

STM32CubeMX

<http://www.st.com/en/development-tools/stm32cubemx.html#getsoftware-scroll>

Soft

STM32 Cube

Open STM32CubeMX

STMCube is a software that generates initialization code for STM processors. It allows you to set up the micro via a GUI interface and not have to write init code yourself.

Select “New Project” and open the Board Selector tab (if you are using a discovery board).

MCU Selector Board Selector

Board Filter

Vendor : STMicroelectronics Type of Board : All MCU Series : All

☐ Initialize all peripherals with their default Mode

Peripheral Selection

Peripherals	Nb	Max
Accelerometer	<input type="checkbox"/>	
Analog I/O	0	3
Arduino Form Factor	0	128
Audio Line In	0	2
Audio Line Out	0	2
Button	0	3
CAN	0	2

Boards List: 90 Items

Type	Reference
Nucleo144	NUCLEO-F303ZE
Nucleo144	NUCLEO-F207ZG
Nucleo144	NUCLEO-F429ZI
Nucleo144	NUCLEO-F446ZE
Nucleo144	NUCLEO-F746ZG
Nucleo144	NUCLEO-F767ZI
Nucleo144	NUCLEO-F412ZG

Since I am using the STM32F100 VL Discovery Board I have filtered for it.

Board Filter

Vendor : STMicroelectronics Type of Board : Discovery MCU Series : STM32F1

☐ Initialize all peripherals with their default Mode

Peripheral Selection

Peripherals	Nb	Max
Accelerometer	<input type="checkbox"/>	
Analog I/O	0	0

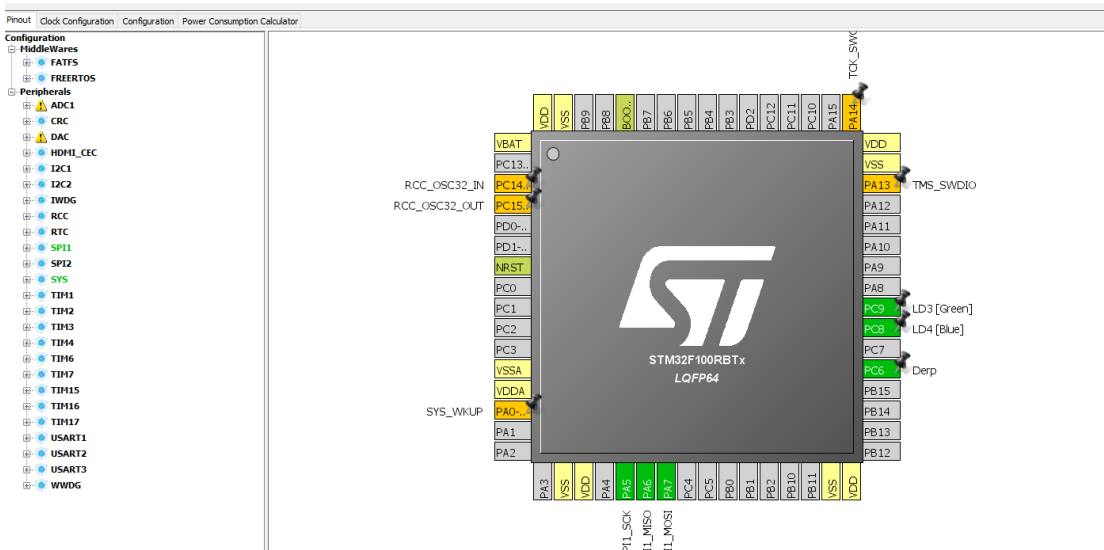
Boards List: 1 Item

Type	Reference	MCU
Discovery	STM32VLDISCOVERY	STM32F100RBTx

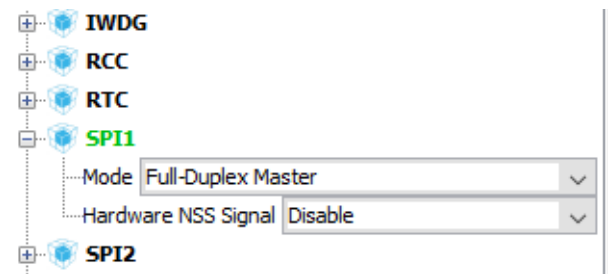
Double click on the board you're using.

Set up all of the pins for what you want to use them for.

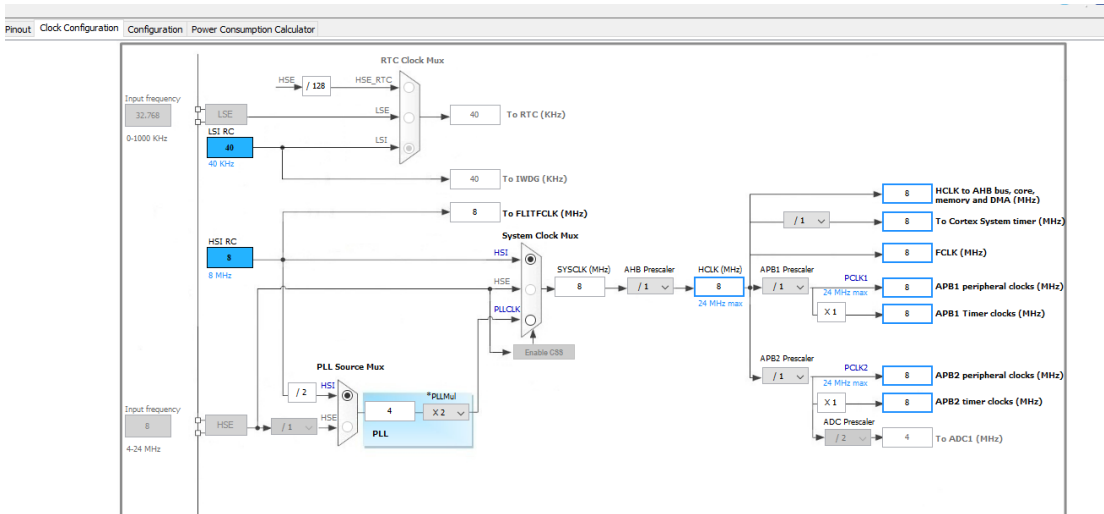
I set PC6 as an output and labeled it “Derp”



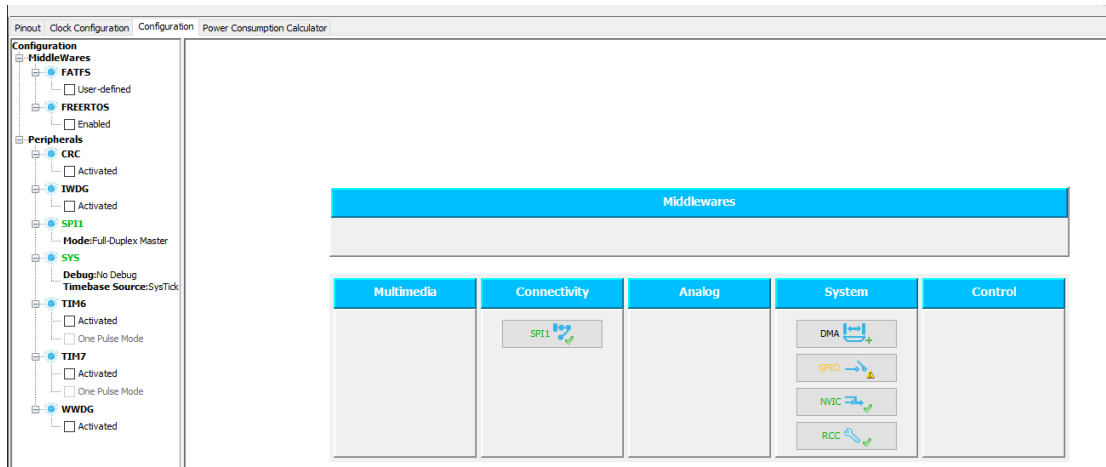
SPI 1 has also been set to “Full-Duplex Master” Mode.



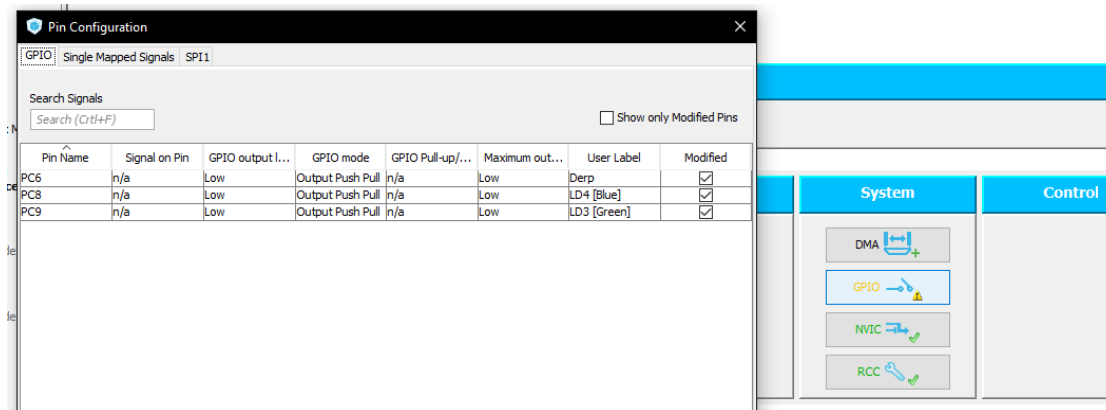
Navigate to the “Clock Configuration” tab and set up the clock.



Navigate to the “Configuration” tab and set up the peripherals that you enabled in the Pinout Tab.

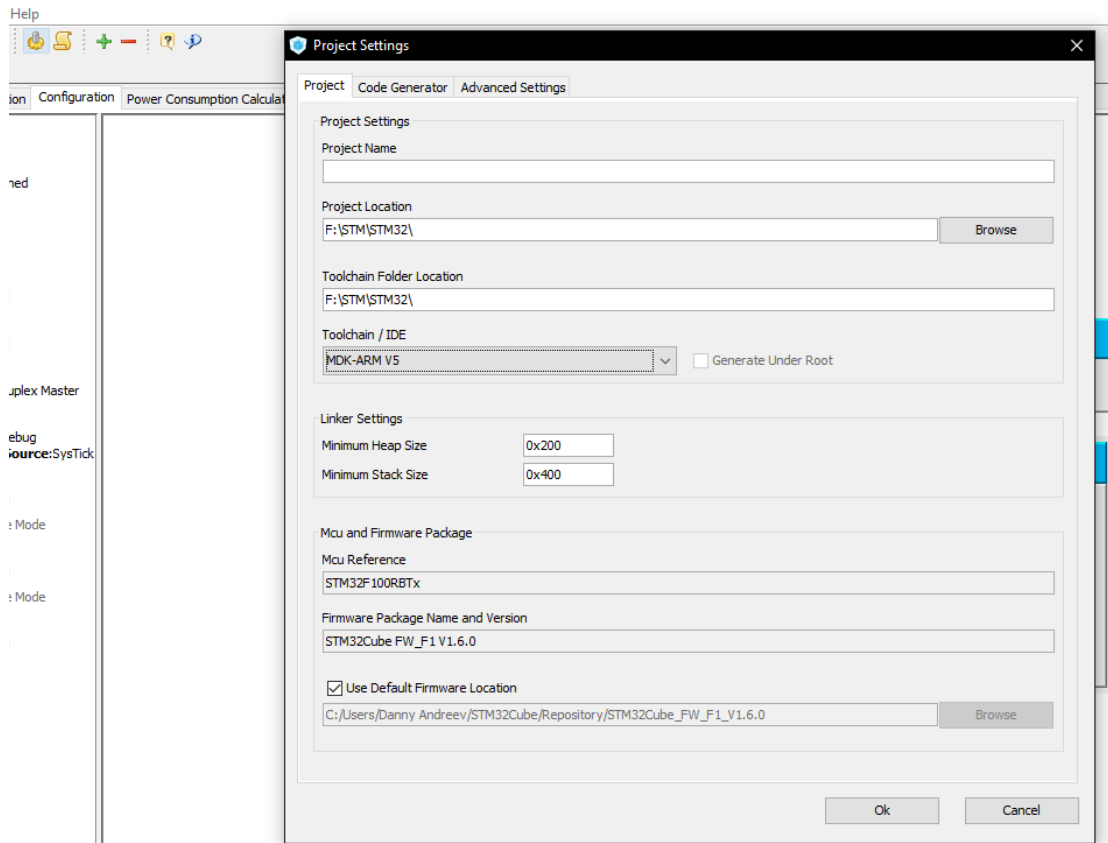


This is where you can set up the GPIO modes for the pins as well as baud rate for the SPI connection.

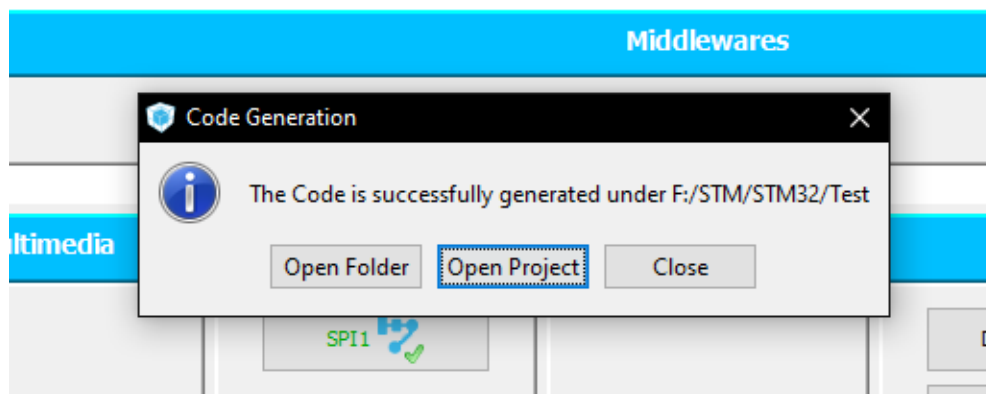


Once happy with the settings hit the "Generate Source Code" Button.

Select a folder to export the code too and make sure that you have selected the correct Toolchain/IDE. We are using Keil V5 so I have selected the MDK-Arm V5 IDE.

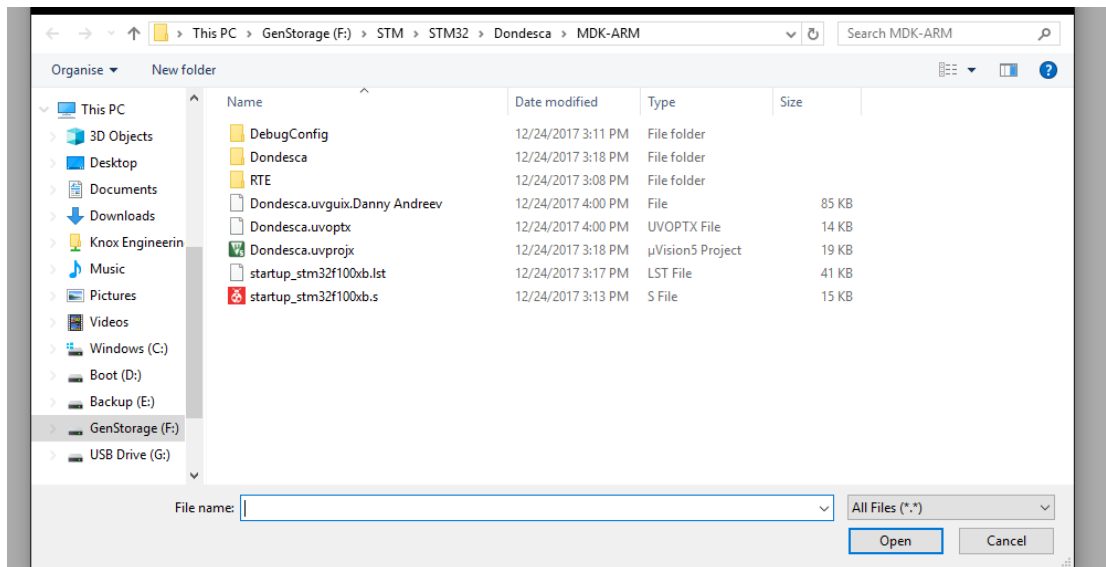


After hitting okay, a popup window will come up once the code is generated. Hit “Open Project” and Keil should automatically open the Project.

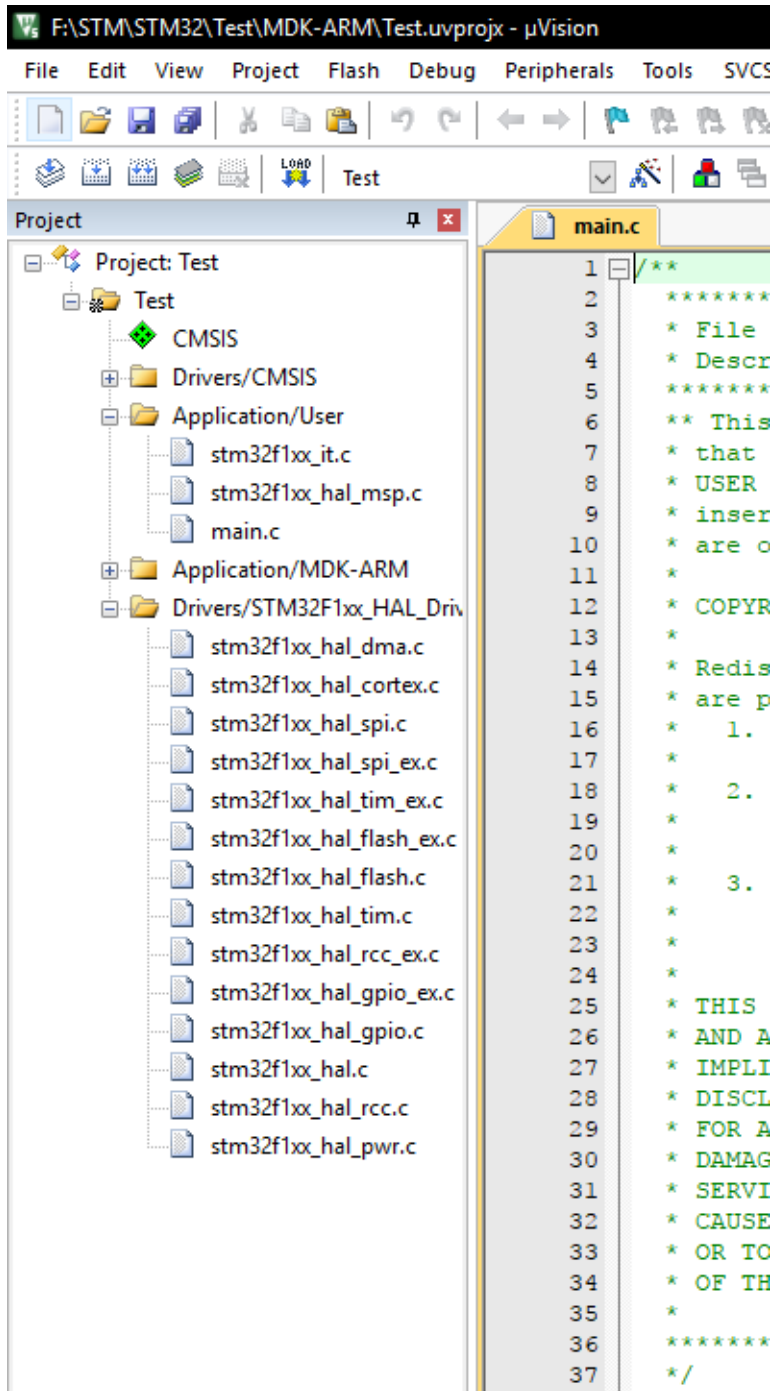


Keil uVision5

The project can also directly be opened by opening the “.uvprojx” file in Keil



Code can be input in the main.c file. Most functions are defined in the files under the Drivers folder. Explore these to learn about functions.



Add the following code in the while loop to turn on the green LED on the discovery board.

Note: All code outside of the `/*USER CODE BEGIN*/ /*USER CODE END*/` will not save and be restored upon opening cube.

```

/* Infinite loop */
/* USER CODE BEGIN WHILE */
while (1)
{
    /* USER CODE END WHILE */

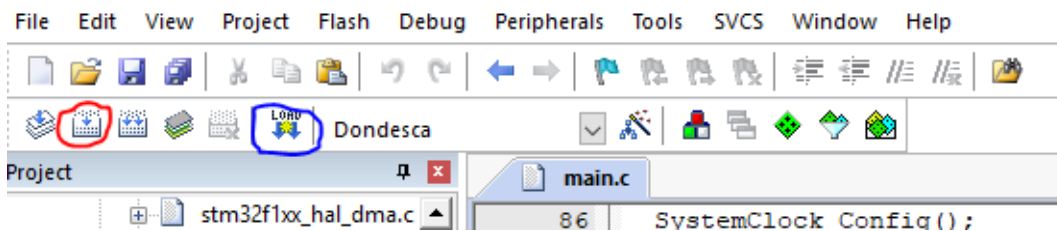
    /* USER CODE BEGIN 3 */
    HAL_GPIO_WritePin(LD3_GPIO_Port, LD3_Pin, 1);
    }
    /* USER CODE END 3 */
}

/** System Clock Configuration

```

To generate the code that will be uploaded to the micro hit the “Build” button circled in red.

Then, hold down the reset button on the Discovery board itself and hit the “Download” button. Release the reset button as soon as the download button is hit.



The code should be successfully uploaded to the Microcontroller.

