

MPLAB™ X IDE 한글 사용설명서

본 문서에서 사용되는 이미지는 MPLAB X IDE의 버전에 따라 다소 상이할 수 있습니다.

목차

다운로드 및 설치

1. 다운로드
2. 설치

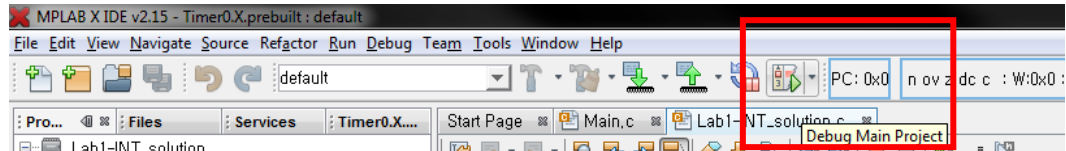
기본 사용법

1. 새 프로젝트 만들기
2. MPLAB™ X IDE 데스크탑
3. 프로젝트 속성(Properties) 조정
4. 디버거, 프로그램, 프로그램 언어 옵션 설정
5. 프로그램 언어 위치 설정
6. 기타 도구 옵션 설정
7. 새 파일 만들기
8. 프로젝트에 파일 추가하기
9. 에디터 사용법
10. 라이브러리/오브젝트 파일 추가 및 셋업
11. 파일/폴더 속성 설정
12. 빌드(Build) 속성 설정
13. 프로젝트 빌드 (Build Project)
14. 코드 실행 (Run Code)
15. 실행 중인 코드 디버깅 (Debug Run Code)
16. 브레이크포인트(Breakpoints) 활용
17. 코드 스텝핑 (Step Code)
18. 와치(Watch)창 활용 – 심볼 값 변경
19. 와치(Watch)창 활용 – 지역변수 값 변경
20. 디바이스 내의 메모리 값 확인 및 변경
21. Call Stack
22. 디바이스 프로그램 (Program Device)

기본 사용법

1. 디버그 동작하는 방법

“디버그 프로젝트(Debug Project)” 버튼을 누른다.



- I. 빌드가 필요할 경우에 프로젝트를 빌드하게 된다.
- II. In-circuit debugger/emulator 를 위해서, 타겟 디바이스 또는 헤더는 자동적으로 디버깅을 실행하기 위한 이미지가 프로그램 되고 디버그 세션이 시작된다.
- III. 시뮬레이터를 디버거로 사용하는 경우에는 다운로드 없이 디버그 세션이 바로 시작된다.

하단의 출력 창(Output window)을 통해서 디버그 진행사항을 확인 할 수 있다.

2. 디버그 매크로 생성(Debug Macros Generated)

MPLAB X IDE 에서는 마이크로칩 언어 툴(Microchip language tools)을 사용하기 위해 디버그 매크로를 발생시킨다. 매크로는 마이크로칩 컴파일러들과 어셈블러에 아래 표와 같이 전달된다.

매크로 이름	관련 툴	기능
<code>__DEBUG</code>	모든 툴	현재 빌드가 디버깅 빌드임을 명시한다.
<code>__MPLAB_REAL_ICE__</code> <code>__MPLAB_ICD3__</code> <code>__MPLAB_PK3__</code> <code>__MPLAB_PICKIT2__</code>	XC8	사용되는 하드웨어 디버그 툴을 명시한다.
<code>__MPLAB_DEBUGGER_REAL_ICE</code> <code>__MPLAB_DEBUGGER_ICD3</code> <code>__MPLAB_DEBUGGER_PK3</code> <code>__MPLAB_DEBUGGER_PICKIT2</code>	XC16 XC32 MPASM	사용되는 하드웨어 디버그 툴을 명시한다.
<code>__MPLAB_DEBUGGER_PIC32MXSK</code>	XC32	사용된 스타터 킷을 명시한다.

사용자는 작성하는 소스코드에 직접 이러한 매크로를 사용 할 수 있다.

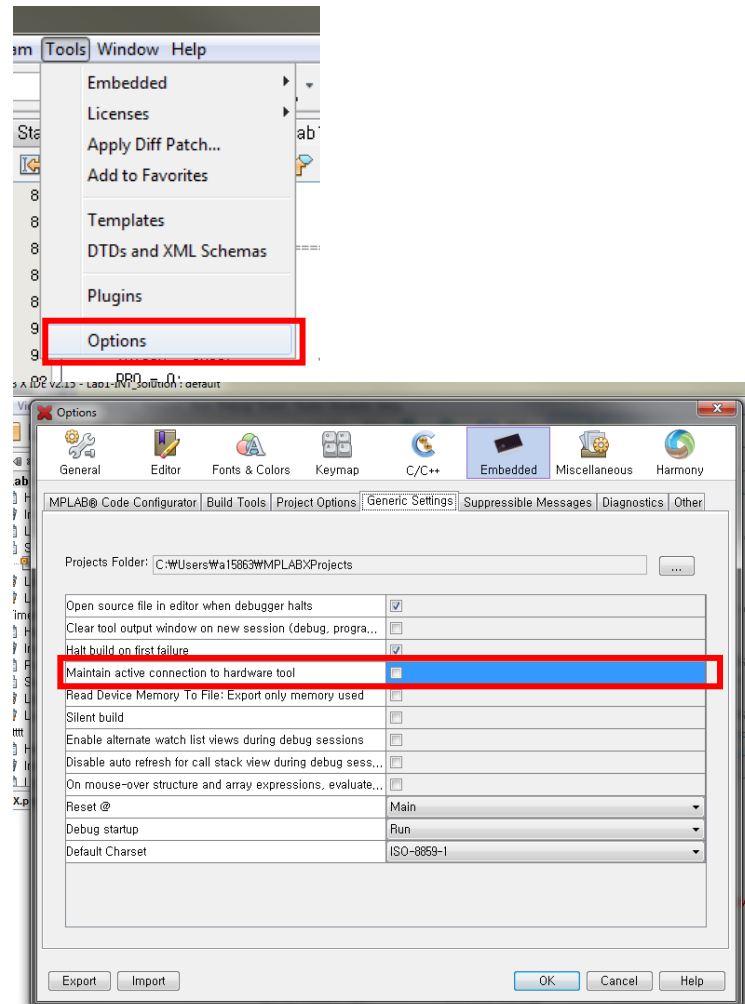
예시)

```
#ifdef __DEBUG
    fprintf(stderr,"This is a debugging message\n");
#endif
```

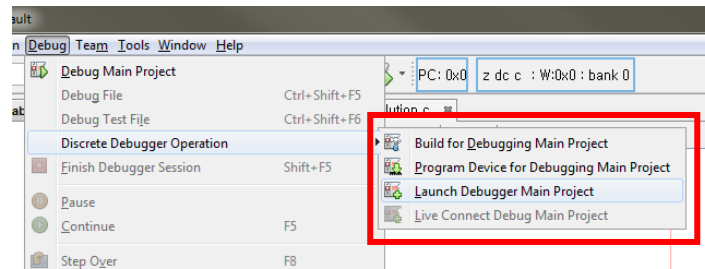
3. 디버그 고려사항(Debug considerations)

사용자가 소스코드를 디버깅 할 때 주의사항이다.

- I. 사용자는 와치(watch)에서 변수 값 보기 또는 메모리 윈도우(Memory window)를 확인하는 많은 디버깅 기능들을 사용하기 위해서 디버깅 세션에 있어야 한다.
- II. MPLAB X IDE 동작은 하드웨어 툴과 런타임(Run-Time, Run or Debug Run)에 연결한다. 이는 MPLAB X IDE 에서 만들어 놓은 설정이 런타임에만 전달 된다는 것을 의미한다. 디버깅이 중단 되었을 동안 (during debug halt) 변경된 설정은 하드웨어가 다시 런타임이 될 때 변경 될 것이다.
항상 연결되게 하기 위해서는 MPLAB IDE 를 사용하던가, 툴(Tool) -> 옵션(Options)에 들어가서 임베디드 버튼(Embedded)을 누르고 일반 세팅(Generic setting) 탭에 들어가서 "하드웨어 툴 연결 유지기능(Maintain active connection to hardware tool)" 체크박스를 체크하면 된다.



- III. 특정 어플리케이션에서 사용자는 디버그 세션을 나누어서 진행해야 할 수 있다. 이럴 때는 Debug 탭에서 빌드, 다운로드, 디버그 실행이 나누어진 아이콘을 통해서 독립된 수행을 진행하면 된다.



4. 사용자의 어플리케이션 코드 디버깅

사용자의 어플리케이션 코드를 디버깅하기 위해서 아래와 같은 단계를 진행한다.

- I. 프로젝트 윈도우에서 사용하려는 프로젝트를 메인 프로젝트로 설정한다.
(프로젝트 명 클릭 후 "Set as main project" 클릭)
- II. 디버그 프로젝트 아이콘 클릭 (또는 디버그(Debug)-> 디버그 프로젝트(Debug Project) 클릭)하여 디버그 실행

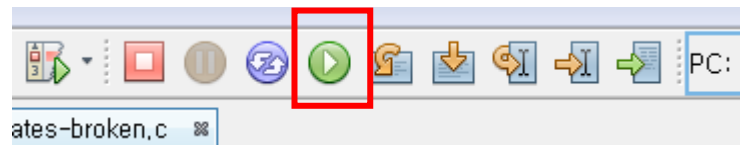
사용자의 어플리케이션 코드를 중단시키기 위해서는 아래와 같은 단계를 진행한다.

- I. "정지(Pause)" 버튼을 누른다. (또는 디버그(Debug)-> 정지(Pause)클릭)



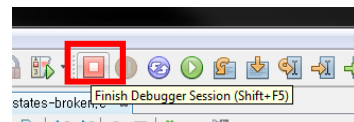
사용자의 어플리케이션 코드를 다시 실행시키기 위해서는 아래와 같은 단계를 진행한다.

- I. "계속(Continue)" 버튼을 누른다.



사용자의 어플리케이션 코드를 실행 종료시키기 위해서는 아래와 같은 단계를 진행한다.

- I. 디버그 세션 종료 아이콘을 클릭한다.



디버거(debugger)를 실행시키기 위해서는 아래와 같은 단계를 진행한다.

- I. 프로젝트가 이미 빌드가 되어 있고 디버거를 실행 시키기 위해서는 아래 디버그 실행 버튼 옆의 작은 화살표를 눌러서 2 번 째 줄에 위치한 디버그 실행(Launch Debugger main Project)를 클릭한다.

