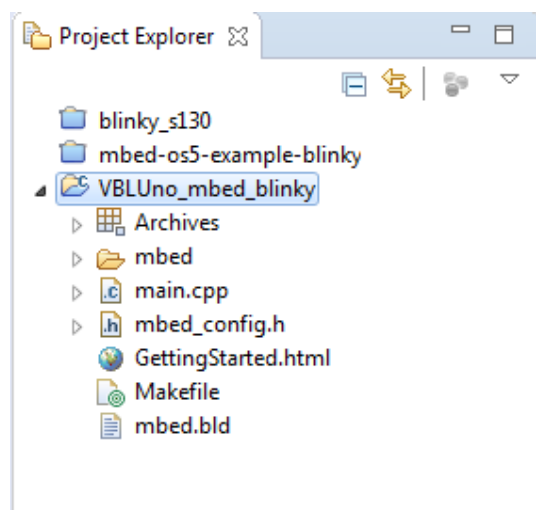
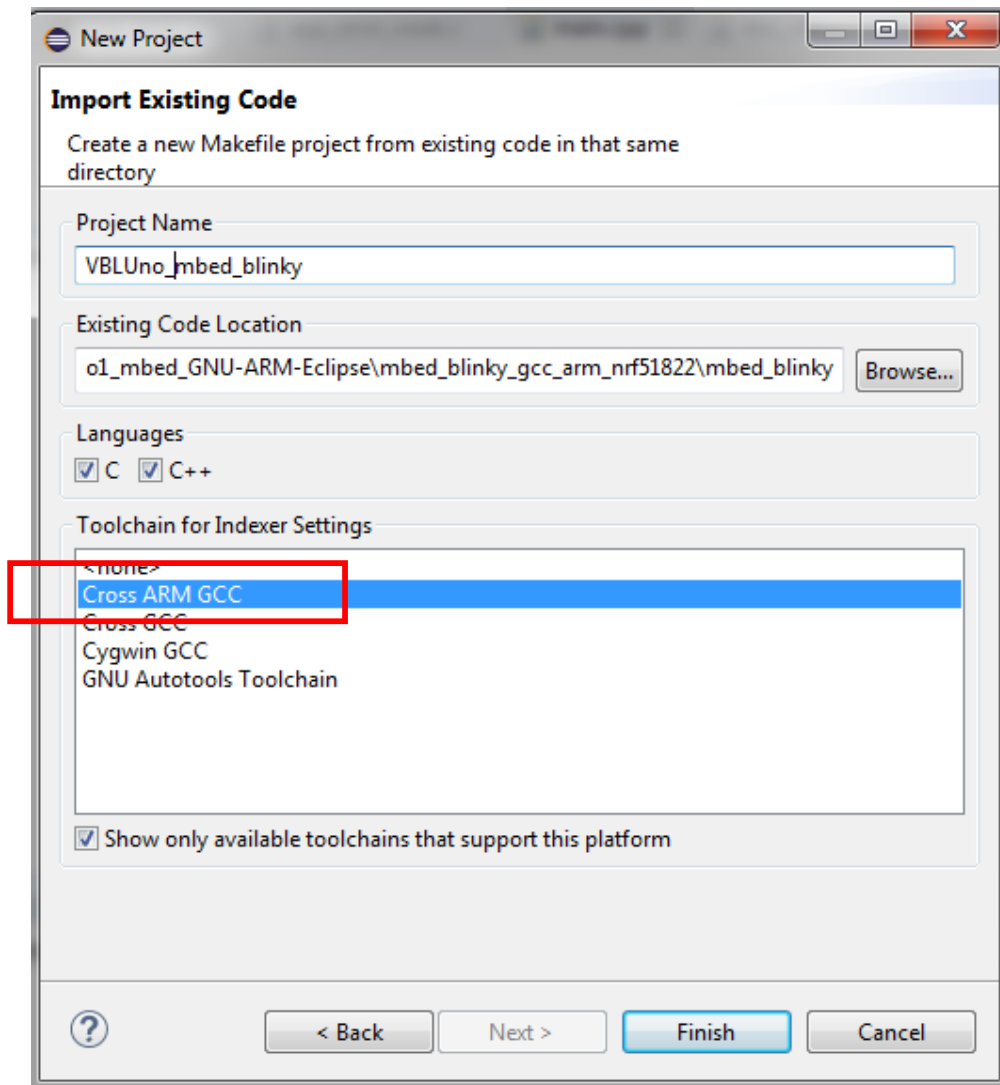
- Bước 2: Tạo project với Eclipse

+ Chọn File -> New -> Project

+ C/C++ -> Makefile Project with Existing Code



TUTORIAL 2: Lập trình ứng dụng cho mạch VBLUno với GNU ARM Eclipse

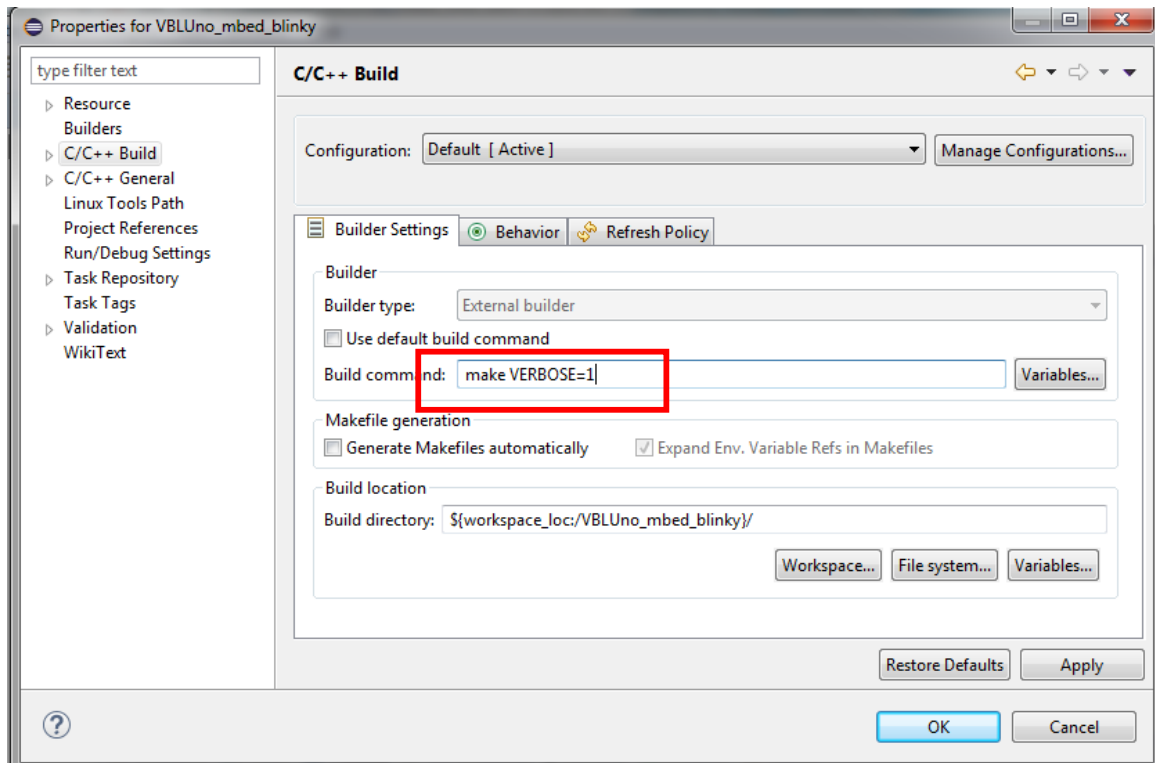


TUTORIAL 2: Lập trình ứng dụng cho mạch VBLUno với GNU ARM Eclipse

- Bước 3: Cấu hình cho project

+ Nhấp chuột phải vào project -> chọn Properties

+ Tìm mục C/C++ Build, chọn thẻ Builder Settings, sửa build command



- Bước 4: Build project

+ Chúng ta không cần tạo Makefile, do đã sử dụng makefile nhúng sẵn vì chúng ta export project từ mbed Online Compiler. Tuy nhiên, makefile nhúng sẵn này dành cho Linux, nếu sử dụng Windows OS bạn cần phải chỉnh sửa Makefile như sau:

Thêm dòng 53 trong hình sau:

```
44 #####
45 AS      = $(GCC_BIN)arm-none-eabi-as
46 CC      = $(GCC_BIN)arm-none-eabi-gcc
47 CPP     = $(GCC_BIN)arm-none-eabi-g++
48 LD      = $(GCC_BIN)arm-none-eabi-gcc
49 OBJCOPY = $(GCC_BIN)arm-none-eabi-objcopy
50 OBJDUMP = $(GCC_BIN)arm-none-eabi-objdump
51 SIZE    = $(GCC_BIN)arm-none-eabi-size
52 SREC_CAT = srec_cat          ##For Linux
53 MERGE_HEX = mergehex        ##For Windows
54
```

TUTORIAL 2: Lập trình ứng dụng cho mạch VBLUno với GNU ARM Eclipse

Thêm dòng 125, 126 trong hình sau:

```
122 merge:
123     $(SREC_CAT) $(SOFTDEVICE) -intel $(PROJECT).hex -intel -o combined.hex -intel -
124
125 merge_window:
126     $(MERGE_HEX) -m $(SOFTDEVICE) $(PROJECT).hex -o combined.hex
127
```

Sửa dòng 76 như hình sau:

```
75
76 all: $(PROJECT).bin $(PROJECT).hex size merge_window
77
```

Sửa dòng 41 như hình sau:

```
--
33 GCC_BIN =
34 PROJECT = mbed_blinky
35 OBJECTS = main.o
36 SYS_OBJECTS = mbed/TARGET_NRF51822/TOOLCHAIN_GCC_ARM/cmsis_r
37 INCLUDE_PATHS = -I../. -I../mbed/. -I../mbed/TARGET_NRF51822/
38 LIBRARY_PATHS = -L../mbed/TARGET_NRF51822/TOOLCHAIN_GCC_ARM
39 LIBRARIES = -lmbed
40 LINKER_SCRIPT = ../mbed/TARGET_NRF51822/TOOLCHAIN_GCC_ARM/NF
41 SOFTDEVICE = ../mbed/TARGET_NRF51822/TARGET_NORDIC/TARGET_MC
42
```

+ Nhấp chuột phải vào project -> chọn Build Project

+ Nếu thành công, kết quả biên dịch như hình dưới

```
CDT Build Console [VBLUno_mbed_blinky]
08:50:49 **** Build of configuration Default for project VBLUno_mbed_blinky ****
make VERBOSE=1 all
arm-none-eabi-g++ -c -Wall -Wextra -Wno-unused-parameter -Wno-missing-field-initializers -fmessage-length=0
arm-none-eabi-gcc -Wl,--gc-sections -Wl,--wrap,main -Wl,--wrap,_malloc_r -Wl,--wrap,_free_r -Wl,--wrap,_
arm-none-eabi-objcopy -O binary mbed_blinky.elf mbed_blinky.bin
arm-none-eabi-size mbed_blinky.elf
   text    data     bss     dec     hex filename
   20756    484      472   21712   54d0 mbed_blinky.elf
mergehex -m ../mbed/TARGET_NRF51822/TARGET_NORDIC/TARGET_MCU_NRF51822/Lib/s130_nrf51822_1_0_0/s1
Parsing input hex files.
Merging files.
Storing merged file.

08:50:50 Build Finished (took 1s.190ms)
```

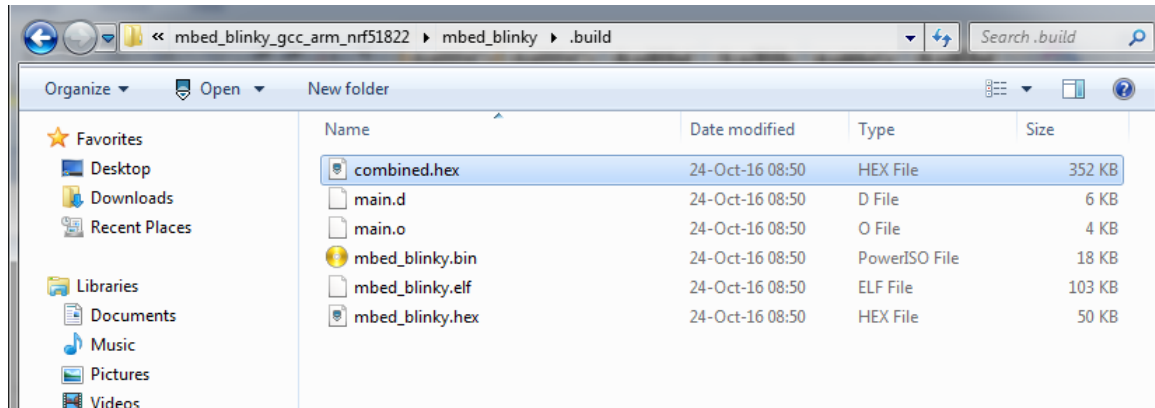
- Bước 5: Nạp file thực thi xuống mạch VBLUno

Chúng ta cần chú ý 2 file thực thi có phần mở rộng (*.hex) trong thư mục .build

+ mbed_blinky.hex: Đây là file thực thi của Application, không gồm softdevice

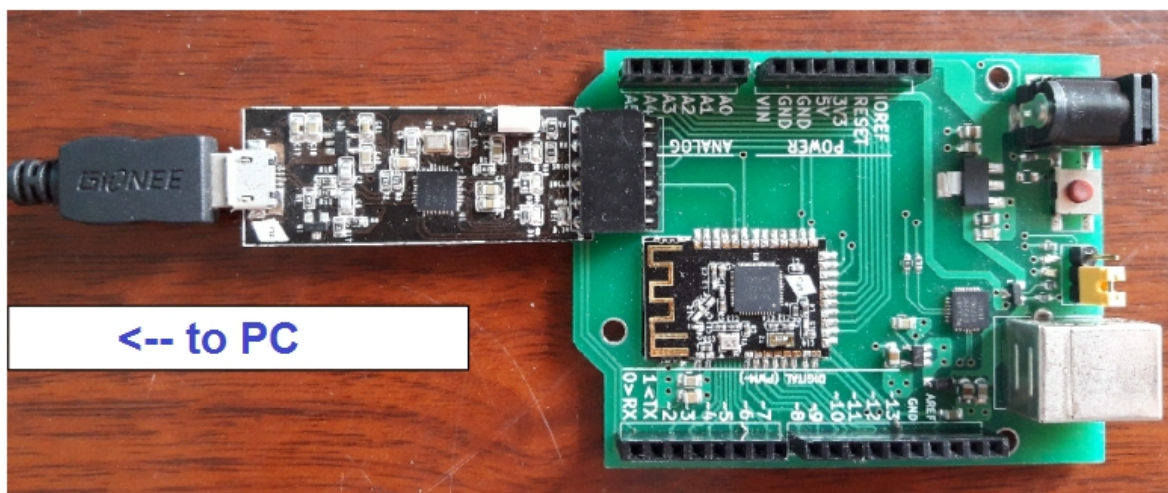
TUTORIAL 2: Lập trình ứng dụng cho mạch VBLUno với GNU ARM Eclipse

+ combined.hex: G m Softdevice và Application



ch c ch n, chúng ta n p file combined.hex xu ng m ch VBLUno b ng cách copy file này vào a c a CMSIS-DAP

S k t n i:



Trong bài hướng dẫn tiếp theo, chúng tôi sẽ trình bày cách nạp chương trình và debug với pyOCD và OpenOCD.

M i th c m cho c ý ki n óng góp b n có th trao i t i ây:

<https://www.facebook.com/bleviet/messages/>

https://github.com/VNGIoTLab/mbed_for_VBLUno-CMSIS-DAP/issues