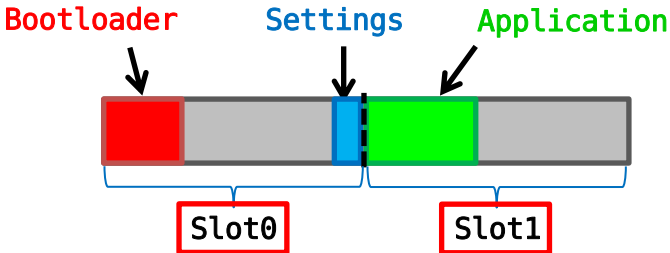


New Directory Structure

```
HexFlashLab/
├── Projects/
│   ├── STM32F103/
│   │   ├── STM32F103C8T6-Slot0/
│   │   └── STM32F103C8T6-Slot1/
│   ├── STM32F407/
│   └── ...
│       ├── STM32F407VET6-Slot0/
│       └── STM32F407VET6-Slot1/
└── Shared/
    ├── Examples/
    │   ├── 01_led_serial_control.c
    │   ├── 02_flash_read_write_erase.c
    │   ├── 03_flash_information.c
    │   ├── 04_flash_to_from_file.c
    │   └── 05_fast_blink.c
    ├── Lib/
    │   ├── flash.c
    │   └── uart.c
    └── HTML/
        ├── 01_led_serial_control.html
        ├── 02_flash_read_write_erase.html
        ├── 03_flash_information.html
        ├── 04_flash_to_from_file.html
        └── 05_fast_blink.html
```

New Directory Structure

```
HexFlashLab/  
├── Projects/  
│   ├── STM32F103/  
│   │   ├── STM32F103C8T6-Slot0/  
│   │   └── STM32F103C8T6-Slot1/  
│   ├── STM32F407/  
│   └── ...  
│       ├── STM32F407VET6-Slot0/  
│       └── STM32F407VET6-Slot1/  
└── Shared/  
    ├── Examples/  
    │   ├── 01_led_serial_control.c  
    │   └── 02_flash_read_write_demo.c
```



How To Select Example

```
└─ Shared/  
    └─ Examples/  
        └─ 01_led_serial_control.c  
        └─ 02_flash_read_write_erase.c  
        └─ 03_flash_information.c  
        └─ 04_flash_to_from_file.c  
        └─ 05_fast_blink.c
```

- Uncomment relevant EXAMPLE_XX define:

```
//#define EXAMPLE_01_LED_SERIAL_CONTROL  
//#define EXAMPLE_02_FLASH_READ_WRITE_ERASE  
//#define EXAMPLE_03_FLASH_INFORMATION  
#define EXAMPLE_04_FLASH_TO_FROM_FILE  
//#define EXAMPLE_05_FAST_BLINK
```

- No need to modify main.c
- When programming the second half of flash, make sure to use the Slot1 project

Slot0 / Slot1

- The Slot0 / Slot1 projects are the same (almost)
- Only two small differences

STM32Fxxxx_FLASH.ld (linker script):

```
/* Memories definition */  
MEMORY  
{  
  RAM (xrw) : ORIGIN = 0x20000000, LENGTH = 20K  
  FLASH (rx) : ORIGIN = 0x8008000, LENGTH = 32K  
}
```

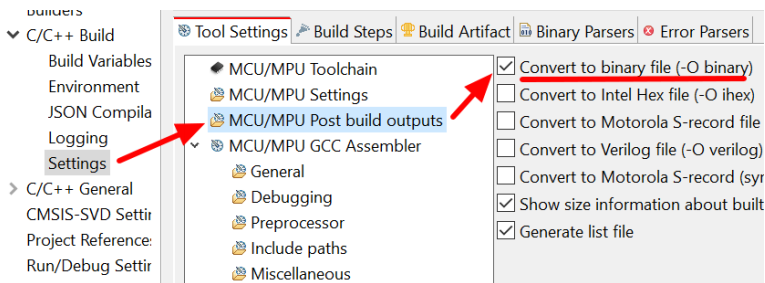
System_stm32fxxxx.c (part of HAL source):

```
#define USER_VECT_TAB_ADDRESS  
...  
#define VECT_TAB_OFFSET          0x00008000U
```

This avoids manual changes.

Slot0 / Slot1

Additionally, Slot 1 project is configured to generate binary files (to be used with flash upgrade demo):



Quick Start Guide

- Open workspace (Set folder to HexFlashLab)
- File ⇒ Open Projects from File System...
- Navigate to HexFlashLab folder, click “Finish”
- For first half of flash use “Slot0” project
- For second half of flash use “Slot1” project
- Select example using defines from examples.h

Bootloader Example

- Open “Slot0” project
- Open examples.h
- Define EXAMPLE_04_FLASH_TO_FROM_FILE
- Flash the STM32 (Run as STM32 Application)
- Close “Slot0” project
- Open “Slot1” project
- Open examples.h
- Define EXAMPLE_05_FAST_BLINK
- Project ⇒ Build Project (or Ctrl+B for all)
- Open 04_flash_to_from_file.html
- Flash BIN file from Debug folder of Example 5

Adding the HexFlashLib

(Only if creating a project from scratch)

- Right Click project name ⇒ New ⇒ Folder
 - Click “Advanced >>>” button
 - Select Link to alternate location (Linked Folder)
 - Browse to Shared folder of HexFlashLab
 - Add **Examples** folder
 - Repeat the same for **Lib** folder
-
- Right Click project name ⇒ Properties
 - C/C++ General ⇒ Paths and Symbols
 - Open “Source Location” tab ⇒ Add Folder
 - Add both **Examples** and **Lib** folders
 - Open “Includes” tab ⇒ Add... ⇒ Workspace...
 - Add both **Examples** and **Lib** folders

Adding the HexFlashLib

- Project Explorer ⇒ Open **main.c**
- Include application code

```
/* USER CODE BEGIN Includes */  
#include "examples.h"  
/* USER CODE END Includes */
```

- Select application from examples.h:

```
//#define EXAMPLE_01_LED_SERIAL_CONTROL  
//#define EXAMPLE_02_FLASH_READ_WRITE_ERASE  
//#define EXAMPLE_03_FLASH_INFORMATION  
#define EXAMPLE_04_FLASH_TO_FROM_FILE  
//#define EXAMPLE_05_FAST_BLINK
```

- No need to copy-paste between projects
- No need to modify **main.c** of each project
- Single **example_xxxx.c** source file for all STM32 projects

Adding the HexFlashLib

- Add **app_boot()** into USER CODE **1**

```
/* USER CODE BEGIN 1 */  
APP_Boot();  
/* USER CODE END 1 */
```

- Add **app_init()** into USER CODE **2**

```
/* USER CODE BEGIN 2 */  
APP_Init(&huart1);  
/* USER CODE END 2 */
```

- Add **app_loop()** into USER CODE **WHILE**

```
/* USER CODE BEGIN WHILE */  
while (1)  
{  
    APP_Loop();  
/* USER CODE END WHILE */
```

Slot1 (Second Half of Flash)

(Only if creating a project from scratch)

- Open linker script (STM32xxxx_FLASH.ld)
- Set **FLASH(rx): ORIGIN** to second half of flash
- Open system_stm32xxxx.c
- Uncomment **USER_VECT_TAB_ADDRESS**
- Set **VECT_TAB_OFFSET** offset to half of flash size
- Right Click project name ⇒ Properties
- Navigate to C/C++ Build ⇒ Settings
- Select MCU/MPU Post build outputs
- Enable **“Convert to binary file”**

Additional Notes

- You don't need to keep the structure of examples.h or use shared folders
- This is just an example of how library and example source code can be reused across multiple STM32CubeIDE projects
- **Make sure to backup your code if you use shared folders**