

=====

mcHFのファームウェアを改造するための環境構築法

=====

自力でファームウェアを改造するために必要な環境とインストール手順を紹介します。mcHFのファームウェアを含めたUHSDRソフトウェアはGitHubですべてのコードが公開されており、コードはC言語で記述されています。最近では、従来のStandard Peripheral Libraryから新しいHALドライバに切り替えて開発進行中です。ソフトウェアはGNU GPLv3で公開されており、パソコン上に開発環境を構築すれば自分なりに自由にコードを改良して再配布が可能になります。

1ファームウェア、ブートローダの開発ビルド環境構築

UHSDRのGitHub内のwiki(<https://github.com/df8oe/UHSDR/wiki>)に、setting up a firmware build environment(<https://github.com/df8oe/UHSDR/wiki/Topics:-UHSDR-SW-Development>)というファームウェアとブートローダのソースコードからビルドする環境構築の手順について記されています。OSはWindows, MacOS, Linuxで構築可能です。mcHFのコードは統合開発環境(IDE)であるEclipseのプロジェクトとしてGitHubで公開されています。UHSDRのwikiではIDEにはGNU MCU Eclipse(<https://gnu-mcu-eclipse.github.io/>)を推奨されています。

Windows系64bit版OS (Windows7以降)でのUHSDRのmcHF用ファームウェアのビルドするための開発環境構築ガイドが公開されていますので、この項ではその手順について解説します。元のURLはこちらです。
<https://github.com/df8oe/UHSDR/wiki/Windows-x64-Compiler-Setup-and-Build-Environment>
ほかのOS環境はwikiを参照してください。

【準備】

いくつか新規にフォルダーを作成します。まず、ユーザーフォルダの下に“uhedr-dev”フォルダーを作成します。さらにそのフォルダー下に“downloads”, “install”フォルダーを作成してさらに作成した“install”フォルダーの下に“node”フォルダーを作成します。

【各ファイルのダウンロード】

(1)から(4)の必要なファイルをダウンロードします。各ダウンロードファイルしたファイルはユーザーフォルダのダウンロードフォルダーに一旦置きます。

(1)JREもしくはJDK

Javaのサイト(<https://java.com/ja/>)から最新版のJREもしくはJDKをインストールします。2019年2月時点での最新版は、Versuin 8 Update 201です。64bit版と32bit版の両方をダウンロードします。

(2)64-bit Git for Windows Setup

<https://git-scm.com/download/win>からダウンロードします。最新版はGit-2.20.1-64-bit.exeになります。

(3)64bit版のGNU MCU Eclipse

<https://github.com/gnu-mcu-eclipse/org.eclipse.epp.packages/releases>からダウンロードします。最新版はGNU MCU Eclipse IDE for C/C++ Developers 2018-12 20190101で、ファイルは20190101-2023-gnumcueclipse-4.5.1-2018-12-R-win32.win32.x86_64.zipです。

(4)Node.js

<https://nodejs.org/en/>から64bit版をダウンロードします。

最新版は<https://nodejs.org/dist/v10.15.0/node-v10.15.0-win-x64.zip>です。

【インストール】

(1)Java

ダウンロードした32bit,64bit版両方各インストーラーを起動してインストールします。

(2)Git for Windows

Git-2.20.1-64.bit.exeインストーラーを起動しインストールを開始しますが、途中に出てくる設定などはすべてデフォルトのまま進めます。動作確認のためuhsdr-devフォルダーを開いて右クリックし、展開したコンテキストメニューの”Git Bash Here”をオープンするとコマンドプロンプトが現れます。lsコマンドを打ち込み”download”と”installs”が現れることを確認します。

(3)パッケージマネージャ (Node.jsとXPM)

Node.jsは、まずダウンロードファイルに置いたnode-v10.15.0-win-x64.zipを展開します。

展開したフォルダ内のすべてのファイルをuhsdr-dev¥installs¥nodeフォルダー内にコピーもしくは移動します。そこで右クリックしコンテキストメニューを開いて”Git Bash Here”を選んでコマンドプロンプトを出します。

XPMは、開いたコマンドプロンプトに、”./npm install xpm”を入力するとXPMのダウンロードとインストールが開始されます。コマンドプロンプトは閉じないで置きます。

(4)GNU MCU packages

Toolchain, build-tools, openocd, qemuはXPMでダウンロードとインストールを行います。開いてあるコマンドプロンプトに以下のように打ち込みます。

```
./xpm install --global @gnu-mcu-eclipse/arm-none-eabi-gcc
./xpm install --global @gnu-mcu-eclipse/windows-build-tools
./xpm install --global @gnu-mcu-eclipse/openocd
./xpm install --global @gnu-mcu-eclipse/qemu
```

(5)GNU MCU Eclipse

GNU MCU EclipseのZipファイルを展開して、展開された”eclipse”フォルダーをまるごとuhsdr-dev¥installs¥に移動します

※注意 EclipseではARM Toolchain Pathが通っていないことがあってコンパイルエラーになってしまうことがあり、設定を確認するようにしてください。

2ソースコードの取得とプロジェクトのインポート

インストールしたEclipseを起動してGitからプロジェクトをインポートします。メ

ニュー・バーの[File]-[Import]メニュー選択すると、図2のようにImportウィンドウが現れます。リストの中から“Git-Projects from Git”を選択して、[Next]ボタンをクリックします。図3のように現れるウィンドウでは“Clone URI”を選択して[Next]ボタンをクリックします。UHSDRのGitHubページ(図4)のCodeタブを押して、[Clone or download]ボタンをクリックして現れたURL(<https://github.com/df8oe/UHSDR.git>)をコピーします。図5に示したEclipseの"Source Git Reposistry"ウィンドウ内のURIのテキスト入力エリアにURLをペーストし[Next]ボタンをクリックします。

次に現れる図6の“Branch Selection”はそのままにして、[Next]ボタンをクリックします。“Load Destination”ではインポート先のDirectoryを図7のように指定して[Next]ボタンをクリックするとcloningを開始します(図8)。

インポートが完了したら図9の"Import Project"ウィンドウでmchf-eclipseのチェックボックスにチェックが入っているのを確認してFinishボタンをクリックすると図10のようにEclipseウィンドウ左のProject

ExploreにUHSDR projectがインポートされます。-----

3mchFファームウェアのビルド -----

mchFファームウェアのビルドには、本体(DebugMCHF)の前にDSPライブラリ DebugLibMCHF)をビルドする必要があります。

まずprojectのルート上にカーソルをあわせて右クリックし、[Build Configuration]-[Set Act4e]-[3 DebugLibMCHF (mchF DSP Library (STM32F4))]を選択してアクティブにします。もう一度project上で右クリックして[Build Project]を選択すると、図11右下のコンソールウィンドウで mchF用DSPライブラリ Libmchf-eclipse.aがビルドが確認て

きます。次にproject上で右クリックして[Build configuration]-[Set Act4e]-[6 DebugMCHF (mchF Firmware (STM32F4),requires DebugLibMCHF build to run before)]を選択してアクティブにします。同様にproject上で右クリックしてBuild Projectを選択すると、図12のコンソールウィンドウで先にビルドしたLibMCHFを取り込みファームウェア fw-mchf.binのビルドが確認できます。ビルドされたファームウェアは ((workspace)¥git¥UHSDR¥mchf-eclipse¥DbugMCHFフォルダ内にあります。

4ブートローダ・ファームウェアの書き換え

ブートローダがv2.2.3以降の新しいバージョンであれば、パソコンに接続しなくてもUSBメモリ経由で書き換えが可能になり作業が楽になります。

(1)ブートローダの書き換え

新しいバージョンのブートローダの書き換えは、下のURLにあるHSDR wiki内の手順を示したPDF

ファイルを参照に行います。

初期のM0KNA bootloaderまたは、DF8OE bootloader 2.2.3以前のブートローダの場合は、本体のジャン

パピン(ve.0.6はP6, ve.0.7はP206)をセットし、DFUモードで起動してWindowsパソコンの場合Dfuse

demo(v.3.0.5)でインストールするという手順になります。詳細については、以下のURLからpdfファイ

ルをダウンロードしてください(英文)

https://github.com/df8oe/UHSDR/blob/act4e-devel/mchf-eclipse/useful_manuals/bootloader_install.pdf

(2)USBメモリを使ったファームウェアの書き換え

図13のようにパソコンでビルドした新しいファームウェア(ファイル名fw-mchf.bin)をそのままUSBメモリにコピーします。ファームウェアをコピーしたUSBメモリを写真12のようにmcHF本体のUSBタイプAコネクタ接続し、BAND-ボタンを押しながら、[Power]ボタンを押して本体を起動させます。[Band]ボタンを押したままにしてしばらくすると写真13のようにLCD表示がFirmwareインストール画面になり、USBメモリ内のfw-mcHF.binを検出してからそのまま自動でファームウェアの書き換えが開始されます。書き換えが終了すると写真14のようにLCD上にFinishと表示されるので、BAND-ボタンから指を離してからUSBメモリを本体から外すと、新しいファームウェアで再起動します。新しいファームウェアに書き換えると、写真15のように起動画面に"Firmware change detected!"と表示されます。