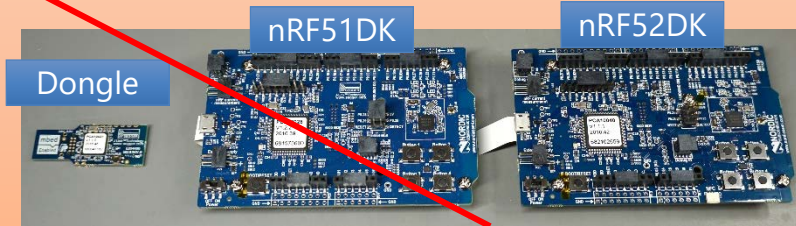


nRF5 開発に使うモノ一覧

ボード



IDE

MDK-ARM
Keil μ Vision

SDK

nRF5 SDK

Shield room ...

Tools (Win App)

nRFgo Studio

Master Control Panel

nRF Connect

J-Link RTT Logger

J-Link RTT Viewer

Tools (iPhone App)

LightBlue

nRF Connect

Tools (Android App)

nRF Connect

MDK-ARM μ Vision (IDE)のインストール

- ARMの統合開発環境
nRF51はCortex-M0, nRF52はCortex-M4がコア
ビルド、ターゲットへの書き込み、デバッグができます
- インストーラー入手先
<https://www.keil.com/demo/eval/arm.htm>
 - 但しフォーム入力が面倒で、後から代理店から営業されるのと、DLに時間が掛かるので、DLしたファイルを以下に置きます
<https://www.dropbox.com/s/sy5rhniursonqm5/MDK-523.EXE?dl=0>
- インストール
MDK-523.exe を実行

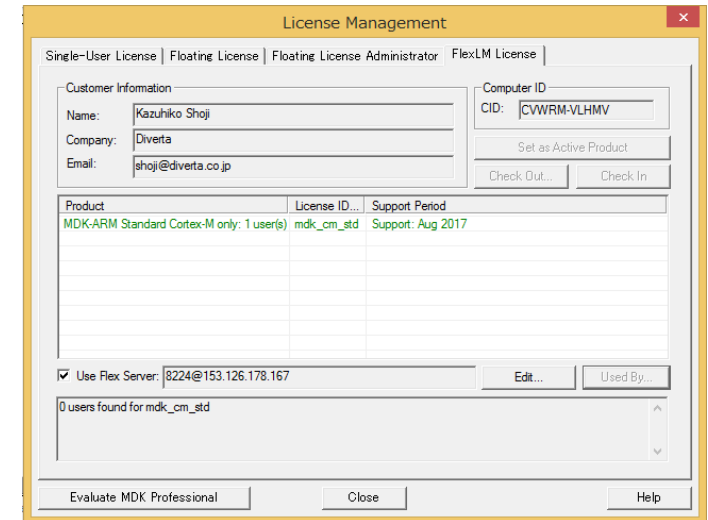
フローティング・ライセンス設定

- μVisonを管理者権限で実行

- File > License Management
FlexLM License Licenseタブを選択し、[Edit] をクリック
- Server (port@hostname): に
8224@153.126.178.167 を入力して[OK]
- ダイアログを [Close]

- ライセンス数

- 今のところ1個のみ
- ライセンス参照タイミング
 - μVision起動時, (多分)ビルド時
複数人で起動は可能、タイミングが被らなければビルドも可能
 - デバッガ起動中
デバッガ起動中はずーっとライセンスを専有するので、他の人はビルドできない



nRF5 SDKのインストール

- SDK
nRF5x用のライブラリ、SoftDevice (BLEプロトコル・スタック)、サンプルコード、各IDE用のプロジェクトファイルが入ってます
- SDK入手先
https://developer.nordicsemi.com/nRF5_SDK/
から取得
 - v13
http://developer.nordicsemi.com/nRF51_SDK/nRF5_SDK_v13.x.x/nRF5_SDK_13.0.0_04a0bfd.zip
又は、
https://www.dropbox.com/s/mi20pr1rwl75hwt/nRF5_SDK_13.0.0_04a0bfd.zip?dl=0
- zipファイルを以下のパスに展開
 - C:¥Nordic¥nRF5_SDK_v13.0.0¥

nRFgo Studioのインストール

- 単独の書き込みツール
通常はμVisionからのWriteでOKですが、SoftDeviceやBootloaderのWrite、Full Eraseなど、時々こちら也使います
- nRFgo Studio
 - インストーラ
 - 32bit : <https://www.nordicsemi.com/eng/nordic/Products/nRFgo-Studio/nRFgo-Studio-Win32/22286>
 - 64bit : <https://www.nordicsemi.com/eng/nordic/Products/nRFgo-Studio/nRFgo-Studio-Win64/14964>
 - ダウンロードしたインストーラーを実行する
μVisionのインストールと同じDLLが含まれるので、置き換えるか？のダイアログが出るが、置き換えなくてOK(多分)

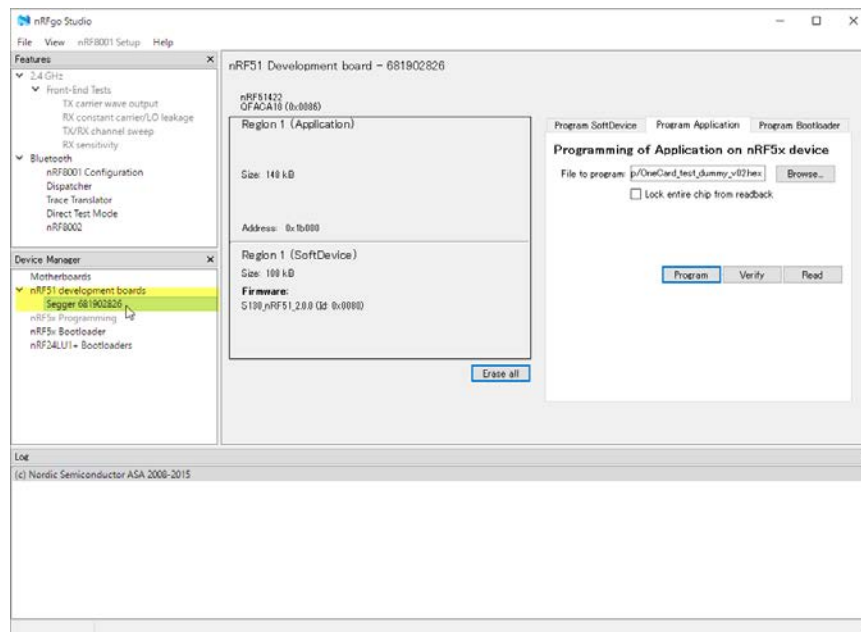
※ nRFgo Studio , μVisionと同時にJ-TAGと接続するUSBドライバがインストールされますが、Arduino IDEのUSBドライバとそのままでは共存できません。もし同じマシンでArduino IDEを使う場合は、共存させる設定が必要ですのでご注意ください。

nRFGo Studioにボードを接続してみる

- 手順

- nRF52DK (PCA10040)をPCにUSBで接続する
→ デバイスマネージャーで、
ポート(COMとLPT) > JLink CDC UART Port
として見える筈

- nRFGo Studioを起動する
→ Device Manager
> nRF52 development boards
> Segger xxxxxxxxxx
のように現れればOK



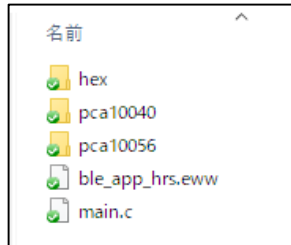
スマートフォンBLEツールのインストール

- BLE Central側の役割をするツール
ここで指定した以外でもBLEツールが色々あります
 - iPhone版
 - nRF Connect (Nordic Semiconductor ASA)
 - Android版
 - nRF Connect for Mobile (Nordic Semiconductor ASA)

基本、同様の機能のAppですが、iOS/Androidの仕様の違いにより、微妙にできることが違うので、できればiPhone/Android両方あると良い。

サンプルプロジェクトをビルドしてみる (1)

- v13 の ble_app_hrs をビルドしてみます
 - サンプルプロジェクトの場所
C:\Nordic\NRF5_SDK_v13.2.0\examples\ble_peripheral\ble_app_hrs
 - フォルダの中身



hex → ビルド済みのバイナリ

pca100xx → 開発ボード毎の設定/プロジェクトファイル

ble_app_hrs.eww → 他のIDEのプロジェクトファイル (使いません)

main.c → ソースコード

- nRF52DK は pca10040 以下のファイルを開きます
pca10040\s132\arm5_no_packs
ble_app_hrs_pca10040_s132.uvprojx ← を開く

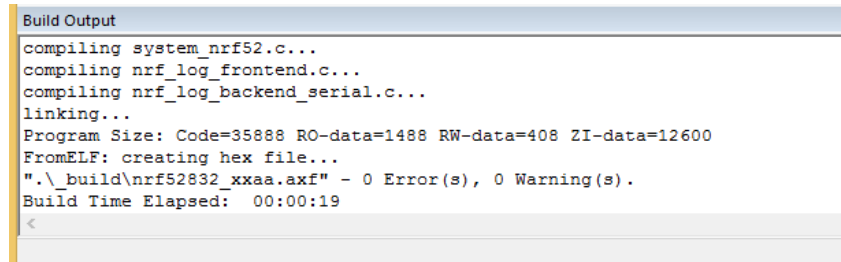
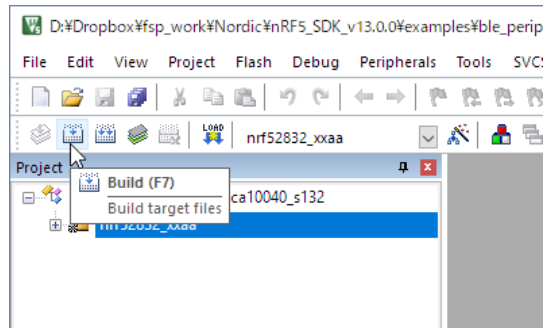
初回は色々パッケージが不足しているので、ダウンロードしてインストールするか？ 訊かれますので、言われるがままインストールします。

サンプルプロジェクトをビルドしてみる (2)

- ビルドする

ビルドターゲット: nrf52832_xxaa のまま Build (F7) クリック

※ターゲット flash_s132_nrf52_4.0.4_softdevice ではビルド実行しないように...
(既にあるsoftdeviceのhexファイルがcleanされて消えてしまいます...)



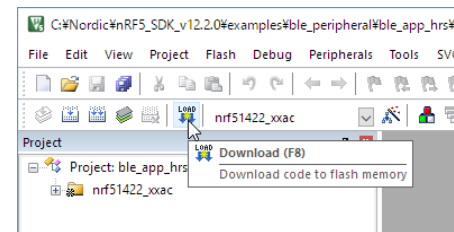
と出ればOK

ボードにDownloadして動かしてみます (1)

- ...と、その前に
先にSoftDeviceを書き込みます
 - SoftDeviceとは？
 - nRF5xのプログラム(Flash領域)は、
SoftDevice / Application / Bootloader
の3層構造になっていて、BLEプロトコルスタックとそのAPIはSoftDeviceに組み込まれています。
ユーザープログラムは通常Applicationのみ書き換える形で開発します。
無線機能に関わるバイナリが変化してしまうと、技適等の各種認証も取り直しになってしまうのですが、SoftDeviceのバイナリをユーザープログラムと分離して固定化することで、この問題を回避できるアーキテクチャになっています。
 - nRFgo Studioを起動して、nRF52DKをUSB接続
 - Device Manager > nRF52 development boards > Segger xxxxxxxxx を選択
 - 一旦、[Erase all]
 - Program SoftDeviceタブ、File to programで
C:\Nordic\nRF5_SDK_v13.0.0\components\softdevice\s132\hex\s132_nrf52_4.0.2_softdevice.hex
を指定して、[Program]

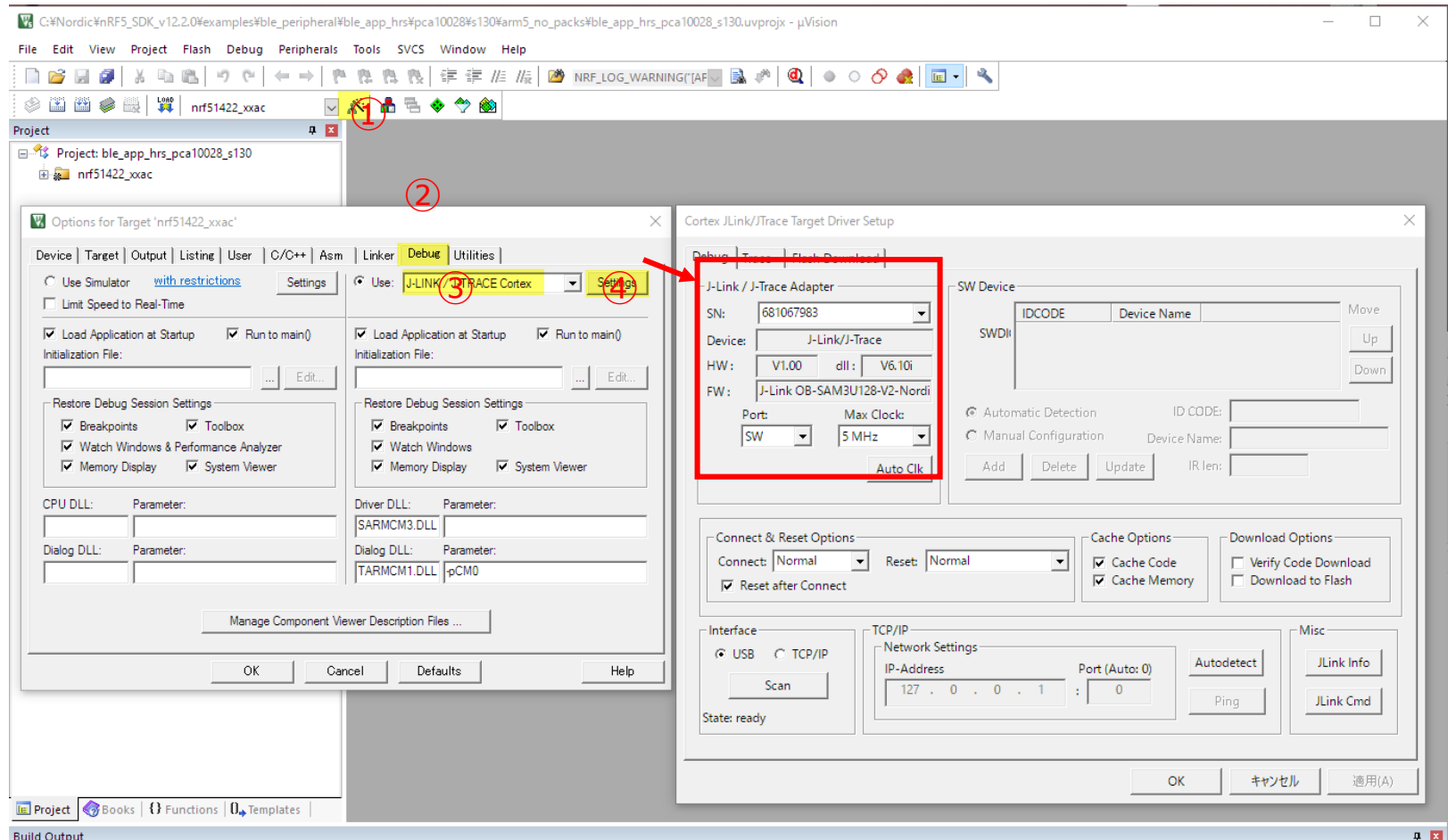
ボードにDownloadして動かしてみます (2)

- 方法2通り
 - µVisionから → ビルドに続けて書き込みする場合はこちらが便利
 - nRFgo Studioから → ビルド済みのHexを書き込む場合
- µVisonからDownload
 - ターゲット選択 nrf51422_xxac
 - LOADボタンを押すとボードにDownloadされます
 - Errorになる場合... Target Driver Setupが必要かも



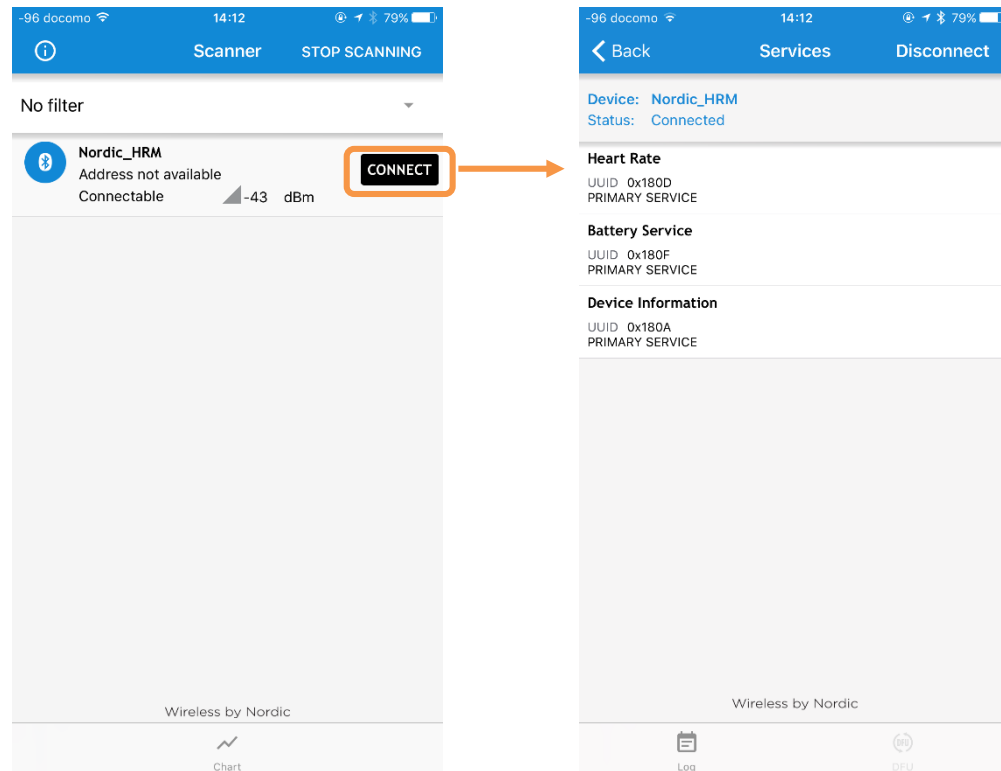
JTrace Target Driver Setup

- Downloadできない時は... (多分この設定)
J-LINK/J-TRACE Cortexを選択し、[Setting]クリック後、J-LINKが認識されてSNが表示されればOK



接続確認

- nRF Connect (iPhone版) の場合...
 - Download完了後、Resetが掛かり即Advertising開始します
 - CentralデバイスからScanすると、デバイス名：Nordic_HRMで検出される筈



ソースファイルの置き場所

- SDKのツリー内、各サンプルと同じ階層に置きます
(プロジェクトファイルでの各SDKコンポーネントへの参照方式がそれぞれ相対パスで記述されている為)
 - 例：nRF5_SDK_v13.0.0¥examples¥diverta¥[projectname]