

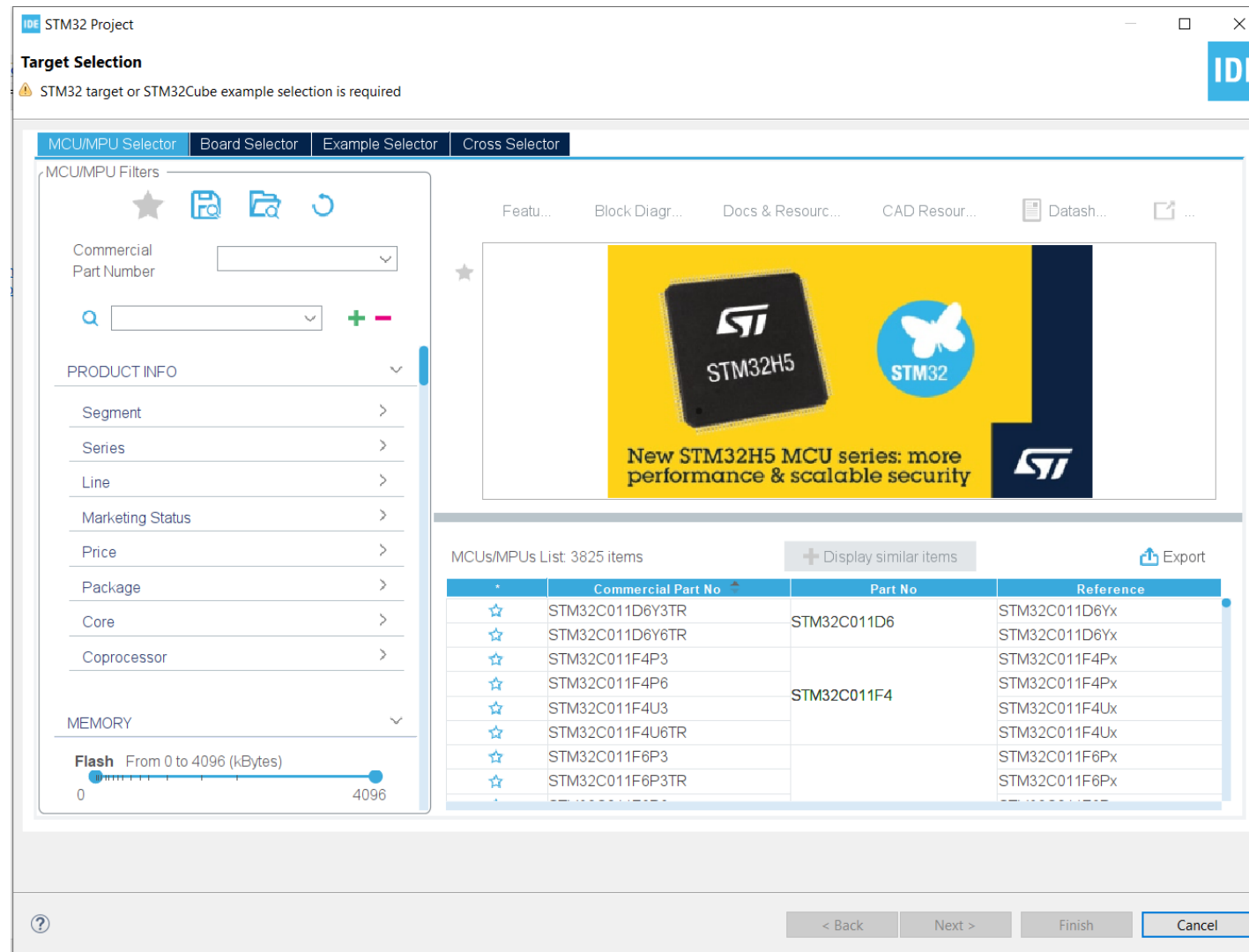
UCSD Embedded C Assignment 6

By

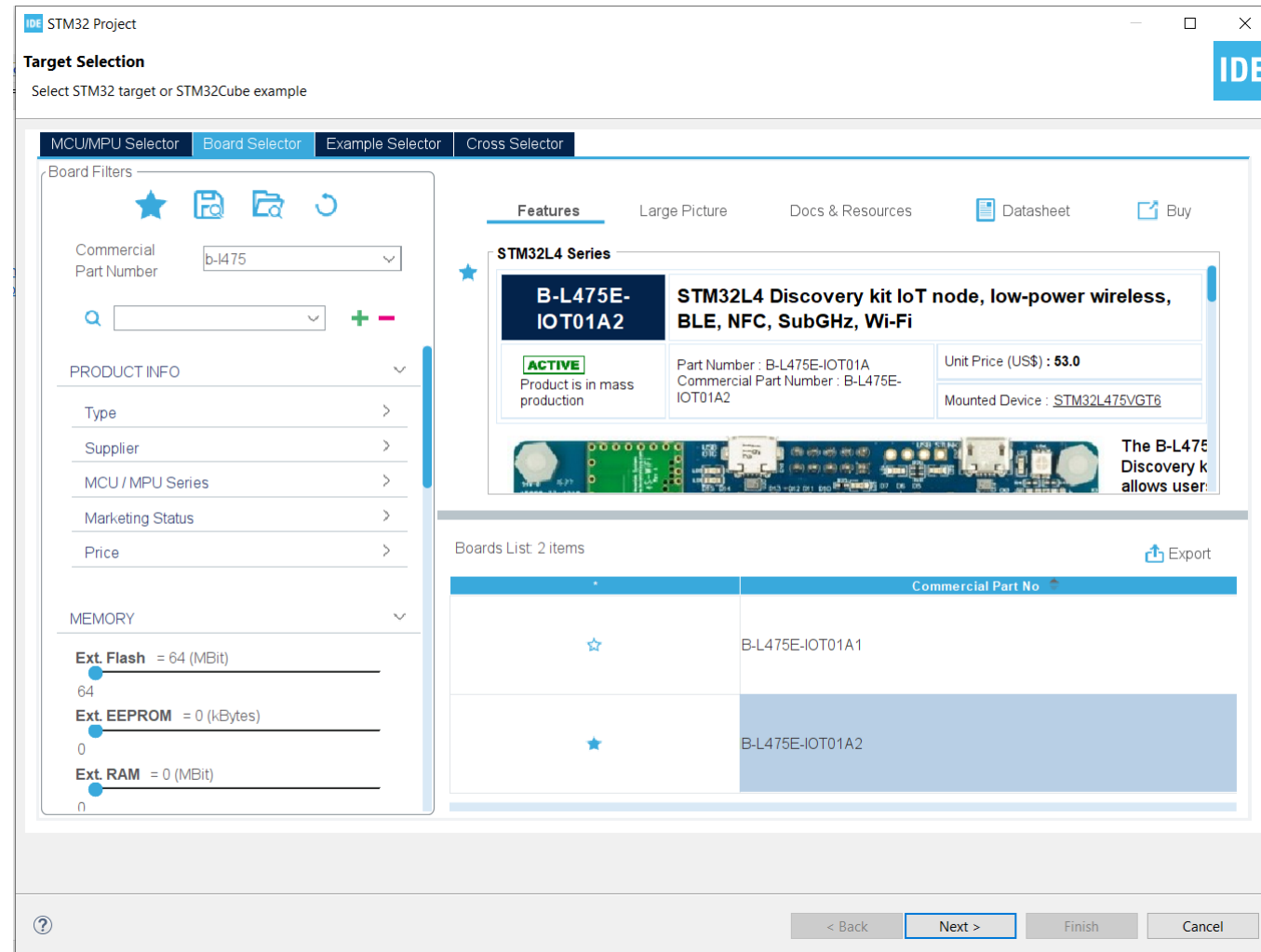
Hsuankai Chang

hsuankac@umich.edu

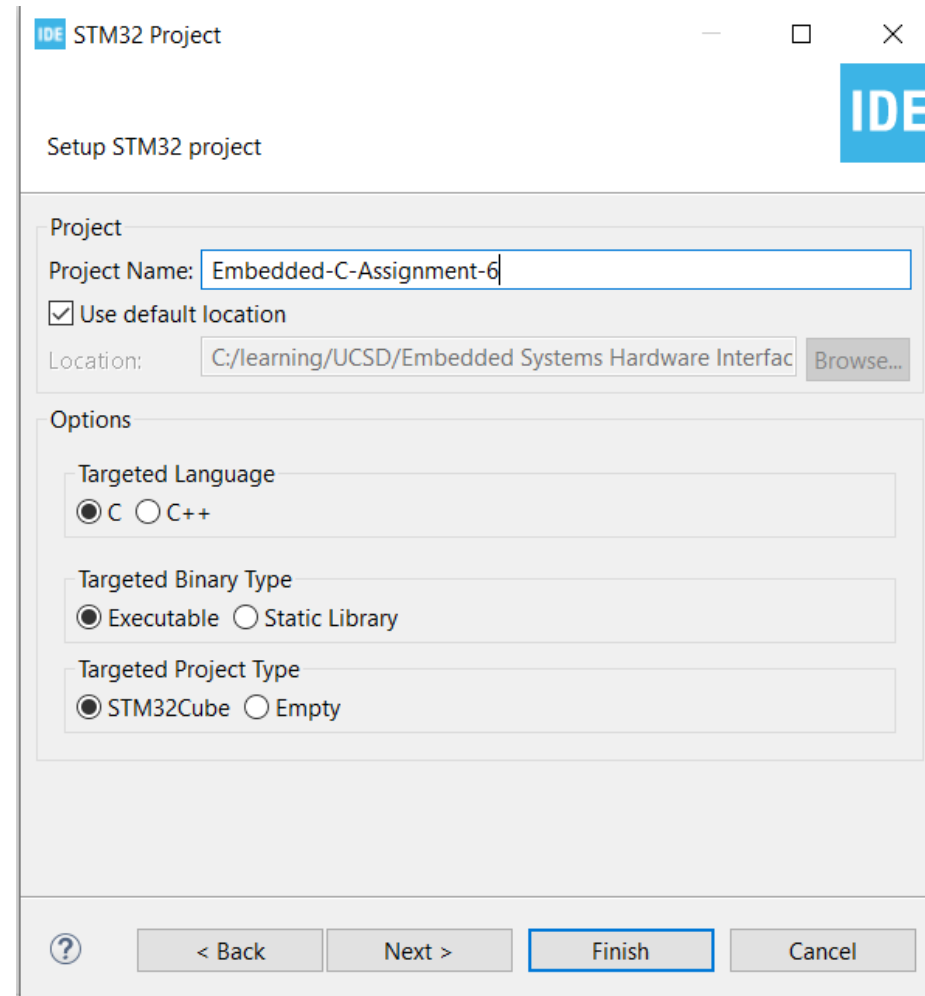
Step 1. Startup STM32CubeIDE and create new STM32 project



Step 2. Access board selector and type in the board you use, click Next



Step 3. Enter the project name then click Next



The image shows a 'Setup STM32 project' dialog box from the IDE. The window title is 'IDE STM32 Project'. The main title is 'Setup STM32 project'. The 'Project' section contains a 'Project Name' field with the text 'Embedded-C-Assignment-6', a checked 'Use default location' checkbox, and a 'Location' field with the path 'C:/learning/UCSD/Embedded Systems Hardware Interfac' and a 'Browse...' button. The 'Options' section contains three sub-sections: 'Targeted Language' with radio buttons for 'C' (selected) and 'C++'; 'Targeted Binary Type' with radio buttons for 'Executable' (selected) and 'Static Library'; and 'Targeted Project Type' with radio buttons for 'STM32Cube' (selected) and 'Empty'. At the bottom, there is a help icon, '< Back' button, 'Next >' button, 'Finish' button, and 'Cancel' button.

IDE STM32 Project

Setup STM32 project

Project

Project Name: Embedded-C-Assignment-6

☒ Use default location

Location: C:/learning/UCSD/Embedded Systems Hardware Interfac Browse...

Options

Targeted Language

☒ C ☐ C++

Targeted Binary Type

☒ Executable ☐ Static Library

Targeted Project Type

☒ STM32Cube ☐ Empty

? < Back Next > Finish Cancel

Step 4. See the firmware package name and version



The image shows a Windows-style dialog box titled "STM32 Project" with a subtitle "Firmware Library Package Setup". The subtitle is followed by the instruction "Setup STM32 target's firmware". The dialog is divided into three sections: "Target and Firmware Package", "Firmware and Software Package Repository", and "Code Generator Options". In the first section, "Target Reference" is set to "B-L475E-IOT01A2" and "Firmware Package Name and Version" is set to "STM32Cube FW_L4" with a dropdown menu showing "V1.17.2". The second section shows the "Location" as "C:\Users\hsuankai.chang\STM32Cube\Repository" and includes a link to the "Firmware Updater". The third section contains three radio button options for code generation, with the third option, "Copy only the necessary library files", being selected. At the bottom, there are buttons for "?", "< Back", "Next >", "Finish", and "Cancel".

IDE STM32 Project

Firmware Library Package Setup

Setup STM32 target's firmware

Target and Firmware Package

Target Reference: B-L475E-IOT01A2

Firmware Package Name and Version: STM32Cube FW_L4 V1.17.2

Firmware and Software Package Repository

Location:
C:\Users\hsuankai.chang\STM32Cube\Repository

See ['Firmware Updater'](#) for settings related to package installation

Code Generator Options

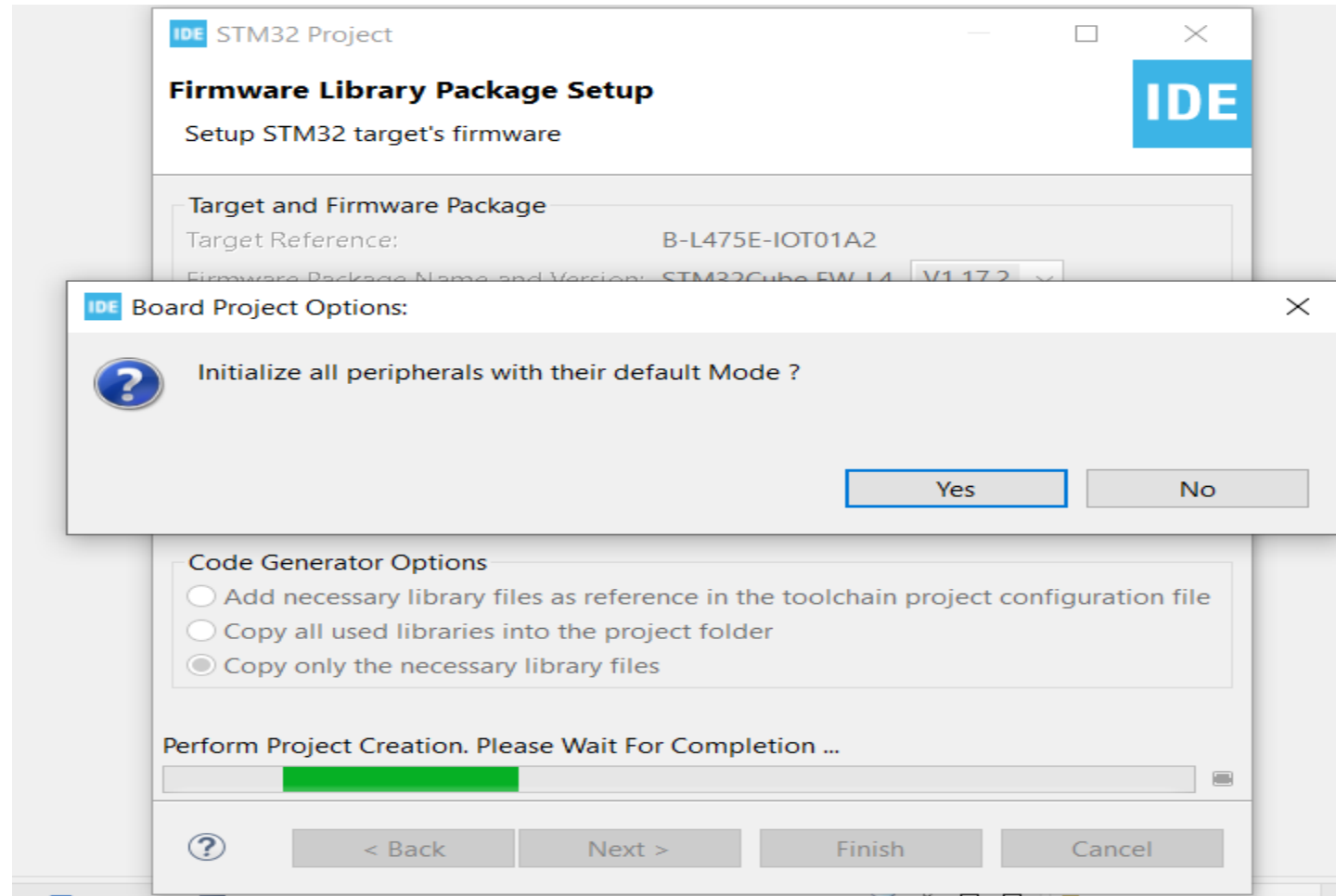
☐ Add necessary library files as reference in the toolchain project configuration file

☐ Copy all used libraries into the project folder

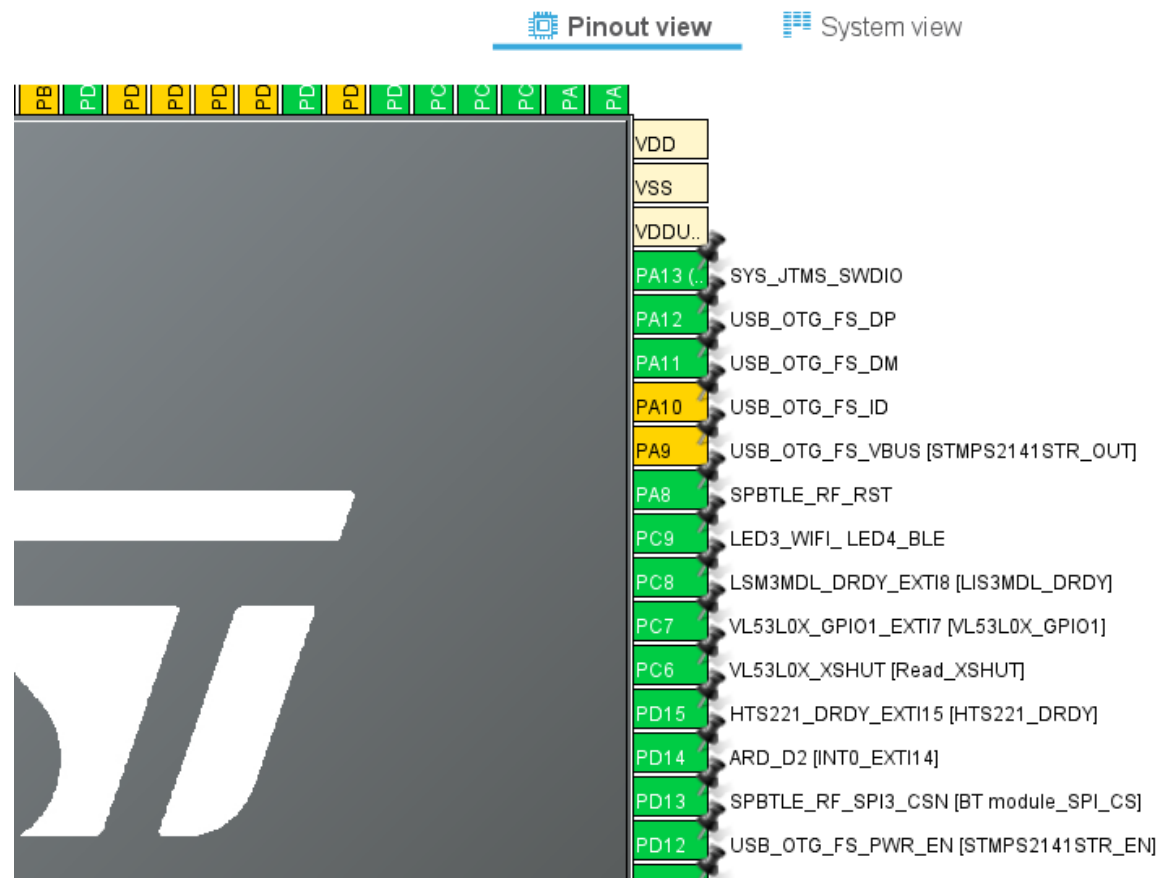
☒ Copy only the necessary library files

? < Back Next > Finish Cancel

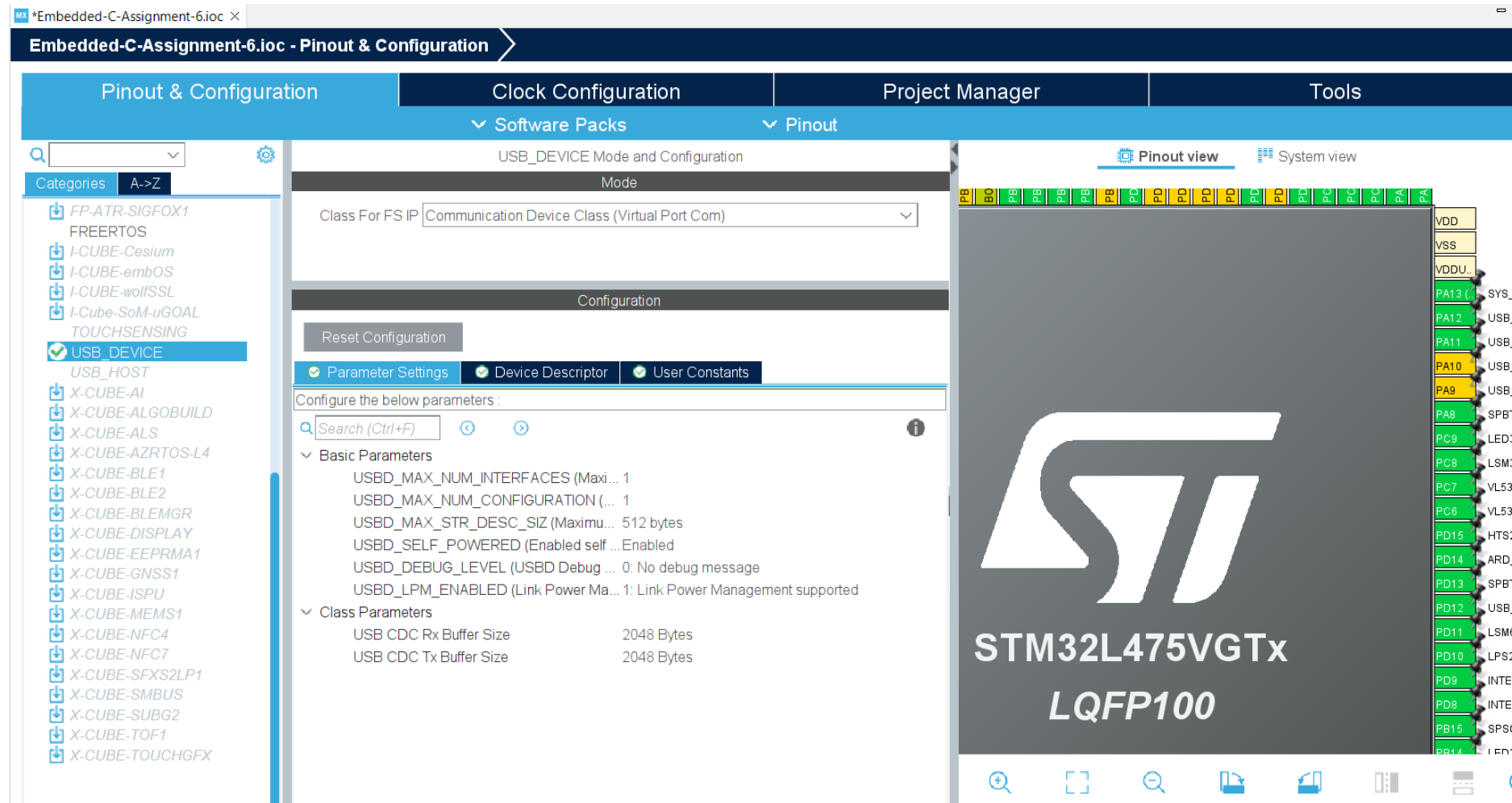
Step 5. Click yes to initialize all peripherals to default



Step 6. When in .ioc file, keep defaults USB OTG settings



Step 7. Add middleware for virtual COM port, and use the default parameter settings



Step 8. Connectivity - USB_OTG_FS -Signal Start of Frame - Enabled

MX Embedded-C-Assignment-6.ioc x

Embedded-C-Assignment-6.ioc - Pinout & Configuration

Pinout & Configuration

Clock Configuration

Project

Software Packs

Pinout

USB_OTG_FS Mode and Configuration

Mode

Mode: Device_Only

Activate_VBUS: Disable

☐ Activate_SOF

☐ Activate_NOE

Configuration

Reset Configuration

☒ Parameter Settings ☒ User Constants ☒ NVIC Settings ☒ GPIO Settings

Configure the below parameters :

Search (Ctrl+F)

Speed	Full Speed 12MBit/s
Low power	Disabled
Link Power Management	Disabled
VBUS sensing	Disabled
Signal start of frame	Enabled

Categories

A-Z

Step 9. Result of code generation

IDE HW6 - Embedded-C-Assignment-6/Core/Src/main.c - STM32CubeIDE

File Edit Source Refactor Navigate Search Project Run Window Help

Project Explorer ×

- Embedded-C-Assignment-6
 - Includes
 - Core
 - Drivers
 - Middlewares
 - ST
 - STM32_USB_Device_Library
 - Class
 - CDC
 - Inc
 - usbd_cdc.h
 - Src
 - usbd_cdc.c
 - Core
 - USB_DEVICE
 - App
 - usb_device.c
 - usb_device.h
 - usbd_cdc_if.c
 - usbd_cdc_if.h
 - usbd_desc.c
 - usbd_desc.h
 - Target

Embedded-C-Assignment-6.ioc main.c ×






```
1 /* USER CODE BEGIN Header */
2 /**
3  *
4  * @file          : main.c
5  * @brief         : Main program body
6  *
7  * @attention
8  *
9  * Copyright (c) 2023 STMicroelectronics.
10 * All rights reserved.
11 *
12 * This software is licensed under terms that can be found in the LICENSE file
13 * in the root directory of this software component.
14 * If no LICENSE file comes with this software, it is provided AS-IS.
15 *
16 */
17 /* USER CODE END Header */
18 /* Includes -----*/
19 #include "main.h"
20 #include "usb_device.h"
21
22
23 /* Private includes -----*/
24 /* USER CODE BEGIN Includes */
25
26 /* USER CODE END Includes */
27
28 /* Private typedef -----*/
```

Step 10. Add code to main.c file

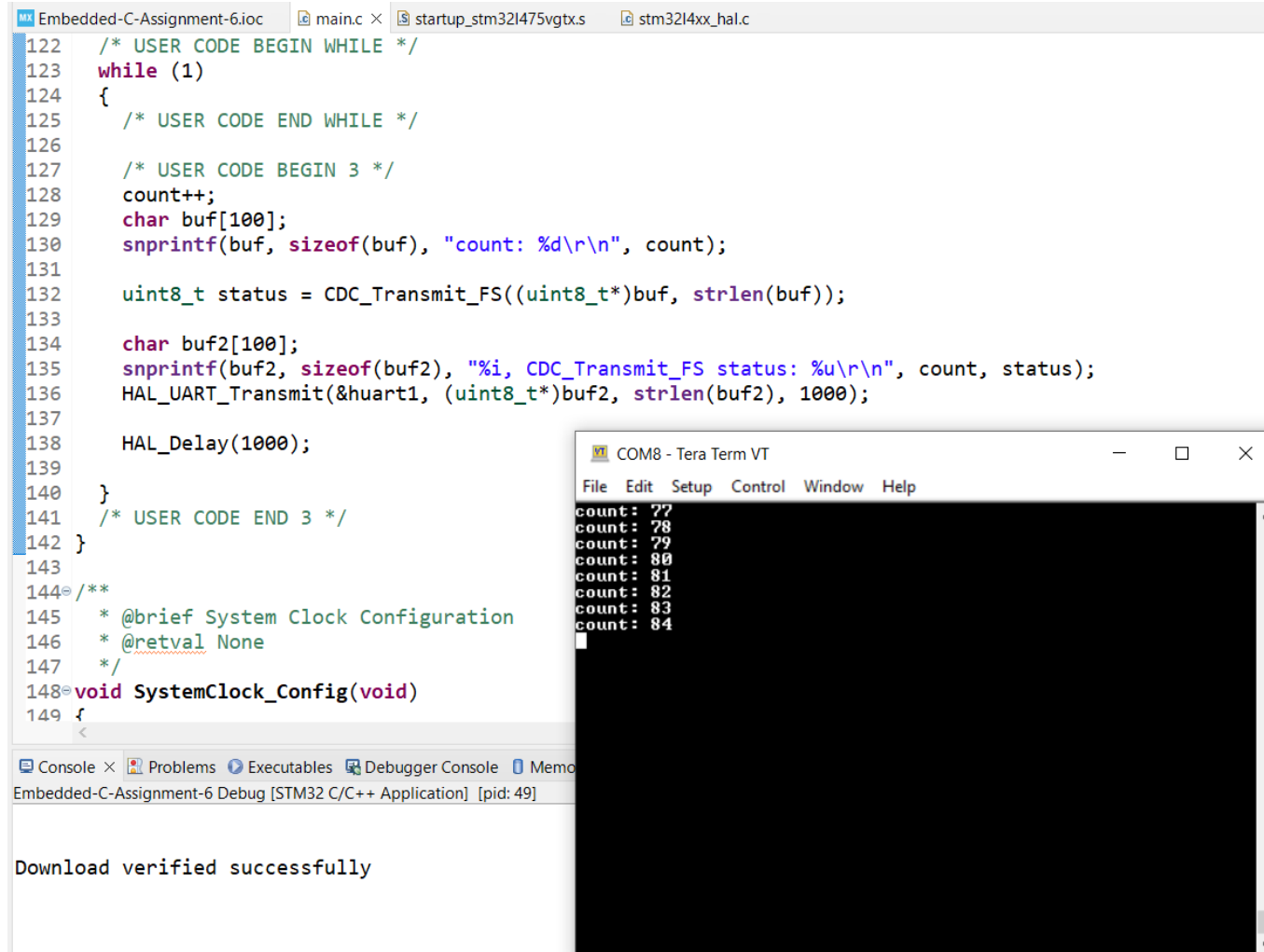
```
Embedded-C-Assignment-6.ioc  main.c ×
112 MX_QUADSPI_Init();
113 MX_SPI3_Init();
114 MX_USART1_UART_Init();
115 MX_USART3_UART_Init();
116 MX_USB_DEVICE_Init();
117 /* USER CODE BEGIN 2 */
118 int count = 0;
119 /* USER CODE END 2 */
120
121 /* Infinite loop */
122 /* USER CODE BEGIN WHILE */
123 while (1)
124 {
125     /* USER CODE END WHILE */
126
127     /* USER CODE BEGIN 3 */
128     count++;
129     char buf[100];
130     snprintf(buf, sizeof(buf), "count: %d\r\n", count);
131
132     uint8_t status = CDC_Transmit_FS((uint8_t*)buf, strlen(buf));
133
134     char buf2[100];
135     snprintf(buf2, sizeof(buf2), "%i, CDC_Transmit_FS status: %u\r\n", count, status);
136     HAL_UART_Transmit(&huart1, (uint8_t*)buf2, strlen(buf2), 1000);
137
138     HAL_Delay(1000);
139
140 }
```

```
18 /* USER CODE END Header */
19 /* Includes -----
20 #include "main.h"
21 #include "usb_device.h"
22
23 /* Private includes -----
24 /* USER CODE BEGIN Includes */
25 #include <stdio.h>
26 #include <string.h>
27 #include "usbd_cdc_if.h"
28 #include "usbd_cdc.h"
29 #include "usbd_core.h"
30 #include "usbd_desc.h"
31 /* USER CODE END Includes */
32
```

Step 11. Build and run the code, see a new COM port appear

- >  Portable Devices
- ✓  Ports (COM & LPT)
 -  Intel(R) Active Management Technology - SOL (COM3)
 -  Standard Serial over Bluetooth link (COM5)
 -  Standard Serial over Bluetooth link (COM7)
 -  STMicroelectronics STLink Virtual COM Port (COM4)
 -  USB Serial Device (COM8)
- >  Print queues

Step 12. Test on the new COM port, Tx is successful



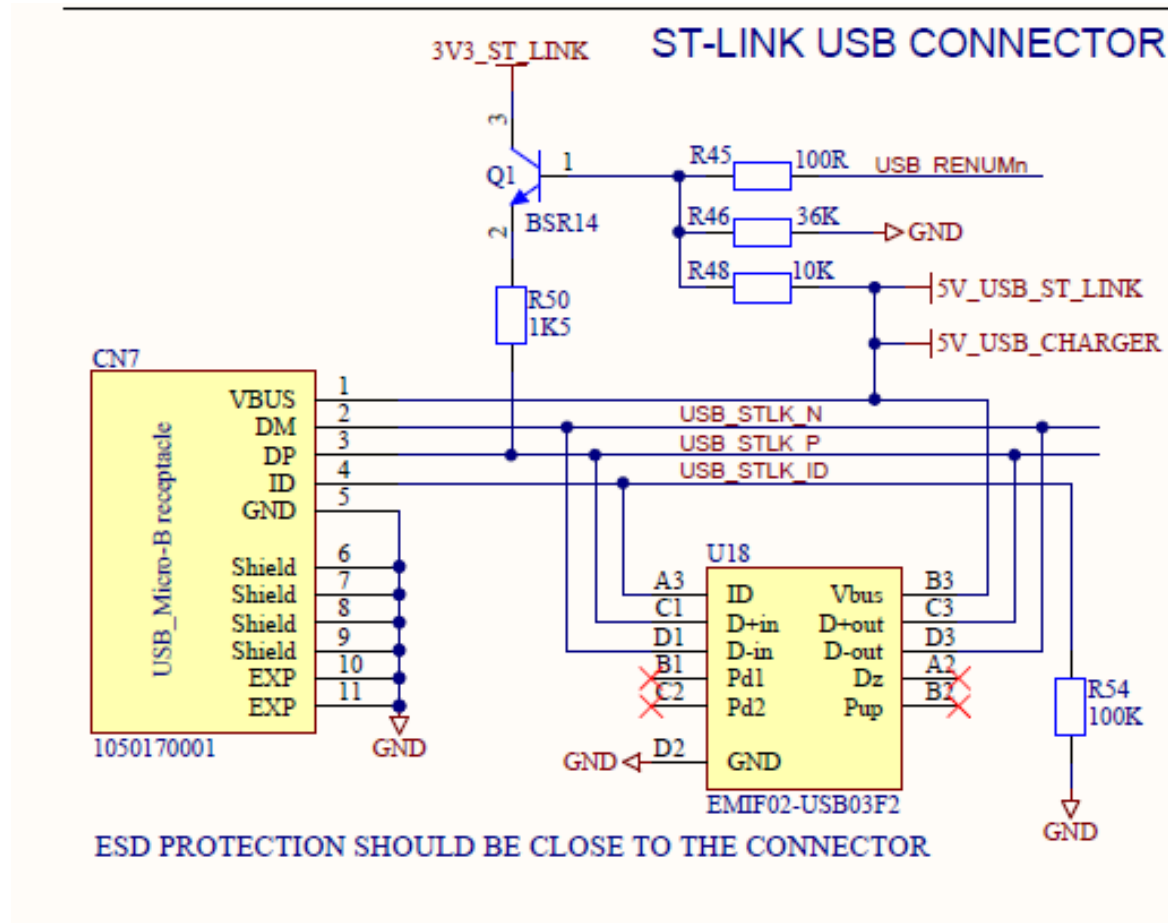
The screenshot shows an IDE with a C code file named 'main.c' and a terminal window titled 'COM8 - Tera Term VT'. The code is a loop that increments a counter and transmits it via UART. The terminal shows the output of the program, displaying the count from 77 to 84.

```
122 /* USER CODE BEGIN WHILE */
123 while (1)
124 {
125     /* USER CODE END WHILE */
126
127     /* USER CODE BEGIN 3 */
128     count++;
129     char buf[100];
130     snprintf(buf, sizeof(buf), "count: %d\r\n", count);
131
132     uint8_t status = CDC_Transmit_FS((uint8_t*)buf, strlen(buf));
133
134     char buf2[100];
135     snprintf(buf2, sizeof(buf2), "%i, CDC_Transmit_FS status: %u\r\n", count, status);
136     HAL_UART_Transmit(&huart1, (uint8_t*)buf2, strlen(buf2), 1000);
137
138     HAL_Delay(1000);
139 }
140 /* USER CODE END 3 */
141 }
142
143 /**
144  * @brief System Clock Configuration
145  * @retval None
146  */
147 void SystemClock_Config(void)
148 {
149     Download verified successfully
```

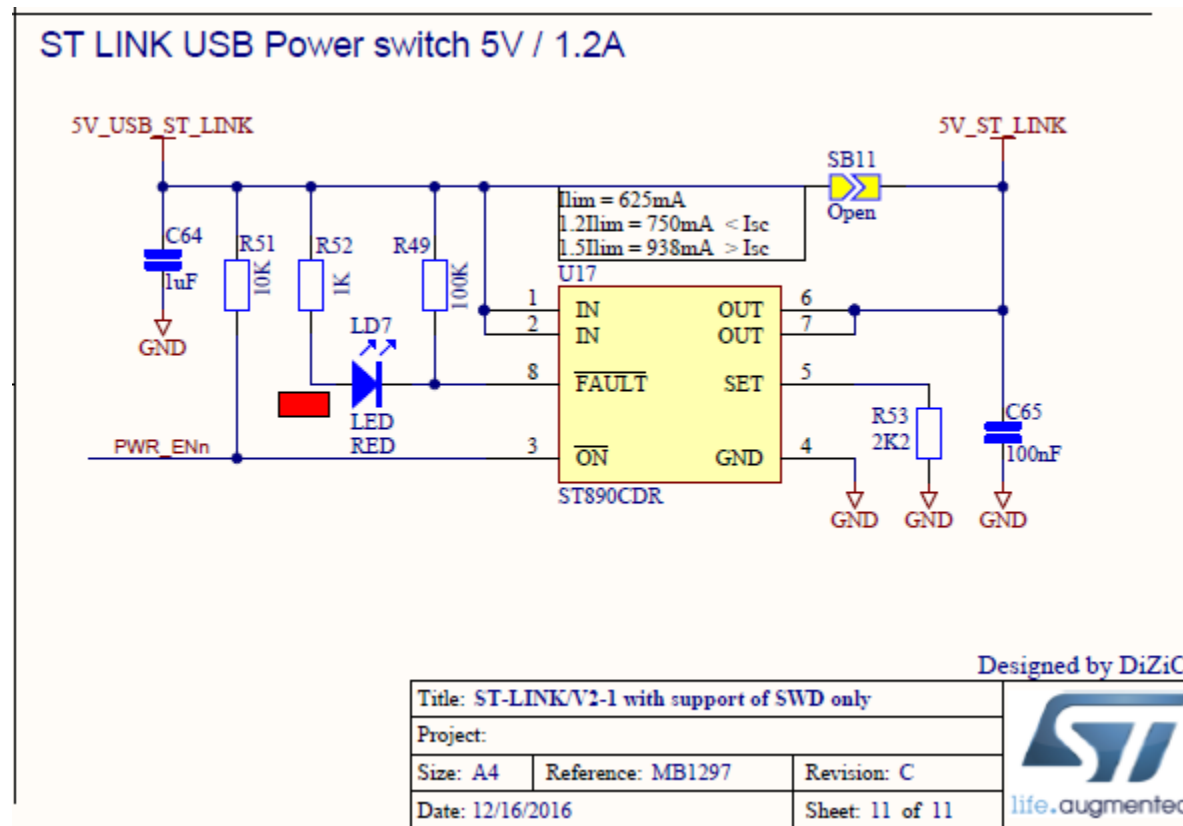
COM8 - Tera Term VT

```
File Edit Setup Control Window Help
count: 77
count: 78
count: 79
count: 80
count: 81
count: 82
count: 83
count: 84
```

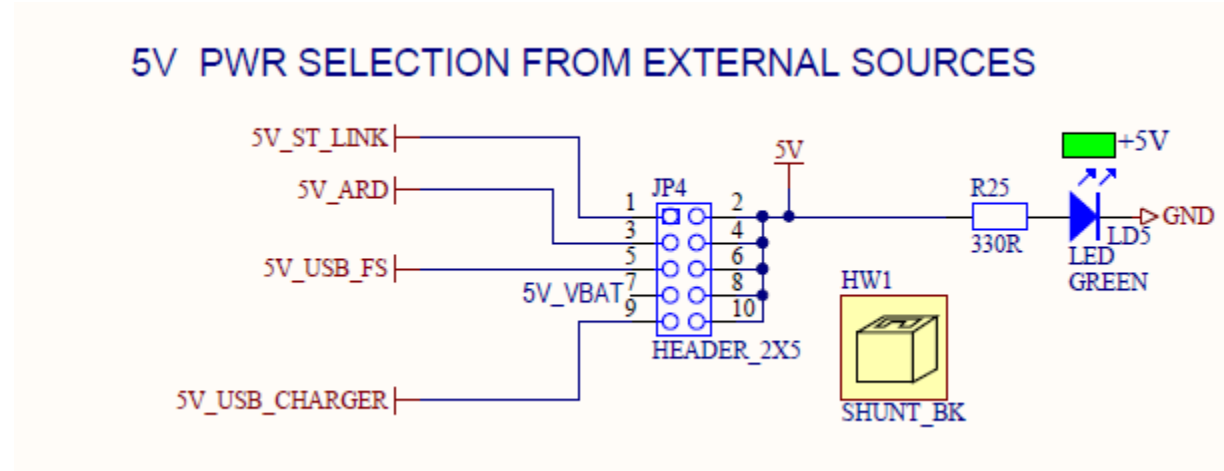
Appendix, schematic for ST-LINK USB Connector



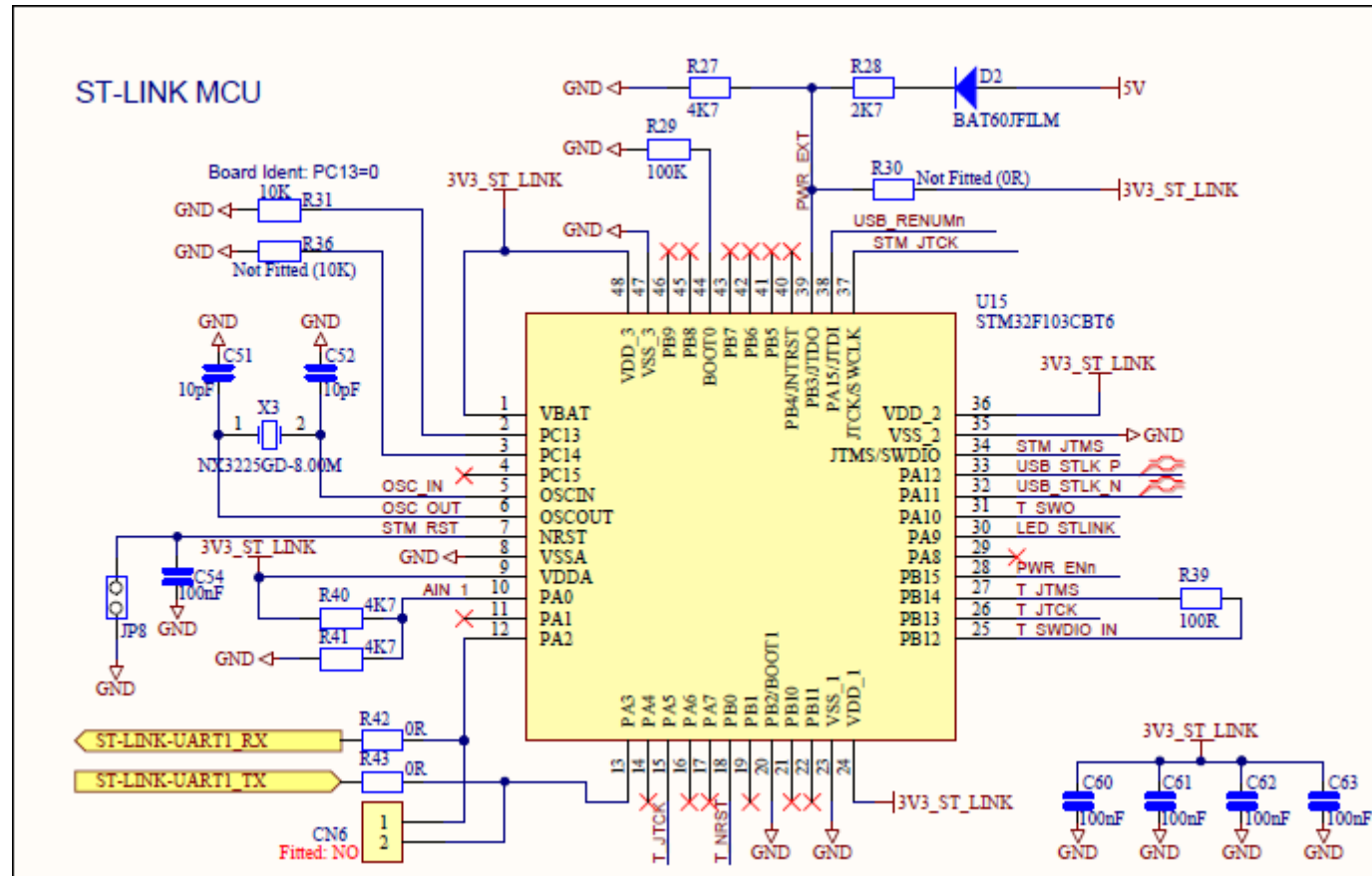
Appendix, schematic for ST-LINK Power Switch



Appendix, schematic for 5V PWR SELECTION



Appendix, schematic for ST-LINK MCU



Appendix, schematic for USB_OTG_FS

