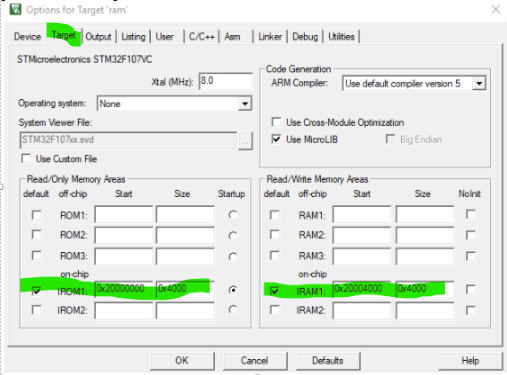
**Title Document: How to run&debug from RAM without flashing flash STM32 devices (Keil)**

If you would like to debug&run program without flashing the device you can follow this procedure:

In this solution, You don't need to Configure the boot pin. not matter what Boot1, Boot0 are, It just works

step 1. Project->Options for Target STM32...->Target

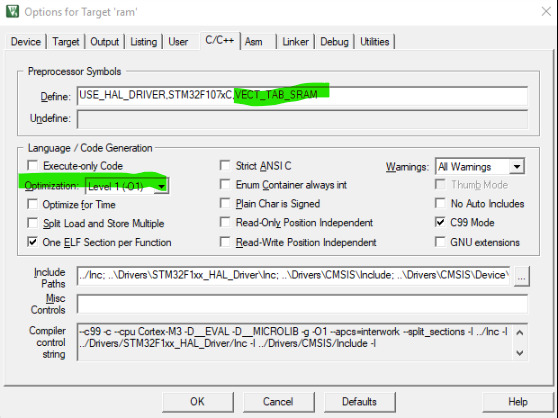


ROM is from 0x2000 0000 - 0x2000 4000 (16KB)

IRAM is from 0x2000 4000 -0x2000 8000 (16KB)

The exact value of ROM and RAM depend on your application.

step 2. Project->Options for Target STM32...->C/C++

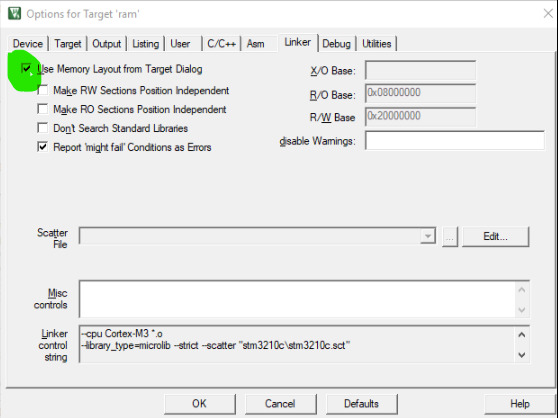


-add the VECT\_TAB\_SRAM to the define field

-optionally you can change the Optimization to O0 or O1 when you want to avoid optimization at O3 level (this helps debugging code)

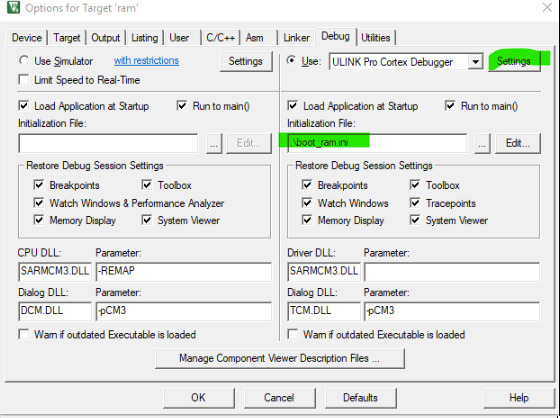
step 3.

Project->Options for Target STM32...->Linker



-make sure the Use Memory Layout from Target Dialog is checked.

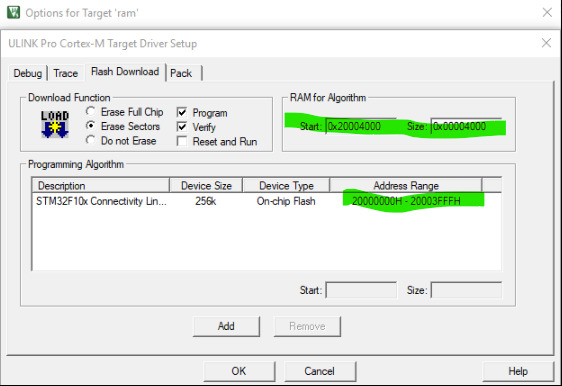
step 4. Project->Options for Target STM32...->Debug

-

Copy the boot\_ram.ini file to the project folder (e.g. [C:\Users\..\MDK-ARM)](file:///C:\MDK-ARM))

-select the boot\_ram.ini as initialization file

-go to settings



make sure the value high lighted in green is exactly same as step 2 we have made.

Up until now, you are done!

To test it work or not, build the project then press Ctrl + F5.

Here is the boot\_ram.ini file.

add the code below and save as **boot\_ram.ini** file.

|  |
| --- |
| /\*----------------------------------------------------------------------  \* Setup() configure PC & SP for RAM Debug  \*----------------------------------------------------------------------\*/  FUNC void Setup (void) {  SP = \_RDWORD(0x20000000); // Setup Stack Pointer  PC = \_RDWORD(0x20000004); // Setup Program Counter  \_WDWORD(0xE000ED08, 0x20000000); // Setup Vector Table Offset Register  }  FUNC void OnResetExec (void) { // executes upon software RESET  Setup(); // Setup for Running  }  load %L incremental  Setup(); // Setup for Running  g, main |