**STM32H750 개발보드 시작하기**

STM32H750 개발보드로 무엇부터 어떻게 시작할 지 모르는 초보자를 위해 아주 간단한 기본적인 것을 정리해 봅니다. 이 문서에서 많은 것을 얘기하지는 못하며 그저 방향만 가르키는 정도일 뿐입니다.

1. **필요한 것**

당연히 STM32H750 보드가 있어야 합니다. 그리고 BOOT0 핀을 조작하여 RS-232 케이블로 펌웨어를 다운로드 하는 등의 방법이 있긴 하지만, 편리함은 둘째 치고라도 디버깅을 위해서 ST-Link 하나쯤은 가지고 시작하는 것이 좋습니다. 이 둘만 있으면 기본적인 것은 테스트해 볼 수 있으며 점퍼 케이블, 빵판(Bread Board)등을 필요에 따라 갖추시면 됩니다.

1. **개발 프로그램**

펌웨어를 코딩하고 다운로드 및 디버깅을 위한 도구는 여러 가지가 있지만, ST 홈에 가시면 거의 모든 필요한 프로그램 및 자료를 무료로 다운받아 사용하실 수 있습니다. 그 중에서 가장 기본이 되는 것이 “STM32CubeIDE”입니다.

<https://www.st.com/en/development-tools/stm32cubeide.html> 에 가시면 ST에 가입 하시고 다운받으실 수 있습니다. 현재 버전은 1.3.0이네요. 설치 후 가끔 실행이 안되면 JAVA를 인스톨해 주면 대부분 해결이 됩니다. 프로그램이 실행 되었으면 메뉴의 Help -> Manage embedded software packages를 선택하여 리소스 패키지를 설치해야 합니다. 사용하려는 MCU에 해당하는 것을 설치하면 되는 데, 우리는 STM32H750VBT6이므로 STM32H7 부분을 클릭하고 가장 최신 버전의 패키지를 선택하고 설치해 줍니다.

이 프로그램만 있으면 일단 시작하실 수 있습니다. 나중에 익숙해 지면 프로그래머, 디버거 등 다른 보조 프로그램들도 관심을 가져 보시기 바랍니다.

1. **하드웨어 연결하기**

STM32H750 보드에는 VCC, SWCLK, GND, SWDIO, RST, SWO 이렇게 6개의 핀이 펌웨어 다운로드 및 디버깅를 위해 존재합니다. VCC는 원래 ST-Link가 타겟 보드에 전원이 잘 들어와 있는 지 보기 위한 핀이지만, 이 핀으로 전원을 공급할 수도 있습니다. 저가의 짝퉁 ST-Link들은 편의를 위해 이 핀으로 전원을 공급하기도 합니다. VCC외에 꼭 연결해야 하는 것은 GND, SWDIO, SWCLK 이렇게 3개의 핀입니다. 나머지는 필요에 따라 연결 하시면 됩니다.

1. **데모 프로그램 실행해 보기**

이 문서와 함께 올린 데모 프로그램들을 다운받아 압축을 풀어 놓습니다. 그 중에서 가장 간단한 H750 LED Blink를 실행해 보겠습니다. 일반 윈도우 프로그램과 프로젝트를 여는 방법이 달라서 조금 당황하실 수도 있습니다.

폴더를 열고 들어가면 몇 개의 파일들이 보이는 데, STM32CubeIDE가 잘 설치되어 있다면 파일 앞에 [IDE]라고 써진 파란 네모에 아이콘이 3개 보이고, 그 중에 .project 라는 파일도 있을 것입니다. 이 것을 더블클릭 합니다. 그러면 STM32CubeIDE가 실행이 되면서 프로젝트를 로드 할 것입니다. 프로그램을 먼저 실행 시키고 File->Open으로 .project 파일을 선택해서 열면 파일 내용만 보여줄 뿐 프로젝트 구조가 열리지 않습니다.

다른 방법으로는 STM32CubeIDE를 먼저 실행시키고 메뉴의 File ->Open Project from File System… 을 실행한 다음 프로젝트 파일들이 있는 폴더를 선택해 줍니다.

C/C++ Projects 창에 H750 LED Blink 가 표시되고 하위 폴더 구조들이 표시되면 프로젝트를 잘 불러온 것입니다. Core 밑의 Src 폴더를 들어가 보면 main.c를 비롯하여 반가운 파일들이 보입니다. H750 LED Blink.ioc 처럼 .ioc 파일을 더블 클릭하면 내장 CubeMX가 실행되어 그래픽 환경으로 I/O 등을 설정할 수 있습니다.

도구상자의 ‘망치’는 컴파일 및 빌드하는 것이며, ‘벌레’는 소스를 빌드하여 타겟 보드에 다운로드 한 후 디버깅을 시작합니다. ST-Link와 STM32H750 보드를 연결하고 ‘벌레’를 눌러봅니다. 중간에 한 두 개의 창이 뜨기도 하겠지만 읽어 보시고 넘어가면, 펌웨어를 다운로드 한 후 main()의 첫 명령에서 디버깅 지시를 기다리고 있을 것입니다. 일단 도구상자에서 빨간 네모를 클릭하여 디버깅을 멈추면 다운로드 된 펌웨어가 실행됩니다. 보드의 하얀 LED가 1초 간격으로 깜빡일 것입니다. 1초에 1000번 불려지도록 설정된 **void** **HAL\_SYSTICK\_Callback**(**void**) 라는 함수에서 매번 500을 세고 HAL\_GPIO\_TogglePin(LED\_GPIO\_Port, LED\_Pin); 에서 LED애 연결된 핀을 토글 시키는 것이 이 프로그램의 전부입니다.

아주 간단한 기능을 구현 하였을 뿐이지만 이것으로 모든 기초는 놓은 것입니다. 이제 여기 저기를 탐험하면서 기능을 익히십시오.

1. **첫 프로젝트 시작하기**

STM32CubeIDE를 실행하면 이전에 종료했을 때의 환경을 복원합니다. 여기서 새로운 프로젝트를 또 시작하여 여러 개의 프로젝트를 동시에 운영할 수도 있습니다만, 깔끔하게 프로젝트를 내리고 싶은 때는 프로젝트 이름을 우클릭 하거나 Edit 메뉴에서 Delelte를 선택할 수 있습니다. 대화상자에서 □Delete project contents on disk(cannot be undone) 에 체크하면 프로젝트가 있던 폴더와 내용을 모두 지워버리니 주의하시기 바랍니다.

메뉴의 File -> New -> STM32 Project 를 선택하면 모든 STM32 MCU 를 선택할 수 있는 창이 실행됩니다. 많은 리스트 중에 원하는 것을 일일이 찾을 수도 있겠지만 돋보기 옆의 입력 창에 h750v 를 한 글자씩 입력해 가면서 필터링 된 것 중에 STM32H750VBTx 를 고르시면 됩니다. 이어지는 창에서 프로젝트 이름과 환경을 설정하고 프로젝트를 시작하시면 됩니다. 제공되는 회로를 참고하여 필요한 패리페럴 등을 설정하고 코딩합니다.

회로와 맞는 지 세심하게 검토하고 하나하나 설정해 가는 것은 많은 노력이 필요합니다. 실수할 가능성도 있습니다. 그래서 좀 더 편한 방법이 있습니다. 제공되는 Temp 프로젝트를 활용하는 것입니다. Temp 프로젝트는 개발보드에 탑제된 기능들의 설정만 했을 뿐 아무런 코딩도 더하지 않은 것입니다. 그러므로 Temp 프로젝트를 올린 다음 필요 없는 것을 지우고 더하여 설정을 완성하는 것입니다. 프로젝트 이름도 Rename 기능을 이용하면 원하는 것으로 바꿀 수 있습니다.

Temp 프로젝트를 이용하면서도 더 기본적인 설정부터 내가 직접 하고싶다면 메뉴에서 File -> New -> STM32 Project from STM32CubeMX .ioc File 을 실행하고 제공되는 Temp.ioc 파일을 선택하시면 됩니다. 저는 이 방법을 추천합니다.