

How to Import the SAMMS Software Package (PDF)

1. Download the [final SAMMS Software Package](#) and unzip.
2. Install the [STM32 Cube IDE](#)
3. Open CubeIDE and create a new STM32 Project (NOT from an existing .ioc)
4. Select STM32F407VG (or whatever MCU you are using) and click Next.

The screenshot shows the STM32CubeIDE MCU/MPU Selector interface. The 'MCU/MPU Filters' panel on the left has 'Part Number' set to 'STM32F407VG'. The 'Features' tab is selected, showing details for the 'STM32F4 Series'. The 'STM32F407VG' is highlighted as the selected device. Below this, a table lists 8 items, with 'STM32F407VG' selected. The bottom navigation bar shows 'Next >' as the active button.

MCU/MPU Selector | Board Selector | Example Selector | Cross Selector

MCU/MPU Filters

Part Number: STM32F407VG

Core > Series > Line > Package > Other > Peripheral >

Features | Block Diagram | Docs & Resources | Datasheet

STM32F4 Series

STM32F407VG

High-performance foundation line, Arm Cortex-M4 core with DSP and FPU, 1 Mbyte of Flash memory, 168 MHz CPU, ART Accelerator, Ethernet, FSMC

ACTIVE Active Product is in mass production

Unit Price for 10kU (US\$) : 5.623

Board: STM32F407G-DISC1

LQFP100

MCUs/MPUs List: 8 items

+ Display similar items

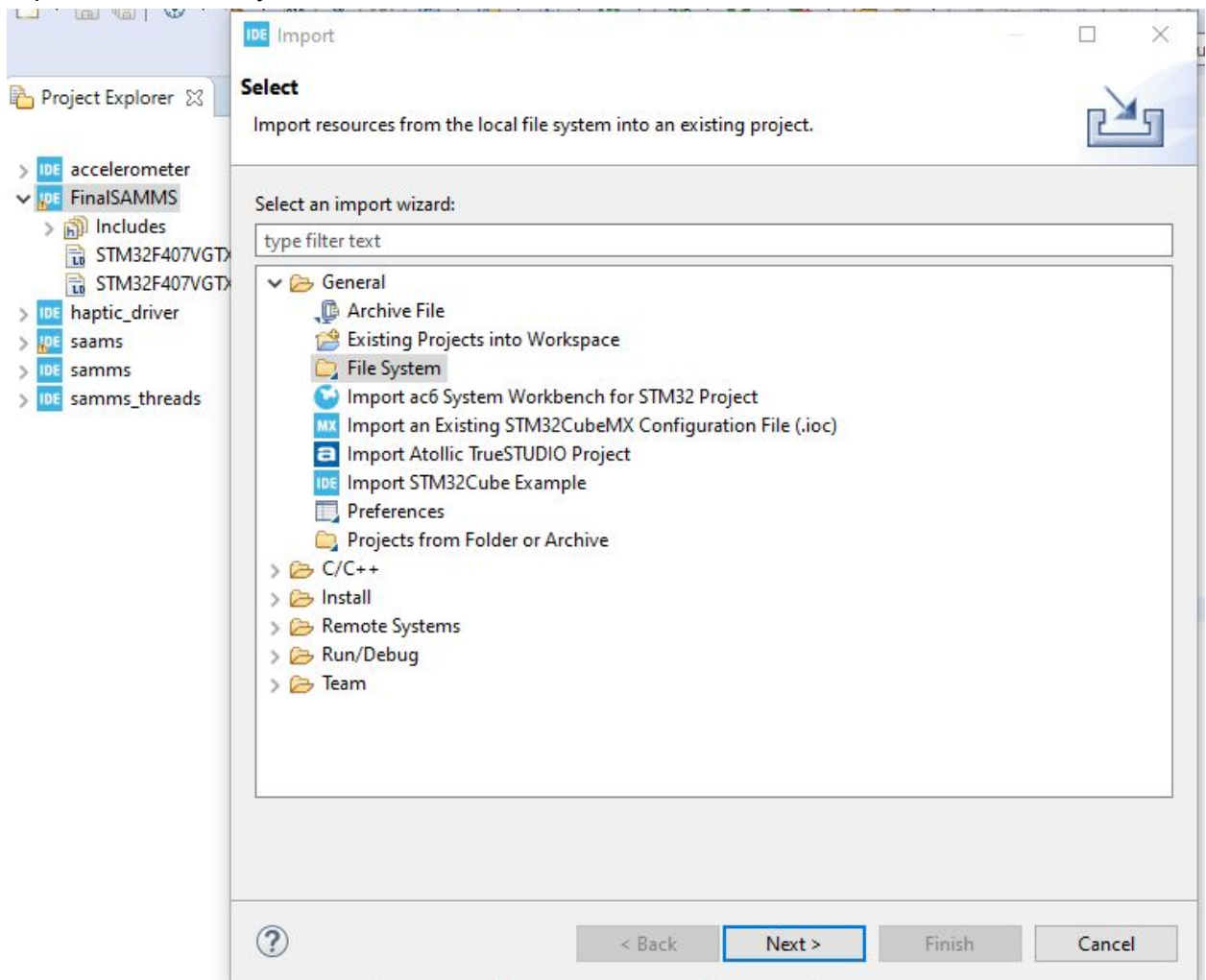
Export

*	Part No	Reference M...	Unit Pric...	Board	Package	Flash	RAM	ID	Freq.
☆	STM32F407IE	STM32F...	Ac... 5.411		LQFP176	512 kByt...	192 kBy...	140	168 MHz
☆	STM32F407IG	STM32F...	Ac... 6.262	STM3240G...	UFPGA...	1024 kB...	192 kBy...	140	168 MHz
☆	STM32F407IG	STM32F...	Ac... 6.262		LQFP176	1024 kB...	192 kBy...	140	168 MHz
☆	STM32F407VE	STM32F...	Ac... 4.772		LQFP100	512 kByt...	192 kBy...	82	168 MHz
☆	STM32F407VG	STM32F...	Ac... 5.623	STM32F407	LQFP100	1024 kB...	192 kBy...	82	168 MHz
☆	STM32F407ZE	STM32F...	Ac... 5.198		LQFP144	512 kByt...	192 kBy...	114	168 MHz
☆	STM32F407ZG	STM32F...	Ac... 6.049		LQFP144	1024 kB...	192 kBy...	114	168 MHz

< Back | Next > | Finish | Cancel

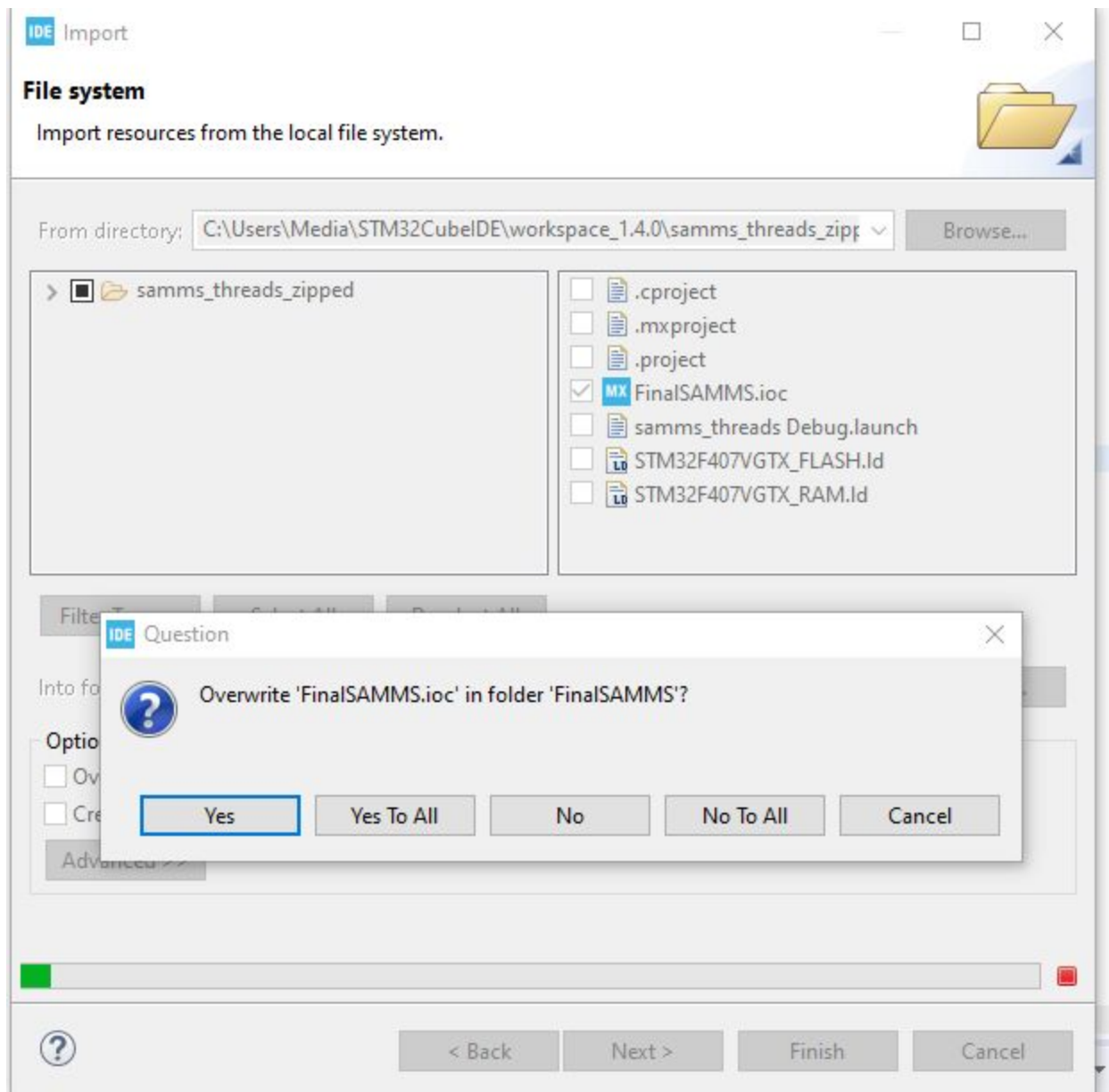
5. Name your project. Click Next and then Finish. The project for this tutorial is named "FinalSAMMS"
6. Open the unzipped final samms project folder. Rename the "samms_threads.ioc" file to "project_name.ioc". In our case, it will be "FinalSAMMS.ioc".

7. Right click on the FinalSAMMS project in the Project Explorer in CubeIDE and choose Import. Select File System and Next.

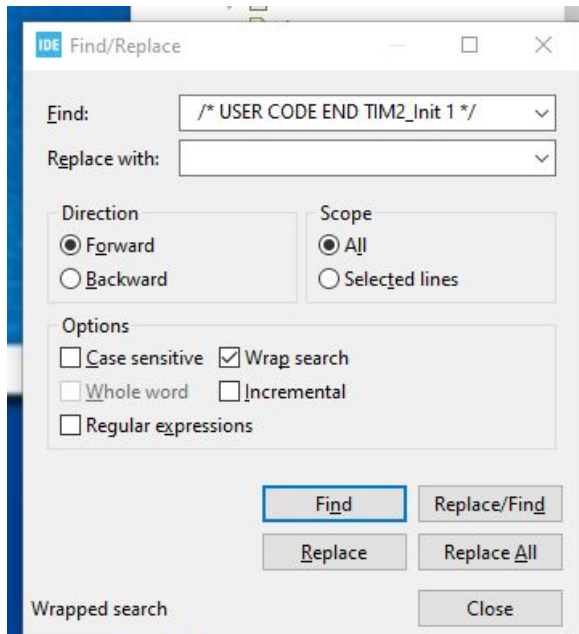


8. Click Browse and navigate to the unzipped final samms project folder. Click OK. Choose "FinalSAMMS.ioc" or "project_name.ioc" from the right pane. Click Finish. If it asks to

overwrite, click Yes To All.

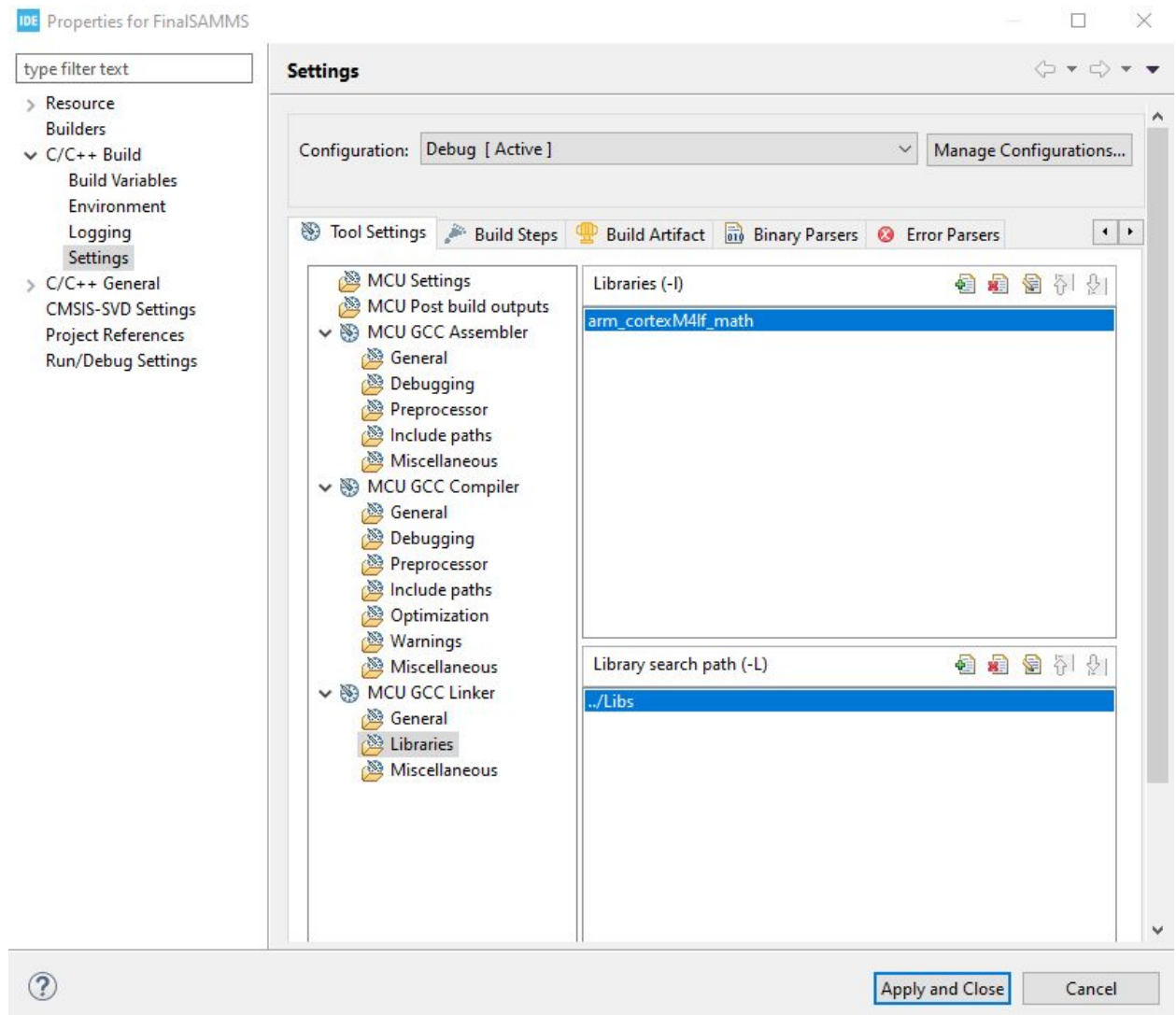


9. Open the Core > Inc folder in the Project Explorer. Import a File System and choose all the files in samms_threads/Core/Inc. Click Finish and Overwrite (Yes to All).
10. Repeat Step 9 but with Core > Src and samms_threads/Core/Src.
11. Right click on the project name and select New > Folder. Name it Libs. Right click on the new Libs folder and choose Import and choose File System. Choose samms_threads/Libs. Click Finish and Overwrite.
12. Double click on the ioc. Click on a random PIN that is gray and choose GPIO_Output. CTRL+S to save and say yes when it asks to generate code. The folders FATFS and Middlewares should appear.
13. CTRL+F and search for "/* USER CODE END TIM2_Init 1 */". Delete the next 4 lines underneath this phrase.



```
870 htim2.Init.Prescaler = 0; //(timer_prescaler - 1)
871 htim2.Init.Period = (timer_clock_frequency / (TIM2_Prescaler + 1));
872 htim2.Init.CounterMode = TIM_COUNTERMODE_UP;
873
874 /* USER CODE END TIM2_Init 1 */
875 htim2.Instance = TIM2;
876 htim2.Init.Prescaler = 0;
877 htim2.Init.CounterMode = TIM_COUNTERMODE_UP;
878 htim2.Init.Period = 4294967295;
879 htim2.Init.ClockDivision = TIM_CLOCKDIVISION_DIV1;
880 htim2.Init.AutoReloadPreload = TIM_AUTORELOAD_PRELOAD_DISABLE;
881 if (HAL_TIM_Base_Init(&htim2) != HAL_OK)
882 {
883     Error_Handler();
884 }
885 sClockSourceConfig.ClockSource = TIM_CLOCKSOURCE_INTERNAL;
886 if (HAL_TIM_ConfigClockSource(&htim2, &sClockSourceConfig) != HAL_OK)
887 {
888     Error_Handler();
889 }
890 if (HAL_TIM_OC_Init(&htim2) != HAL_OK)
891 {
892     Error_Handler();
893 }
894 sMasterConfig.MasterOutputTrigger = TIM_TRGO_UPDATE;
895 sMasterConfig.MasterSlaveMode = TIM_MASTERSLAVEMODE_DISABLE;
896 if (HAL_TIM_MasterConfigSettable(&htim2) != HAL_OK)
897 {
898     Error_Handler();
899 }
```

14. At the top toolbar, click on Project. Then go to Project > Properties > C/C++ Build > Settings > Tool Settings > MCU GCC Linker > Libraries. Add the libraries shown in the screenshot then click Apply and Close.



15. Click Project on the top toolbar and then Build Project. It should compile without errors. You're ready to go!