



不可能への挑戦

株式会社日昇テクノロジー

低価格、高品質が不可能？
日昇テクノロジーなら可能にする

ARM Cortex-M3 RedBull-STM32F103ZET6

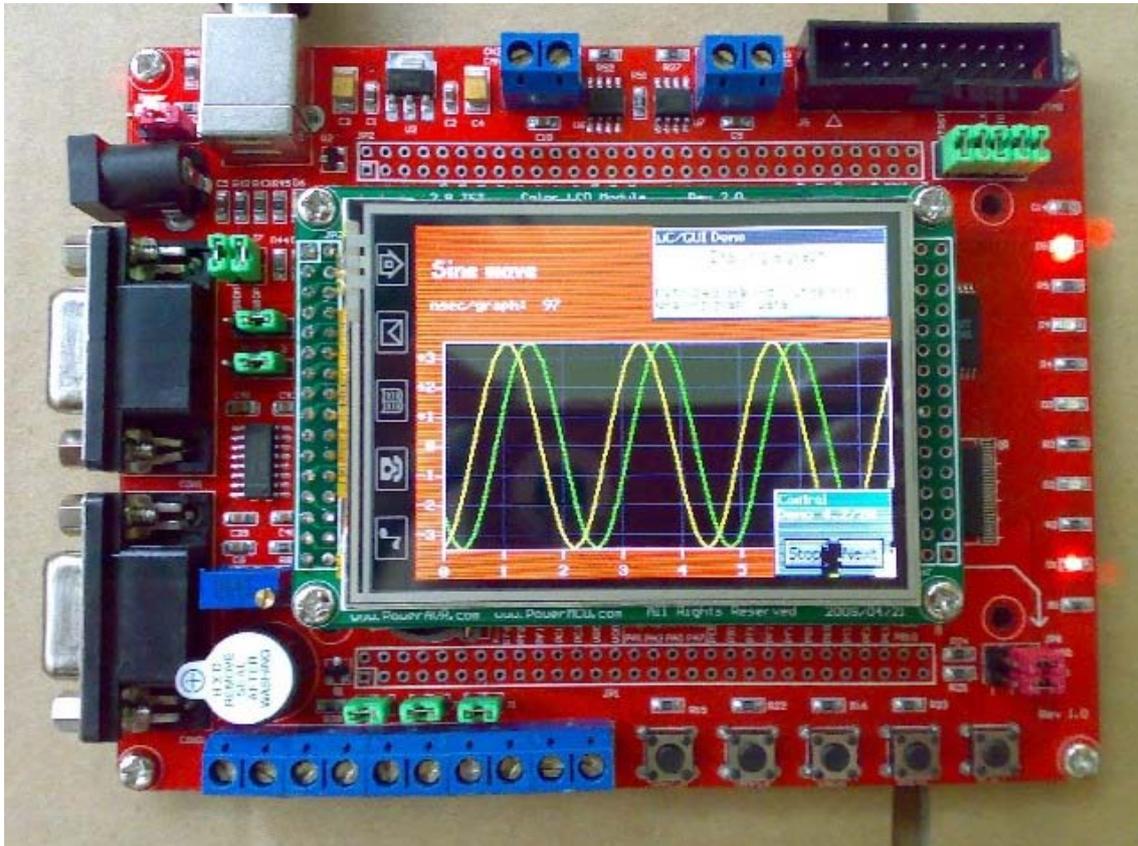
マニュアル

株式会社日昇テクノロジー

<http://www.csun.co.jp>

info@csun.co.jp

2010/05/18



copyright©2010



• 修正履歴

NO	バージョン	修正内容	修正日
1	Ver1.0	新規作成	2010/05/18

※ この文書の情報は、文書を改善するため、事前の通知なく変更されることがあります。最新版は弊社ホームページからご参照ください。

「<http://www.csun.co.jp>」

※ (株)日昇テクノロジーの書面による許可のない複製は、いかなる形態においても厳重に禁じられています。



目次

第一章 RedBull-STM32F103ZE 概要.....	5
1.1 主な特徴.....	5
第二章 回路の説明.....	6
2.1 Power Supply (電源供給)	6
2.2 Boot option (起動オプション)	6
2.3 Clock source	7
2.4 リセット方法.....	7
2.5 アナログ信号のインプット	7
2.6 PWM アウトプット	7
2.7 DAC アウトプット	7
2.8 USB	7
2.9 LCD インタフェース.....	8
2.10 SRAM.....	8
2.11 NAND Flash.....	8
2.12 NOR Flash.....	8
第三章 インタフェース定義 (Connectors)	10
3.1 アナログインプット、PWM アウトプット、DAC アウトプット IF CN1 の定義.....	10
3.2 CAN Bus インタフェース CN2 の定義.....	10
3.3 RS485 通信インタフェース CN3 の定義.....	10
3.4 RS232 通信インタフェース CON1、CON2 の定義.....	10
3.5 JTAG デバッグインタフェース J5 の定義.....	11
3.6 USB 2.0 インタフェース定義.....	11
3.7 JTAG と SWD デバッグモード選択ジャンパ定義 JP6.....	11
第四章 タッチパネル付けの 3.2 インチ/2.8 インチ TFT 液晶.....	13
4.1 タッチパネル付けの 3.2 インチ TFT 液晶.....	13
4.2 タッチパネル付けの 2.8 インチ TFT 液晶.....	14
第五章 実行ファイルの書き込み	15
5.1 シリアルポートで書き込む	15
5.2 OpenLink で書き込む.....	20
5.2.1 ドライバのインストール.....	20
5.2.2 J-FLASH ARM で実行ファイルを書き込む.....	23
5.3 H-JTAG で実行ファイルを書き込む.....	28



第六章	OpenLink でデバッグ	34
6.1	J-Link command でデバッグ	34
第七章	開発ツール KEIL の応用	35
7.1	KEIL のインストール	35
7.2	既存のプロジェクトから	37
7.3	新しいプロジェクトの作成	41



第一章 RedBull-STM32F103ZE 概要

ARM コア新型プロセッサCortex-M3 を採用した ST マイクロエレクトロニクス社の STM32F103ZET6 (LQFP144)。

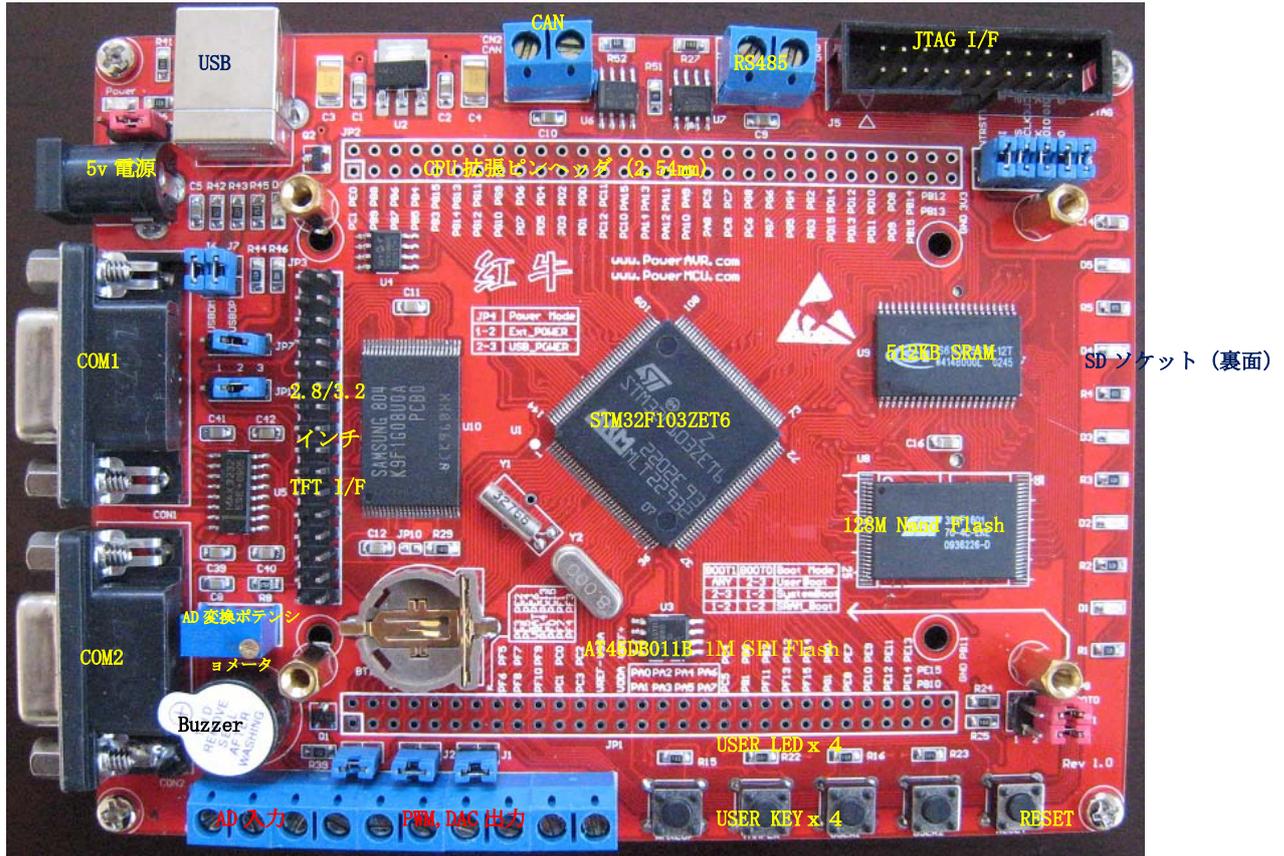
標準外付け：512K flash、64KRAM、12BitADC、DAC、PWM、CAN、USB、SDIO、FSMCなど。

豊富なハードウェアの上、色々なサンプルソースを提供しているので、初心者最適です。

1.1 主な特徴

- STM32F103ZE、LQFP144、512KB FLASH/64KB RAM内蔵。
- 外付け 512KB SRAM、2M NOR FLASH、大容量のデータ採集、処理と分析ができる。更に大容量に拡張可。
- 128MB Nand Flash、画像などのデータを出来る。大容量に拡張可。
- RS485 x 1、RS232 x 2 (DB9)
- CAN BUS x 1、SN65VHD230
- GUI、3.2或いは2.8インチ、320*240、26 万色TFT-LCD、8/16BitのBUSをサポートする、16Mbit SPI Flash(AT45DBxxx)未実装、SDソケット付き、タッチパネル (ADS7843) 付き
- ブーザ x 1、LED x 5、電源LED x 1、USB通信指示LED x 1
- ユーザーボタン x 4、RESET x 1
- SPI インタフェース、AT45DB011B(DATA FLASH)
- IIC インタフェース、24LC02 (EEPROM)
- SDIO式SDインタフェース
- ADC x 3
- DAC x 2
- PWM x 2
- ポテンショメータ入力アナログ信号 x 1
- 標準JTAG/ICE デバッグ用インタフェース (20pin)
- 5V電源、またはUSB ポート或いはJLink給電、ジャンパで選択
- CPU のすべての IO を 2.54mm 拡張ピンヘッダで引き出される
- 外形寸法：130×100 (mm) ※突起物は除く

第二章 回路の説明



2.1 Power Supply (電源供給)

ジャンパ JP4 より 5V 外部電源あるいは 5V USB で電源供給する。

Jumper JP4	説明
1 - 2	5V 外部電源供給
2 - 3	USB で電源供給

2.2 Boot option (起動オプション)

下記三つの方法がある：

- Embedded user Flash (デフォルト)
- System memory with boot loader for ISP
- Embedded SRAM for debugging

BOOT1 (JP8) と BOOT0 (JP9) で制御する。

BOOT1 (JP8)	BOOT0 (JP9)	
ANY (1-2、2-3 or open)	2-3	Embedded user Flash (デフォルト) モード
2-3	1-2	System memory モード



1-2	1-2	Embedded SRAM モード
-----	-----	-------------------

2.3 Clock source

FireBull ボードは五つのクロックでシステム、RTC、USB HOST、Ethernet、Mp3 クロックを発生する。

- Y1, 32.768kHz クリスタル、RTC のクロックを発生する。
- Y2, 8MHz クリスタル、システムのクロックを発生する。

2.4 リセット方法

本ボードでは低電圧信号でリセット発生する。

方法は下記二つ：

- Reset (S1) キー操作。
- JTAG エミュレータよりリセット信号を発生させる。

2.5 アナログ信号のインプット

CN1 の AIN0、AIN1、AIN2 を STM32F103ZE のアナログ信号のインプットピン PC0、PC1、PC2 と接続する。

2.6 PWM アウトプット

CN1 の PWM0、PWM1 を STM32F103ZE のタイマーアウトピン PB0、PB1 と接続する。

2.7 DAC アウトプット

CN1のDAC0、DAC1をSTM32F103ZEのDACアウトプットピンPA4、PA5と接続する。DAC0、DAC1アウトプットピンはSPI1_NSS、SPI1_SCKとしても利用されているので、DAC0、DAC1アウトプットピンとして利用したい場合はJ1、J2をオープンする必要。J1、J2はデフォルトではショートでSPI1に設定されている。

2.8 USB

本ボードはUSB B型のUSB2.0インタフェース(J4)を提供している。このUSBで最大500mAの電流を提供する。

1.5kΩ抵抗をUSB+と直接接続するか或いはIO制御通じて1.5kΩ抵抗をUSB+と接続するか、二種類ある。ジャンパJP7で選択する。

ジャンパ JP7	説明
1-2	1.5kΩ抵抗をUSB+と直接接続する
2-3	ソフトでIO(PB5)を通じて1.5kΩ抵抗をUSB+と接続或いは不接続する



2.9 LCD インタフェース

320X240 TFTカラー LCDデータ線はSTM32F103ZEのFSMCのbank1 NOR/PSRAM4と接続する。5個の赤色のLED (LD1、2、3、4、5) は標準IOのPF6、7、8、9、10と接続する。

JP3 カラーTFT LCD インタフェース

Pin	信号説明	IO	Pin	信号説明	IO	Pin	信号説明	IO
1	3V3	電源	2	GND	GND	3	DB00	PD14
4	DB01	PD15	5	DB02	PD0	6	DB03	PD1
7	DB04	PE7	8	DB05	PE8	9	DB06	PE9
10	DB07	PE10	11	DB08	PE11	12	DB09	PE12
13	DB10	PE13	14	DB11	PE14	15	DB12	PE15
16	DB13	PD8	17	DB14	PD9	18	DB15	PD10
19	CS	PG12	20	RS	PF0	21	WR	PD5
22	RD	PD4	23	RESET	RESET	24	NC	PA1
25	MISO	PB14	26	INT	PG7	27	MOSI	PB15
28	LE	PG8	29	SCLK	PB13	30	F_CS	PG11
31	TP_CS	PB12	32	SD_CS	PG15			

2.10 SRAM

256kx16 SRAMはSTM32F103ZEのFSMCのbank1 NOR/PSRAM3と接続する。8-bitと16-bitをサポートする。

2.11 NAND Flash

1Gbitx8或いは2Gbitx8 NAND FlashをSTM32F103ZEのFSMCのNAND bank2と接続する。NAND Flashのready/busy信号はジャンパJP11でSTM32F103ZEのFSMCのWAIT信号或いはFSMC_INT2信号と接続する。

ジャンパ JP11	説明
1-2	ready/busy 信号は FSMC の WAIT 信号と接続する (デフォルト)
2-3	ready/busy 信号は FSMC_INT2 信号と接続する

2.12 NOR Flash

16Mbit (2MByte) NOR FlashをSTM32F103ZEのFSMCのbank1 NOR/PSRAM2と接続する。NOR Flashはプルアップ抵抗でBYTEピンと接続して16-bit操作モードを選択する。NOR Flashの書き込み保護はジャンパJP12をショートして実現する。



不可能への挑戦

株式会社日昇テクノロジー

低価格、高品質が不可能？
日昇テクノロジーなら可能にする

ジャンパ JP12	説明
ショート	書き込み禁止
オープン	書き込み可 (デフォルト)

第三章 インタフェース定義 (Connectors)

3.1 アナログインプット、PWM アウトプット、DAC アウトプット IF CN1 の定義

Pin number	Description	Pin number	Description
1	AIN0	6	PWM1
2	AIN1	7	GND
3	AIN2	8	DAC0
4	GND	9	DAC1
5	PWM0	10	GND

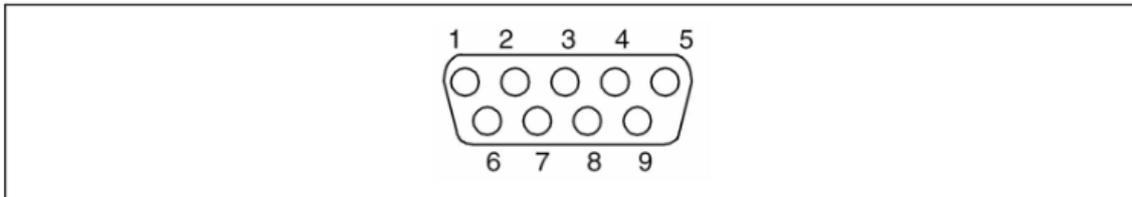
3.2 CAN Bus インタフェース CN2 の定義

Pin number	Description	Pin number	Description
1	CANH	2	CANL

3.3 RS485 通信インタフェース CN3 の定義

Pin number	Description	Pin number	Description
1	485B	2	485A

3.4 RS232 通信インタフェース CON1、CON2 の定義



CON1 定義

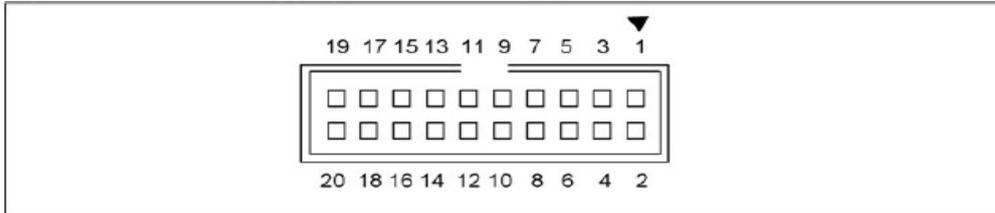
Pin number	Description	Pin number	Description
1	NC	6	NC
2	USART1_PA9	7	NC
3	USART1_PA10	8	NC
4	NC	9	NC
5	GND		

CON2 定義

Pin number	Description	Pin number	Description
1	NC	6	NC
2	USART2_PA2	7	NC
3	USART2_PA3	8	NC

4	NC	9	NC
5	GND		

3.5 JTAG デバッグインタフェース J5 の定義



Pin number	Description	Pin number	Description
1	3.3V power	2	3.3V power
3	PB4	4	GND
5	PA15	6	GND
7	PA13	8	GND
9	PA14	10	GND
11	RTCK	12	GND
13	PB3	14	GND
15	RESET#	16	GND
17	DBGRRQ	18	GND
19	DBGACK	20	GND

3.6 USB 2.0 インタフェース定義

USB-A 型はHost用、USB-B 型はデバイス用。

No.	説明	No.	説明
+	DP	—	DM
V	VBUS	G	GND

3.7 JTAG と SWD デバッグモード選択ジャンパ定義 JP6

JTAG モードを選択する場合は JP6 の五つを全てショートする必要。

SWDモードの場合はMS (SWDIO) とTCK (SWDCLK) の二つをショートすれば良い。



Pin number	Description	Pin number	Description
1	NTRST (PB4)	2	TDI (PA15)
3	TMS (PA13)	4	TCK (PA14)
5	TDO (PB3)		

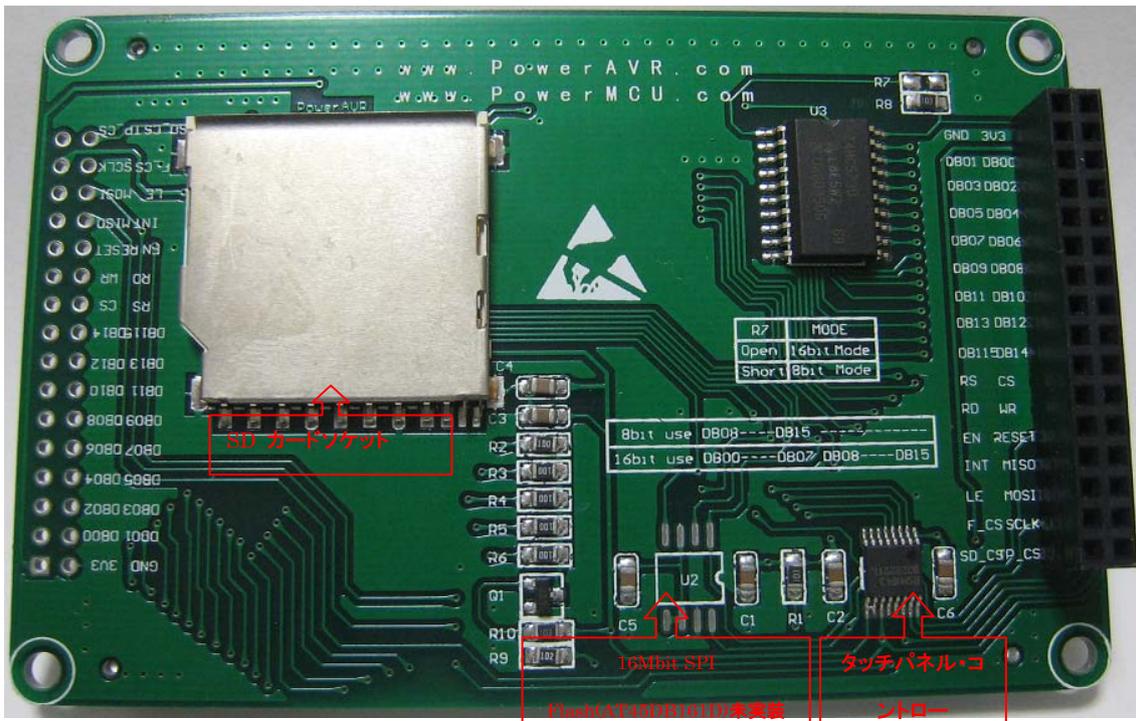
第四章 タッチパネル付けの 3.2 インチ/2.8 インチ TFT 液晶

4.1 タッチパネル付けの 3.2 インチ TFT 液晶

表面：



裏面：



- ・ 3.2 インチTFT 液晶、解像度は240(W)*320(H)
- ・ 8/16bit パラレルインタフェース
- ・ タッチパネル・コントローラADS7843 或いはTSC2046 (SPI インタフェース)
- ・ 16Mbit SPI Flash(AT45DB161D)未実装



- ・ SD カードソケット
 - ・ 使いやすい2.54mm コネクタ。
 - ・ 外形寸法：95×62(mm) ※突起物は除く
- 各ピンの詳細な説明は 2.9 LCD インタフェース を参照する事。

4.2 タッチパネル付けの 2.8 インチ TFT 液晶

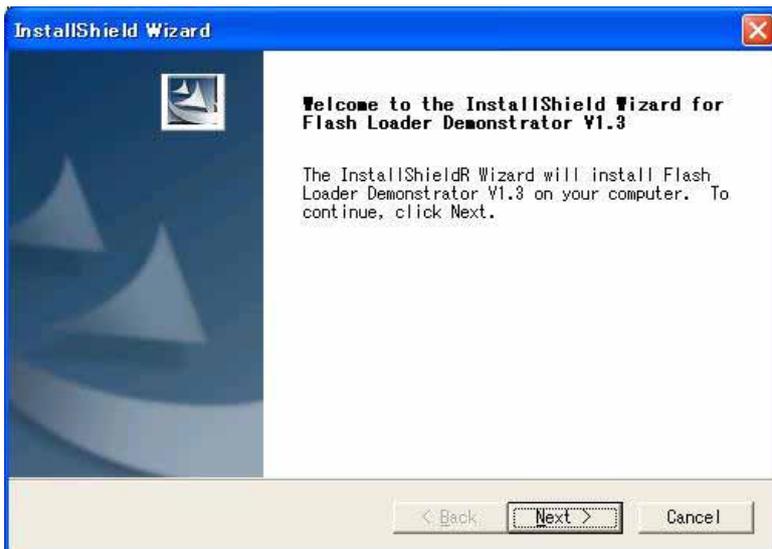
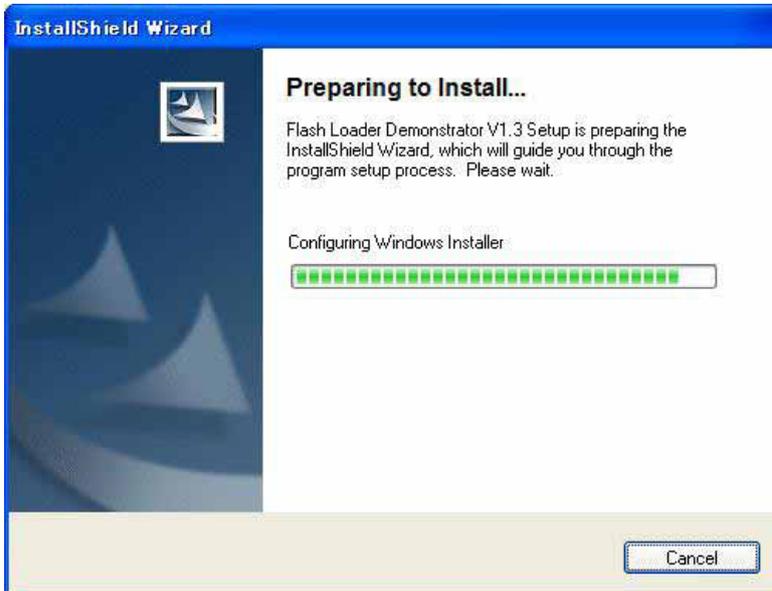
- ・ 2.8 インチTFT 液晶、解像度は240(W)*320(H)
 - ・ 8/16bit パラレルインタフェース
 - ・ タッチパネル・コントローラADS7843 或いはTSC2046 (SPI インタフェース)
 - ・ 16Mbit SPI Flash(AT45DB161D)未実装
 - ・ SD カードソケット
 - ・ 使いやすい2.54mm コネクタ。
 - ・ 外形寸法：82×55(mm) ※突起物は除く
- 各ピンの詳細な説明は2.9 LCDインタフェース を参照する事。
- 具体的な仕様は弊社HPに提供している各種チップの仕様書をご参照ください。

第五章 実行ファイルの書き込み

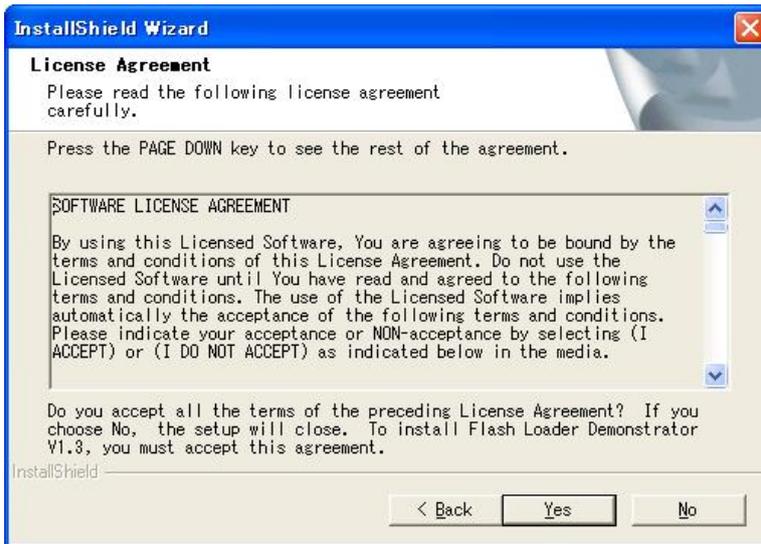
5.1 シリアルポートで書き込む

Flash Loader Demonstrator_V1.3_Setup.exeはシリアルポートでSTM32マイコンのFlashを更新するツールである。

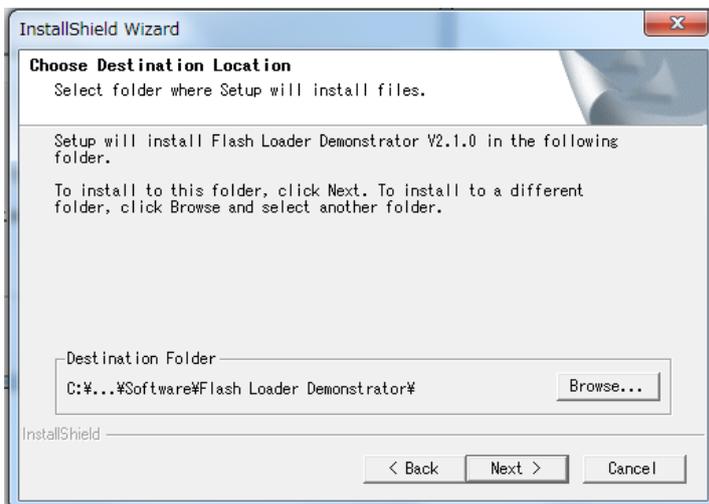
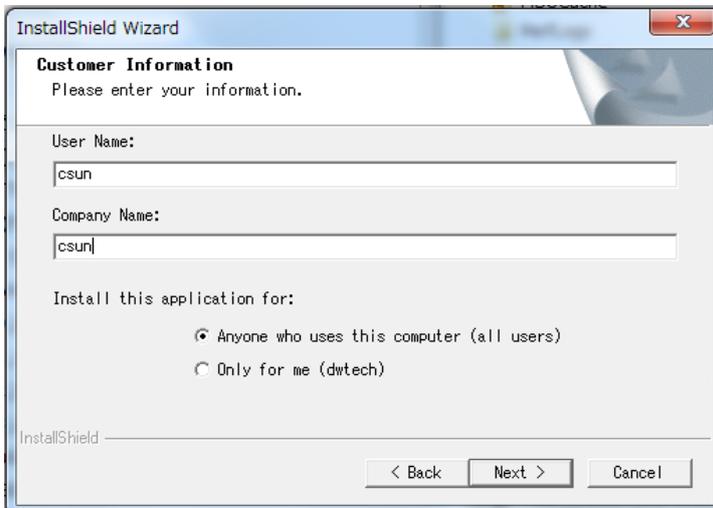
Flash Loader Demonstrator_V1.3_Setup.exeを実行する。



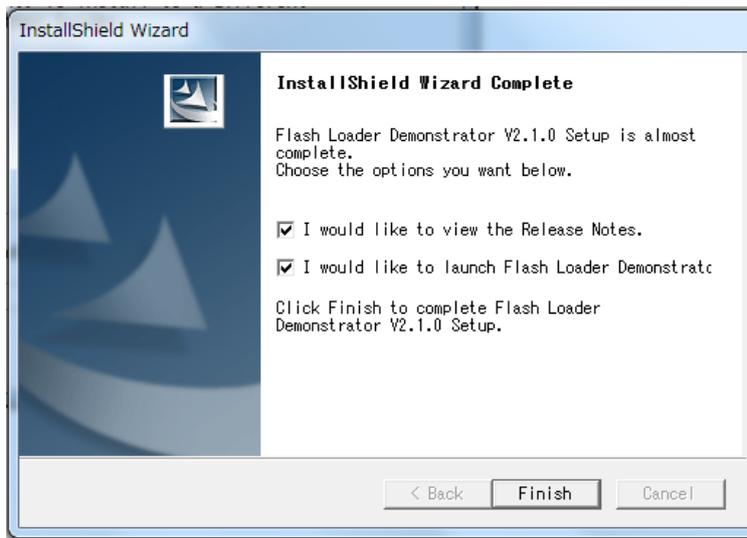
「Next」ボタンを押すと、英文のライセンスが出てきます。同意できる場合は、「Yes」ボタンを押す。



ユーザー名と会社名を入力して、「Next」ボタンを押す。



インストール先フォルダを変更せず、そのまま進んでください。



最後に「Finish」をクリックすると、ウィザードが閉じてインストールが終了。

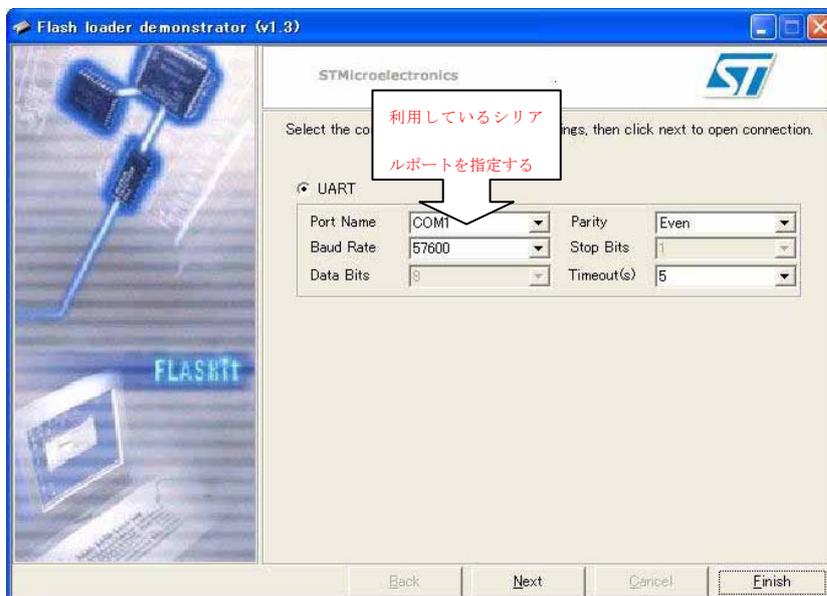
書き込む前にボードのJP9(Boot0)を1-2に設置する。

直接RS232 ケーブルでボードのCOM1 をパソコンと接続して、電源を入れる。

パソコン側に RS232 インタフェースがない場合は USB RS232 変換ケーブルで接続する。

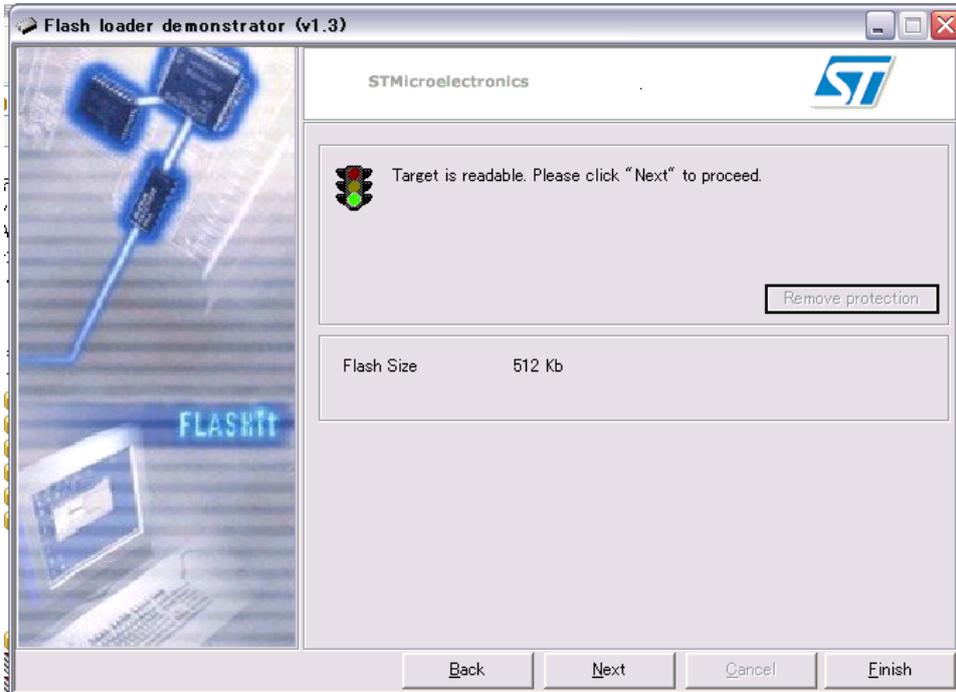
USB RS232変換ケーブル製品紹介URL : (<http://www.csun.co.jp/SHOP/2010040601.html>)

上記準備終わったら、Windowsのメニュー「スタート」→「STMicroelectronics」→「Flash Loader Demonstrator」→「Flash Loader Demo」を選択して起動する。

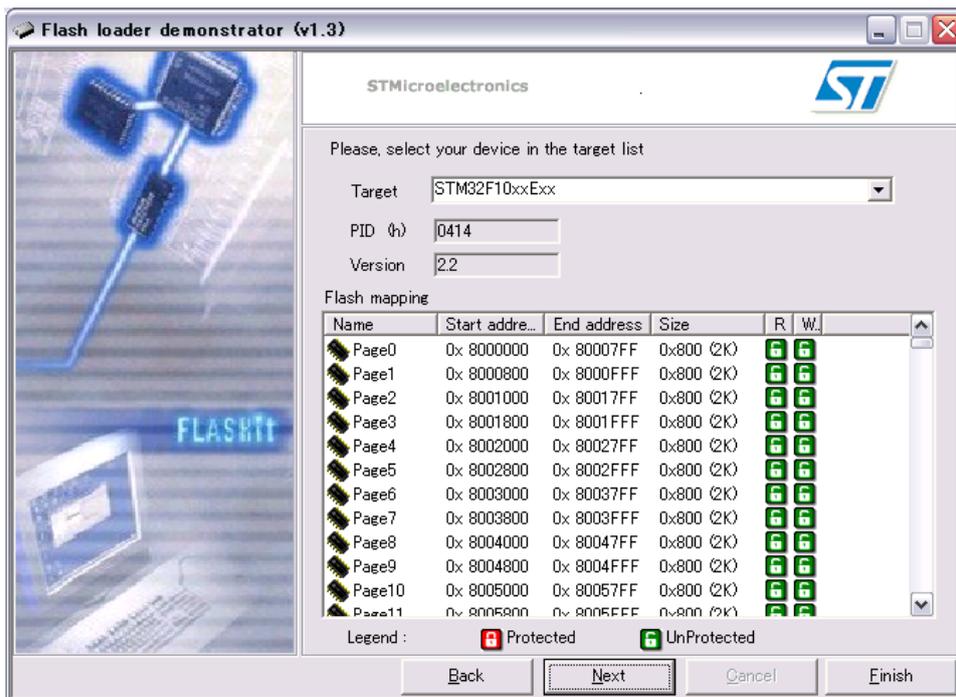


書き込み用のシリアルポートを選択して、「Next」ボタンを押す。

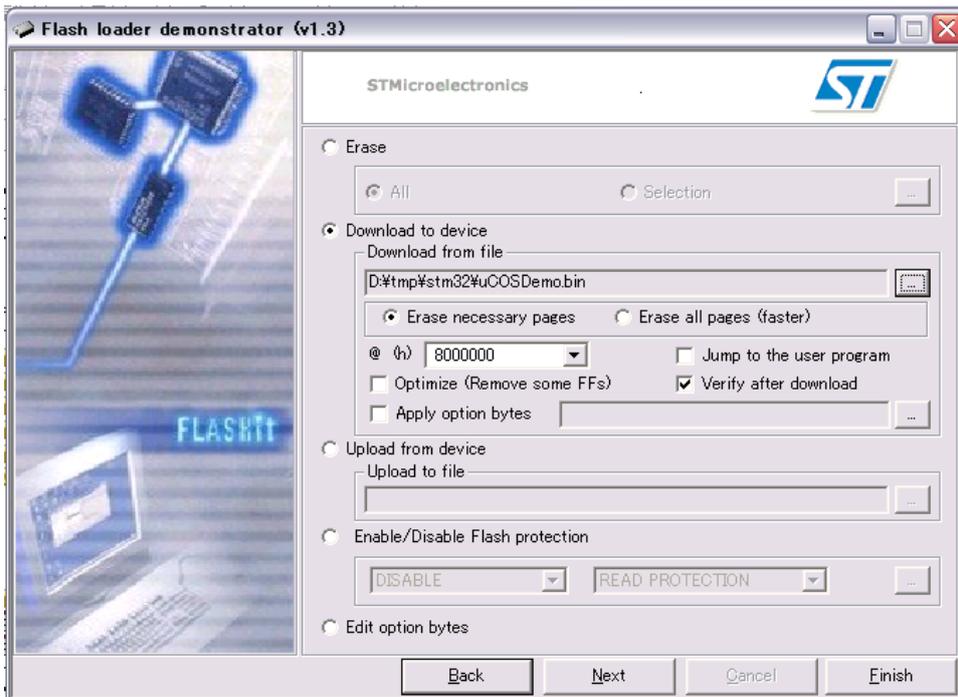
(USB-RS232ケーブルを利用している場合は、そちらの設定と合わせて設定する。)



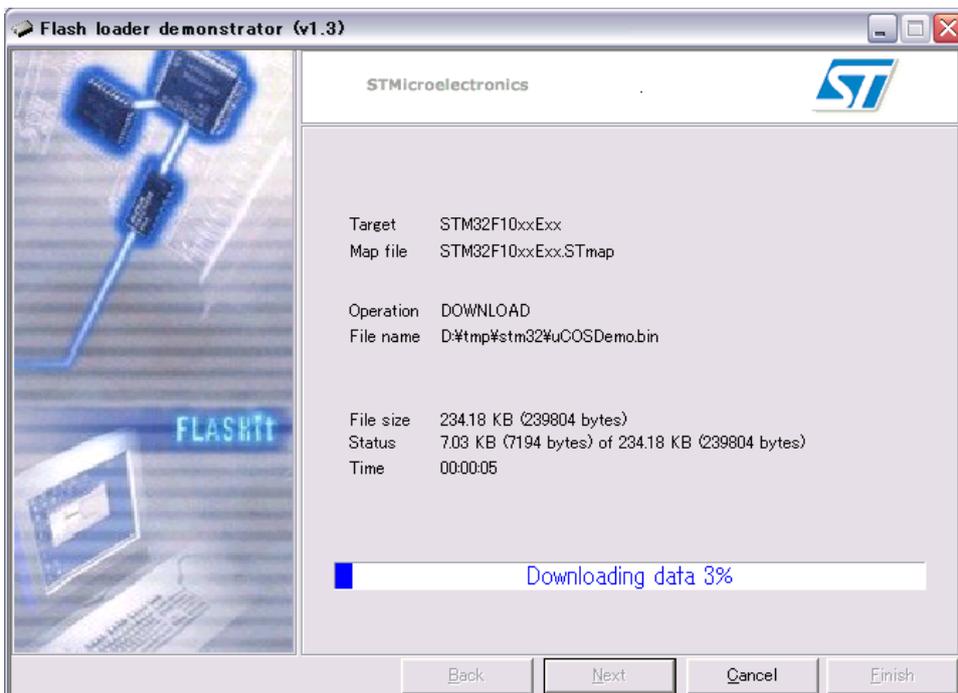
「Next」ボタンを押す。

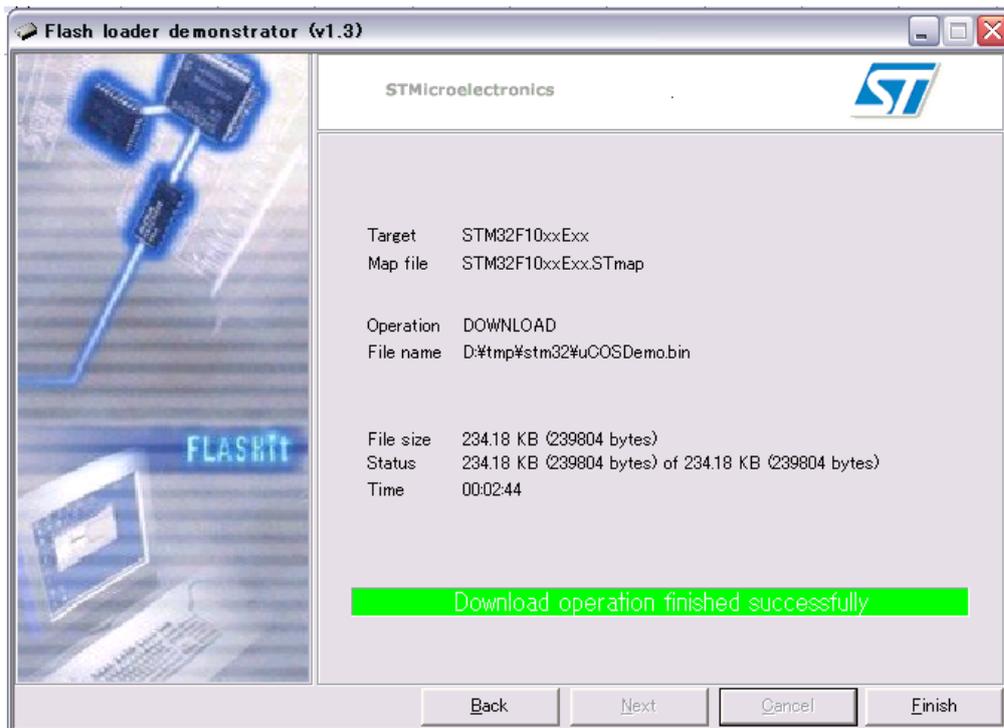


そのまま「Next」ボタンを押す。



書き込む*. Binファイルを選択して、「Next」ボタンを押す。





最後に「Finish」をクリックすると、ウィザードが閉じて書き込みが終了。

5.2 OpenLink で書き込む

弊社は **OpenLink** のハードウェアを提供しております（製品紹介 URL：<http://www.csun.co.jp/SHOP/2009121901.html>）。

5.2.1 ドライバのインストール

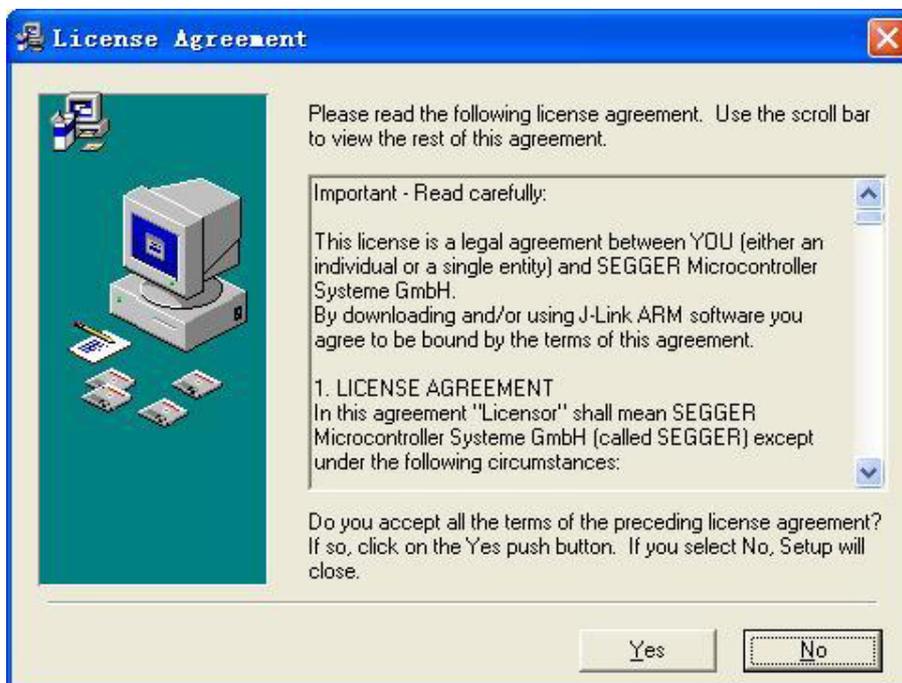
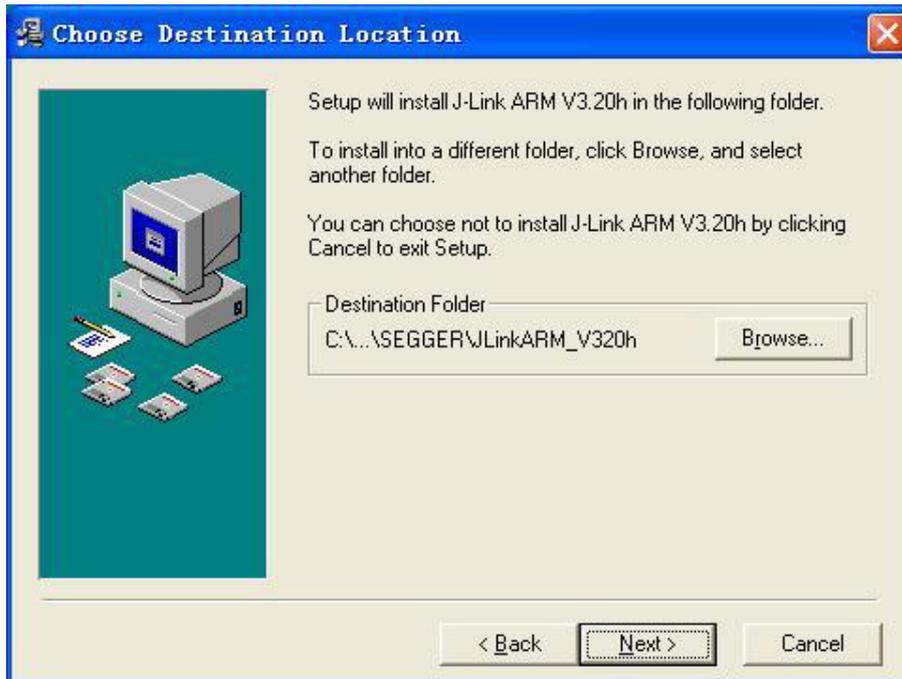
ドライバインストール用のファイルは弊社ホーム下記 URL からダウンロードできる。

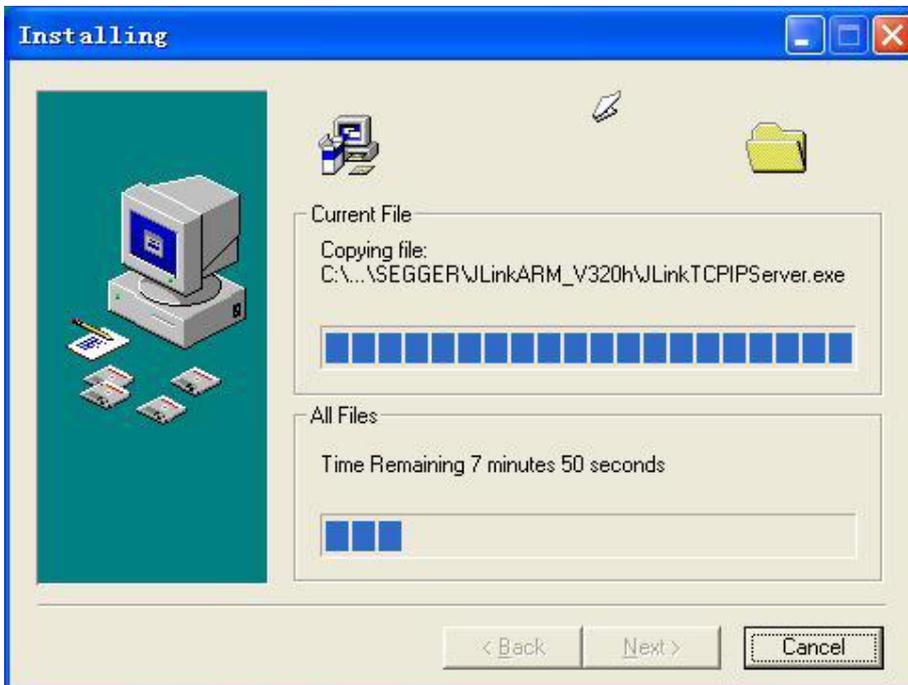
http://www.dragonwake.com/download/open-link/Setup_OpenLinkARM.zip

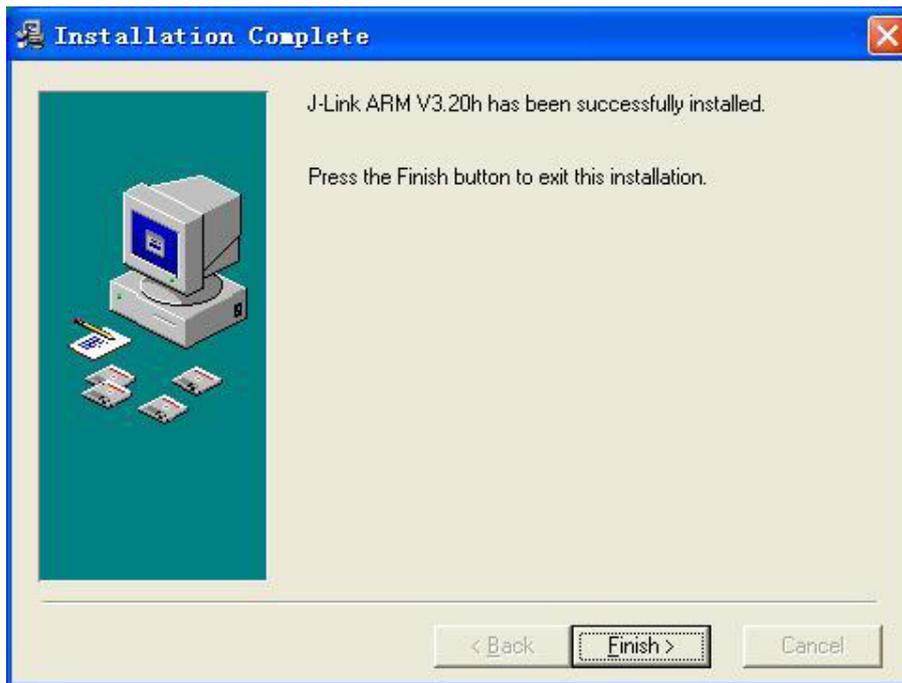
SEGGER 社様のソフトウェアを利用しておりますので、直接 SEGGER 社様ホームページから最新の USB ドライバをダウンロードできる。

<http://www.segger.com/cms/jlink-software.html>

インストールの際に、ダウンロードした ZIP ファイルを解凍し、デフォルトのままで行ってください。

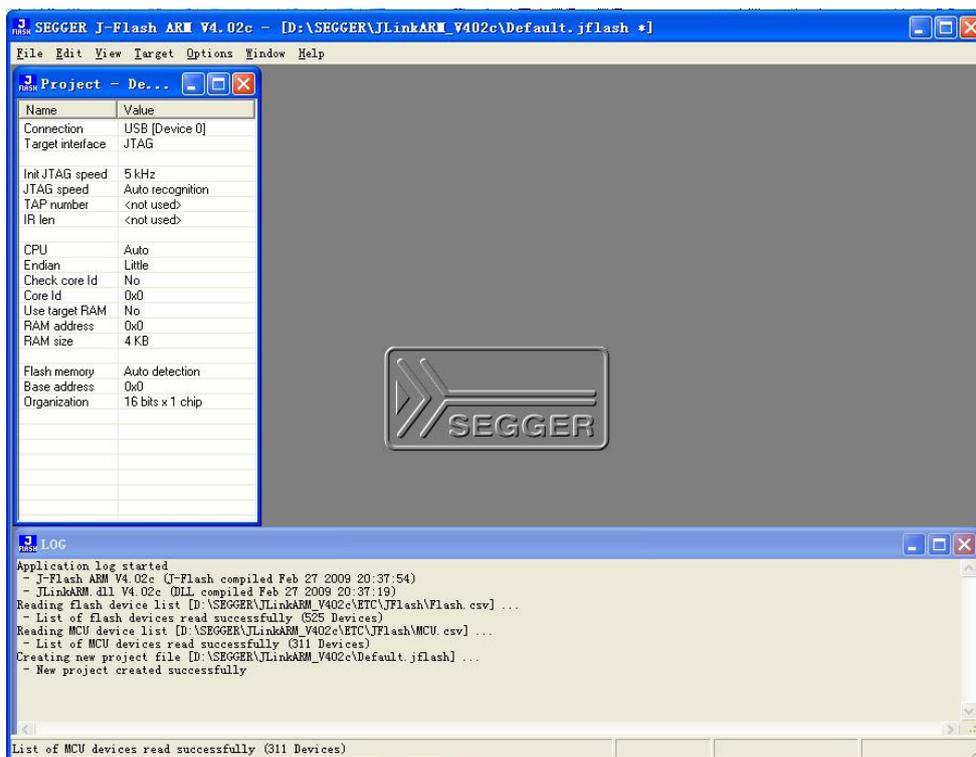




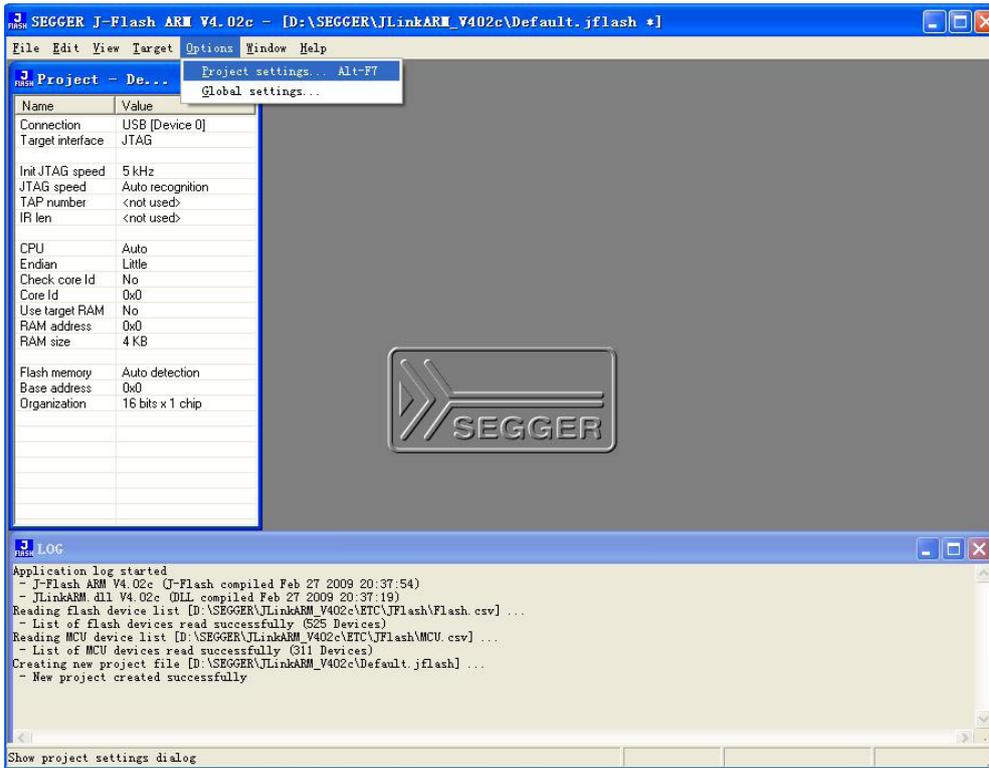


5.2.2 J-FLASH ARM で実行ファイルを書き込む

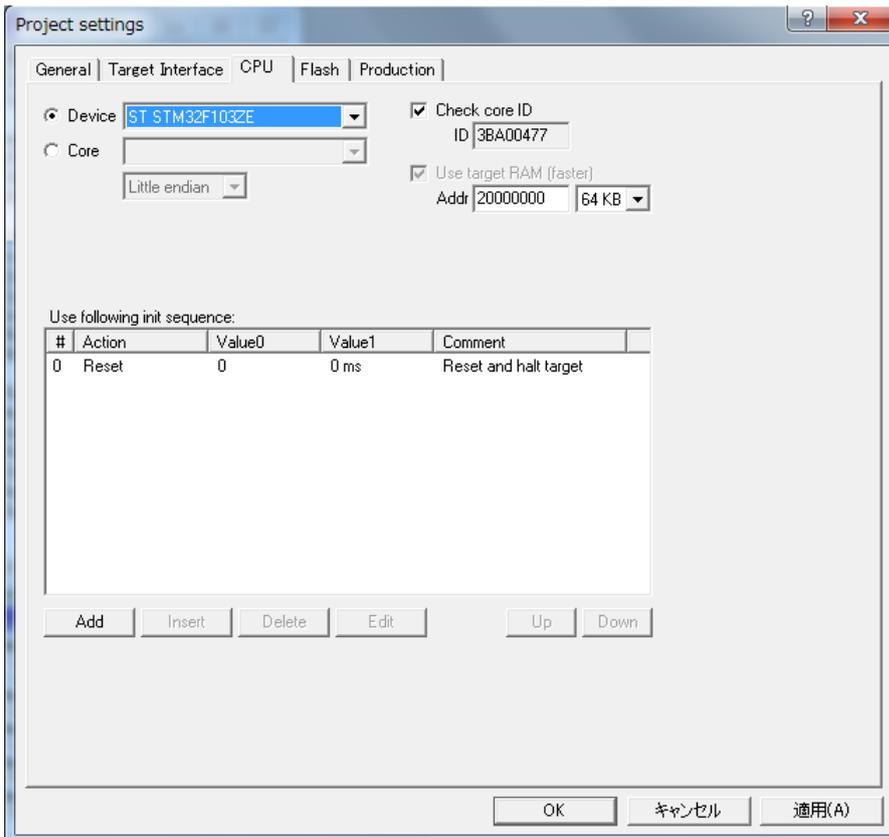
J-FLASH ARM を実行する。



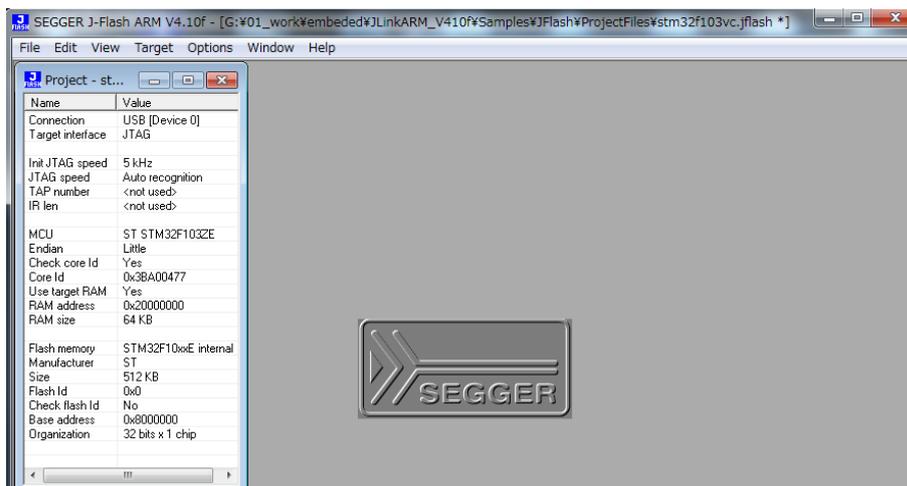
書き込む前に必要な設定 (Options->project settings...) :



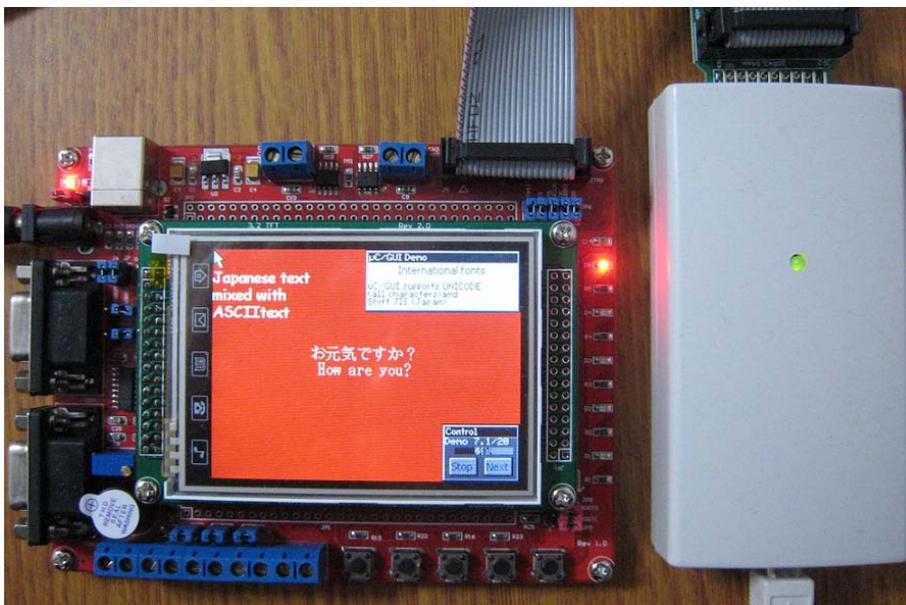
STM32 ボードの CPU 型番を選択する。



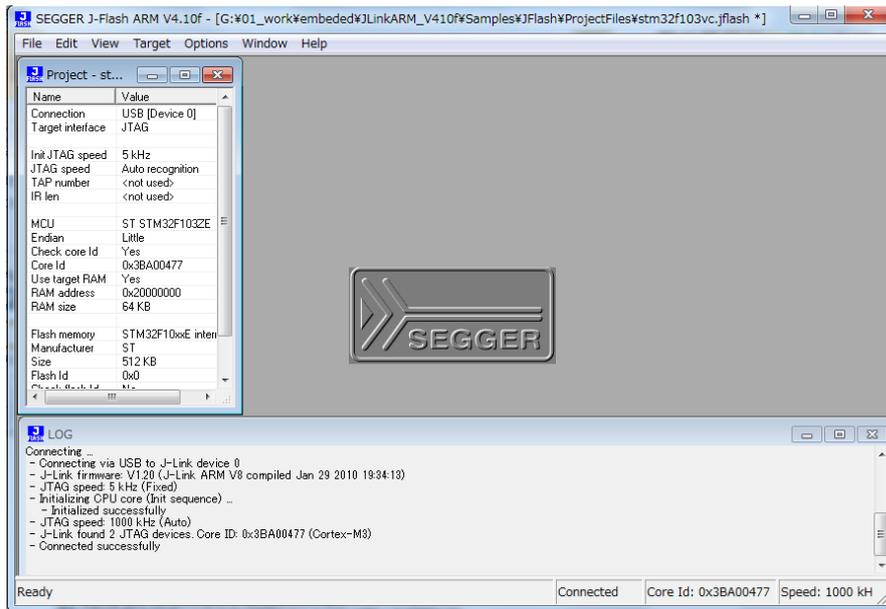
設定後、左側に書き込み情報が表示される。



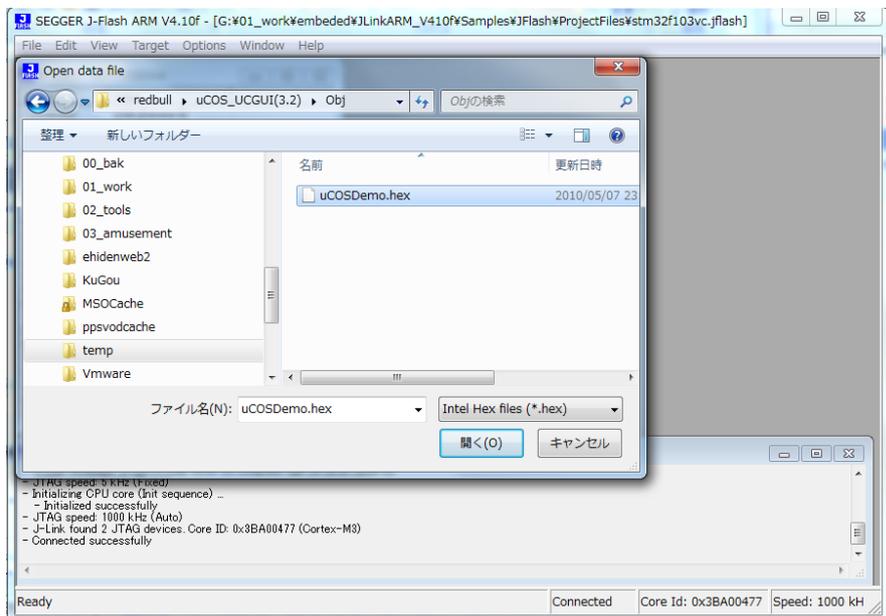
ボードを接続する。



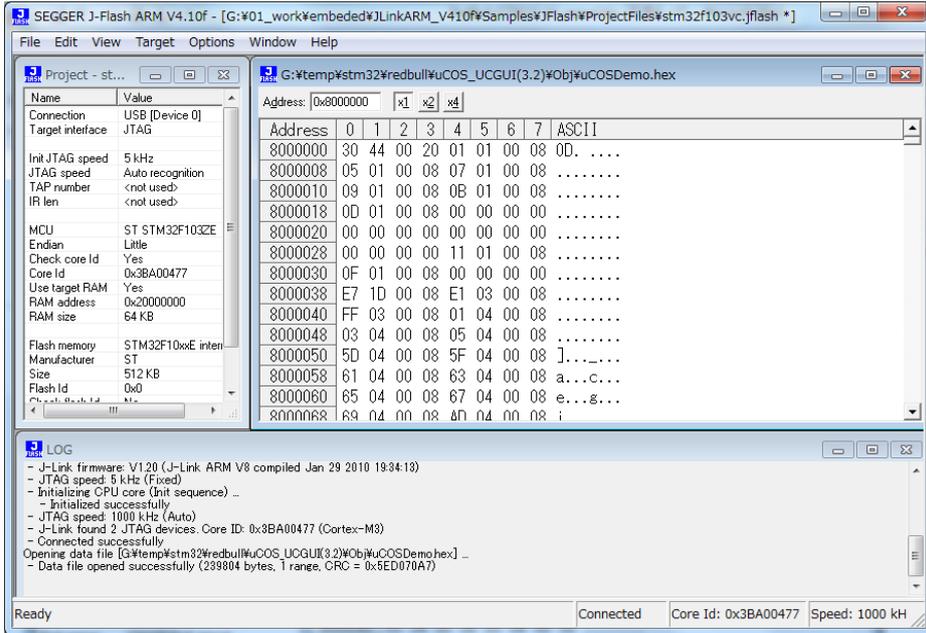
ソフト側も接続する (Target→Connect)。



File->open で実行ファイルを選択する。

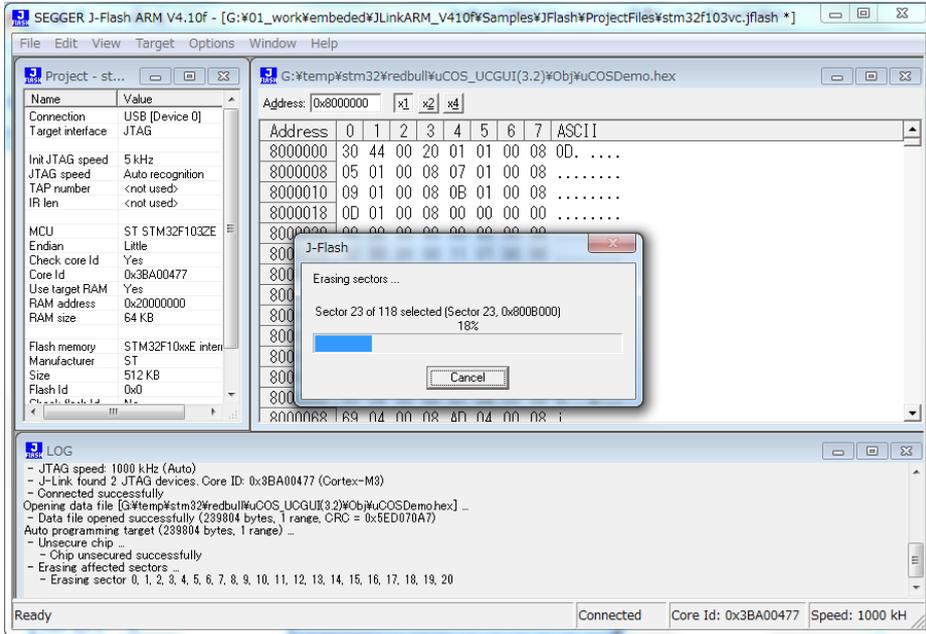


「開く (O)」をクリックする。

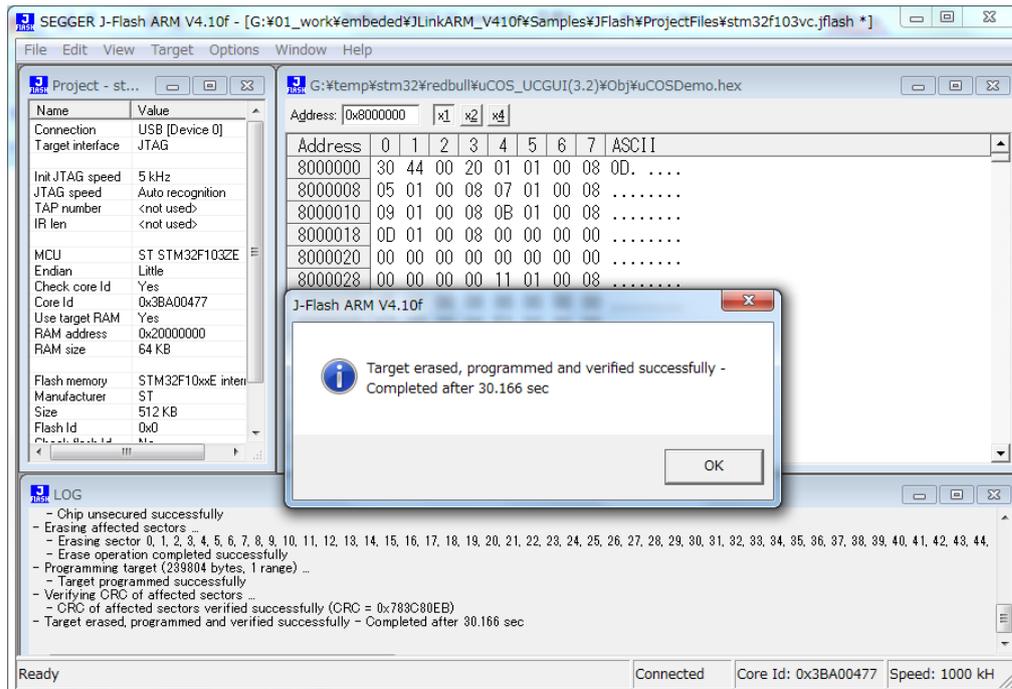


Target->Auto あるいは F7 で書き込み開始する。

書き込み中：



書き込み完了：



5.3 H-JTAG で実行ファイルを書き込む

H-JTAG は ARM の為の JTAG エミュレータです。AXD 又は keil をサポートします。デバッグのスピードも速いです。詳しい情報はこちらです。

<http://www.hjtag.com>

弊社は H-JTAG のハードウェアを提供しております（製品紹介 URL：<http://www.csun.co.jp/SHOP/200806151.html>）。パソコン側には LTP が必要です。

(1) H-JTAG をダウンロードしてインストールする。

ホームページ <http://www.hjtag.com> から最新版をダウンロードできます。

H-JTAG の特性：

- RDI 1.5.0 & 1.5.1 をサポートします；
- ARM7 & ARM9 (ARM9E-S と ARM9EJ-S を含む) ；
- thumb & arm 命令；
- little-endian & big-endian；
- semihosting；
- 実行環境 WINDOWS 9. X/NT/2000/XP；
- flash の書き込み

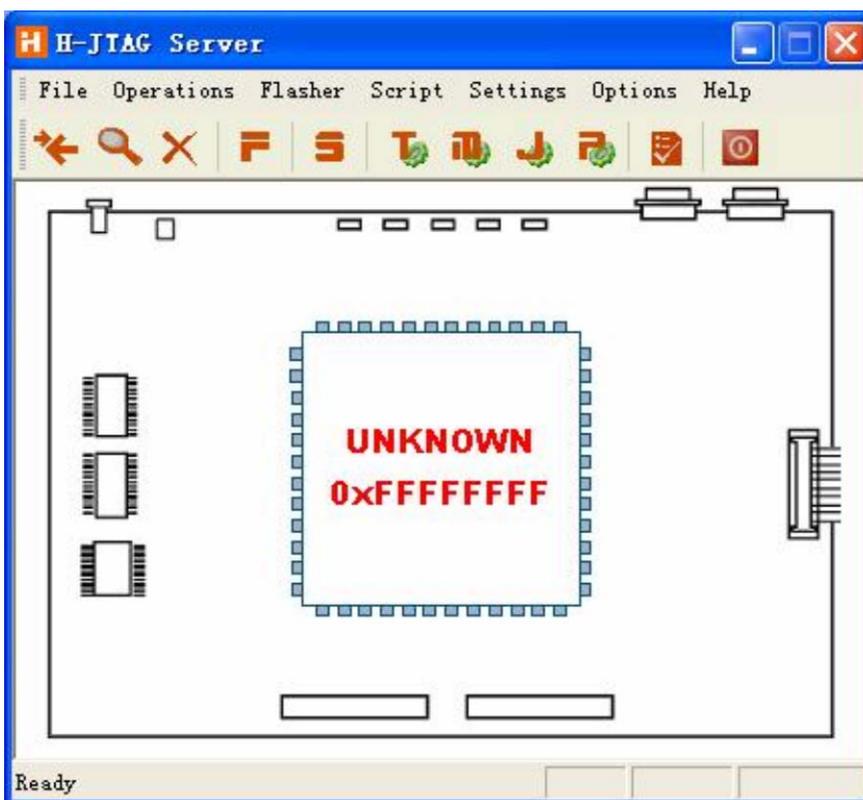
デフォルト設定のままインストール完了させて、デスクトップで H-JTAG と H-Flasher が生成される。

H-JTAG を実行する前に、まず、H-JTAGでSTM32ボードとパソコンを接続する。STM32 ボードに電源を入れてください。

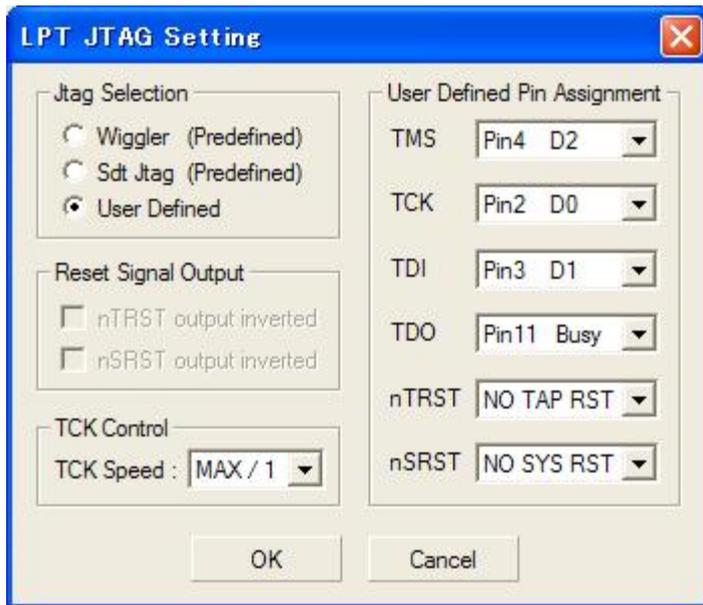
初めて H-JTAG を実行する時、次の画面のエラーメッセージが出て来る。



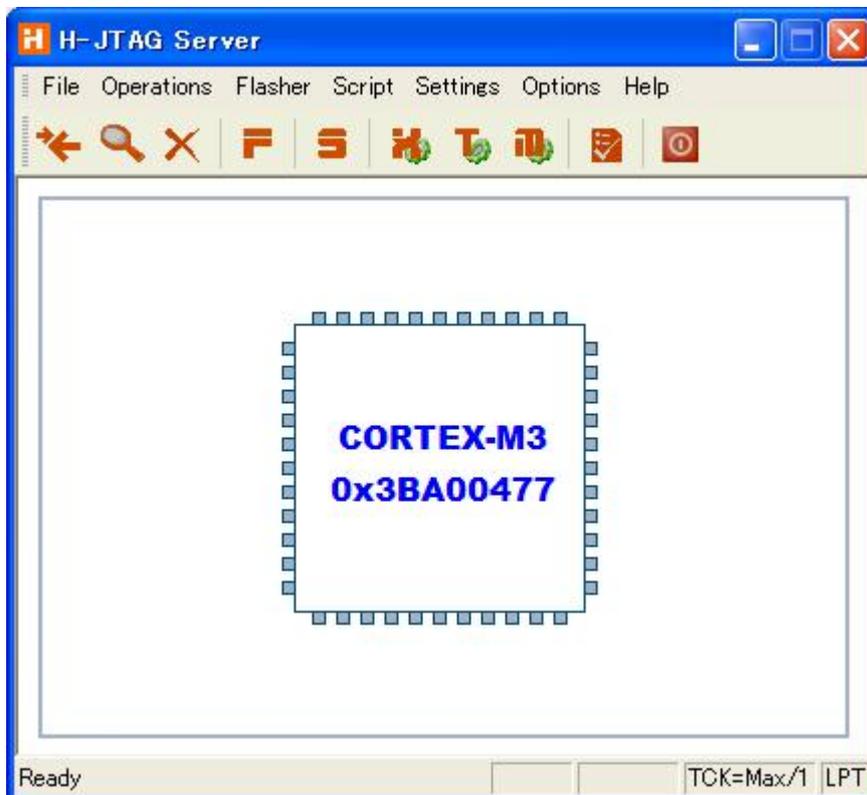
“確定”ボタンをクリックすると、初の画面が出て来る。



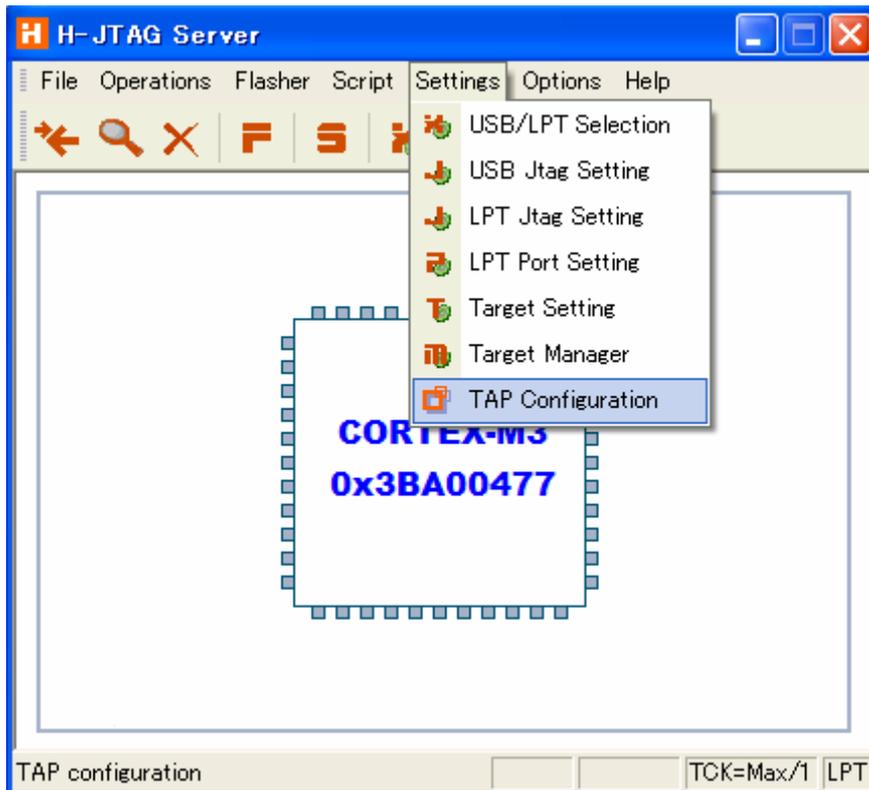
H-JTAG のメニュー : Setting → LPT Jtag Setting



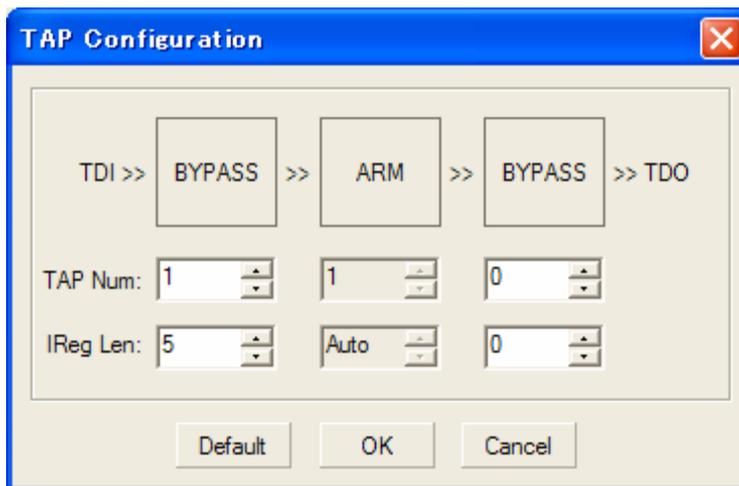
上記画面の様に設定して、“Ok”ボタンをクリックすると CORTEX-M3 が認識される。



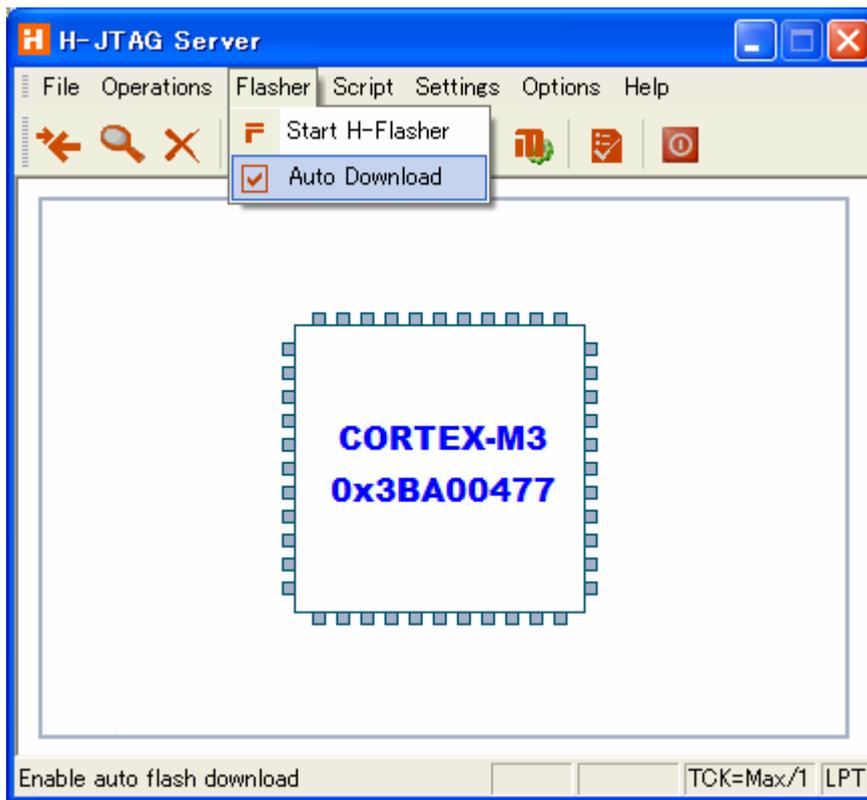
メニュー「Settings」→「TAP Configuration」を選択する。



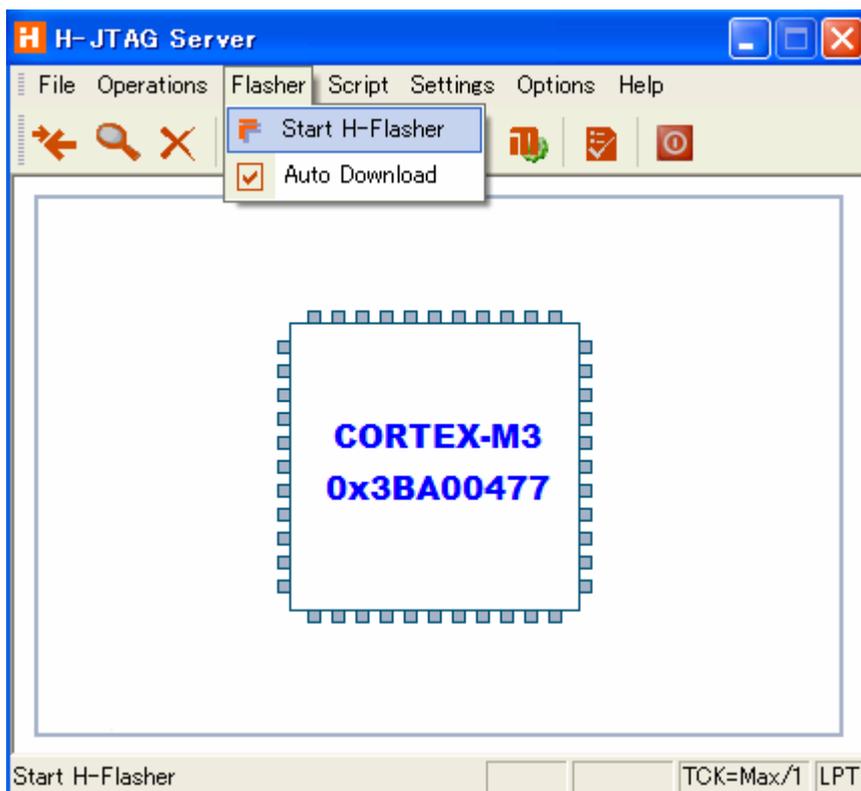
下記画面の通りに設定する。



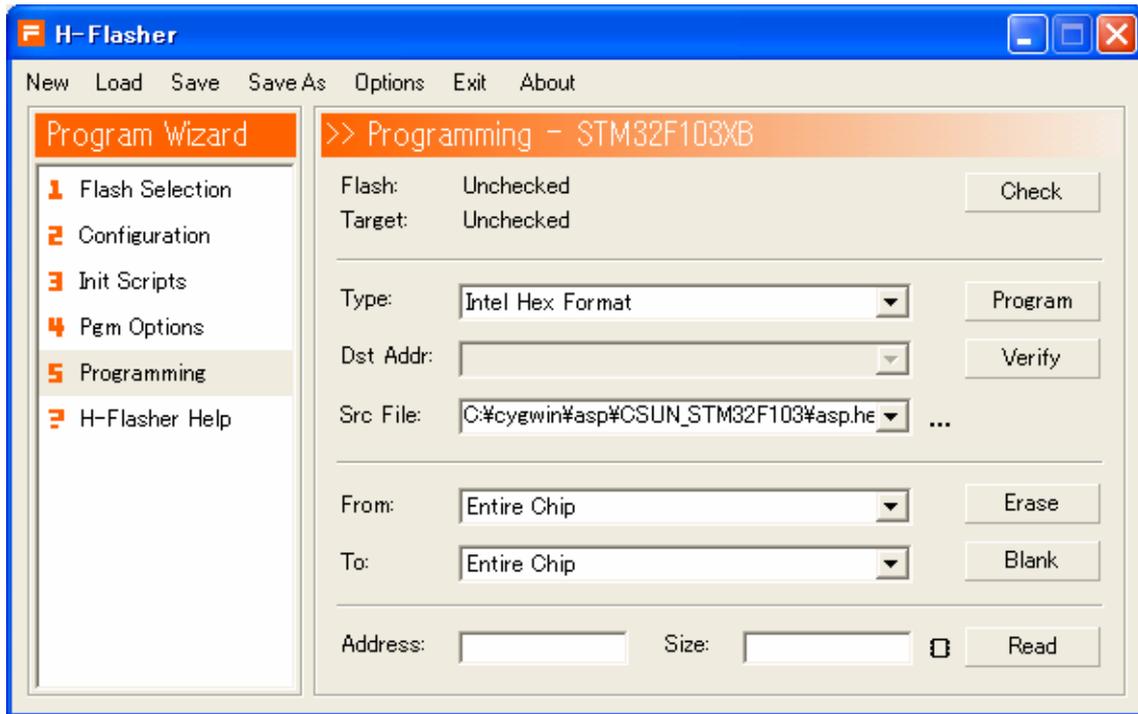
メニュー「Flasher」→「Auto Download」にチェックを入れる。



メニュー「Flasher」→「Start H-Flasher」を選択する。



STM32F103ZE を選択する。



ファイルのフォーマットを「Intel Hex Format」を設定して、実行ファイル*.hex を選択して、「Program」ボタンをクリックする。

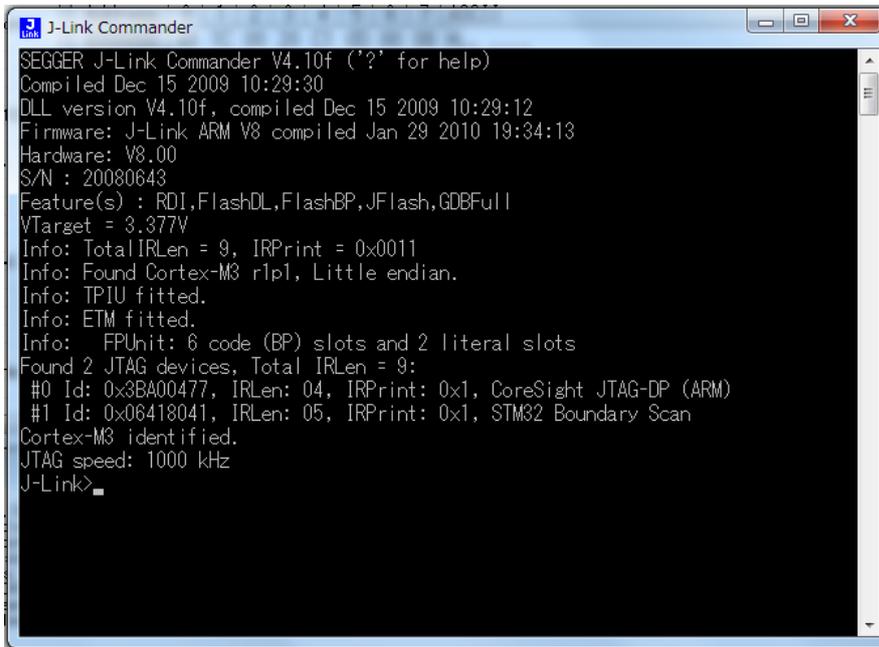
第六章 OpenLink でデバッグ

OpenLink エミュレータ使い環境のインストール手順は「5.2.1 ドライバのインストール」をご参照ください。

6.1 J-Link command でデバッグ

コマンドラインでコマンドを入力して実行する。

J-Link command を起動すると、JLINK のバージョン情報が表示される。ターゲットを接続している場合は、ターゲットの状態と CPU などの情報が表示される。



```
SEGGER J-Link Commander V4.10f ('?' for help)
Compiled Dec 15 2009 10:29:30
DLL version V4.10f, compiled Dec 15 2009 10:29:12
Firmware: J-Link ARM V8 compiled Jan 29 2010 19:34:13
Hardware: V8.00
S/N : 20080643
Feature(s) : RDI,FlashDL,FlashBP,JFlash,GDBFull
VTarget = 3.377V
Info: TotalIRLen = 9, IRPrint = 0x0011
Info: Found Cortex-M3 r1p1, Little endian.
Info: TPIU fitted.
Info: ETM fitted.
Info: FPUunit: 6 code (BP) slots and 2 literal slots
Found 2 JTAG devices, Total IRLen = 9:
#0 Id: 0x3BA00477, IRLen: 04, IRPrint: 0x1, CoreSight JTAG-DP (ARM)
#1 Id: 0x06418041, IRLen: 05, IRPrint: 0x1, STM32 Boundary Scan
Cortex-M3 identified.
JTAG speed: 1000 kHz
J-Link>
```

J-Link command では豊富なデバッグ、検索などのコマンドを持っている。詳しい内容は J-Link command で ? を入力してエンタリすると説明が表示される。

第七章 開発ツール KEIL の応用

MDK315B.exe は開発ツール KEIL の無償評価版です。

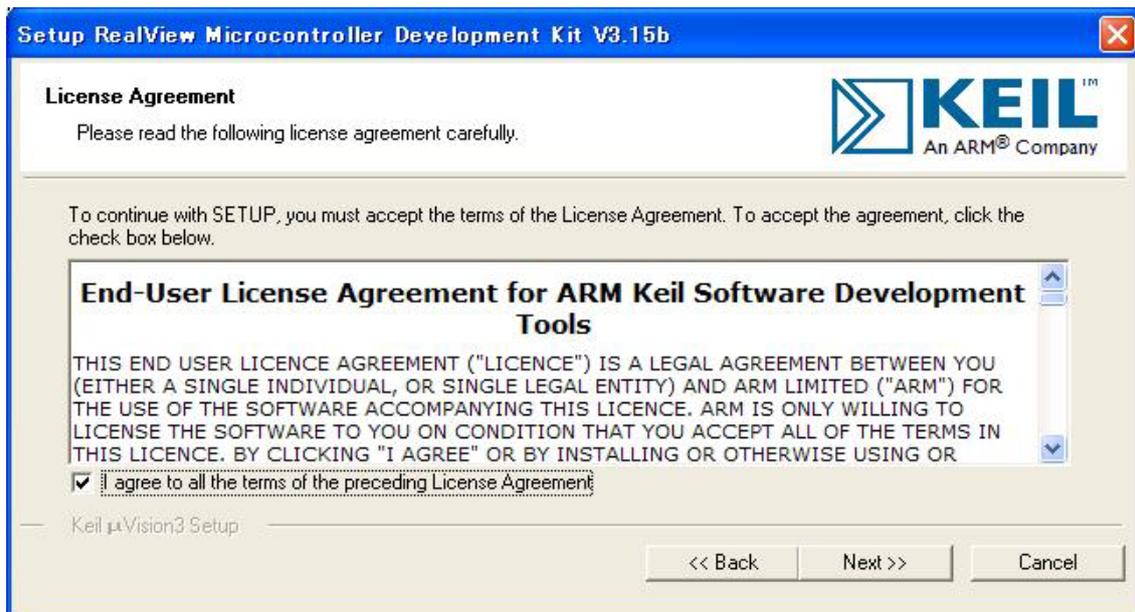
Keil 社の HP (<http://www.keil.com/>) から最新版がダウンロード出来ます。

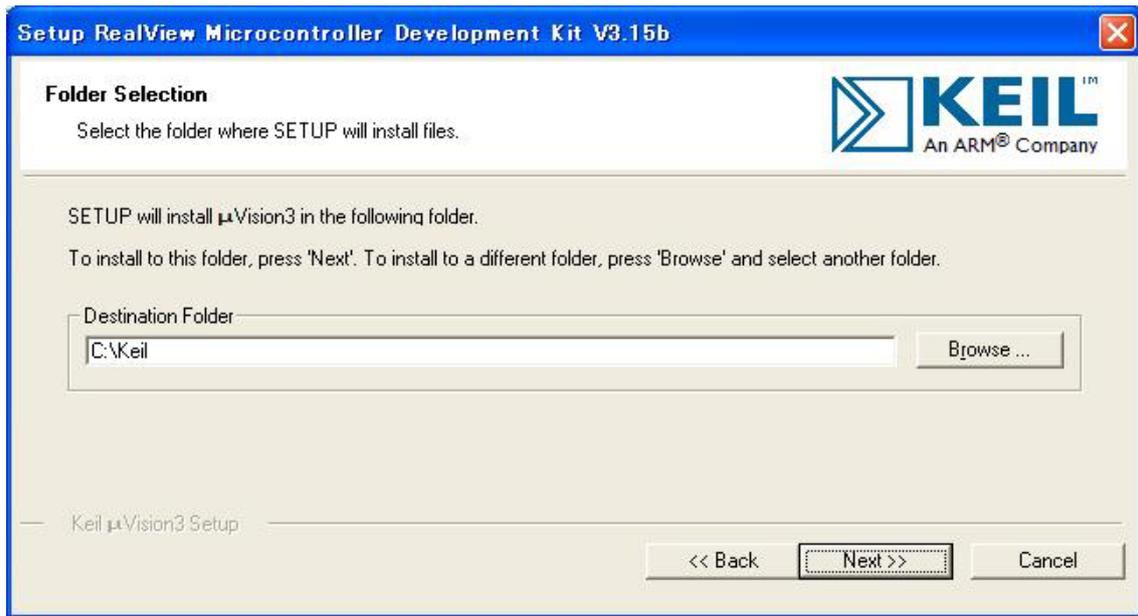
7.1 KEIL のインストール

MDK315B.exe を実行して、KEIL3.15 をインストールする。



「Next」ボタンを押すと、英文のライセンス契約画面が表示される。同意できる場合は、「I accept the terms of the license agreement」を選択して、「Next」ボタンを押す。

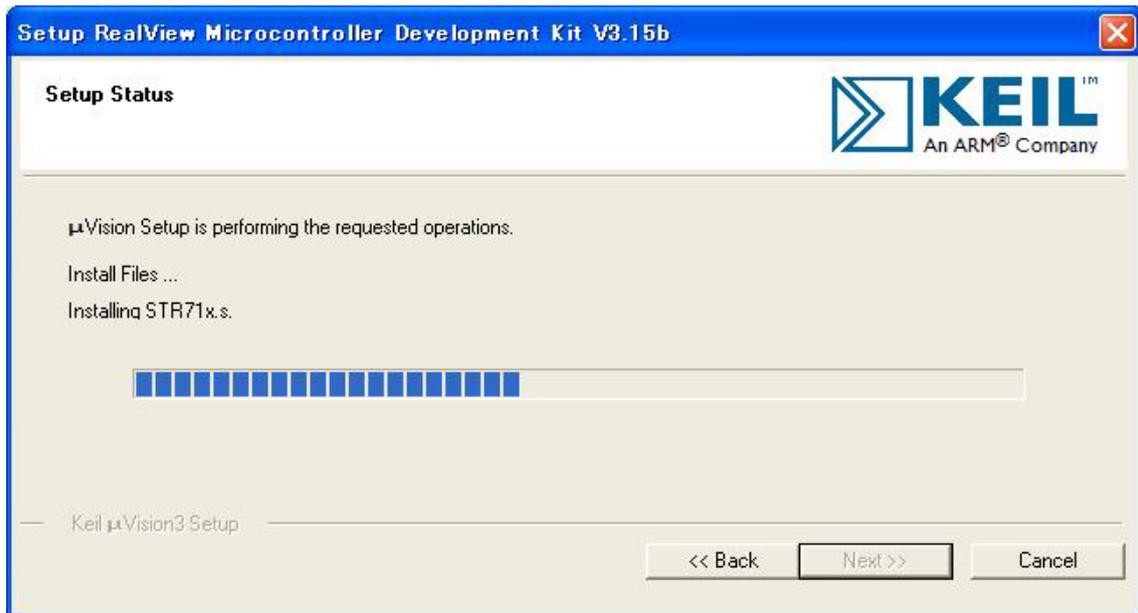




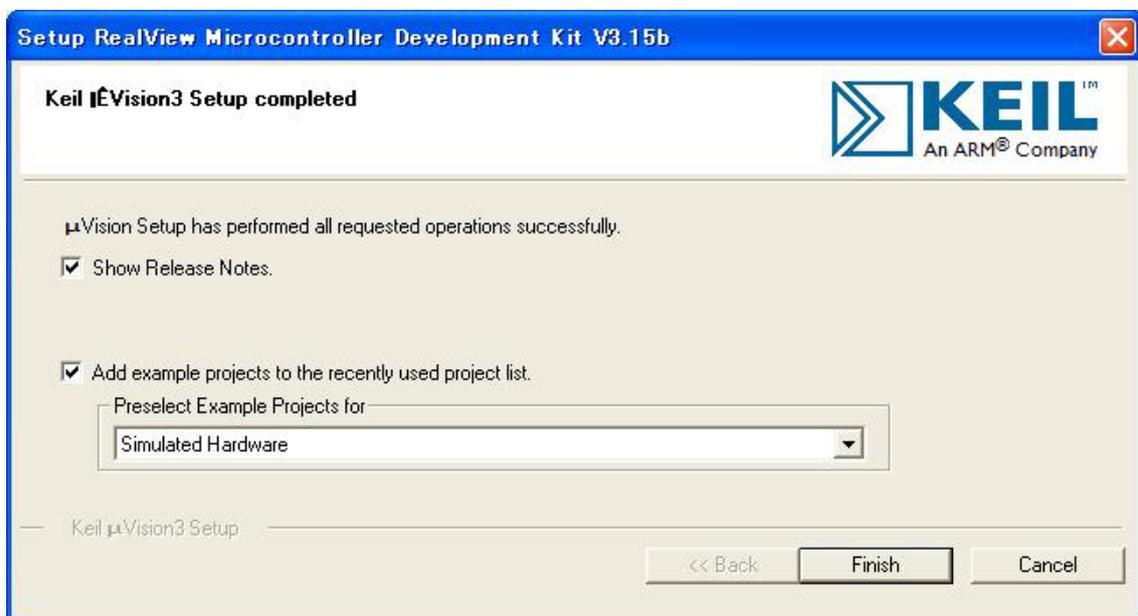
インストール先フォルダを変更せず、そのまま進んでください。



使用者の名前と所属会社名を入力するダイアログが表示される。名前は半角のアルファベットで入力してください。



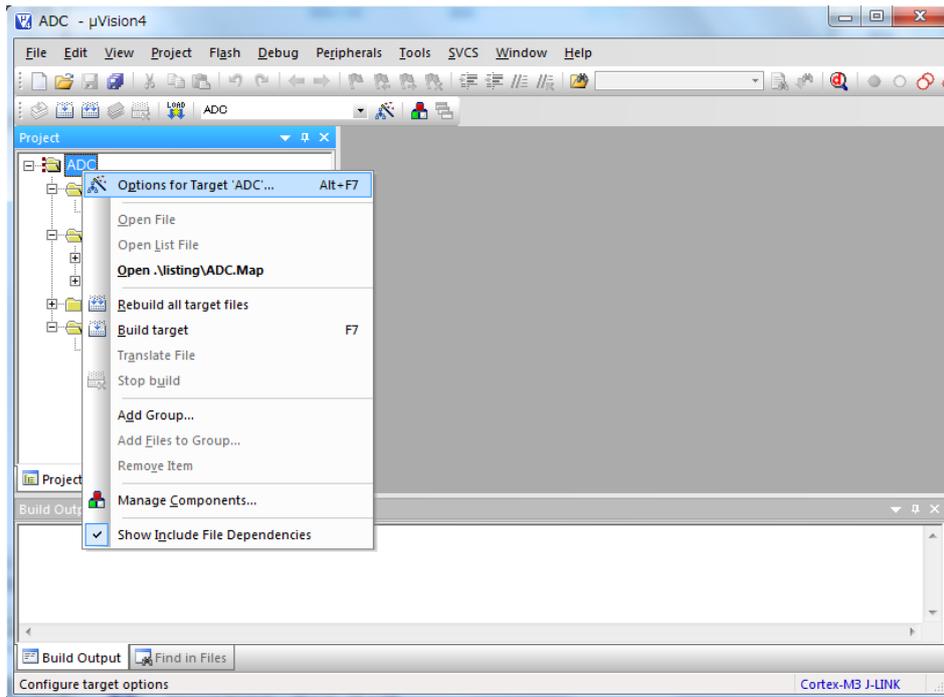
インストール中の画面です。



最後に「Finish」をクリックすると、ウィザードが閉じられてインストール終了。
デモ版ではライセンスがないので、プログラムのサイズ制限があります。ライセンスを取得するにはKeil社の日本代理店と連絡する事。

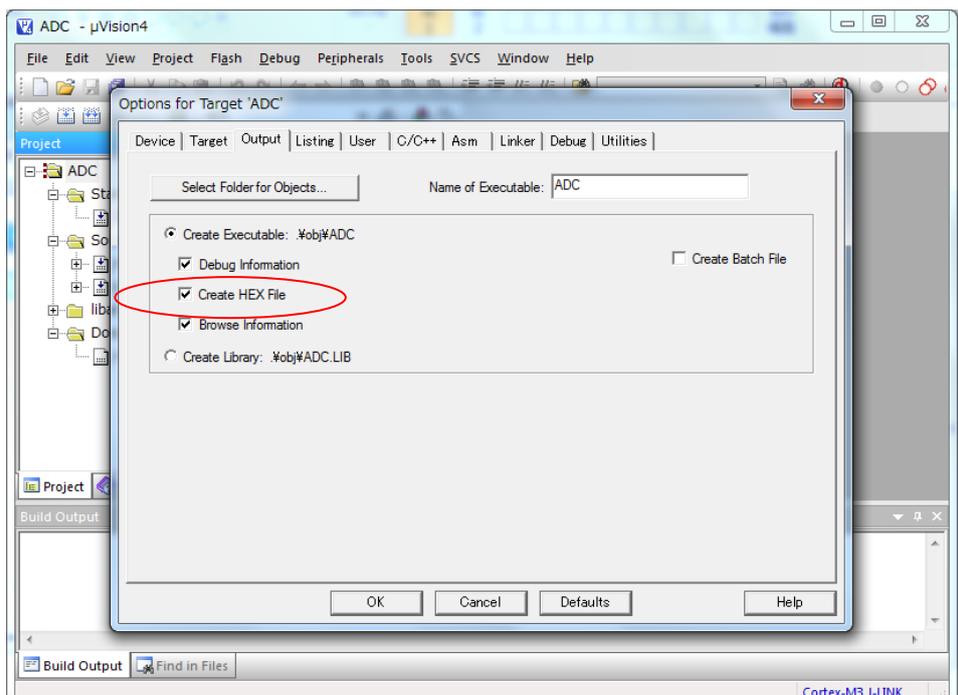
7.2 既存のプロジェクトから

プロジェクトファイルExample/ADC_test/ADC_Uv2をダブルクリックする。或いはKEILのメニューでProject→Open Project…でADC_Uv2を選択する。

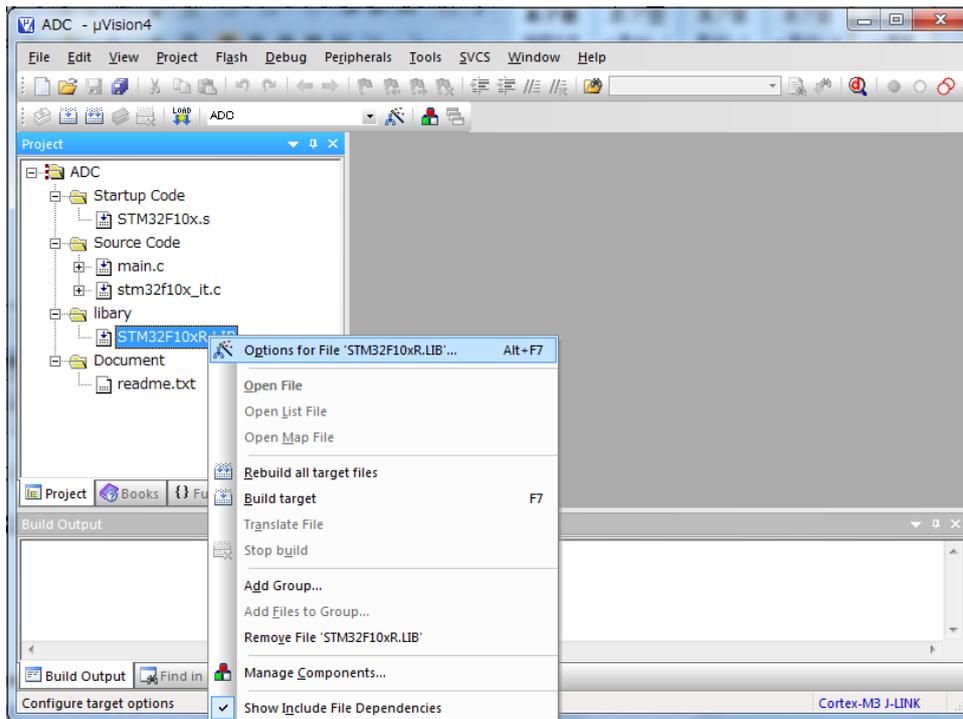


プロジェクト ADC を右クリックして「Options for Target ‘Target 1’ …」をクリックする。

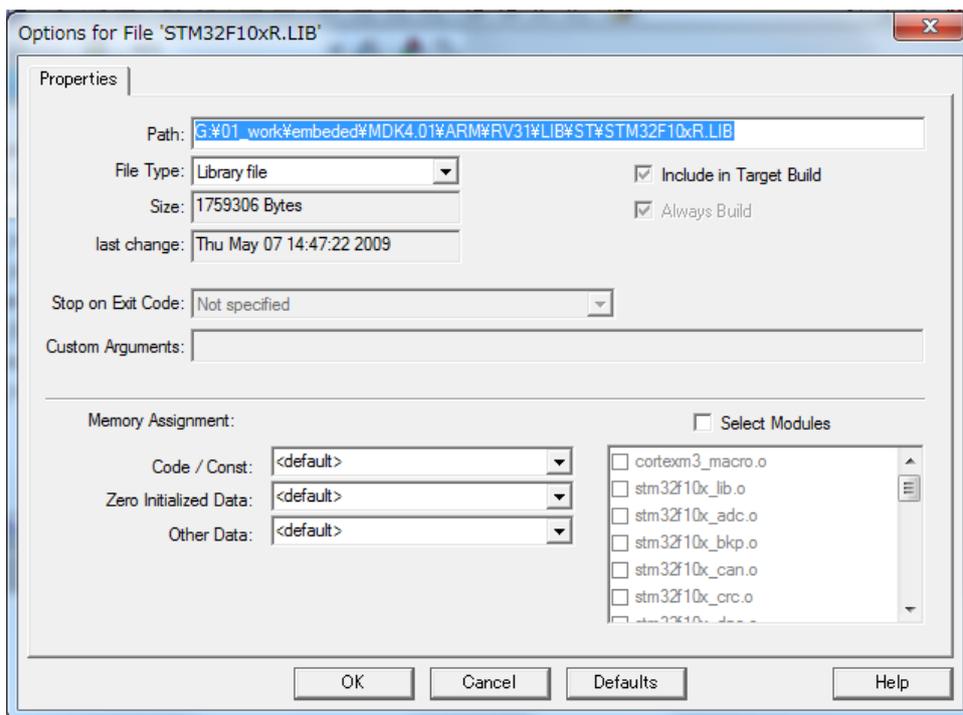
Options for Target ‘Target 1’ の画面が出て来る。「output」タブを選択する。

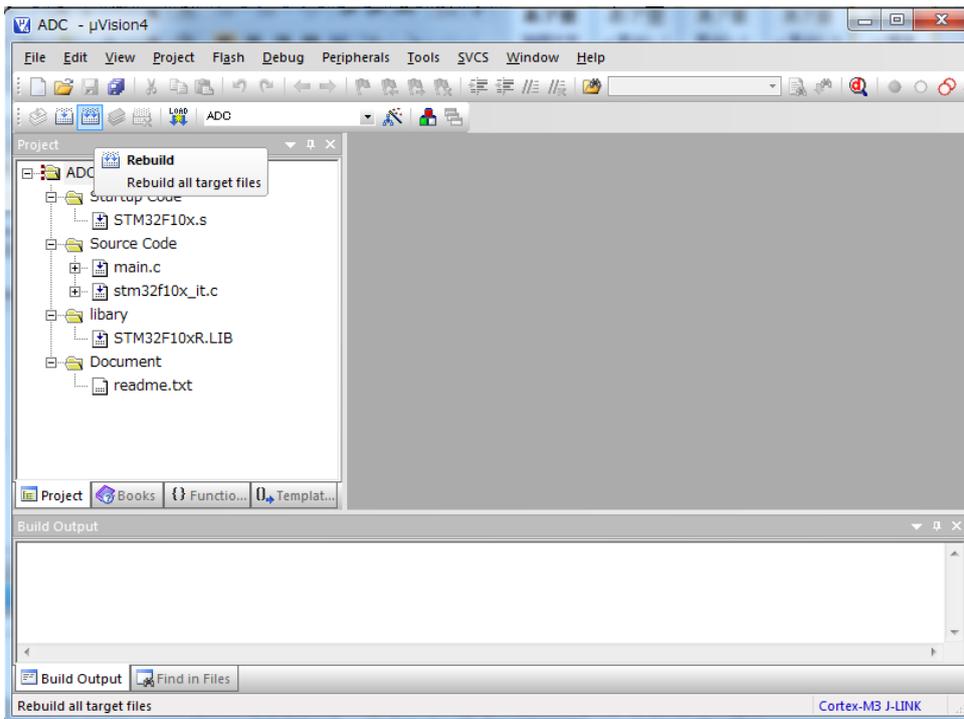


Create HEX Fileの所のをチェックを入れて「OK」ボタンをクリックする。
STM32F10xR.LIBが見つけない場合は、右クリックして「Options for File ‘STM32F10xR.LIB’ …」をクリックする。

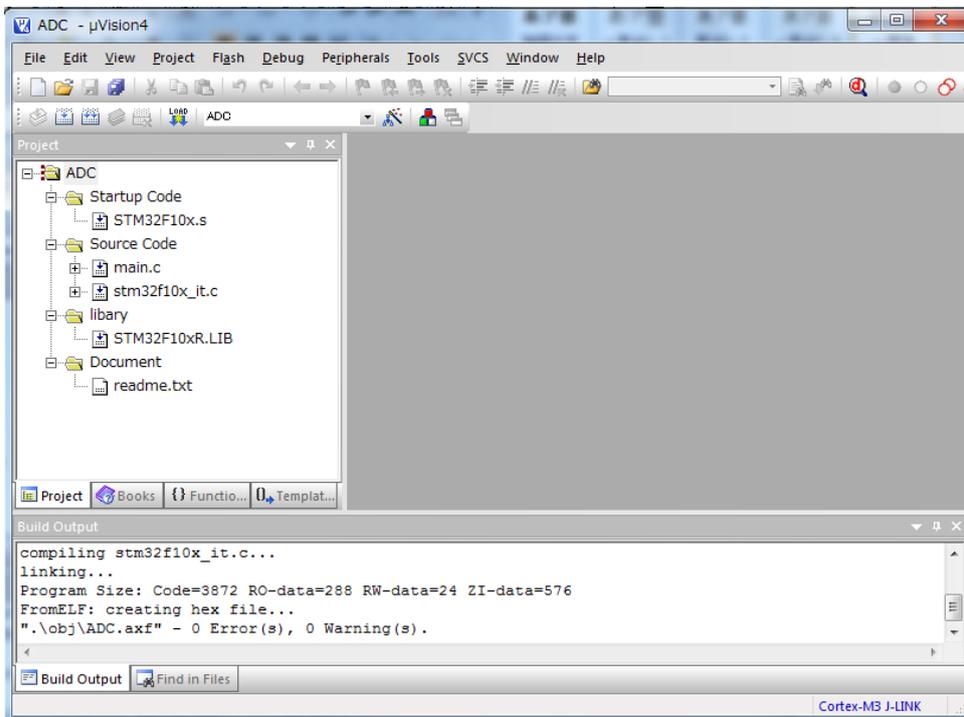


Pathの所に“Keilのインストールフォルダ¥ARM¥RV31¥LIB¥ST”を入力してOKを押す。





ツールバーの「Rebuild all target files」を押すと、ビルドが開始する。



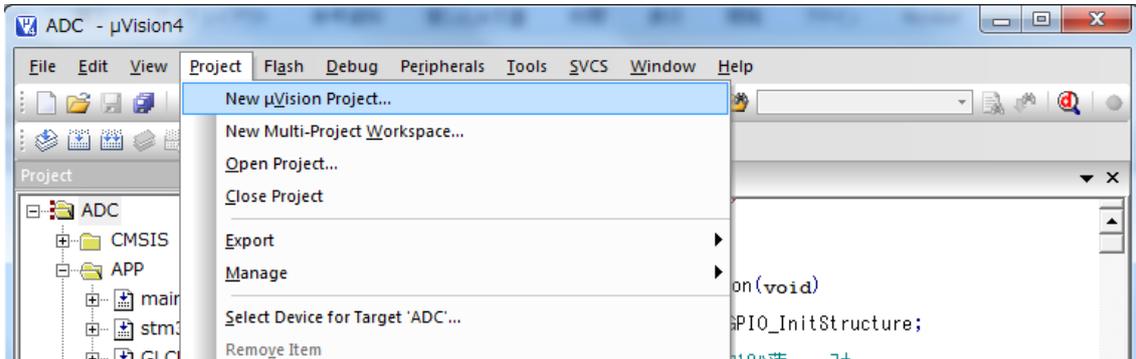
ビルドが成功したら、プロジェクトのoutputフォルダにADC.hex ファイルを生成される。
このHEXファイルをSTM32F103 ボードに書き込む。

実行する前にPC側のハイパーターミナル（115200(B)、8(D)、なし(P)、1(S)、なし(F))を
起動する。

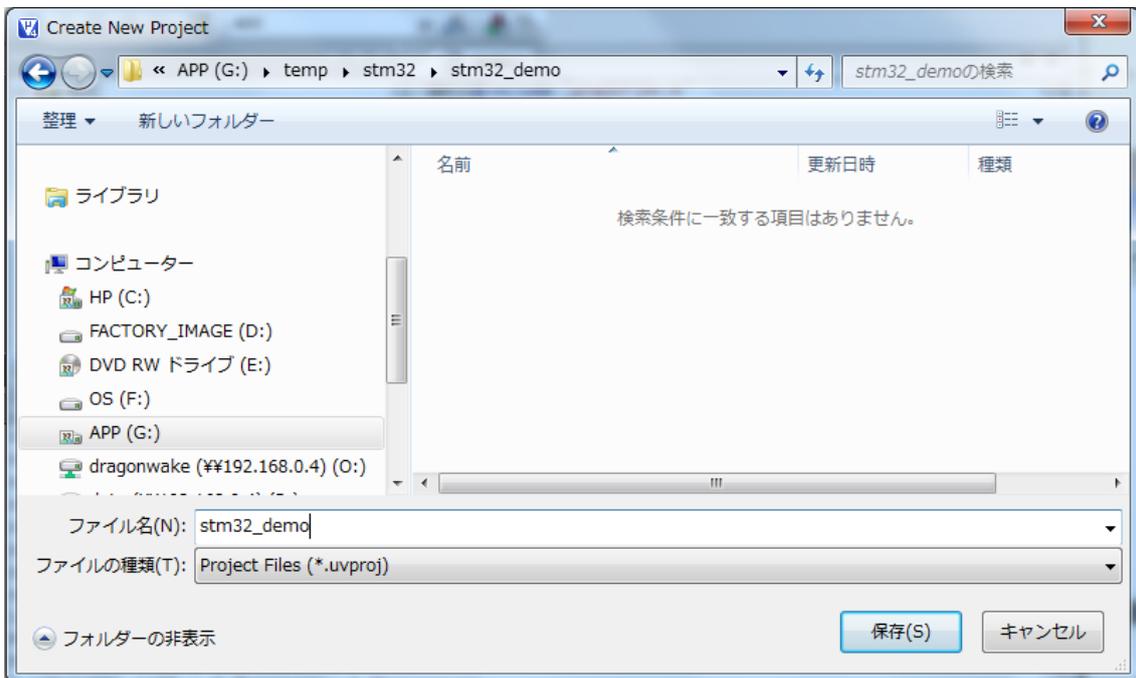
※VR1のボリュームを調整するとハイパーターミナルの画面で数値が変化する。

7.3 新しいプロジェクトの作成

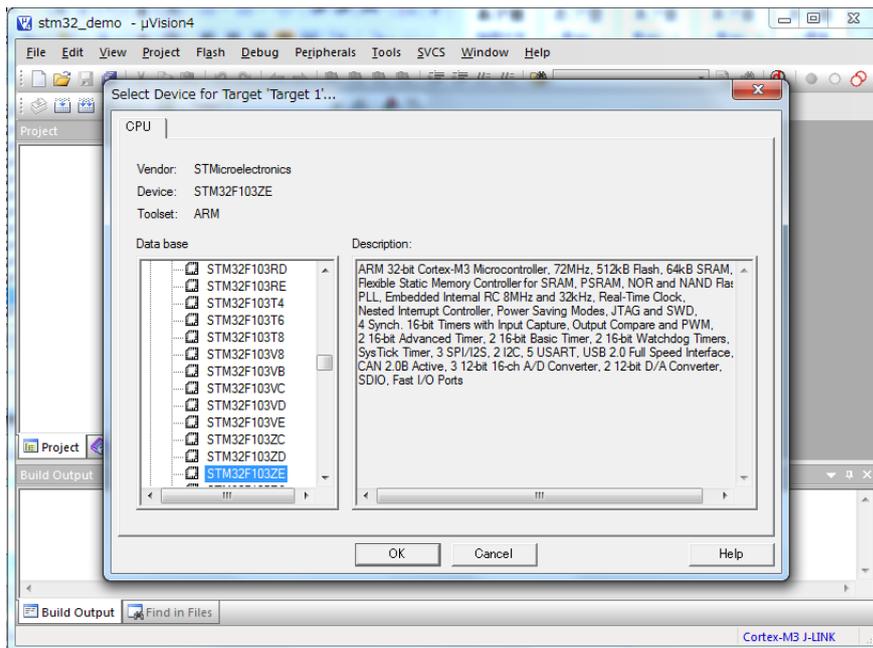
Keil のメニュー「Project」→「New uVision Project…」を選択する。



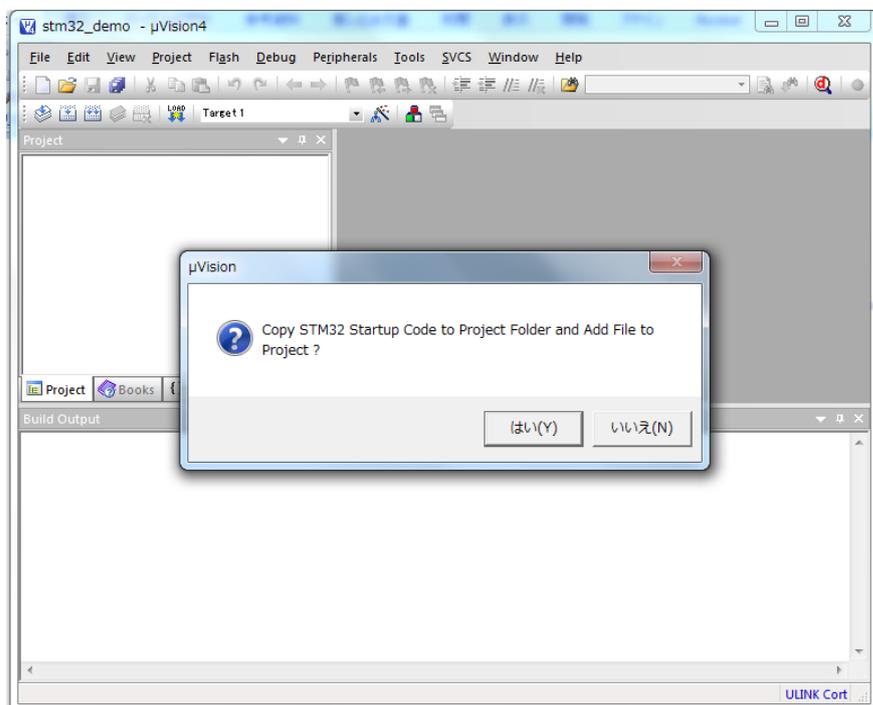
プロジェクト名を入力して、保存する。



CPU 選択画面が出て来る。選択肢 STMicroelectronics を開いて STM32F103ZE を選択する。

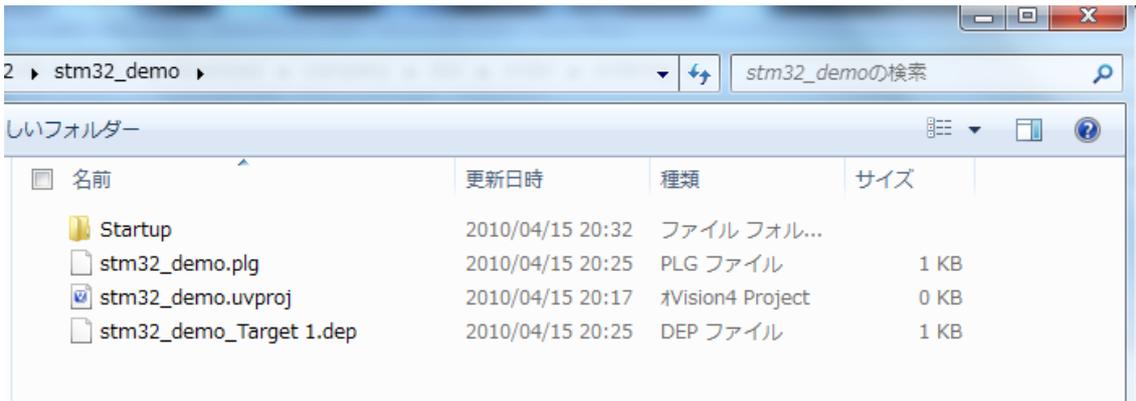


「OK」ボタンをクリックすると下記画面が表示される。

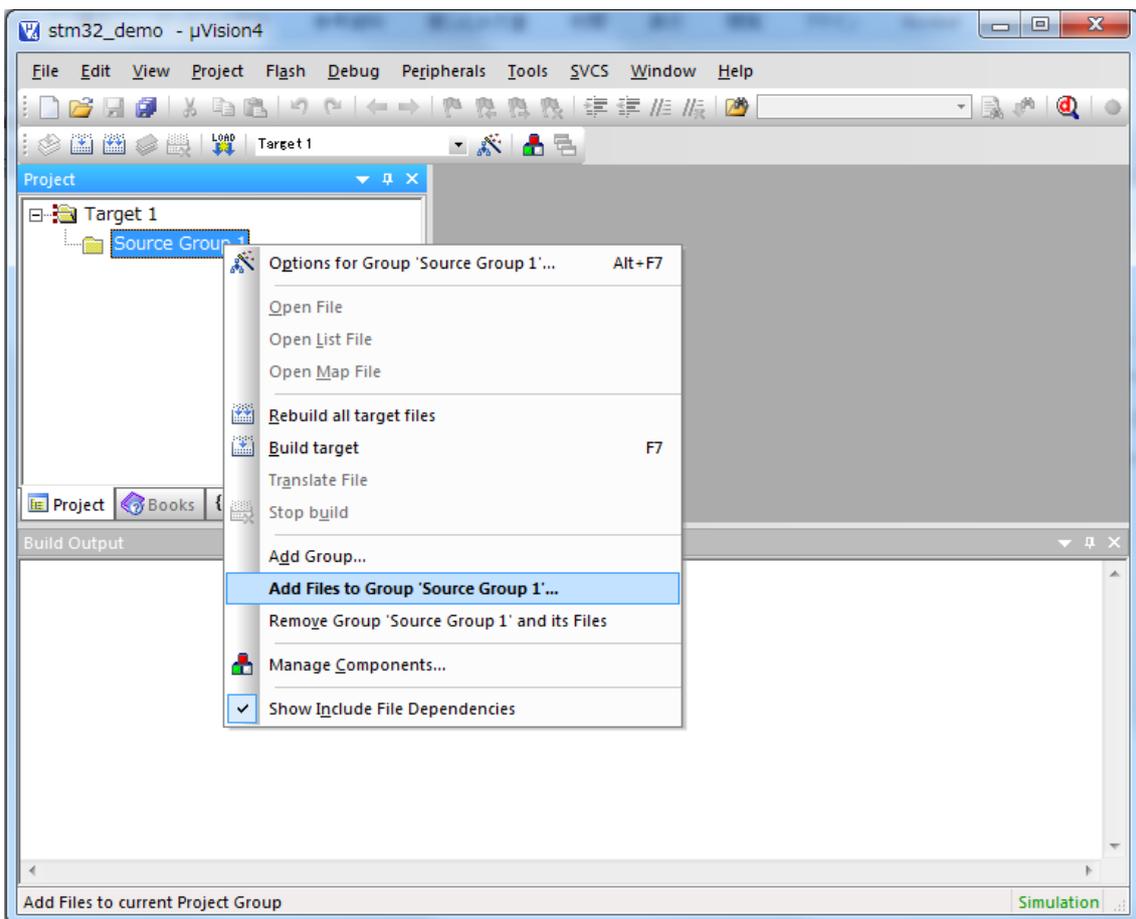


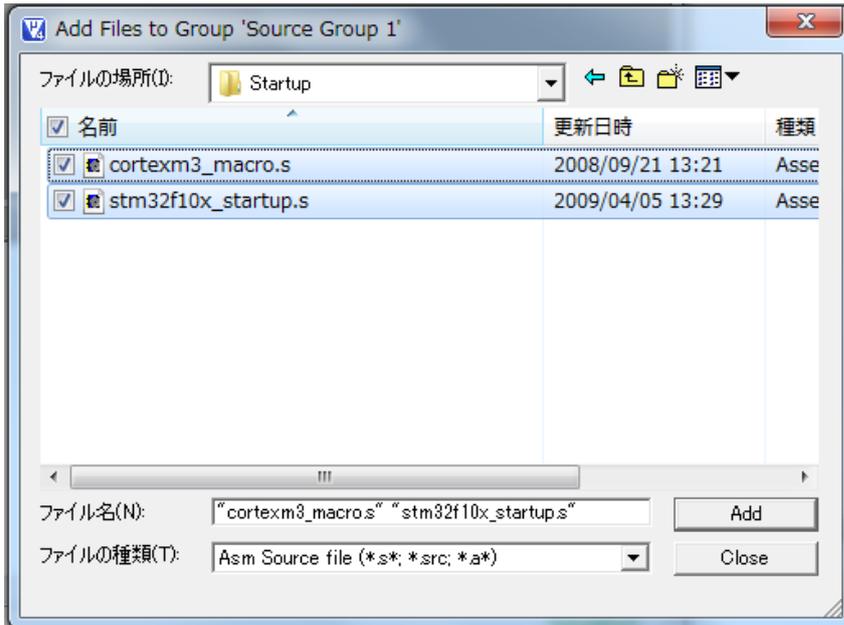
「いいえ」ボタンを押してください。

弊社 HP で提供している tools.rar にある Startup フォルダをプロジェクトにコピーする。

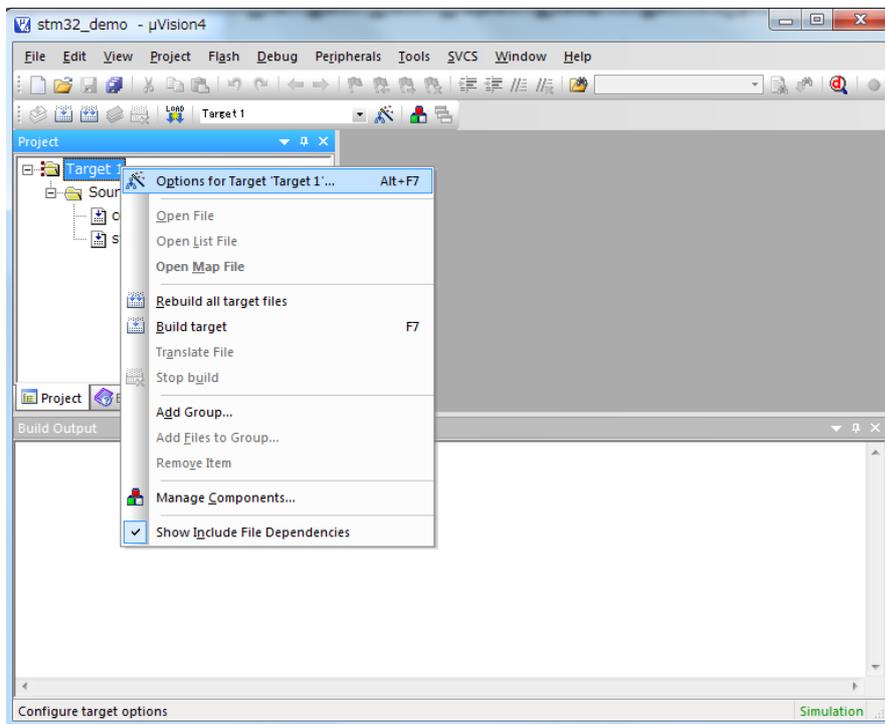


プロジェクトの「Source Group 1」でマウスを右クリックしてメニューから「Add Files To Group 'Source Group 1' ...」をクリックしてファイルを添加する。

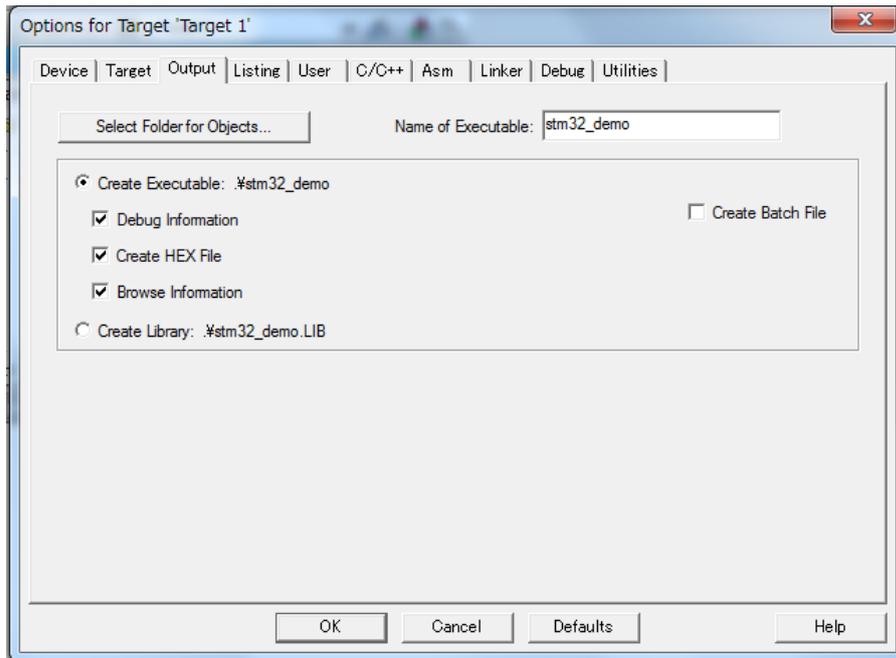




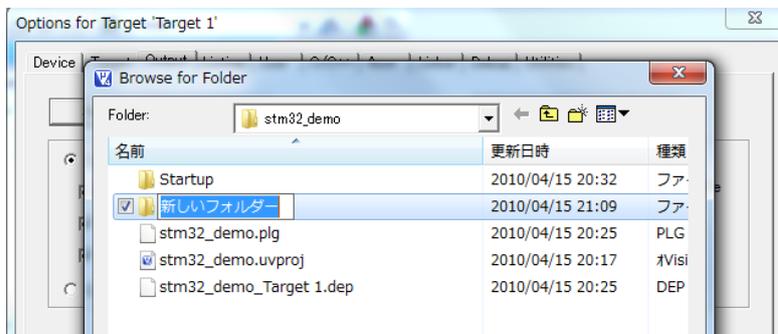
スタートアップファイルを添加される。
プロジェクトのオプションを設定する。



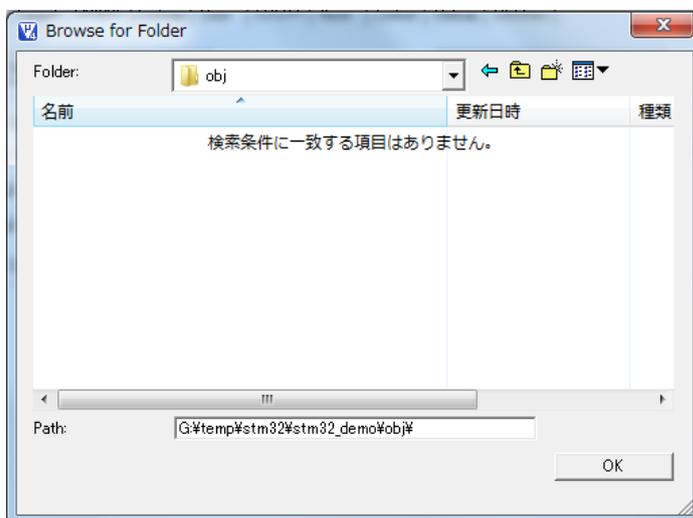
オプション設定画面で「output」タブを選択して、Hex ファイルを作成する選択肢にチェックを入れる。



上記画面で「Select Folder For Objects」ボタンを押して、出力フォルダを指定する。

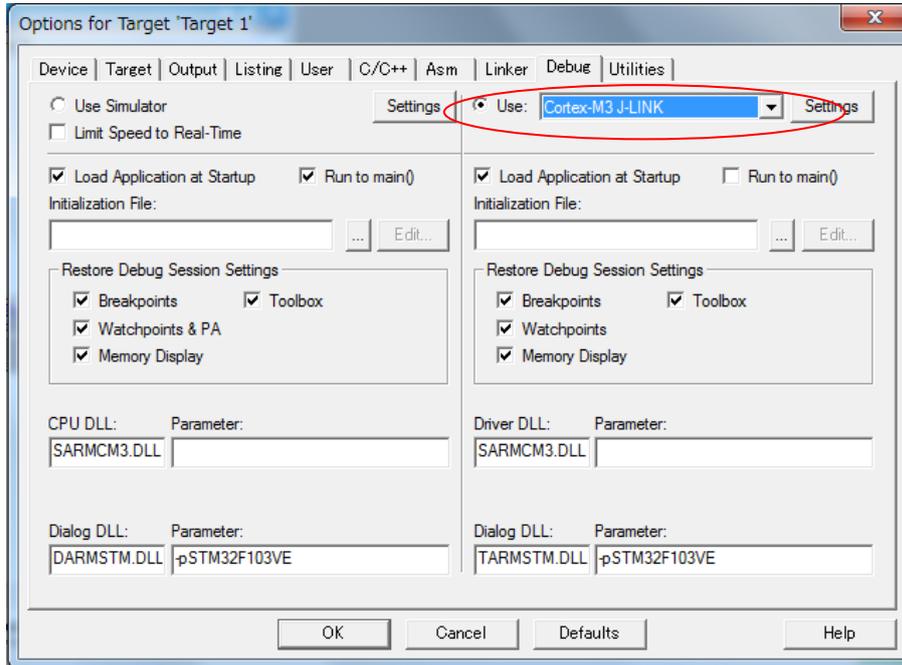


「obj」フォルダを作成して指定する。

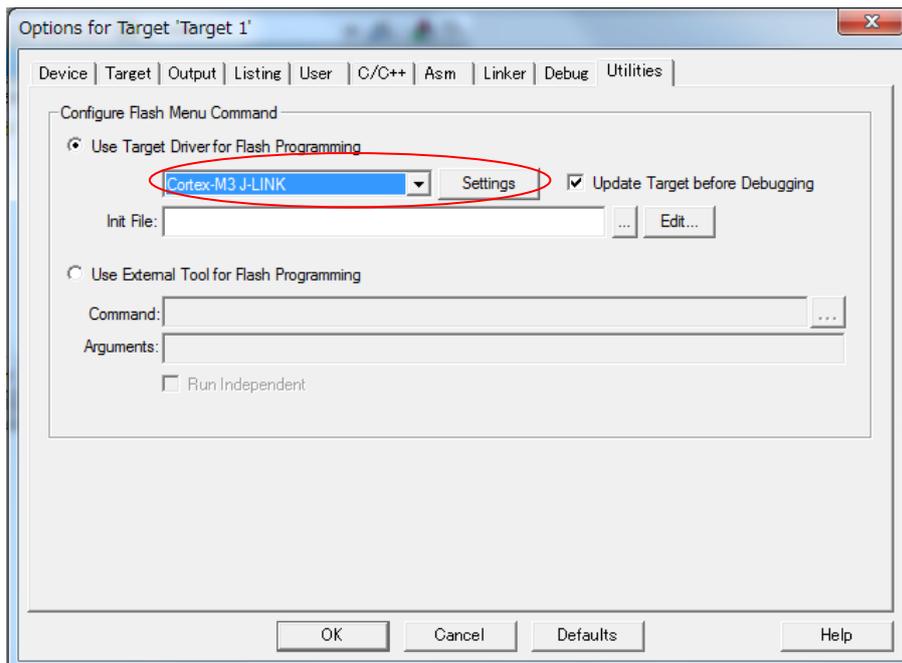


「OK」押してオプション設定画面に戻る。

「Listing」タブを選択して、上記と同じ手順で list フォルダを作成する。
次は「Debug」タブを選択して、利用している JTAG を選択する。シミュレータでデバッグする場合はデフォルトの Use Simulator のままで良い。



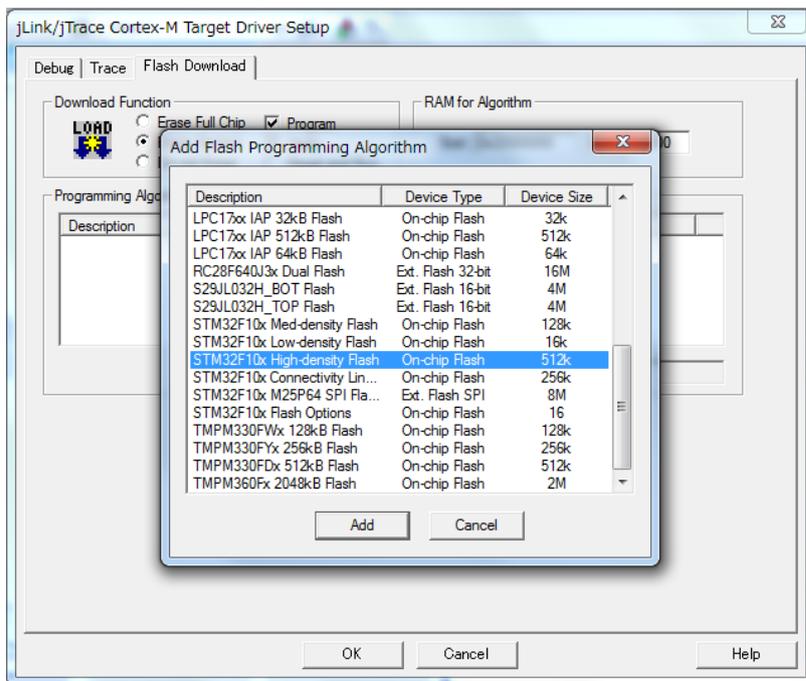
最後は「Utilities」タブを選択して、「Use Target Diver for Flash Programming」を選択する。ここは Debug タブで選択した JTAG と合わせて設定する。



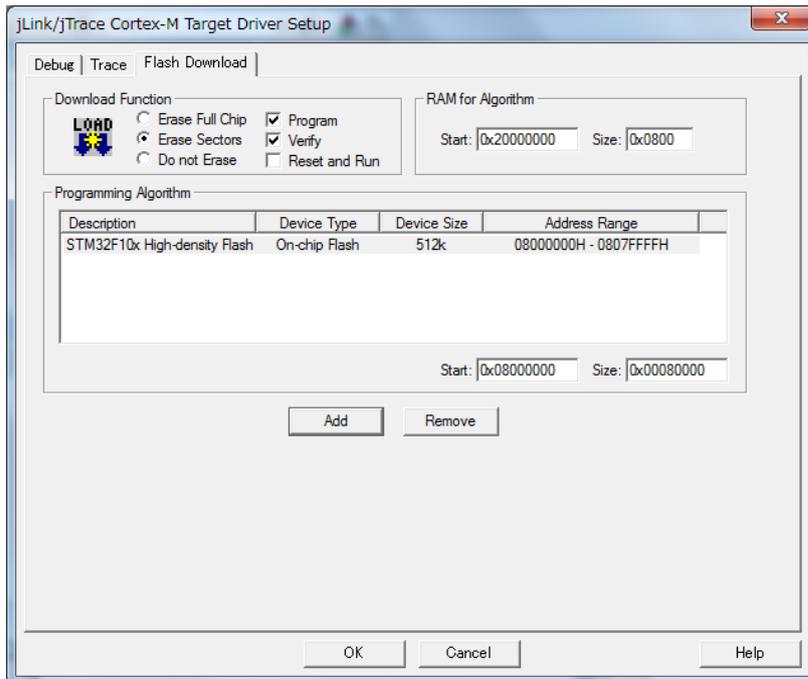
「Setting」ボタンを押すと、次の画面が表示される。



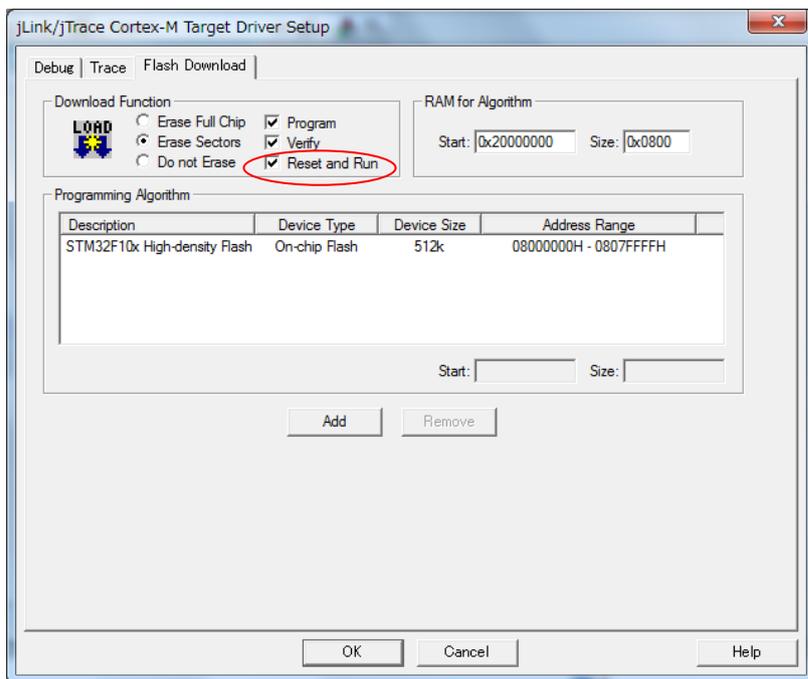
「Add」ボタンを押して、プログラムの書き込みアルゴリズムを設定する。



「Add」ボタン押すと、次の画面になる。

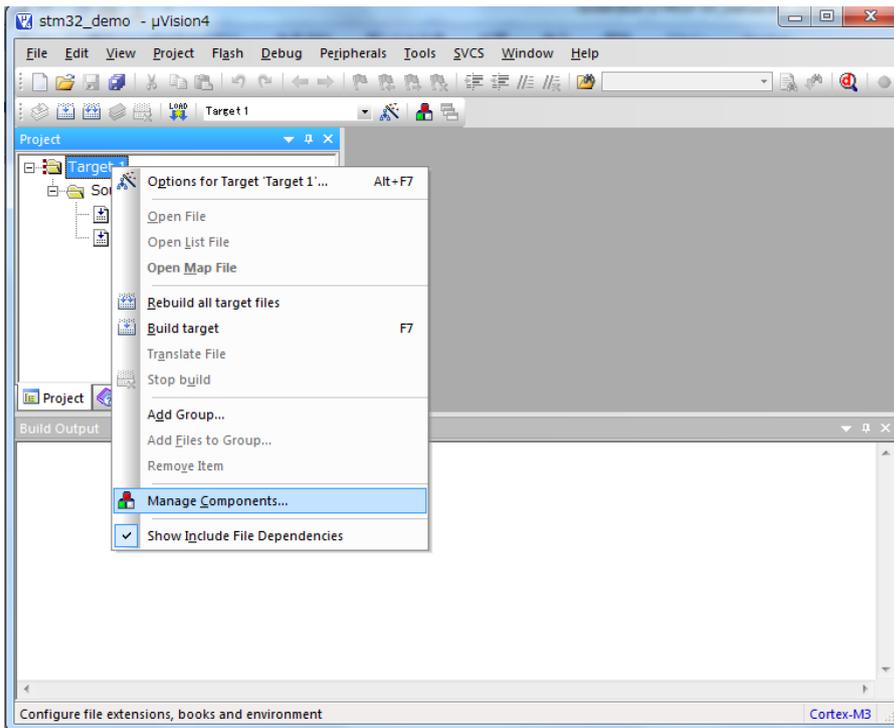


「Reset and Run」の所にチェックを入れて「OK」ボタンを押す。

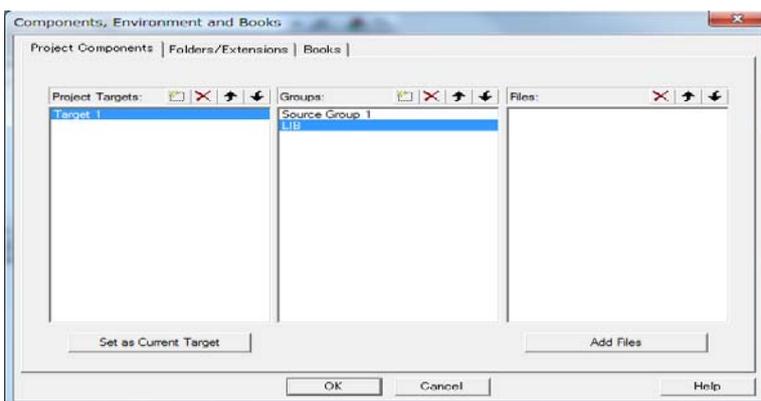
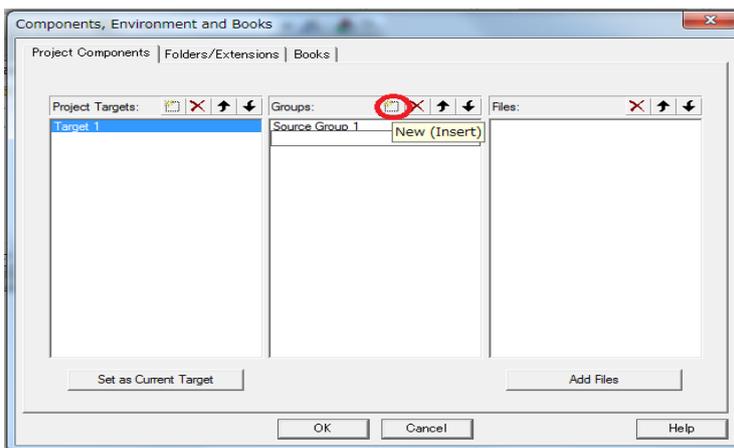


オプション設定画面に戻して「OK」ボタンを押す。

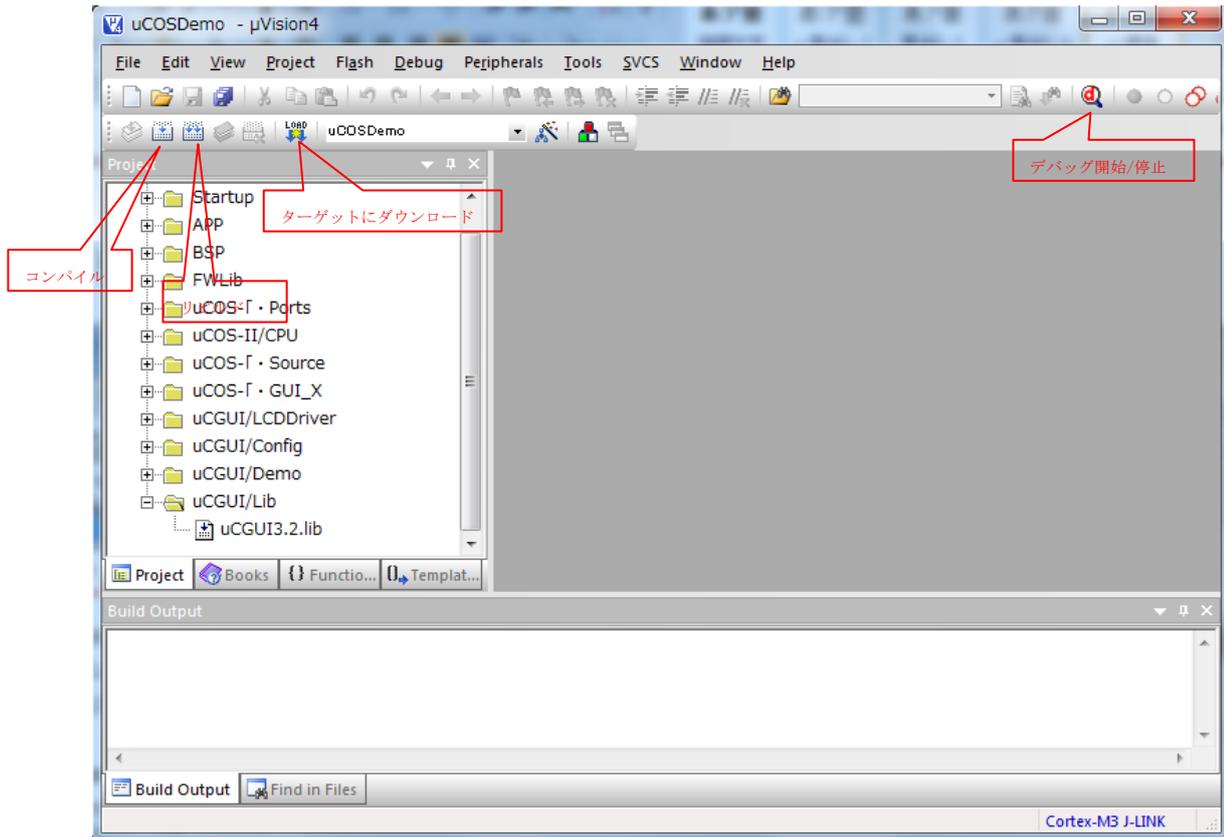
Target1でマウスを右クリックして” Manage Components” を選択する。



必要に応じてグループフォルダを追加する。LIB、APP など。



コンパイル、ビルド、ダウンロード、デバッグなどの操作。



以上