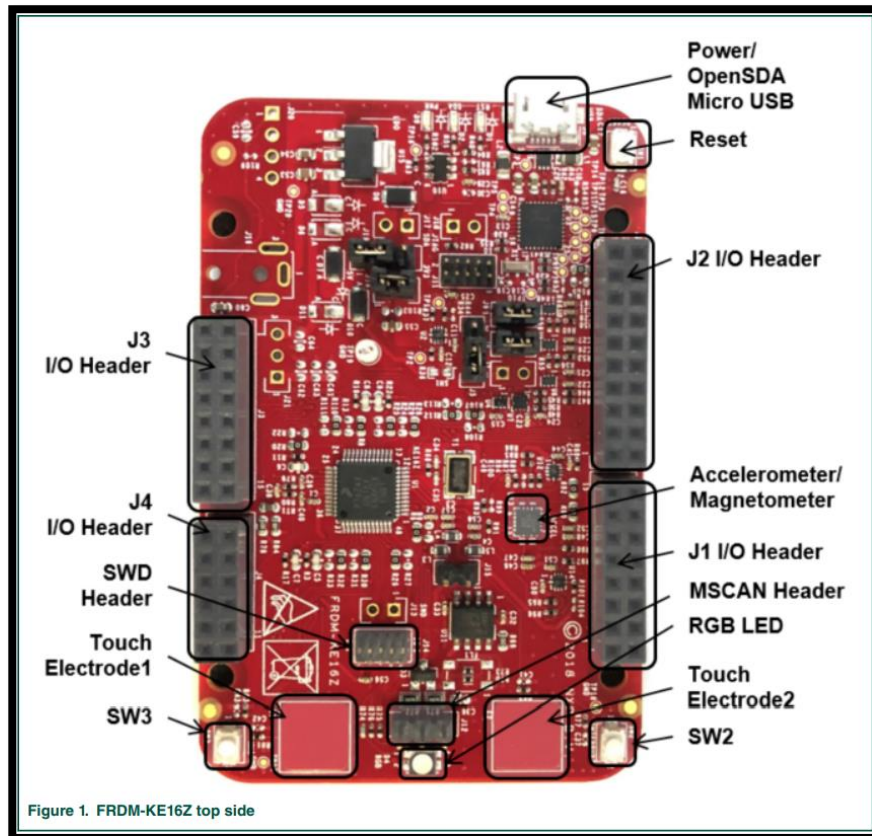
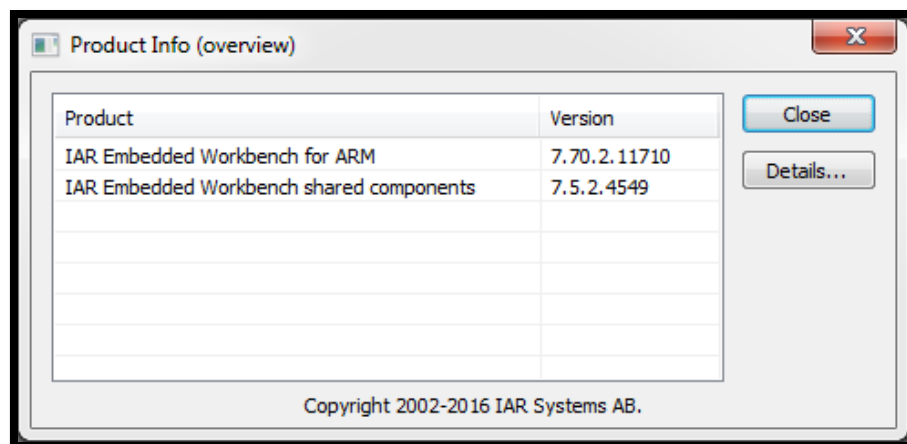


Các bước tạo và debug một chương trình với IAR cho board FRDM-KE16Z



Các thao tác dưới đây được thực hiện trên phần mềm IAR với phiên bản như sau:

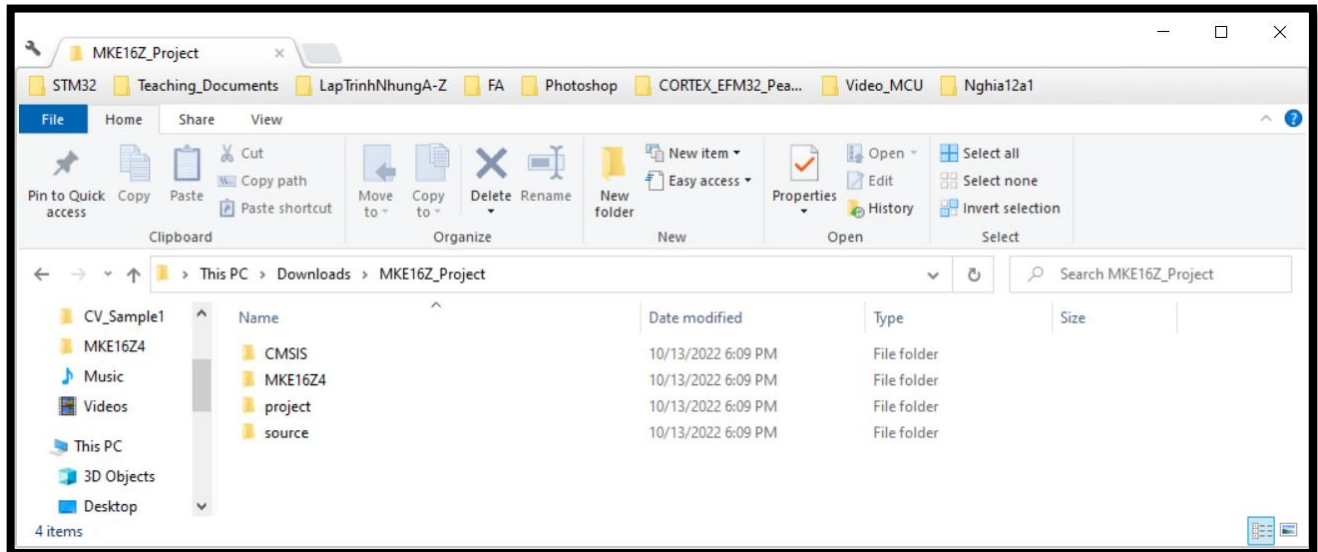


B1: Tạo folder tên là "LedBlinking" cho bài tập nhảy led

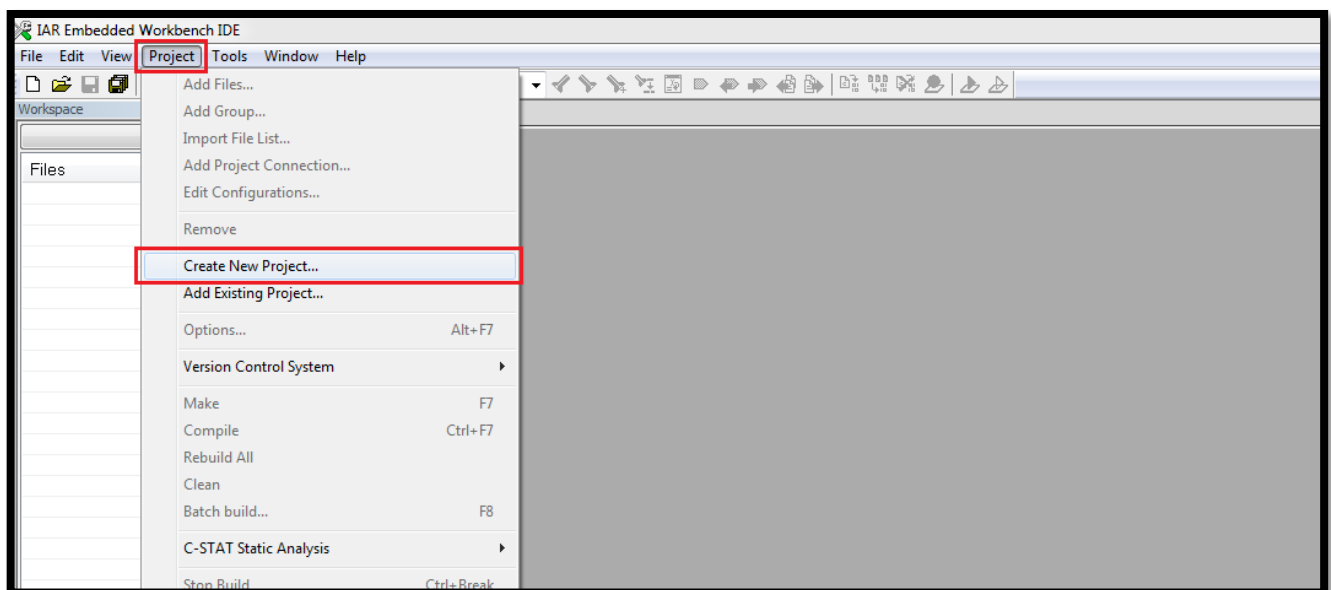
B2: Trong folder "LedBlinking" tạo ra các folder sau:

- “source”: thư mục này sẽ chứa file "main.c" là file cần viết mới cho mỗi bài tập
- “project”: Thư mục này sẽ chứa các file do IAR tạo ra (các file cấu hình cho project)

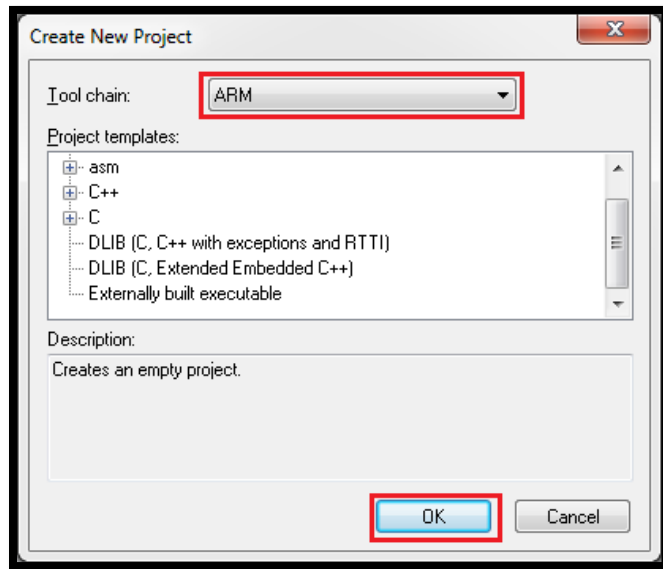
B3: Copy 2 folder mẫu "CMSIS" và "MKE16Z" tới thư mục "LedBlinking". Sau bước này sẽ được một folder như sau:



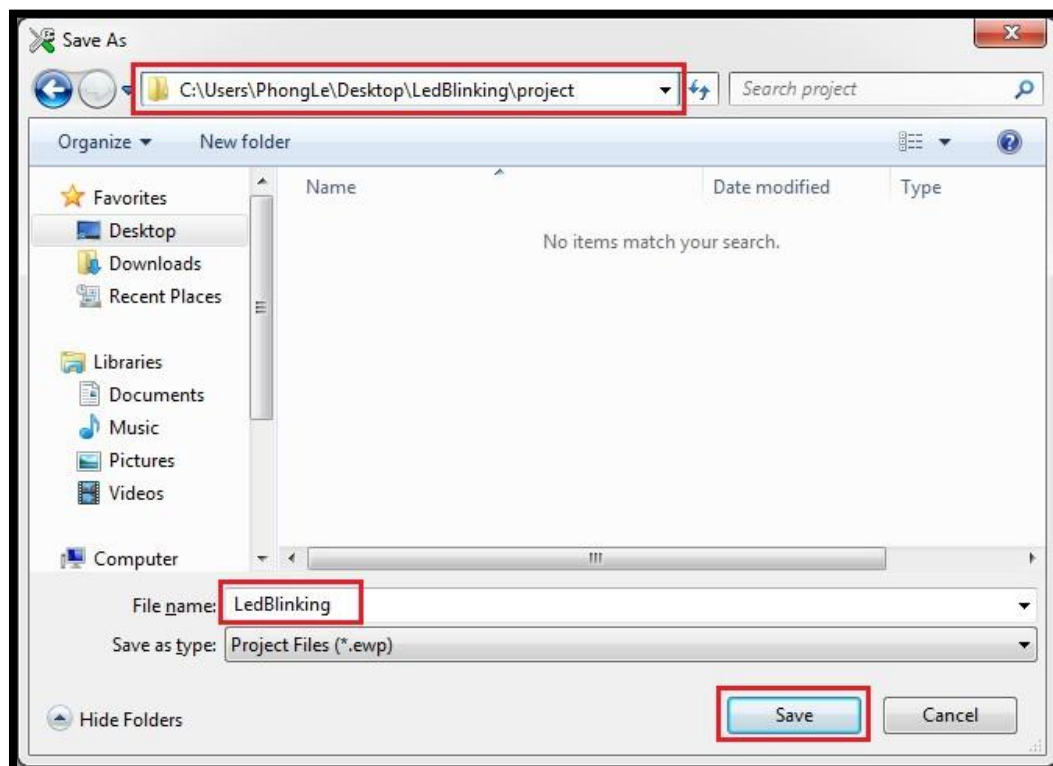
B4: Mở IAR, vào **Project -> Create New Project**



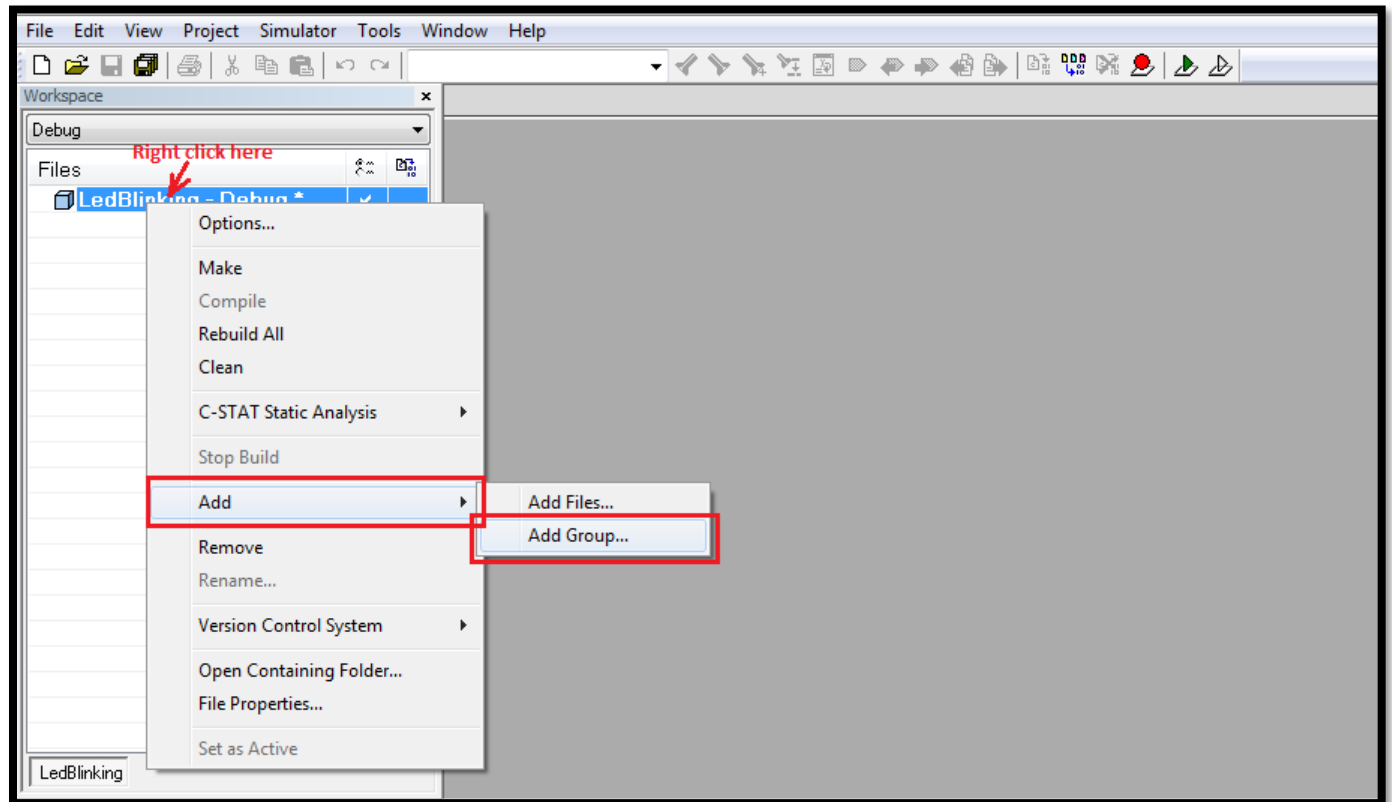
Chọn "ARM" ở mục "Tool chain" rồi nhấn OK



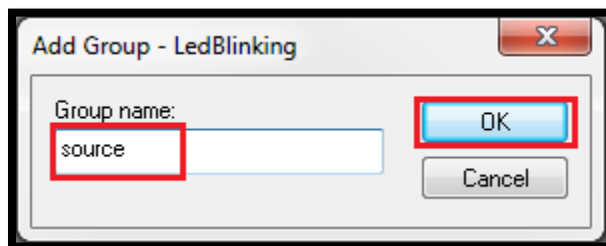
B5: Đặt tên project là "LedBlinking" và lưu tại folder "**project**" vừa tạo ở bước trên. Sau bước này, một project trống sẽ được add vào IAR.



B6: Tạo 2 group "source" và "startup" trong IAR để quản lý các file bằng cách sau:



- Chuột phải vào project "LedBlinking" chọn Add -> Add Group

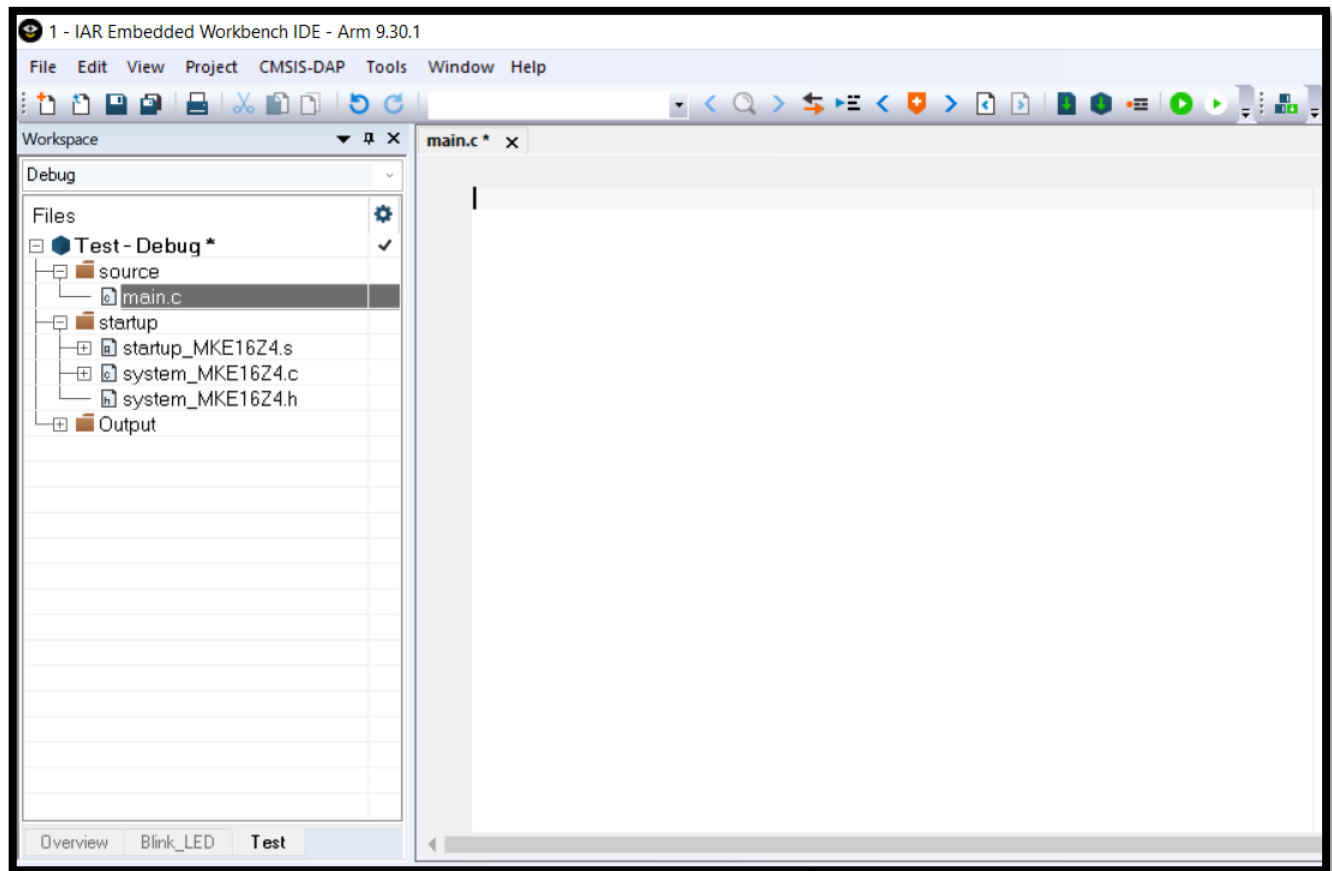


- Nhập tên group "source" rồi nhấn OK.
Tương tự tạo thêm group "startup".

B7: Add các file cần thiết tới project:

- Add 3 file "startup_MKE16Z4.s", "system_MKE16Z4.c", "system_MKE16Z4.h" trong thư mục "MKE16Z4" trong thư mục dự án vào group "**startup**". Để add file vào group chỉ cần kéo thả file cần add vào group tương ứng hoặc **chuột phải vào group -> Add -> Add Files**.

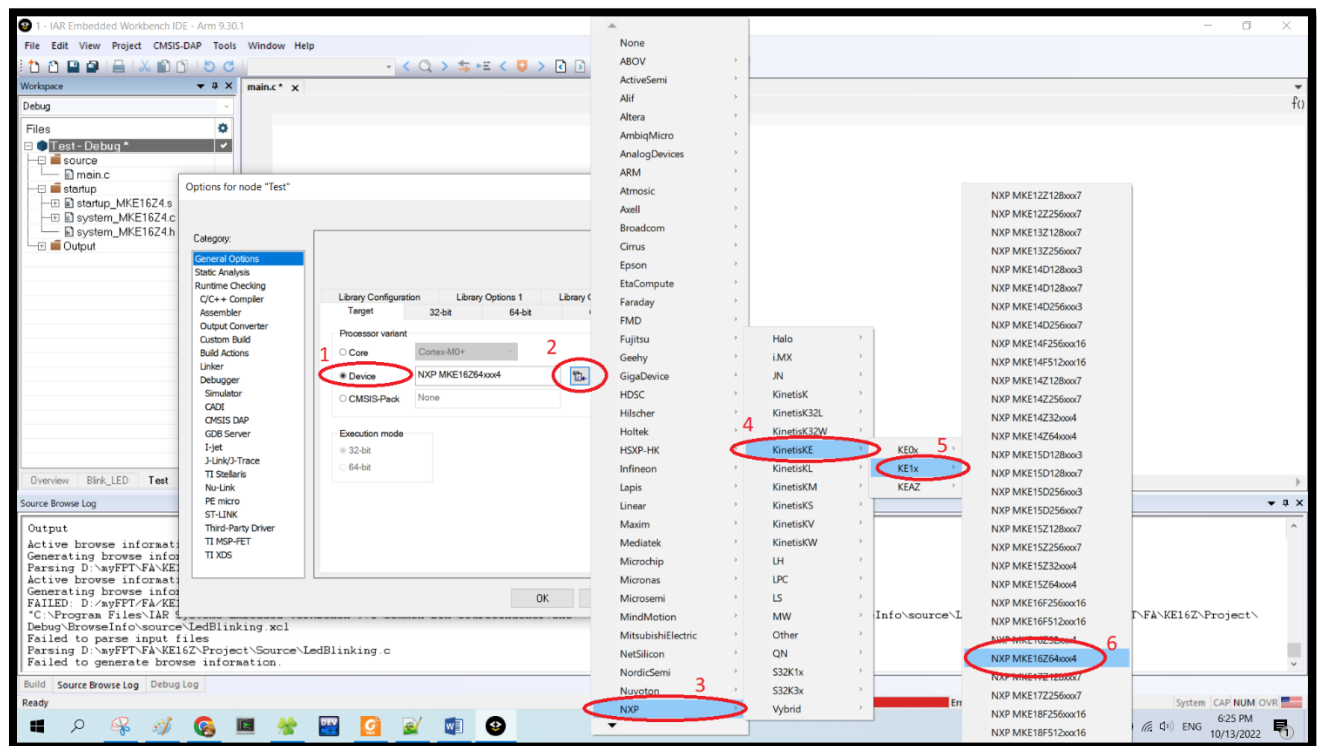
- Tạo file "**main.c**", lưu ở thư mục "source" rồi add vào group "source" trong project. Sau bước này, ta sẽ được một project như sau:



B8: Cấu hình cho project: Chuột phải vào project chọn Option để mở cửa sổ "Options for node LedBlinking"

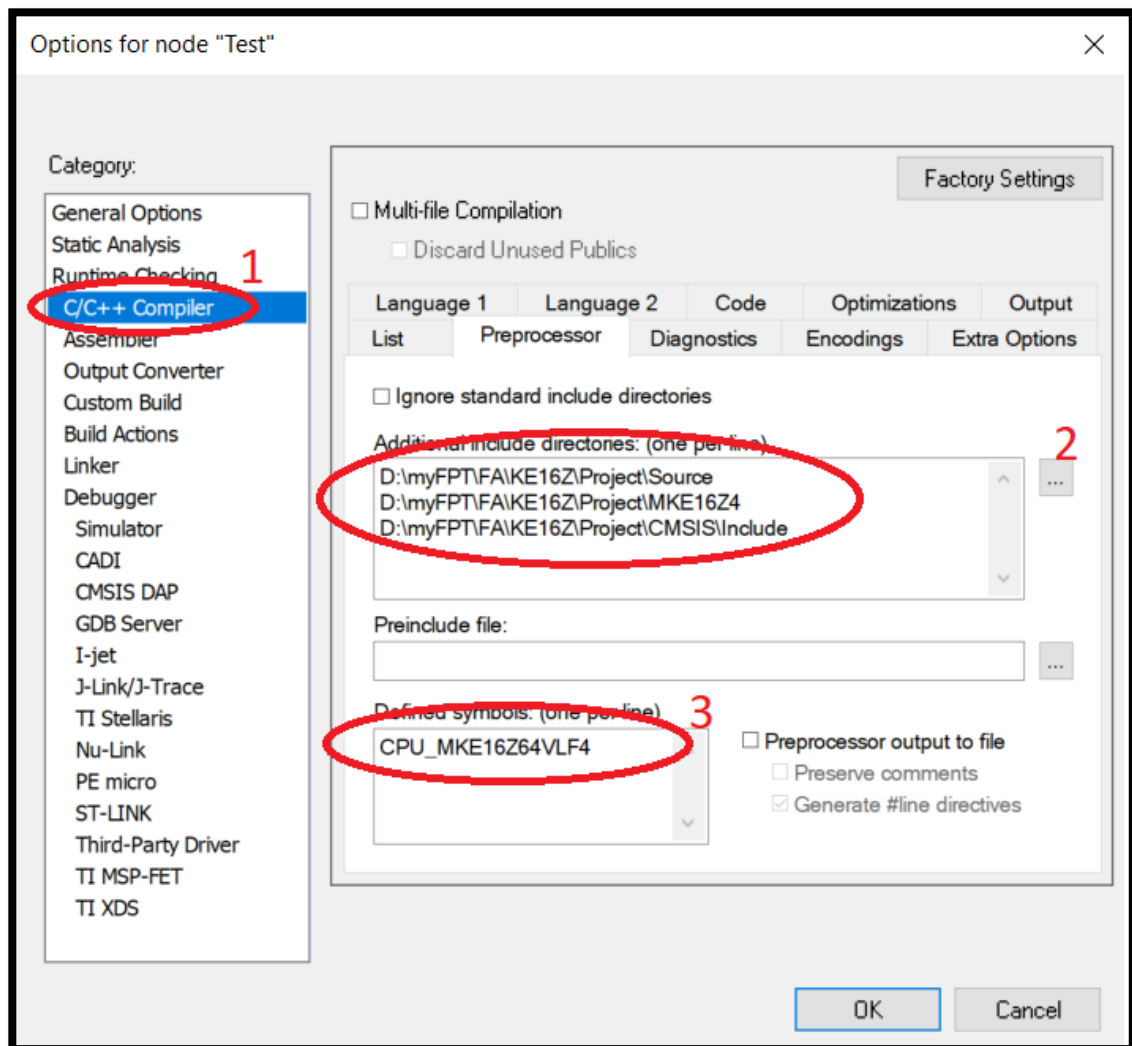
- **Chọn chip cho project:**

Tại thẻ "General Options", chọn chip như hình sau:

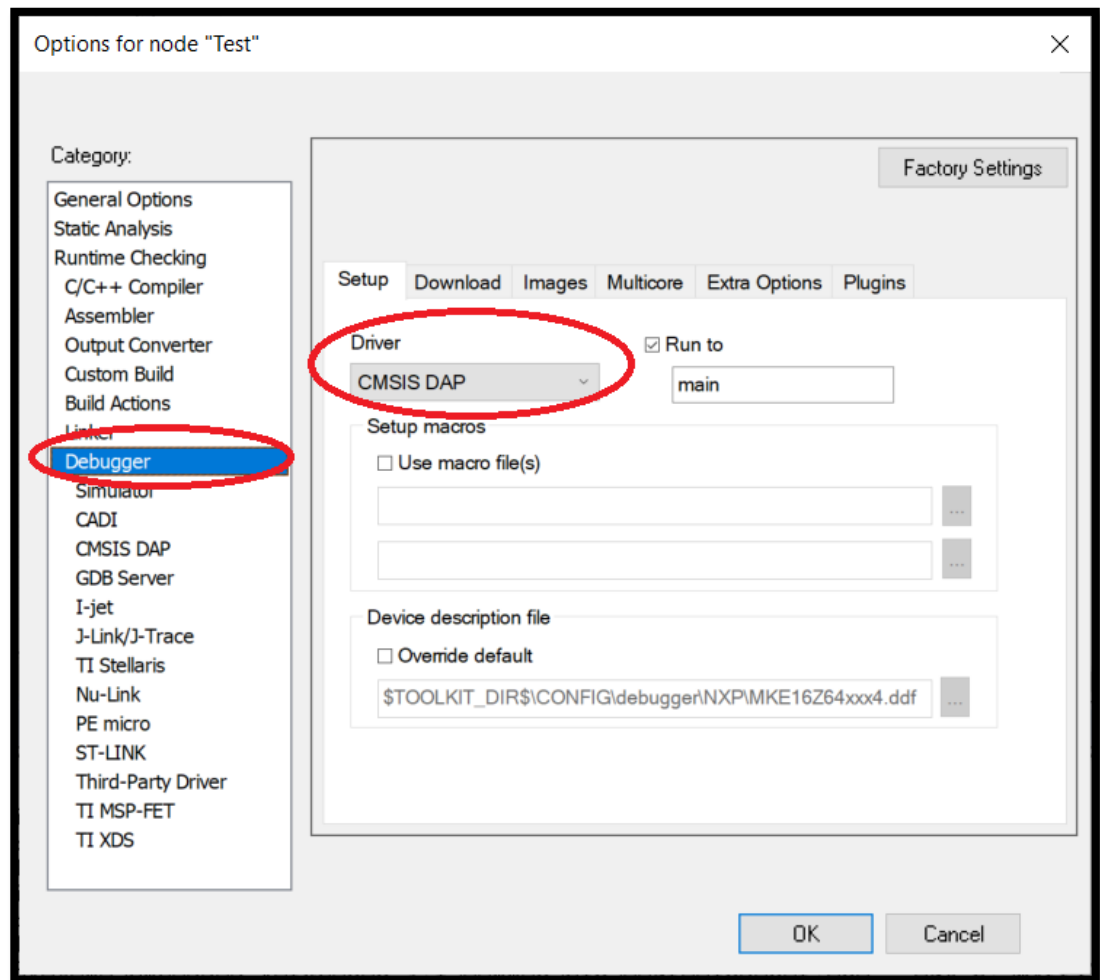


➤ Cấu hình include path và defined symbols:

- Chọn các “include directories” là các folder chứa các header file trong project.
- Mục “Defined symbols” để định nghĩa một Macro đặc trưng cho từng dòng chip, phục vụ cho thư viện CMSIS.



➤ Cấu hình debugger:



B9: Viết code cho file main.c

Tự viết code cho file main.c hoặc sử dụng ví dụ mẫu nháy Led được viết trong file **LedBlinking.c**

B10: Build, Chạy và Debug:

- Chuột phải vào project, chọn “Rebuild All”, IAR sẽ yêu cầu lưu lại workspace trước khi build. Đặt tên workspace là “LedBlinking.eww” và lưu tại thư mục “project” của dự án. Sau khi lưu lại workspace, IAR sẽ thực hiện build project.
- Nếu quá trình build không có lỗi, một file **LedBlinking.out** sẽ được tạo ra trong thư mục “project\Debug\Exe\LedBlinking.out”, đây chính là file binary dùng để nạp xuống board.

- Vào **Project -> Download and Debug** (hoặc phím tắt Ctrl+D) để download file “LedBlinking.out” xuống board. Chạy chương trình bằng cách nhấn F5.

Tạo project bằng cách sửa đổi một project có sẵn.

Việc tạo một project mới theo cách trên sẽ mất nhiều thời gian vì vậy để tiết kiệm thời gian, sau khi đã tạo được một project như phía trên có thể thực hiện copy project trên và sửa đổi tên project để tạo ra project mới. Các bước sau sẽ thực hiện việc tạo một project mới có tên là **Button** từ project **LedBlinking** đã tạo phía trên:

B1: Copy toàn bộ folder LedBlinking sang folder mới và đổi tên folder mới thành **Button**.

Trong folder **project**, xóa hết các file và folder do IAR tạo ra, chỉ giữ lại 3 file **LedBlinking.ewd**, **LedBlinking.ewp**, **LedBlinking.eww** và đổi tên thành **Button.ewd**, **Button.ewp**, **Button.eww** tương ứng.

B2: Mở 2 file **Button.ewp** và **Button.eww** bằng trình soạn thảo văn bản (ví dụ Notepad++). Tìm và thay thế tất cả chuỗi “LedBlinking” thành “Button” (Thực tế mỗi file chỉ chứa 1 chuỗi “LedBlinking”).

Mở file **Button.eww** bằng IAR, một project có tên là Button sẽ được add vào IAR với cấu hình tương tự như project LedBlinking. Lúc này chỉ cần sửa đổi nội dung file main.c và thực hiện build, debug và Run.