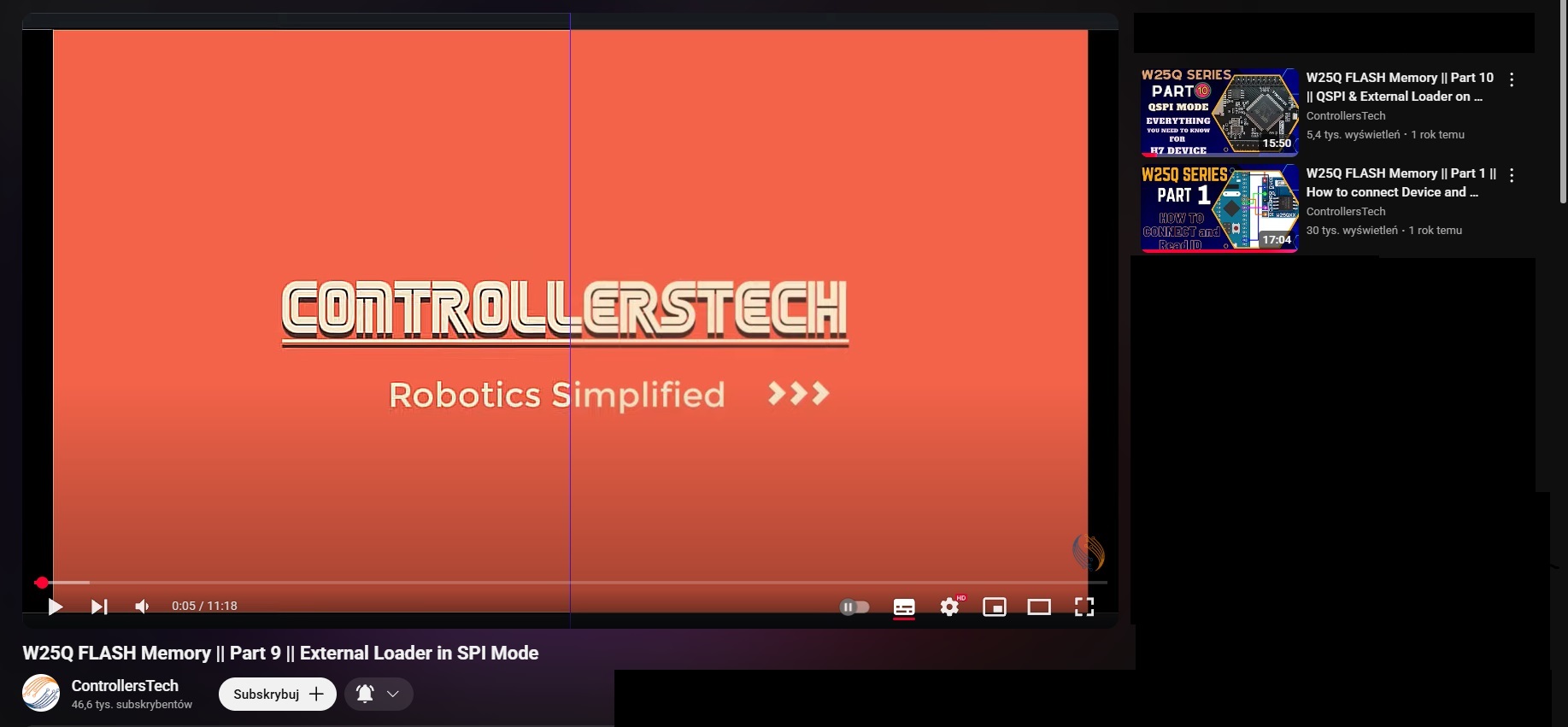
BASED ON THE BLUE PILL SPI EXTERNAL FLASH LOADER.

At the first need send many thanks to Controllerstech which explain this subject on the him YT channel:

https://www.youtube.com/watch?v=Gil\_LbT5UVw



First you need make new STM32 project for STM32F103C8T6 and don’t forgot about exist 2 blue pills – old blue pill and new blue pill PLUS. Which can have W25Qxx SPI FLASH on board. Also if you want blinking led it have LED on the PC13 for old blue pill. New blue pill PLUS got the led on PB2.

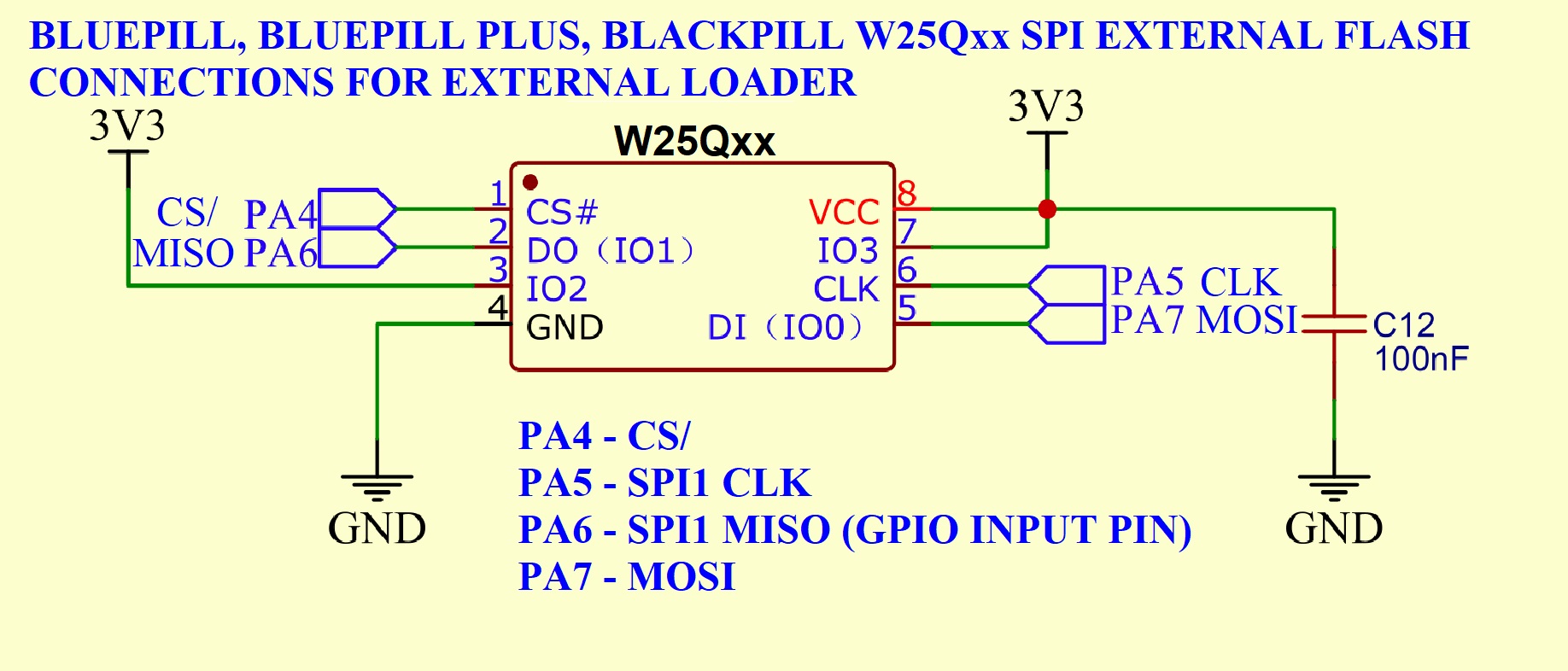
- FLASH is connected to SPI1:

PA4 as CS/

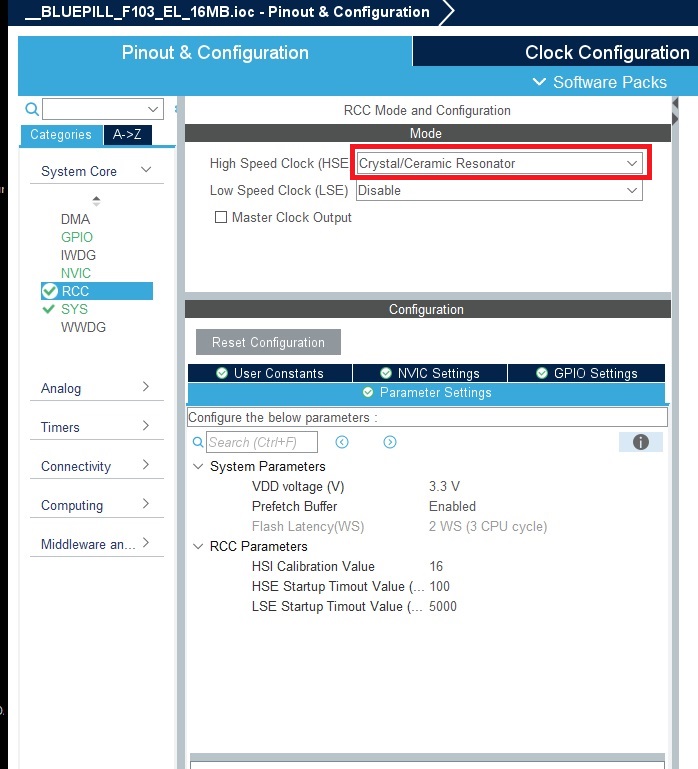
PA5 as SPI1 CLK

PA6 as SPI1 MISO (GPIO INPUT PIN)

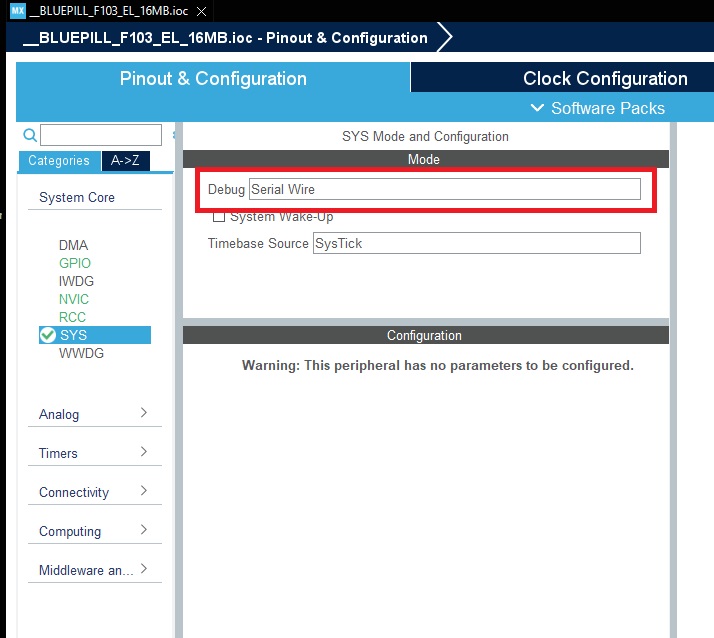
PA7 as MOSI



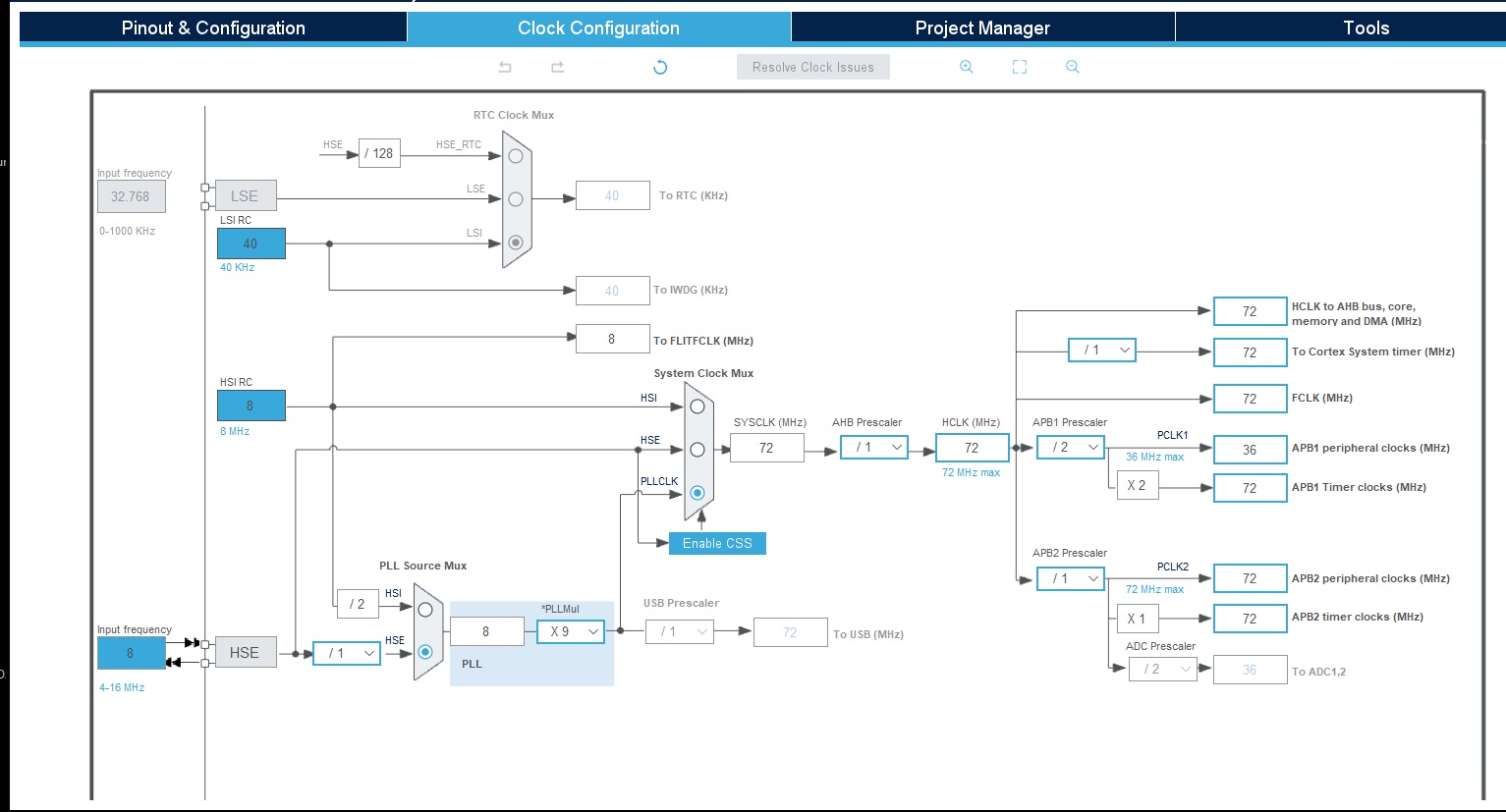
So let create a new STM32 project for STM32F103C8T6 and enable RCC:



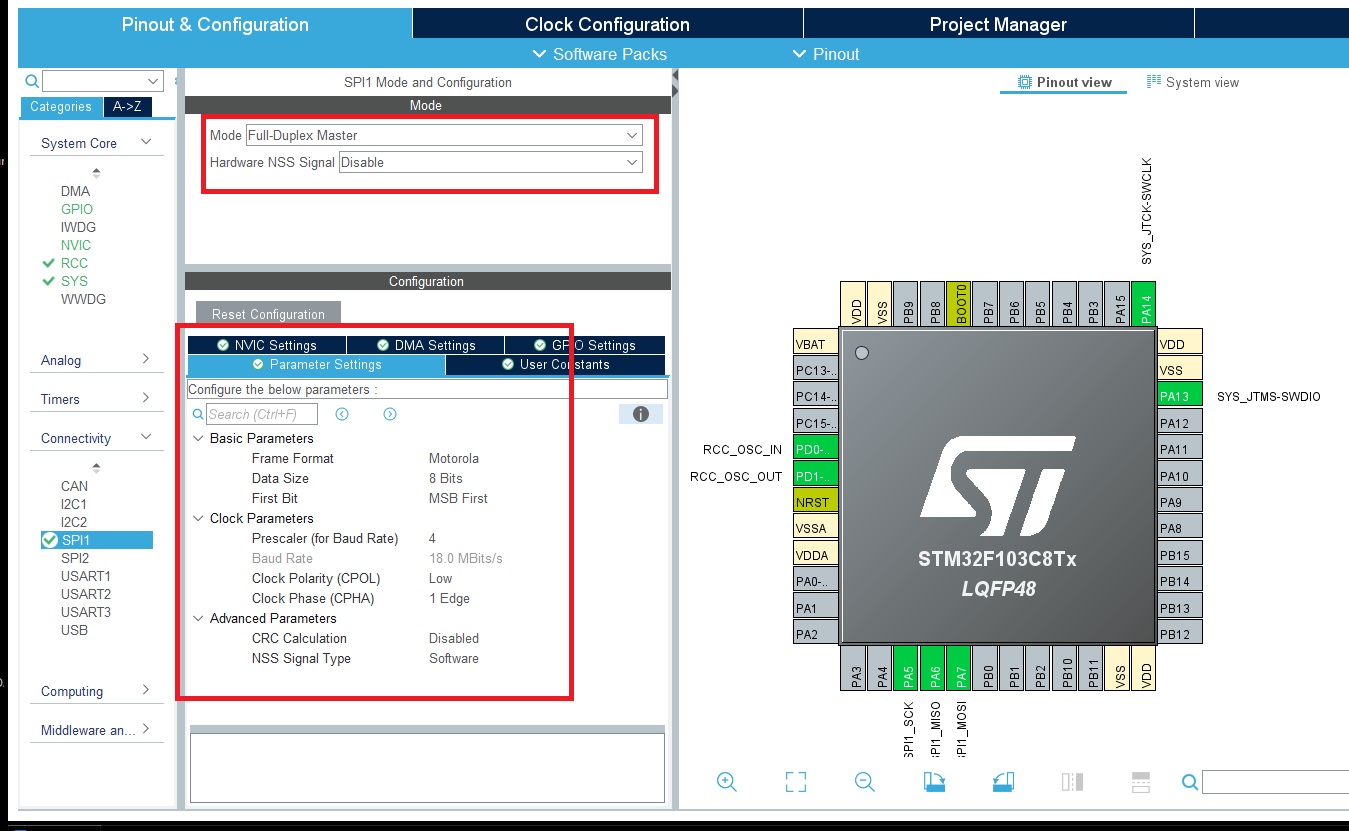
Enable Debug Serial Wire in SYS section.



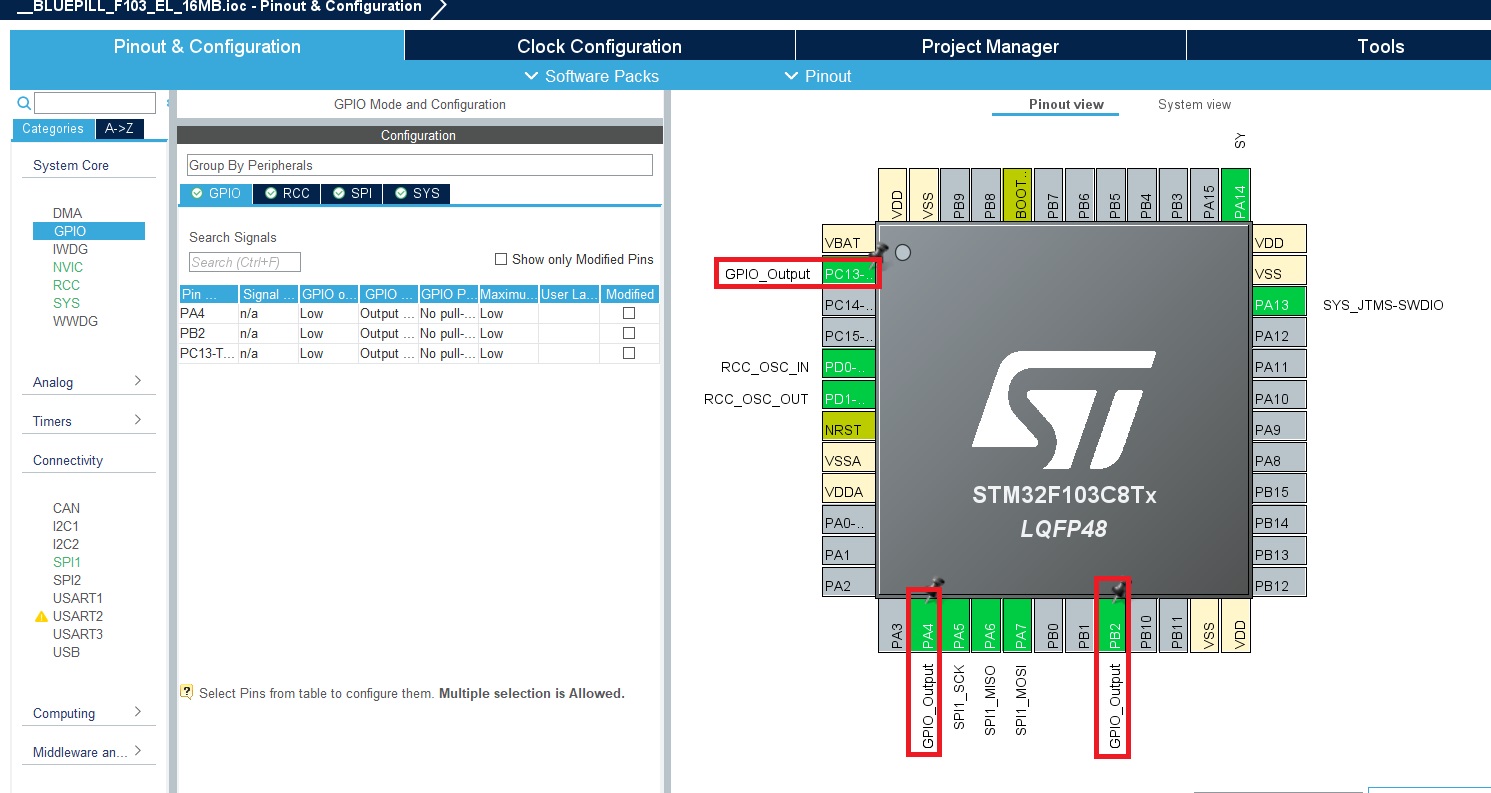
Now set the system clock:



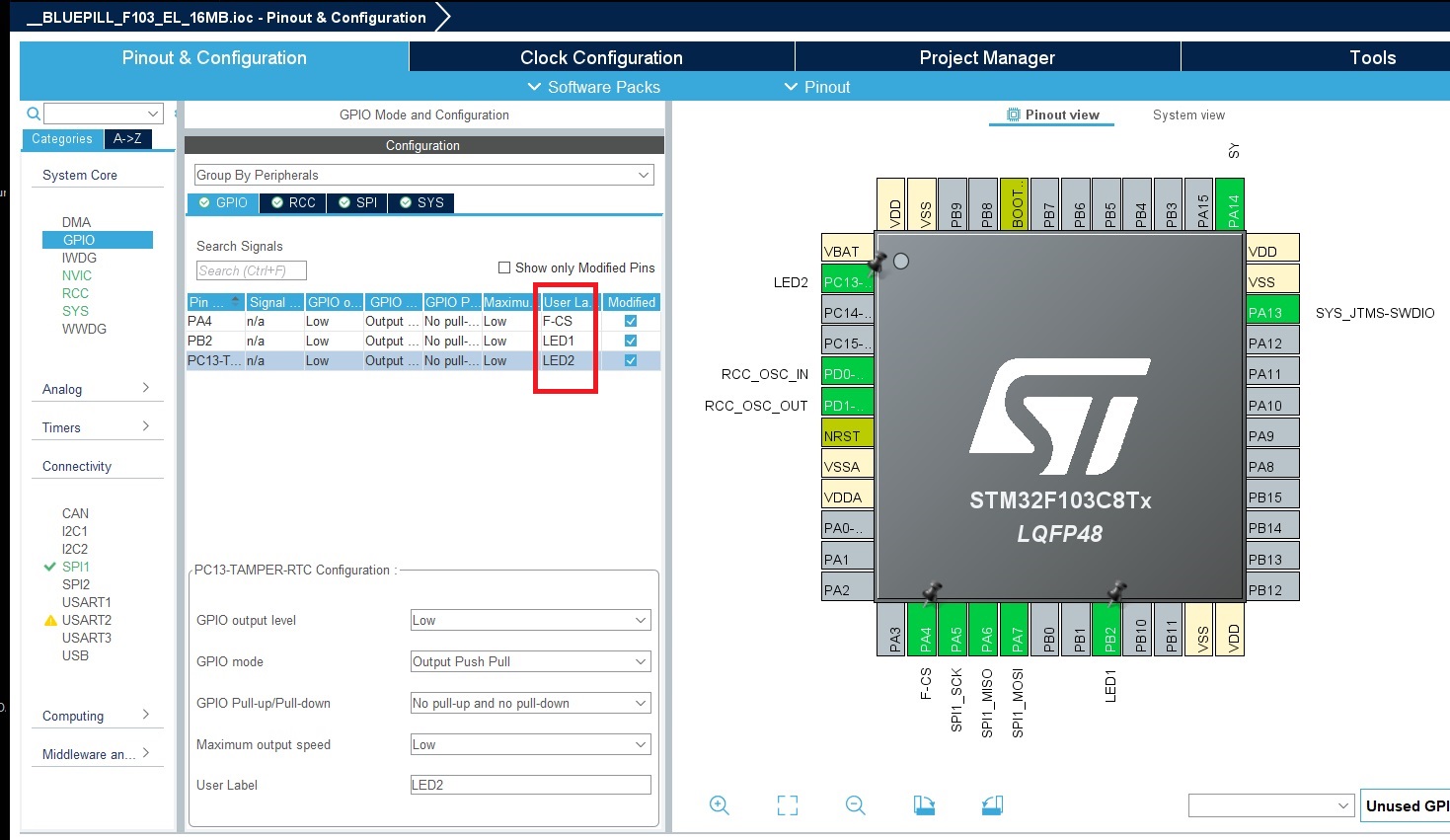
And enable SPI1:



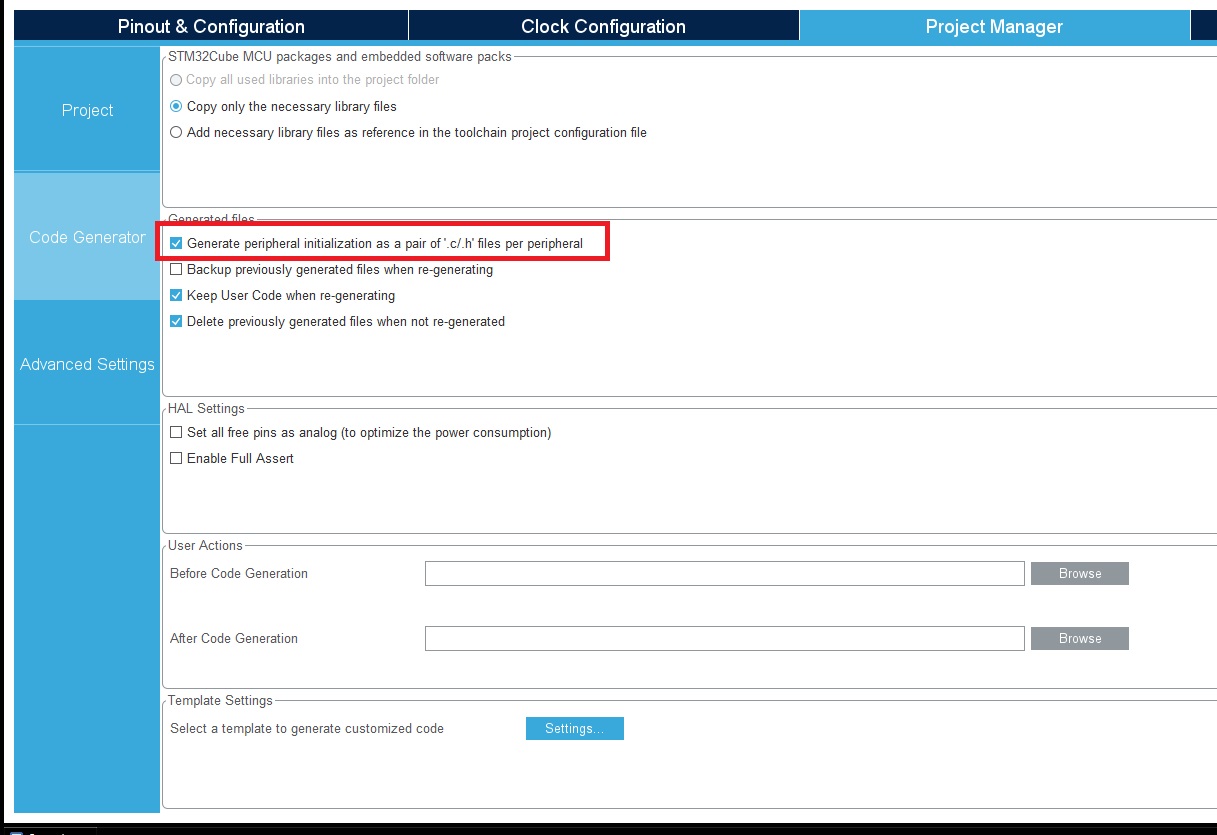
Enable PA4, PB2 and PC13 as GPIO Output:



Set the User labels for this 3 pin as follow:



In Project Manager set the genrate peripherial initializations files as pair c/h:



And generate project.

Now you need copy 6 files into the project.

Into the **root** folder copy file:

**linker.ld**

Into the **inc** folder:

**Dev\_Inf.h**

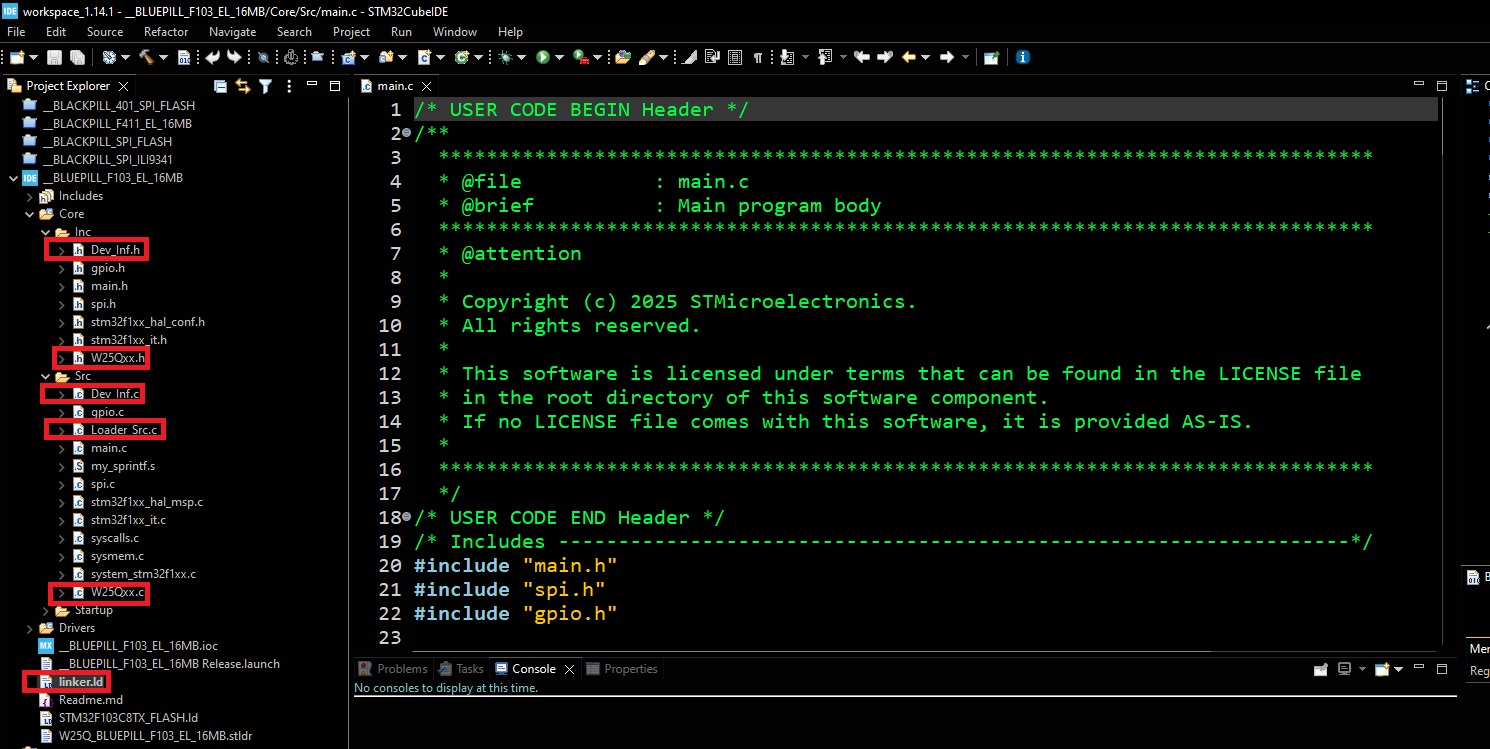
**W25Qxx.h**

Into the **src** folder:

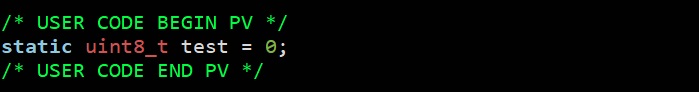
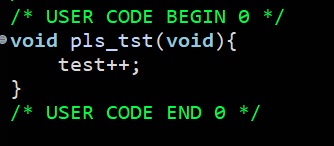
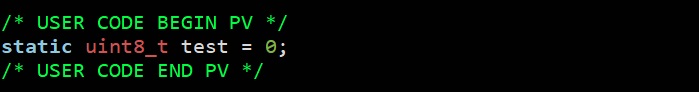
**Dev\_Inf.c**

**Loader\_Src.c**

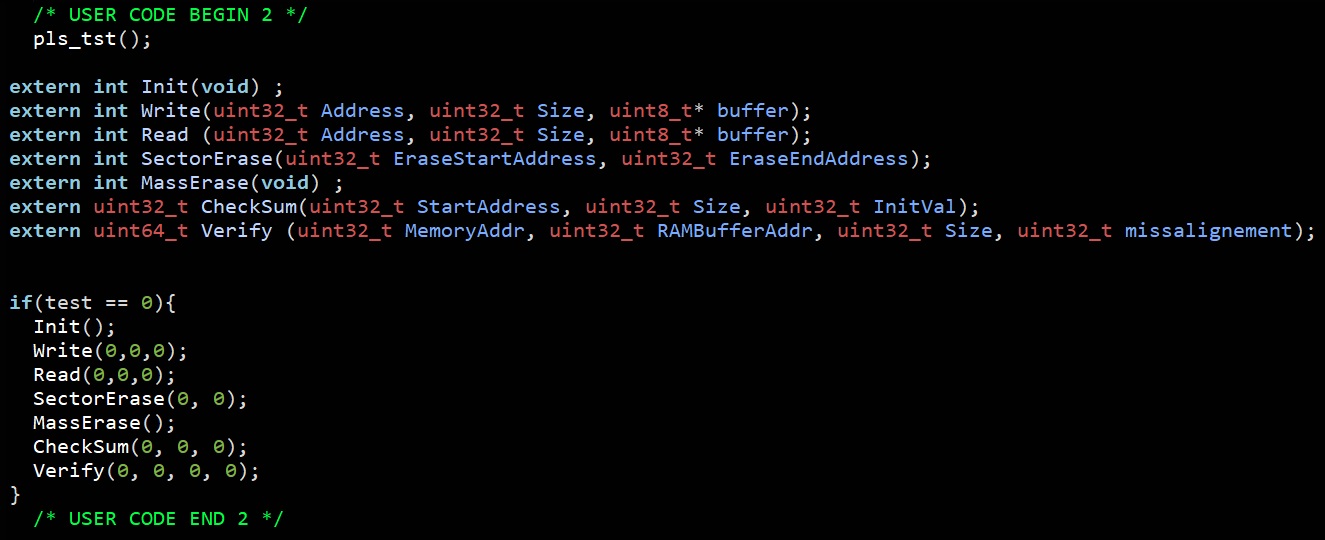
**W25Qxx.c**



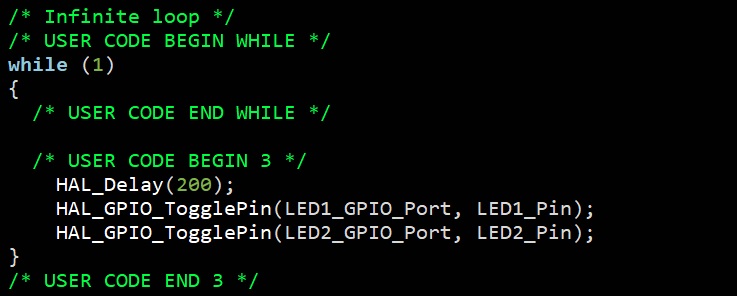
Now need add a few changes in main.c file:



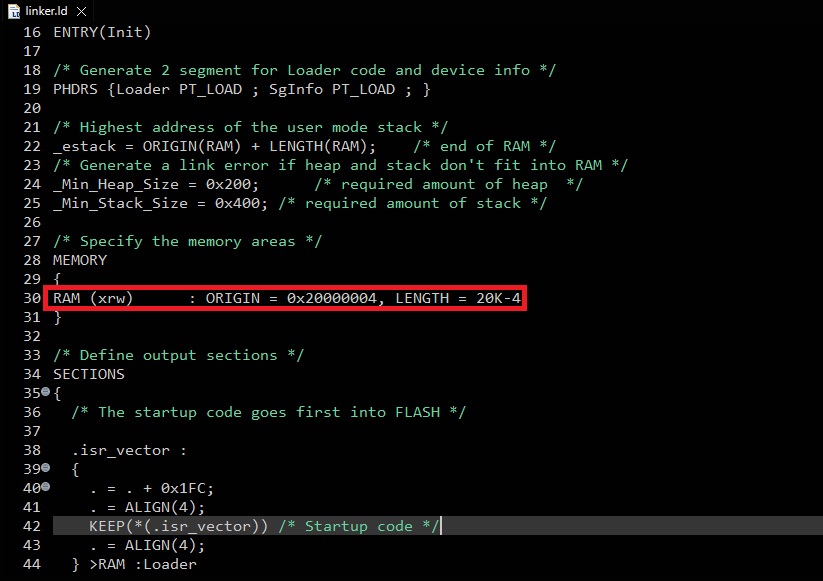
For minimize output file and cheating compiler:



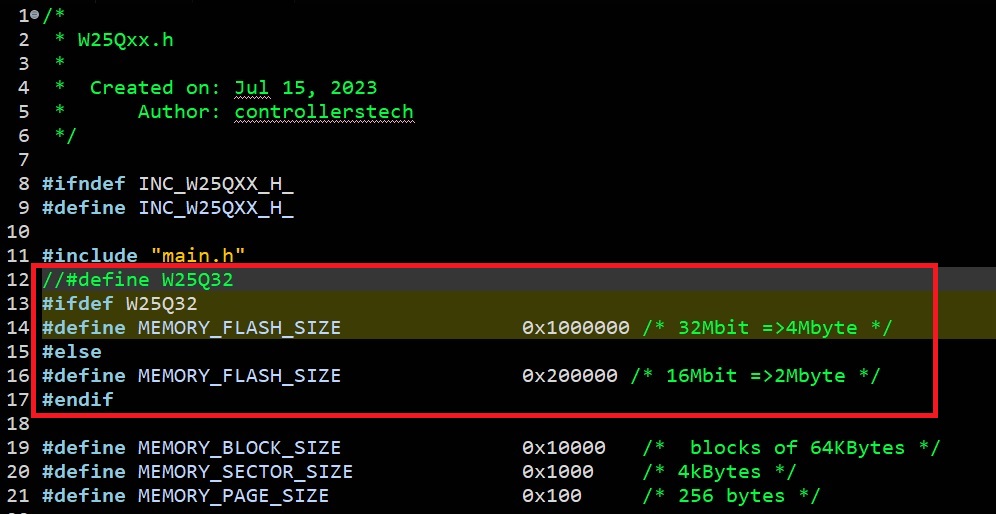
Last one in main.c not mandatory – when loader working this code never reaching this point.



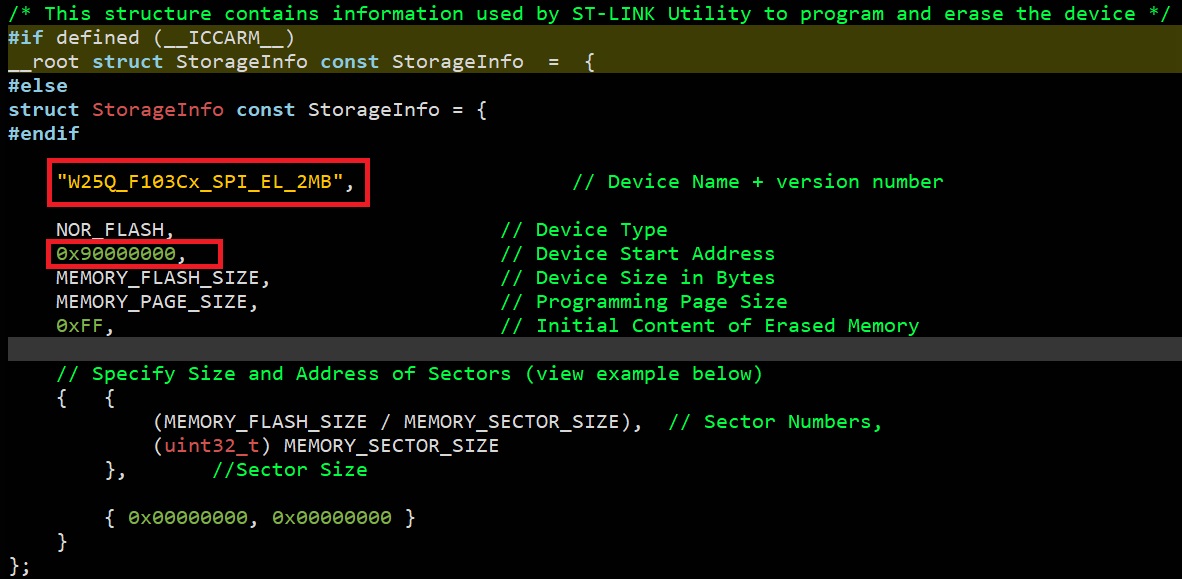
In the **linker.ld** file need only one change to do:



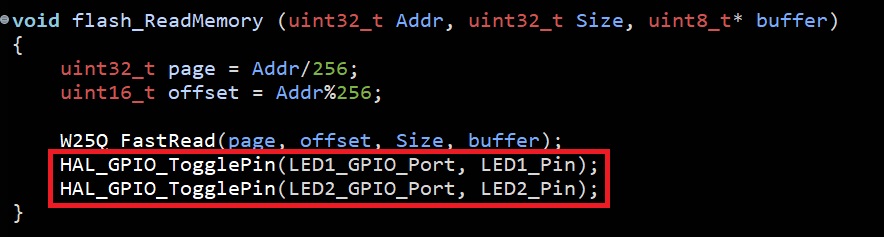
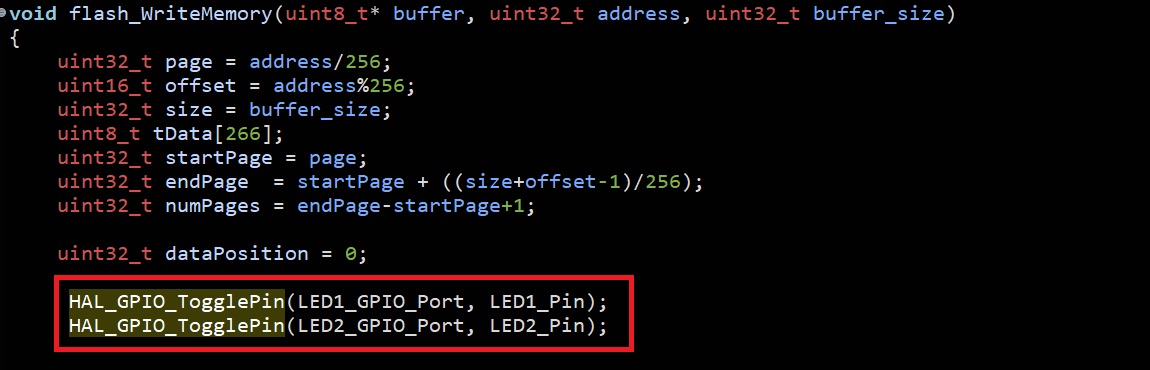
In the W25Qxx.h file take care about this lines for 2MB flash:



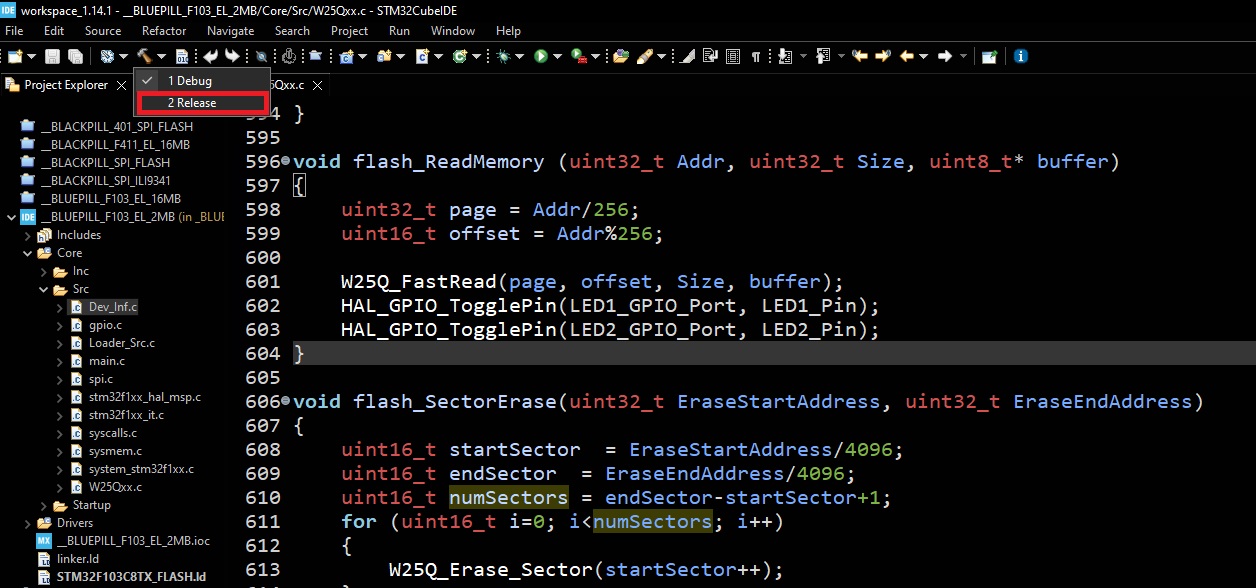
For Dev\_Inf.c file need change:



In file W25Qxx.c two changes for led blinking under work external SPI flash loader:



After all of this from hammer icon choose release build:



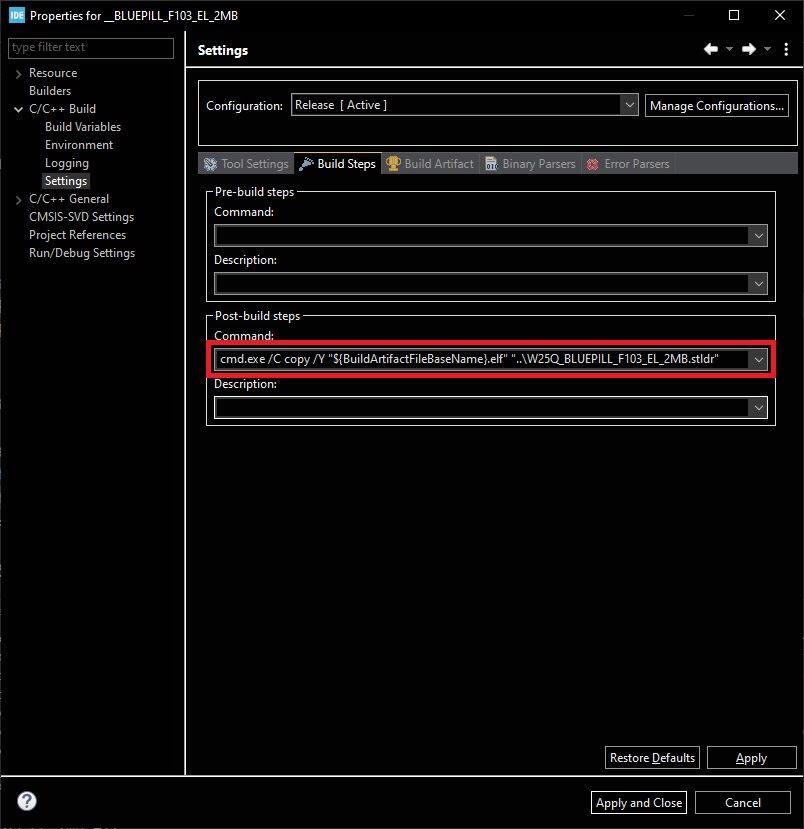
The loader should be build like bellow:



In Project->Properties->C/C++ Build->Settings->Build Steps->Post-build steps->Command:

Put the command like:

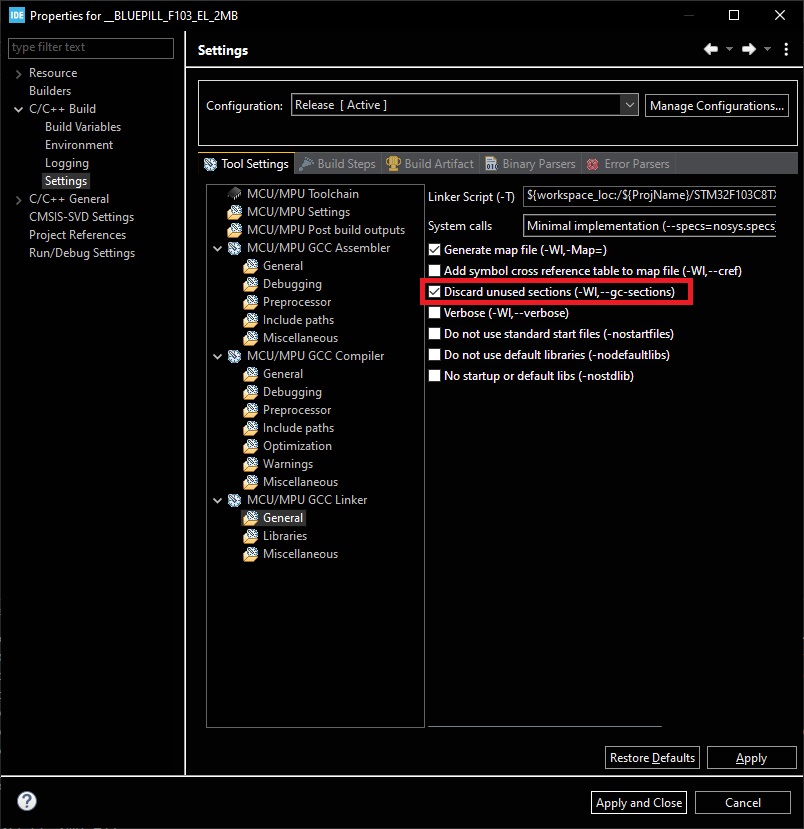
cmd.exe /C copy /Y "${BuildArtifactFileBaseName}.elf" "..\W25Q\_BLUEPILL\_F103\_EL\_2MB.stldr"



And click **Apply and Close** buton.

In Project->Properties->C/C++ Build->Settings->Tool Settings->MCU/MPU GCC Linker->General

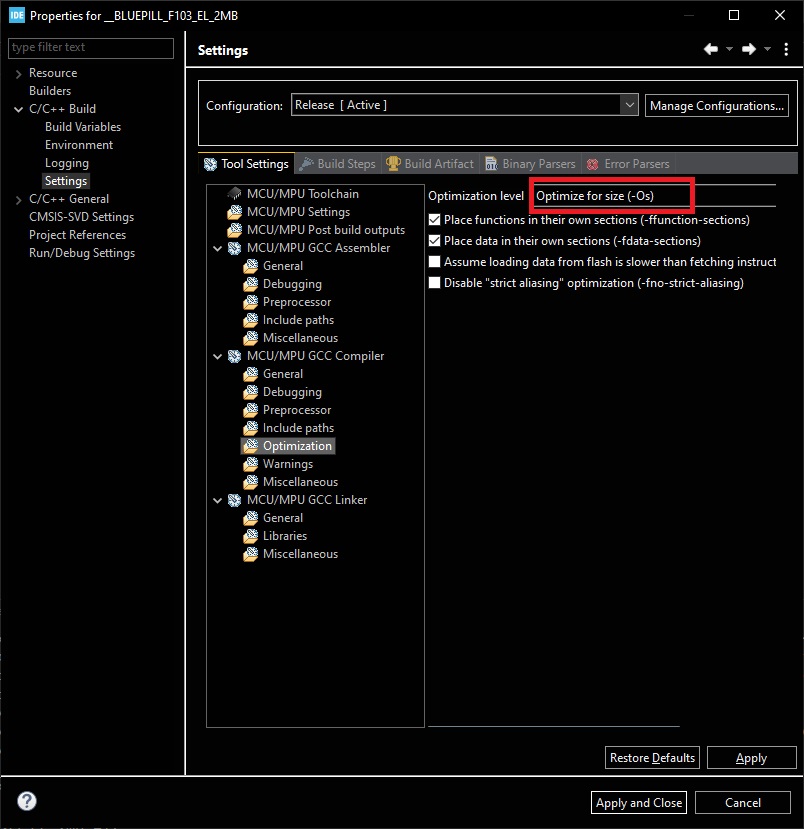
Make sure the „Discard unused sections (-Wl, --gcc-sections)” is checked for make smallest as possibility output execute file.



And click **Apply and Close** buton.

In Project->Properties->C/C++ Build->Settings->Tool Settings->MCU/MPU GCC Compiler->Optimization

Make sure the Optimization level is set to: „Optimize for size (-Os)” is choosed for make smallest as possibility output execute file.



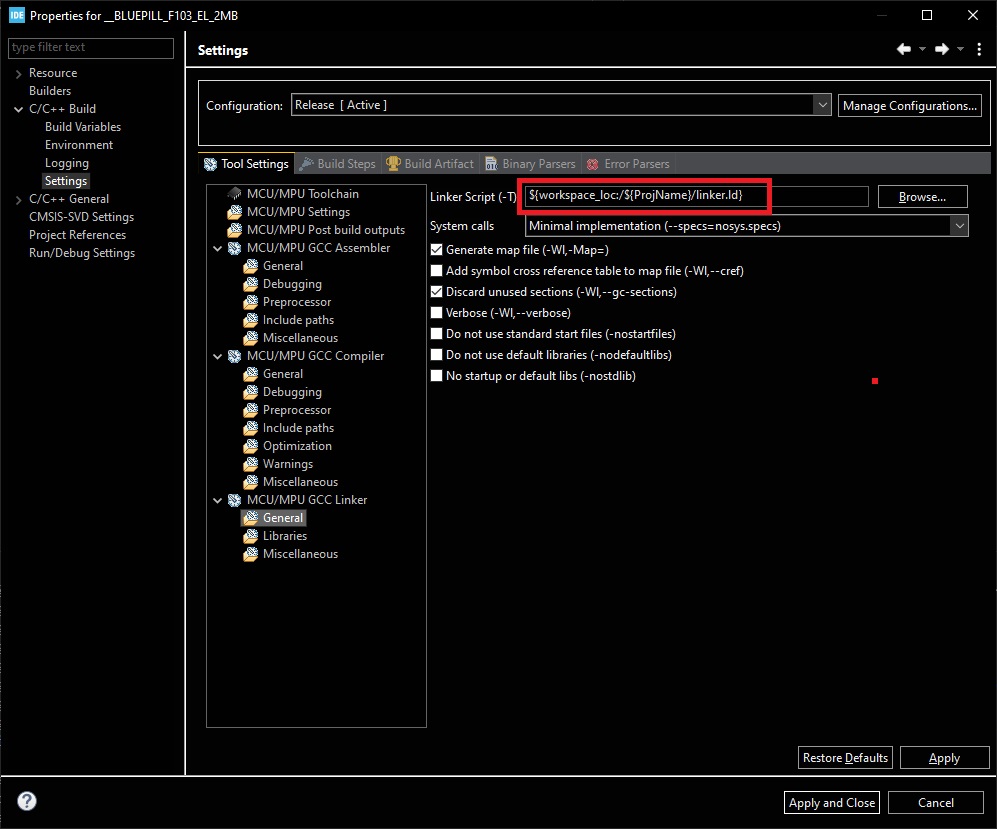
And click **Apply and Close** buton.

In Project->Properties->C/C++ Build->Settings->Tool Settings->MCU/MPU GCC Linker->General

In the Linker Script (-T) text edit field put the text:

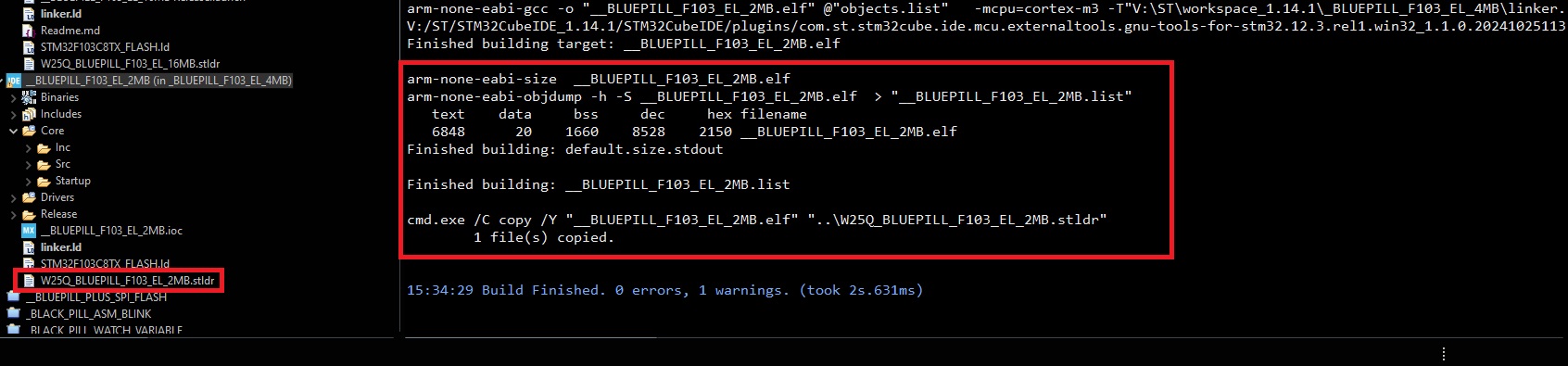
${workspace\_loc:/${ProjName}/linker.ld}

Which choose our **linker.ld** file.

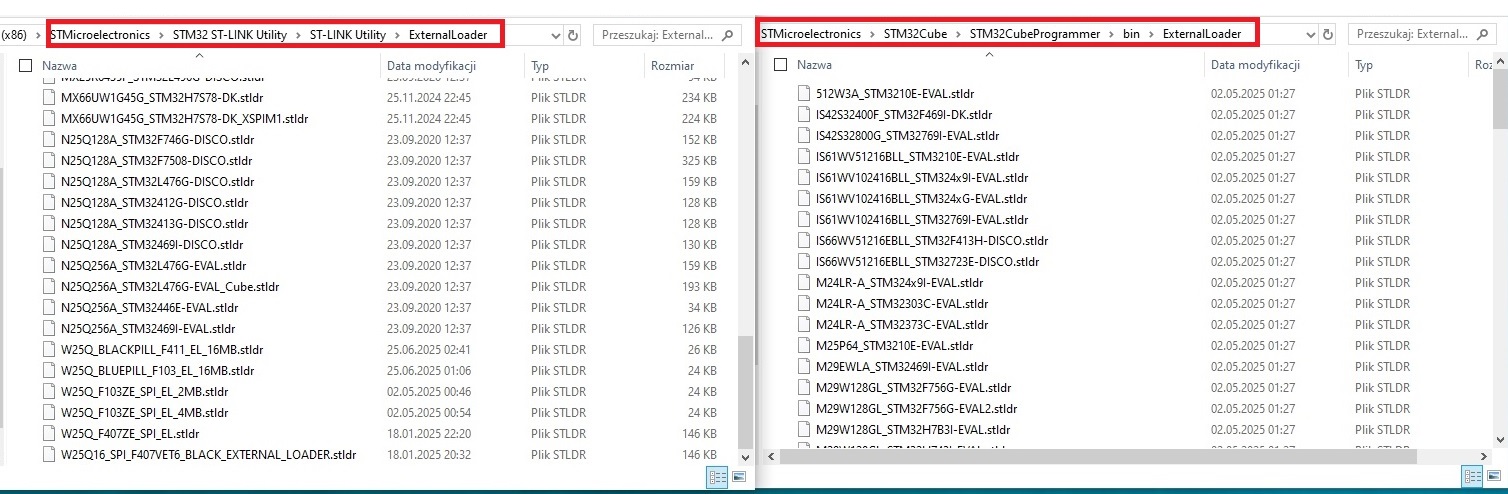


And click **Apply and Close** buton.

Now from menu **Project** choose **Clean** option, and after this choose **Build Project** option. What should gave final effect:



At the end you need copy the stldr file int the ST-Link Utility or STM32CubeProgrammer folder for external loaders.



After this you can use this external loader when you choose him in EL menu.