

FW_upgrade

開発環境	Ac6 System Workbench for STM32
基板	STM32F4DISCOVERY
CPU	STM32F407VGT6
ARM	Cortex-M4

STM32F4-Discovery_FW_V1.1.0に含まれる、FW_upgrade(Upgrading STM32F4DISCOVERY board firmware using a USB key)を、Ac6 System Workbench for STM32を用いて、動かすことが出来たので紹介する。

FW_upgradeは、USBメモリ(マストレージ)から、DiscoveryのFlashROMプログラムを書き込んだり、読み出したりできるプログラムである。

私は最初、USB Keyという言葉に惑わされて、USB Keyboard(HID)にまつわるプロジェクトだとばかり思い込んでいた。あるいは、Upgradingという言葉に惑わされて、何か特別なパッチを当てるためのものかと思い込んでいた。

ユーザスイッチを押しながら立ち上げると、USBメモリのルートにあるimage.binというファイルが書き込まれる。

このプログラムで書き込もうとするプログラムは、0x08008000番地から開始するものである必要がある。

STM32F4-Discovery_FW_V1.1.0のプロジェクトのBinaryフォルダには、～～～_0x08008000.binという名前のファイルがあらかじめ用意されているので、それをリネームして使うことが出来る。

ユーザスイッチを押さずに立ち上げると、正当なプログラムが書き込まれていると、それが実行される。

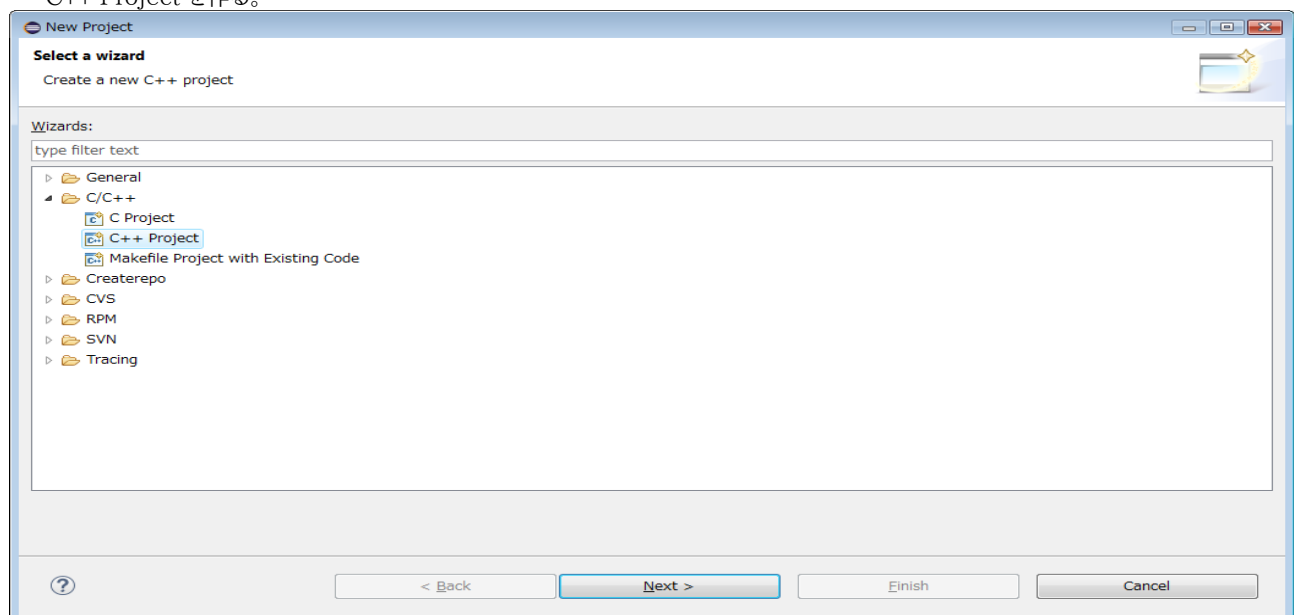
起動時にユーザスイッチを3秒以上推すと、書き込んだあと、UPLOAD.BINというファイルに読み出してくれる。

★但しこれにはバグがある。書き込みが3秒以下で終了すると、読み出しは絶対に行なわれない。
ので、書き込みの終了を、少なくとも3秒待つように、修正を加えている。

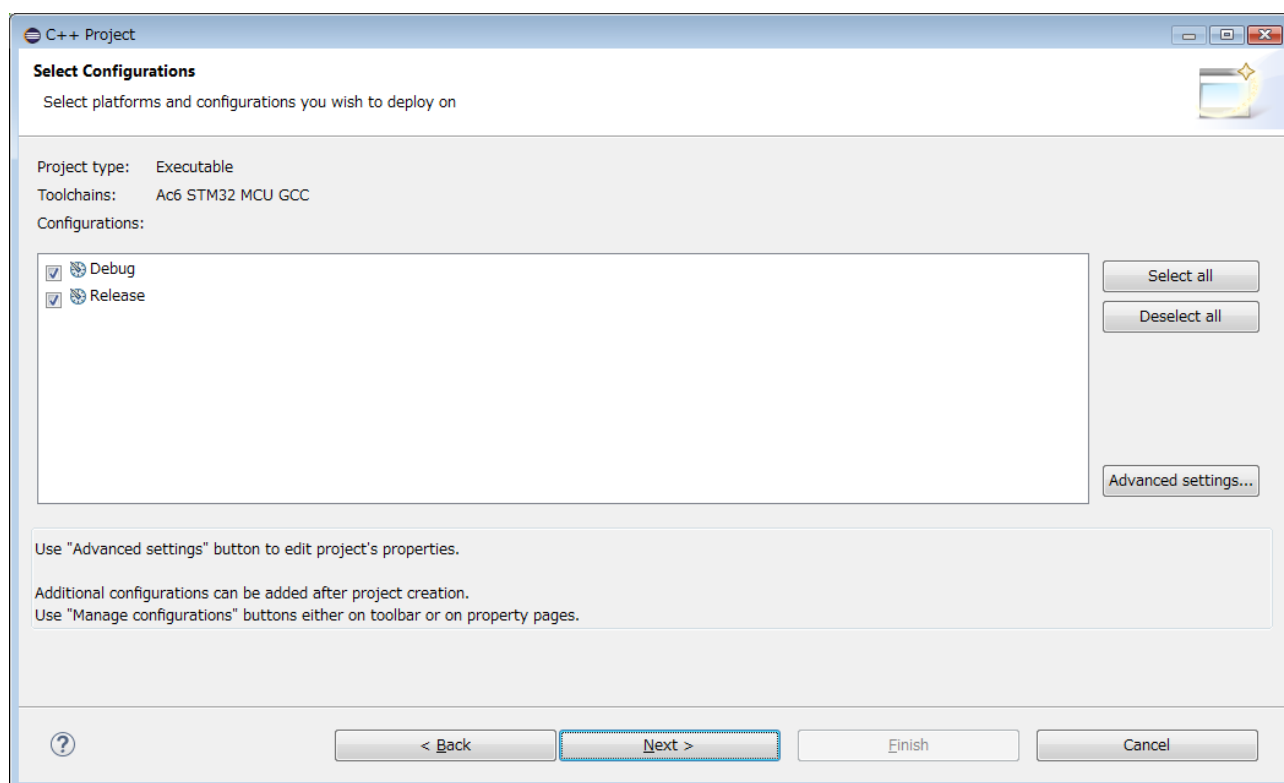
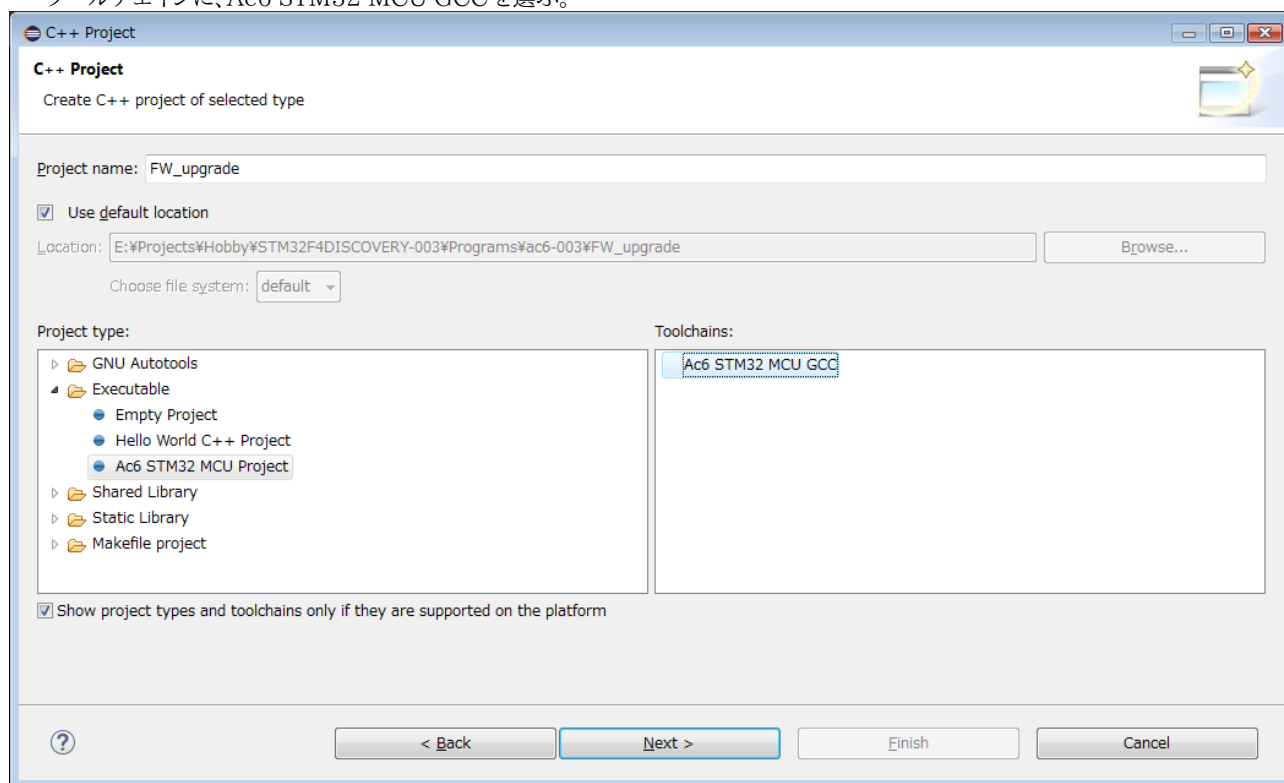
★このプロジェクトは、デバッグモードでは、FlashROMへの書き込みがうまく行なえない。
リリースビルドを焼き、スタンドアローンで実行させる必要がある。

■以下の要領で、プロジェクトをつくる。

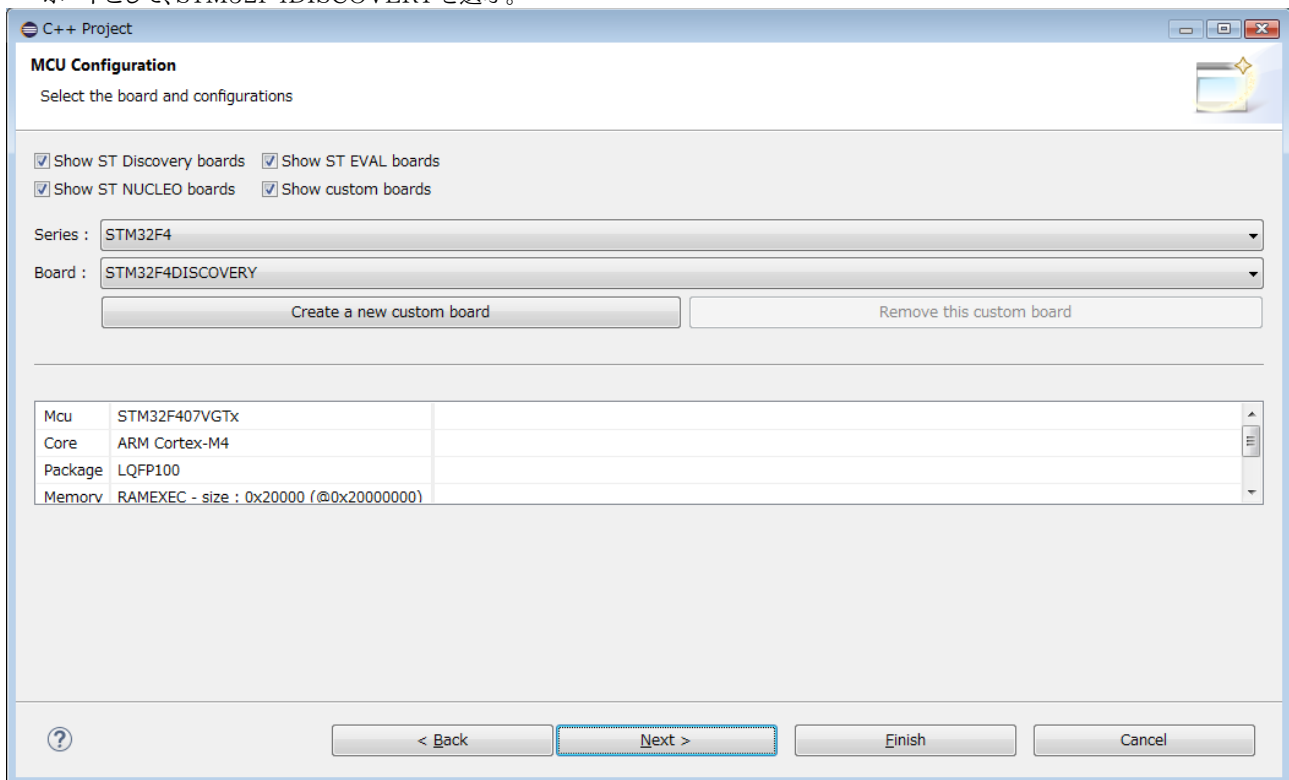
・C++ Projectを作る。



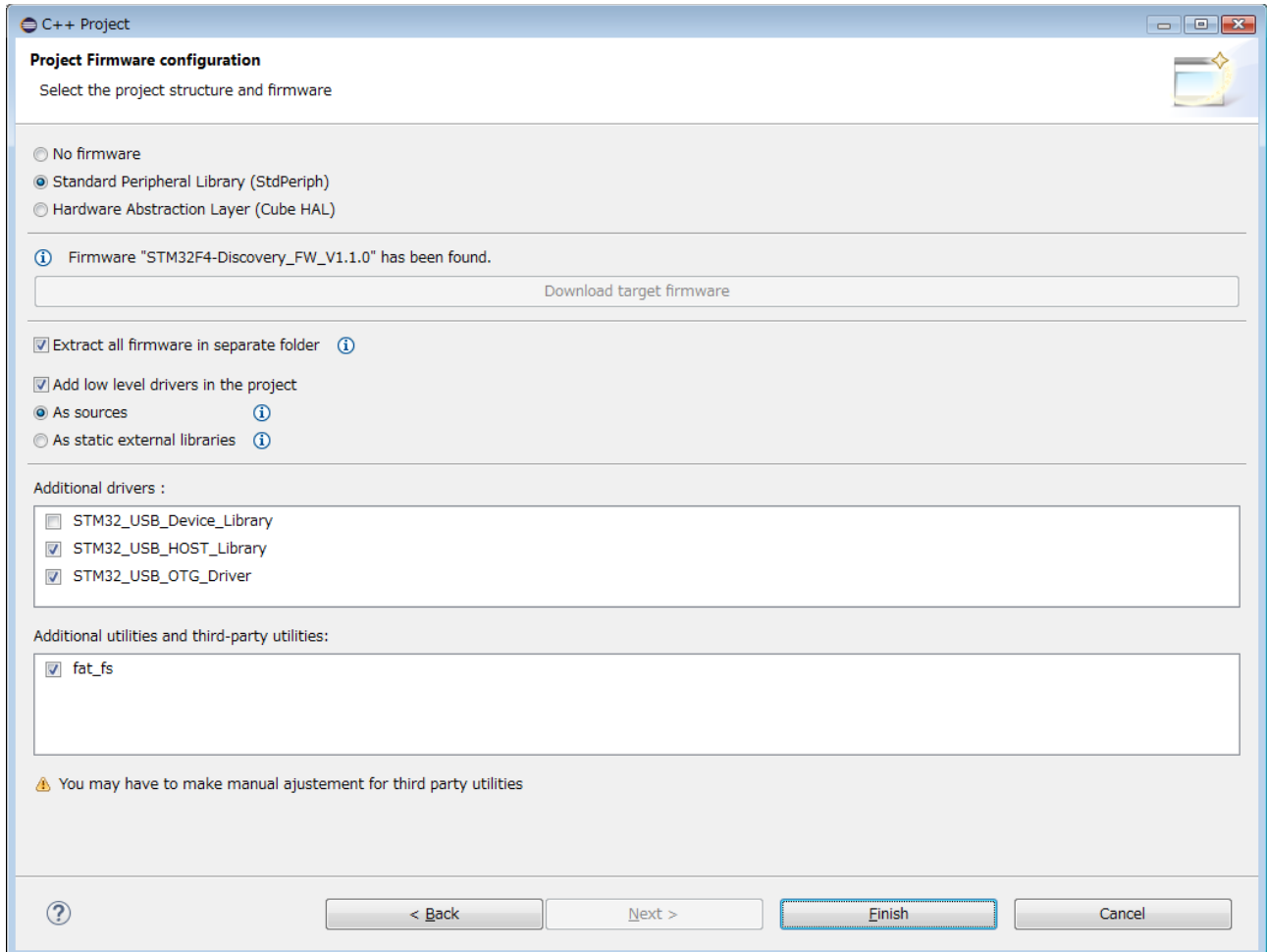
・ツールチェーンに、Ac6 STM32 MCU GCCを選ぶ。



・ボードとして、STM32F4DISCOVERYを選ぶ。



- ・STM32_USB_HOST_Libraryを追加する。
- ・STM32_USB_OTG_Driverを追加する。
- ・fat_fsを追加する。
- ・ソースコードを生成させる。
- ・全ソースコードを別のフォルダーに生成させる。



■プロジェクトの修正

- FW_upgrade のファイルをコピーしてくる。

	inc	STM32F4-Discovery_FW_V1.1.0¥Project¥FW_upgrade¥inc
	src	STM32F4-Discovery_FW_V1.1.0¥Project¥FW_upgrade¥src

- つぎのファイルをプロジェクトから外す

- ・拡張子を違うものにするだけで、ビルドから外れる。

usb_bsp_template.c usb_dcd.c usb_dcd_int.c usb_otg.c	USB_OTG_Driver¥src
---	--------------------

ccsbcs.c syncobj.c diskio.c	fat_fs¥src¥option
-----------------------------------	-------------------

(ソースコード変更)

usb_conf.h (@inc)

```
#ifndef USE_USB_OTG_FS
//<yf>
#define USE_USB_OTG_FS
#endif /* USE_USB_OTG_FS */

#ifndef USE_EMBEDDED_PHY
//<yf>
#define USE_EMBEDDED_PHY
#endif /* USE_EMBEDDED_PHY */
```

■書き込みに3秒かからなくても読み出しが行なわれるようにするパッチ

usbh_usr.c (@src)

```
int USBH_USR_MSC_Application(void)
{
    EXTI_InitTypeDef EXTI_InitStructure;

    switch (USBH_USR_ApplicationState)
    {
        <<省略>>

        case USH_USR_IAP:
            <<省略>>

            /* Writes Flash memory */
            COMMAND_DOWNLOAD();

            while(TimingDelay);           //<yf>

            /* Initialize User_Button on STM32F4-Discovery in the GPIO Mode -----*/
            STM_EVAL_PBInit(BUTTON_USER, BUTTON_MODE_GPIO);
```