How to create a File System on a SD card using STM32CubeIDE

ST AME Support NF
ST Employee

on 2021-09-24 12:17 AM - edited on 2023-12-15 12:24 AM by ADMIN **Laurids_PETERSEN**

How do I Create a File System on a SD card using STM32CubeIDE?

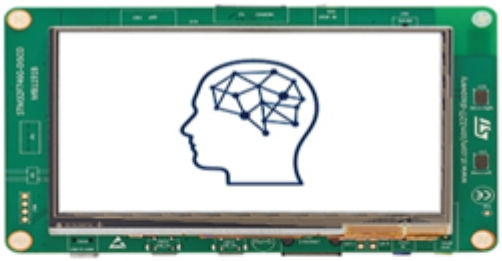
Introduction

Data logging applications require storing large amounts of data over a period of time. SD cards are a convenient solution for storing data and many STM32 products include the proper hardware interface. Using a standard file system to write data on an SD card ensures that the data is easily accessible on another device or computer. Adding a file system along with an SD card driver is easy to do using the various ST tools available for the STM32 family of microcontrollers.
This article shows you how to create a file system on a SD card using STM32 and ST Toolsets.

Pre-requisites

STM32 Discovery Kits as well as evaluation boards include an SD card socket. While this example uses the STM32F746G-DISCO board, any other STM32 board with an SD card socket can be used along with appropriate changes made in the software configuration for a different target board.

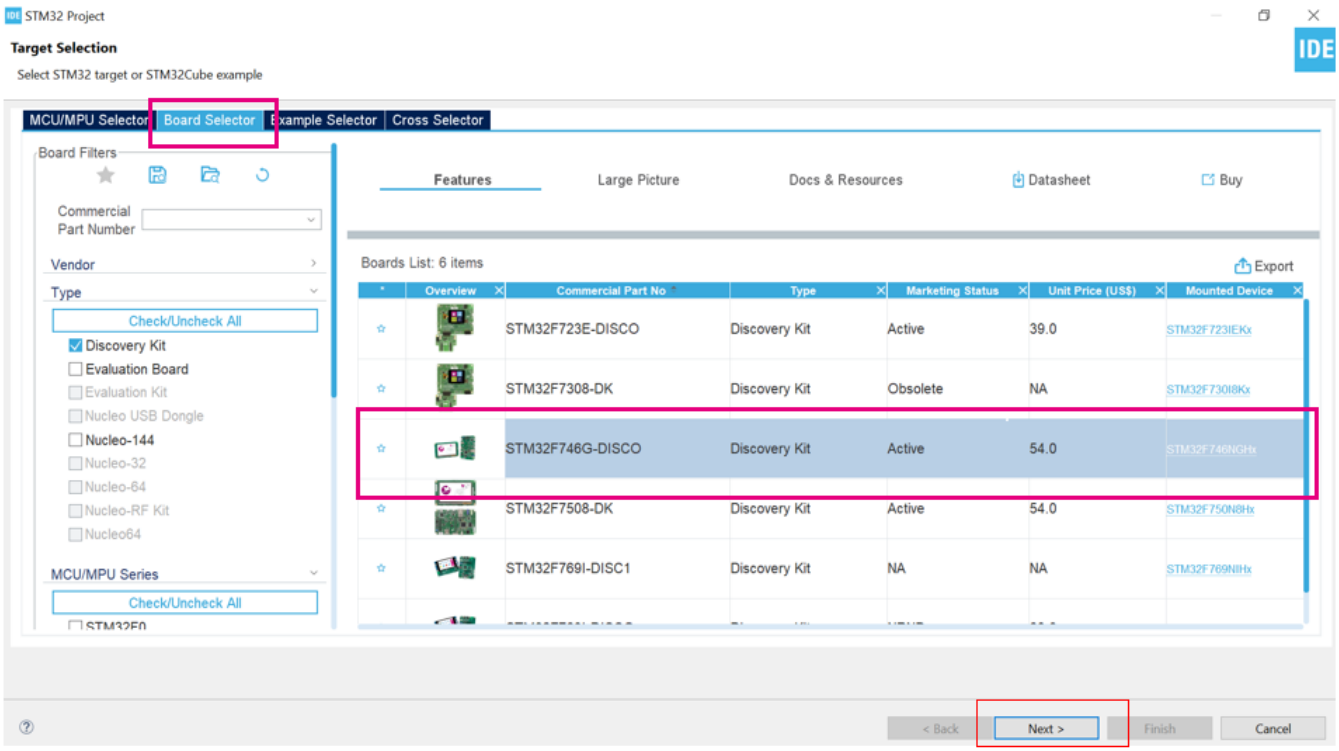
- Hardware: STM32F746G-DISCO



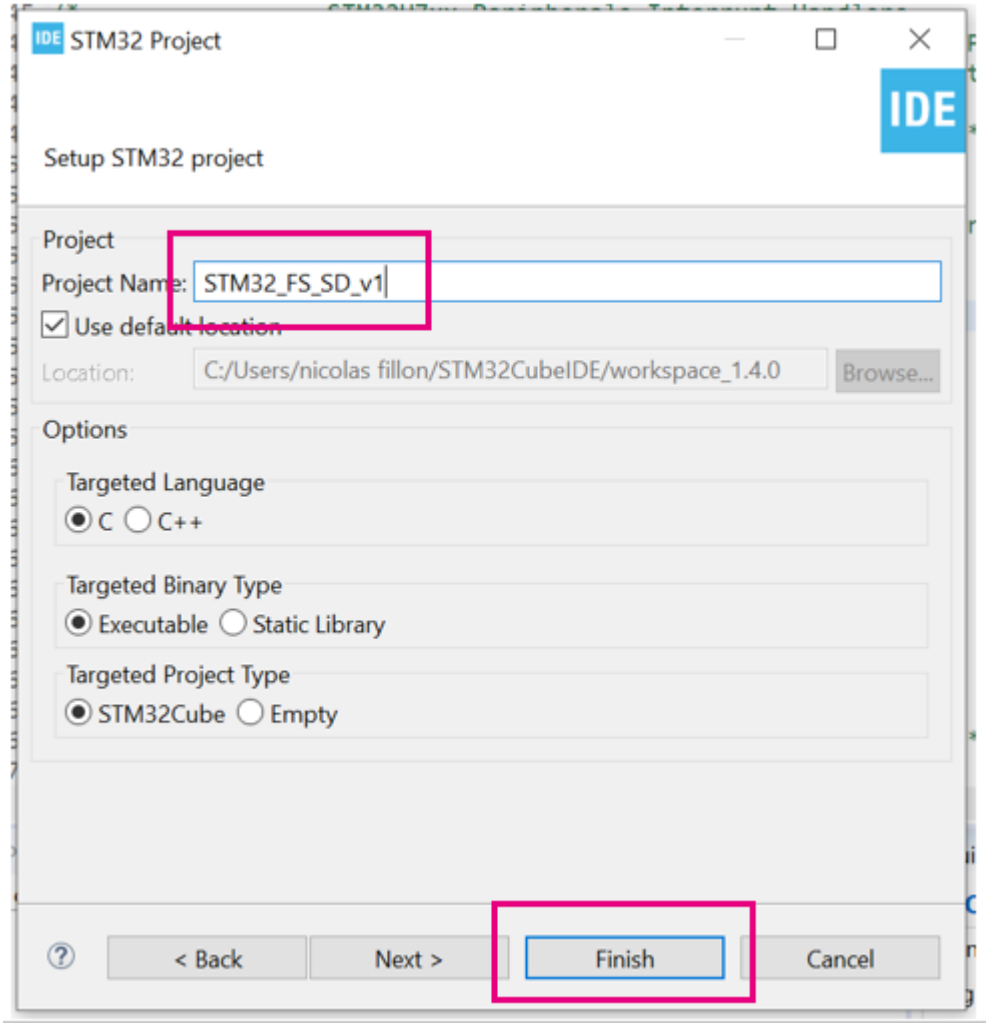
- Software: STM32CubeIDE

Steps

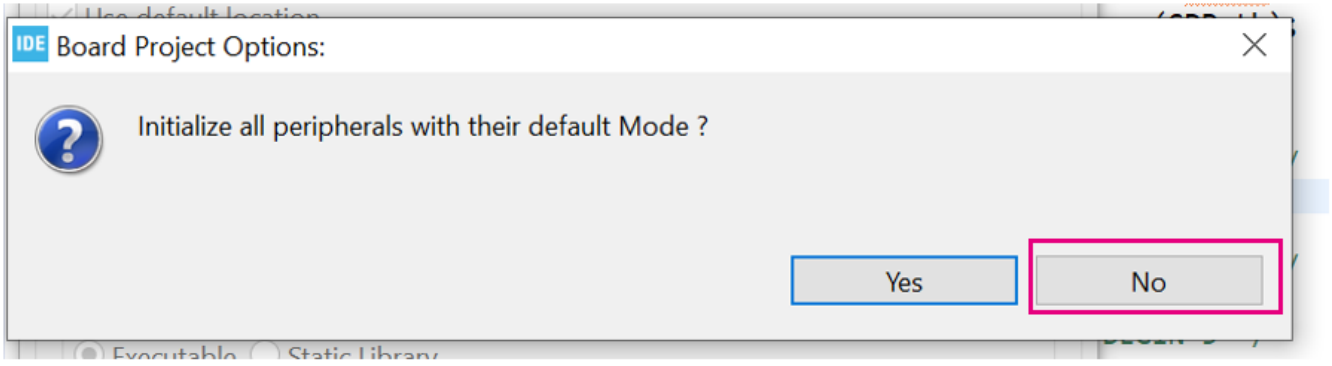
1. Open STM32CubeIDE
2. Create a new STM32 project and select the STM32F746G-DISCO template



3. Give a name to your project and then click finish

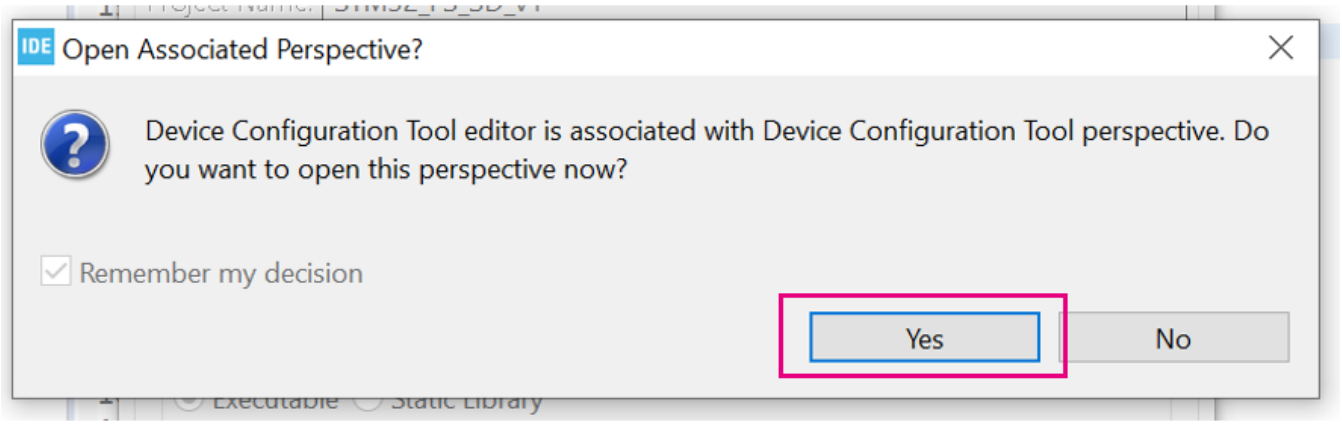


4. Select No for the Board Project Options



We will use pre-set peripheral configuration from the board configuration file.

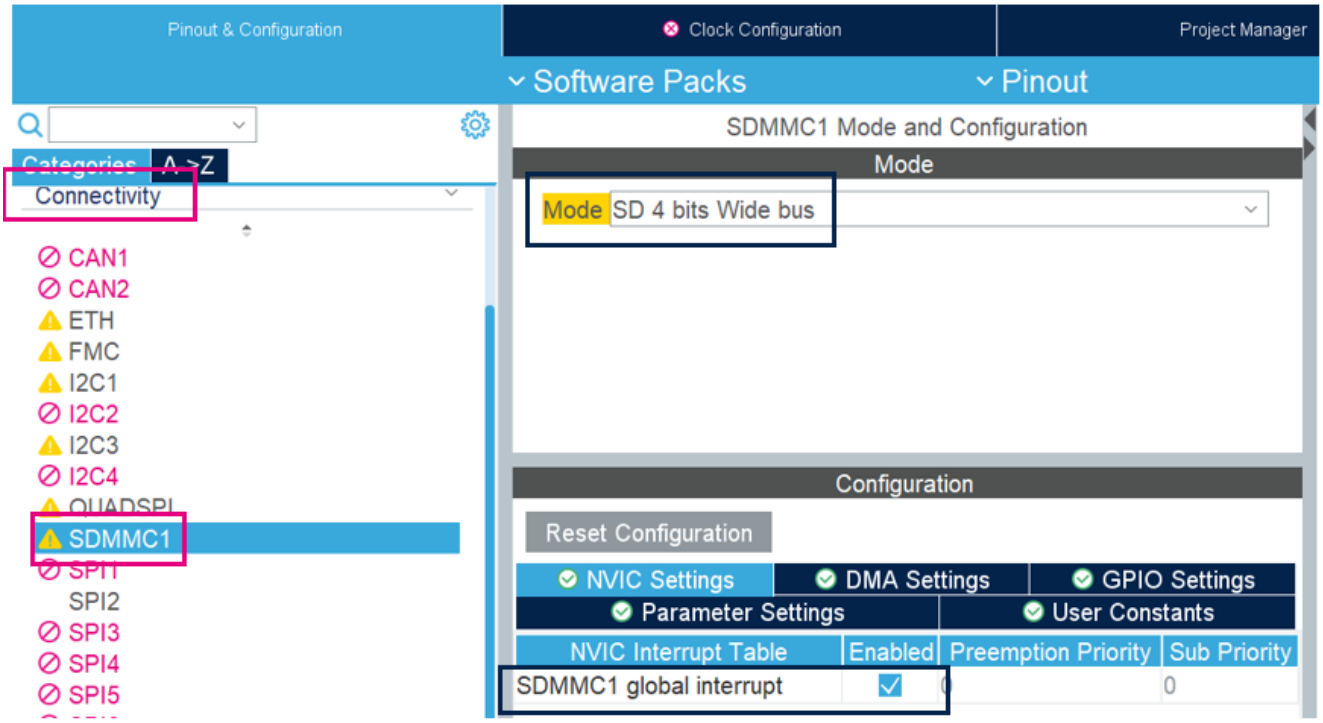
5. Open Associated Perspective, click Yes



6. SDMMC configuration

SDMMC is a peripheral that can be used to interface to a SD card.

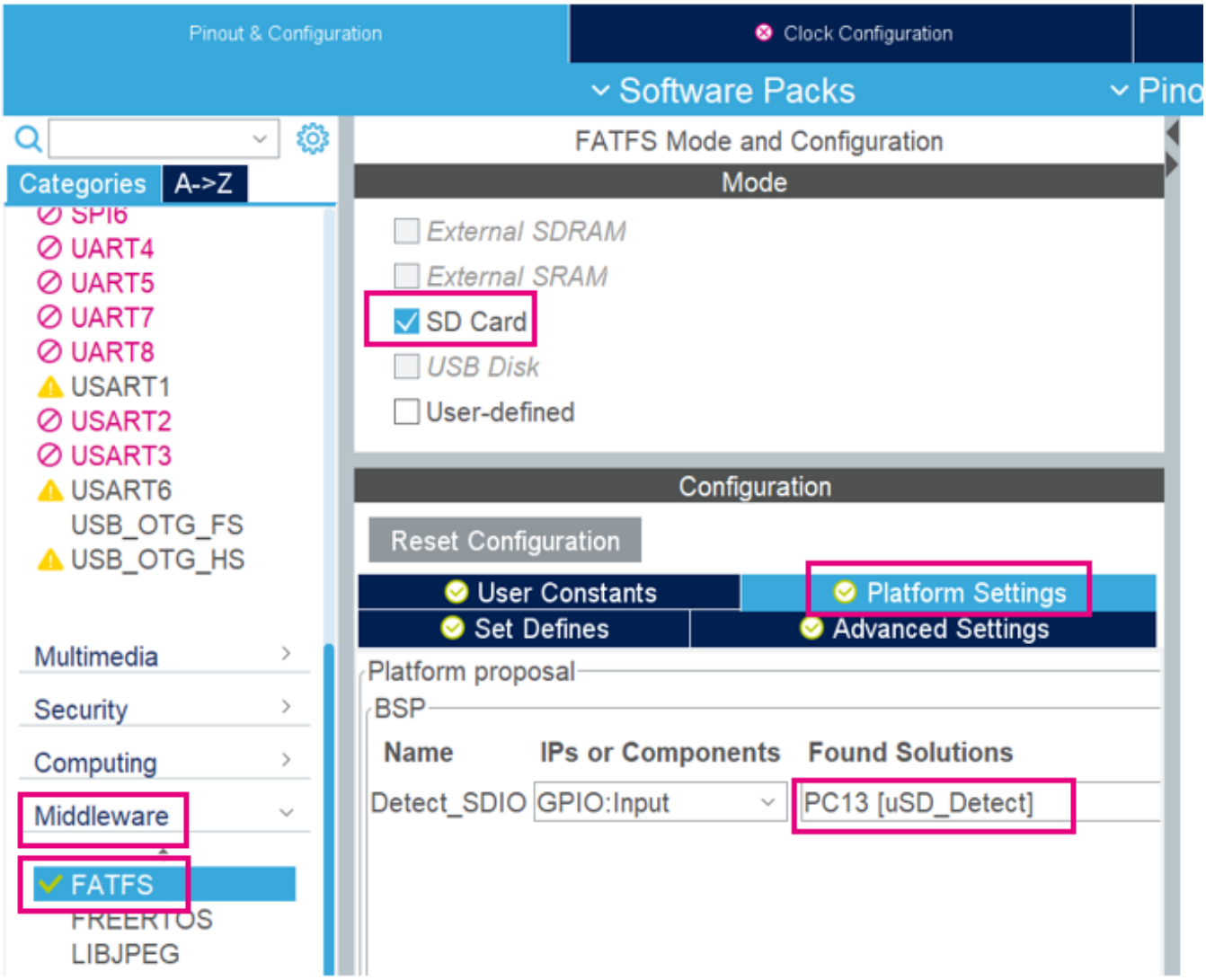
- Enable “SD mode” in “SDMMC1” in “Connectivity” and enable the global interrupt



7. FatFs configuration

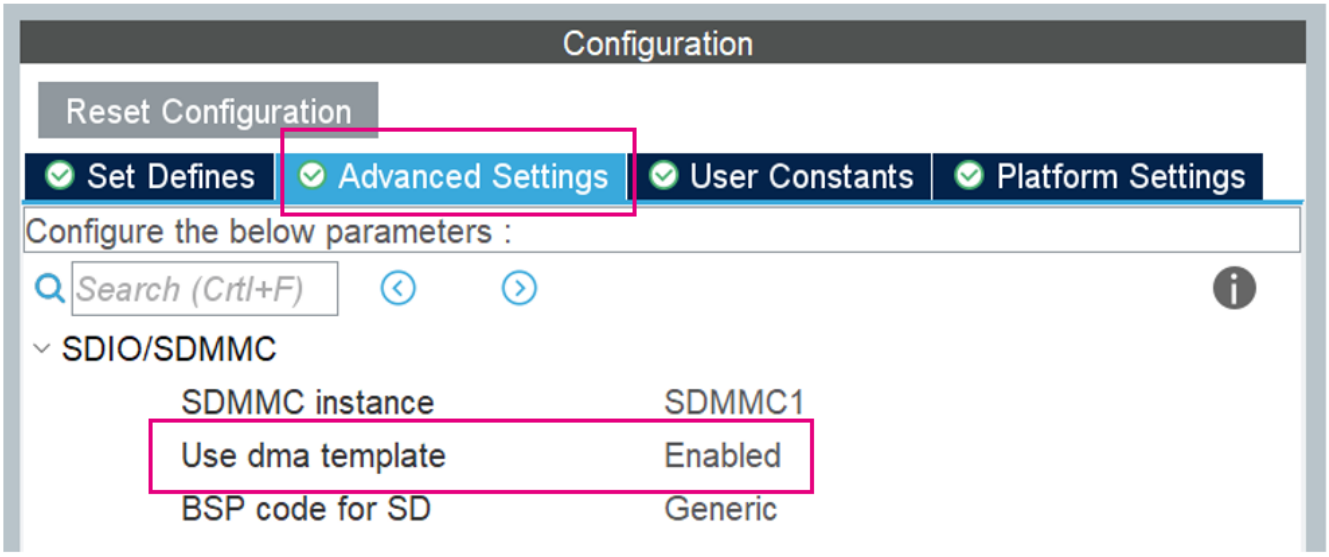
FatFs is an open-source file system middleware. This is integrated in STM32 Cube Libraries.

- Configure FatFs as SD Card mode in “MiddleWare”
- Select PC13 as Detect_SDIO in Platform Settings



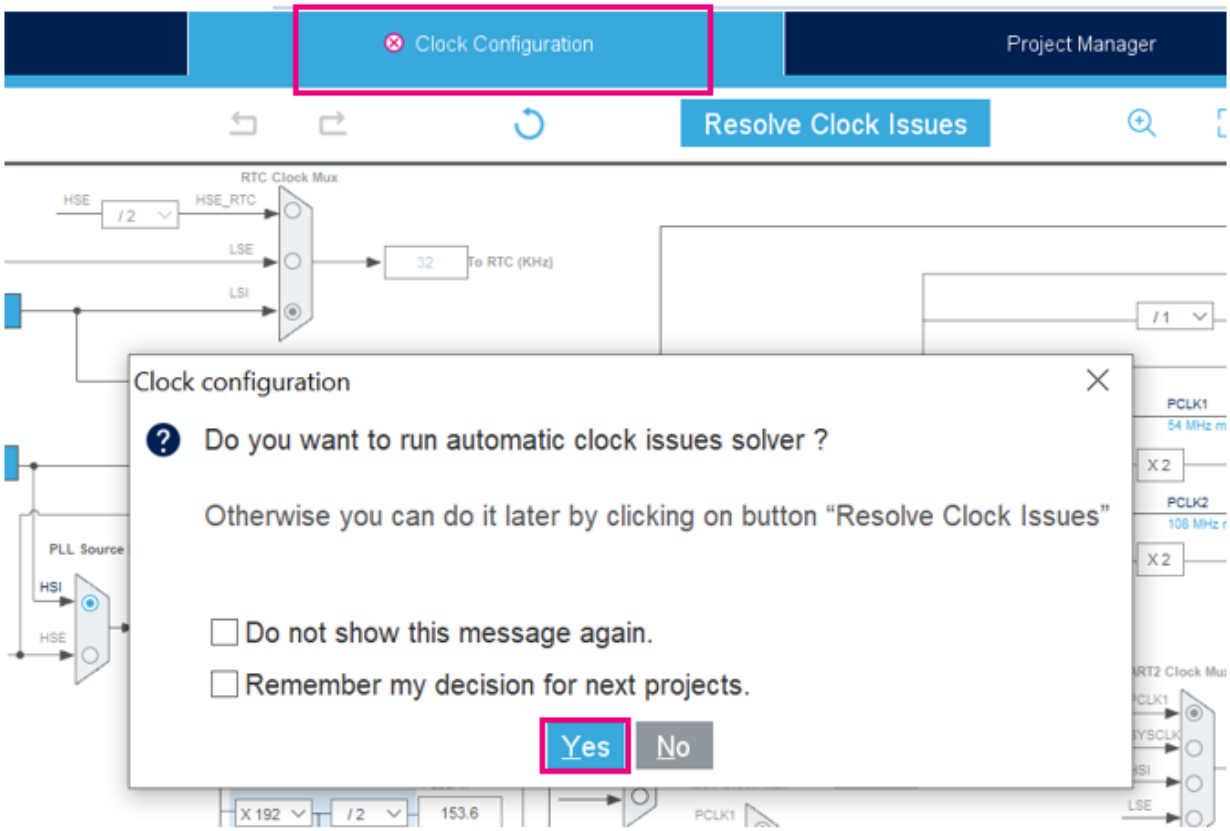
- Add DMA for TX and RX with default settings

This will permit to achieve best performance.



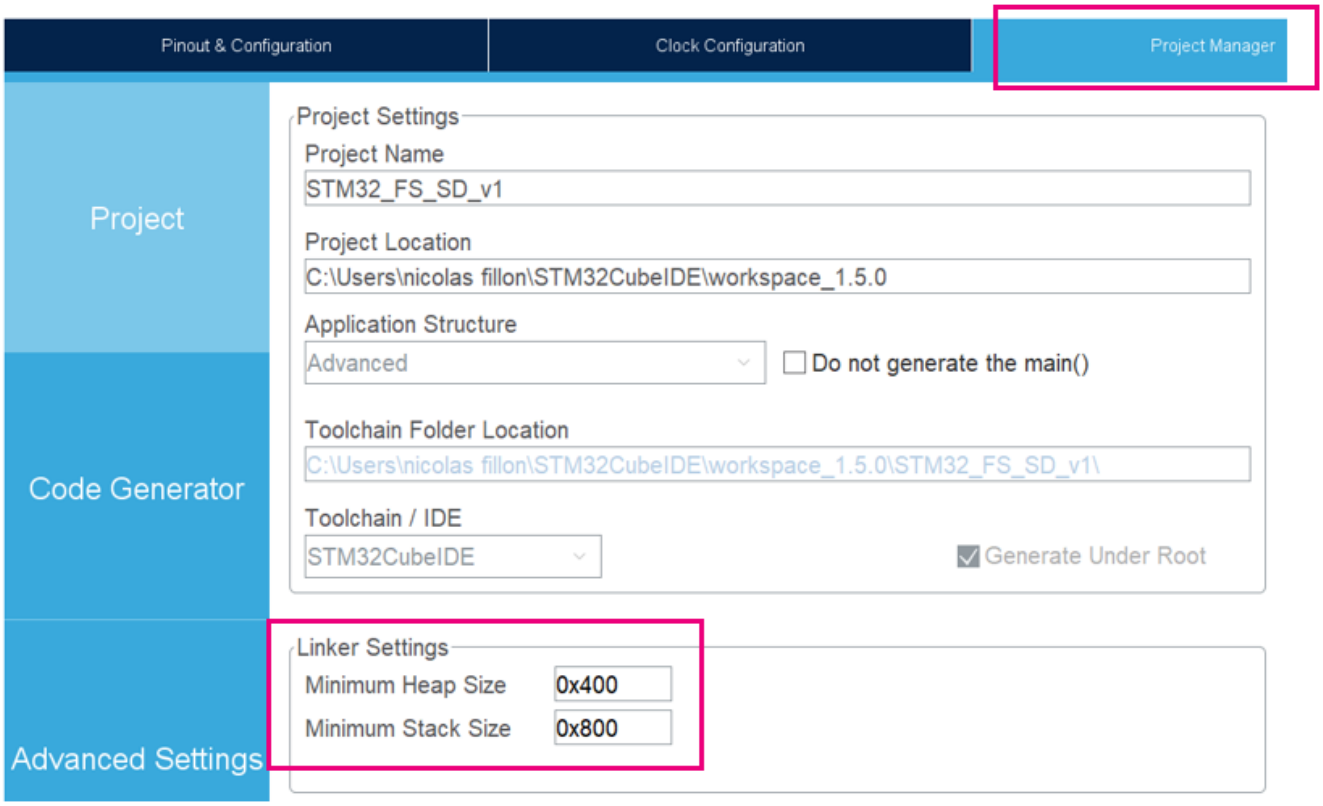
8. Clock Configuration

- Go to Clock Configuration Tab and Press Yes to resolve the clock issues



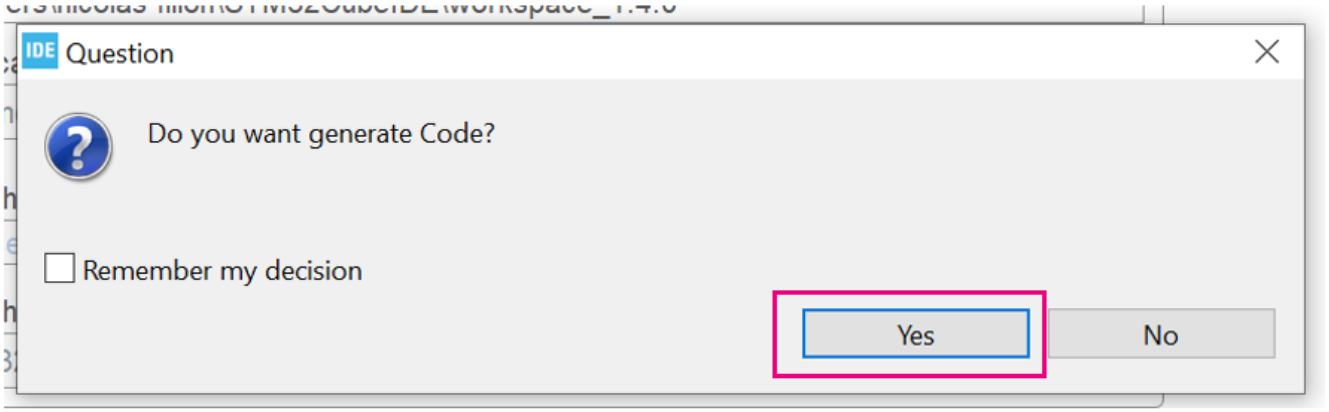
9. Project Manager Settings:

- In the Project Manager Tab, increase the Heap and Stack size because we are using FatFs Middleware that requires more memory:

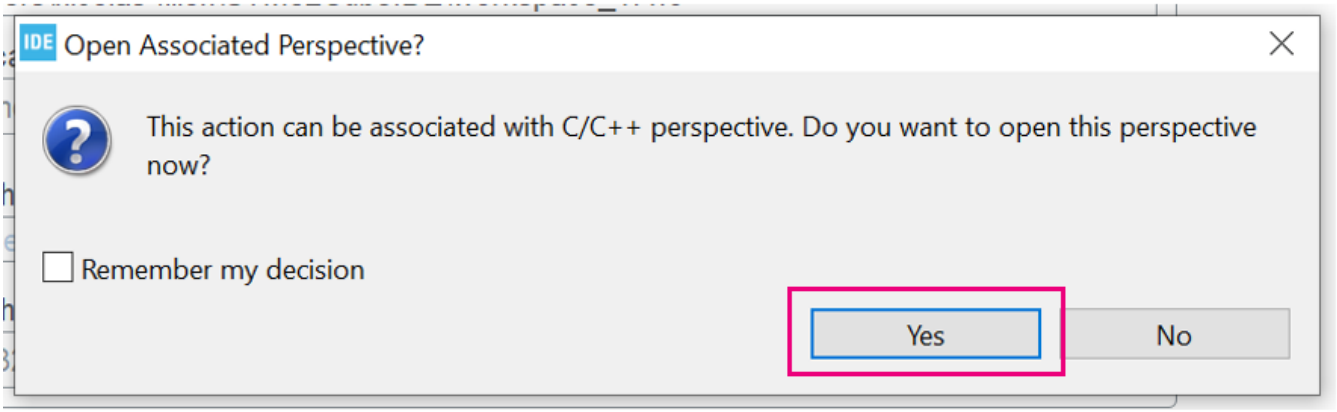


10. Code generation

- Now Save the project to generate the code:



11. Open Associated Perspective


Open Associated Perspective?
This action can be associated with C/C++ perspective. Do you want to open this perspective now?
Remember my decision
Yes
No

12. Add code

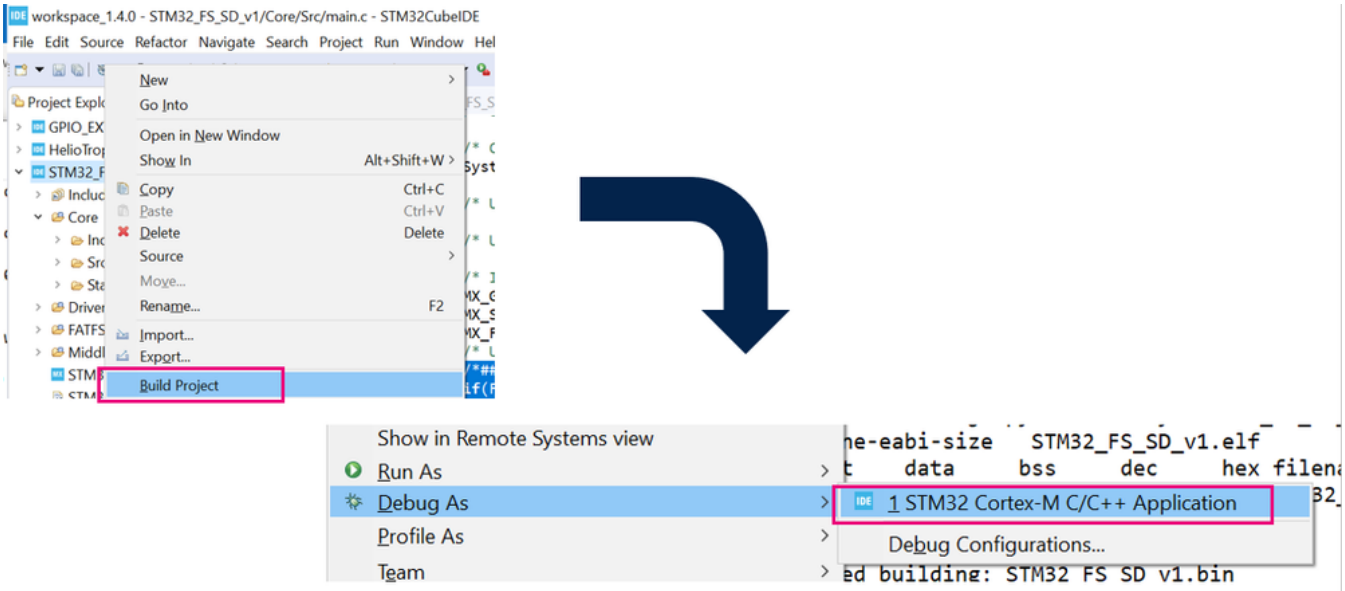
```
/* USER CODE BEGIN 1 */
FRESULT res; /* FatFs function common result code */
uint32_t byteswritten, bytesread; /* File write/read counts */
uint8_t wtext[] = "STM32 FATFS works great!"; /* File write buffer */
uint8_t rtext[_MAX_SS];/* File read buffer */
/* USER CODE END 1 */
  ...
/* USER CODE BEGIN 2 */
        if(f_mount(&SDFatFS, (TCHAR const*)SDPath, 0) != FR_OK)
        {
                Error_Handler();
        }
        else
        {
                if(f_mkfs((TCHAR const*)SDPath, FM_ANY, 0, rtext, sizeof(rtext)) != FR_OK)
            {
                        Error_Handler();
            }
                else
                {
                        //Open file for writing (Create)
                        if(f_open(&SDFile, "STM32.TXT", FA_CREATE_ALWAYS | FA_WRITE) != FR_O
                        {
                                Error_Handler();
                        }
                        else
                        {

                                //Write to the text file
                                res = f_write(&SDFile, wtext, strlen((char *)wtext), (void *
                                if((byteswritten == 0) || (res != FR_OK))
                                {
                                        Error_Handler();
                                }
                                else
                                {

                                        f_close(&SDFile);
                                }
                        }
                }
        }
```

```
        f_mount(&SDFatFS, (TCHAR const*)NULL, 0);
/* USER CODE END 2 */
```

13. Build and flash the code



When the code is executed, the SD card will be formatted and a file will be written, you can check the content with an SD card reader.

Links

STM32CubeIDE:

STM32CubeIDE - Integrated Development Environment for STM32 - STMicroelectronics

FatFs

FatFs - Generic FAT Filesystem Module (elm-chan.org)

Video

This is the video that explains how to create a File System on SD card using STM32CubeIDE as explained in this article.

STM32 – Creating a File System on a SD card - YouTube

STM32 MCU Products **STM32CubeIDE**

1 Kudo