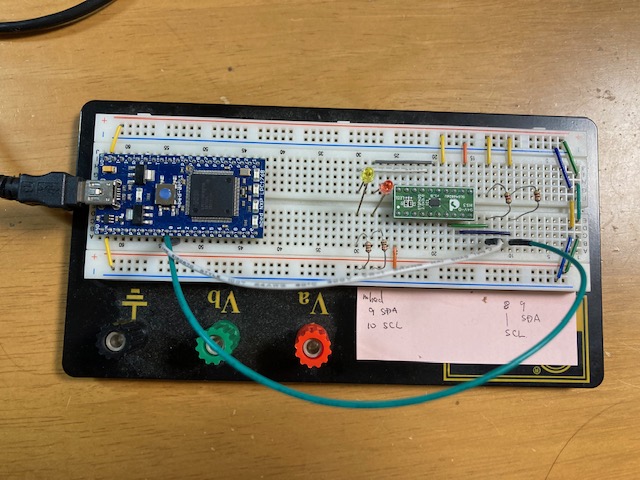
GreenPak(SLG46826V-DIP) Programaer説明書

# 概要

mbed(LPC1768)を使ってGreenPak(SLG46826V-DIP)に設計データを書き込むツールです。

mbed(LPC1768)はPCに接続するとUSBメモリとして認識されるので、GreenPakの設計データであるHexファイルをそのままmbedに転送して、GreenPakに書き込むことができます。

GreenPakには回路データと内蔵EEPROMデータの２種類のHex fileがありますがどちらも書き込みができます。



# 準備

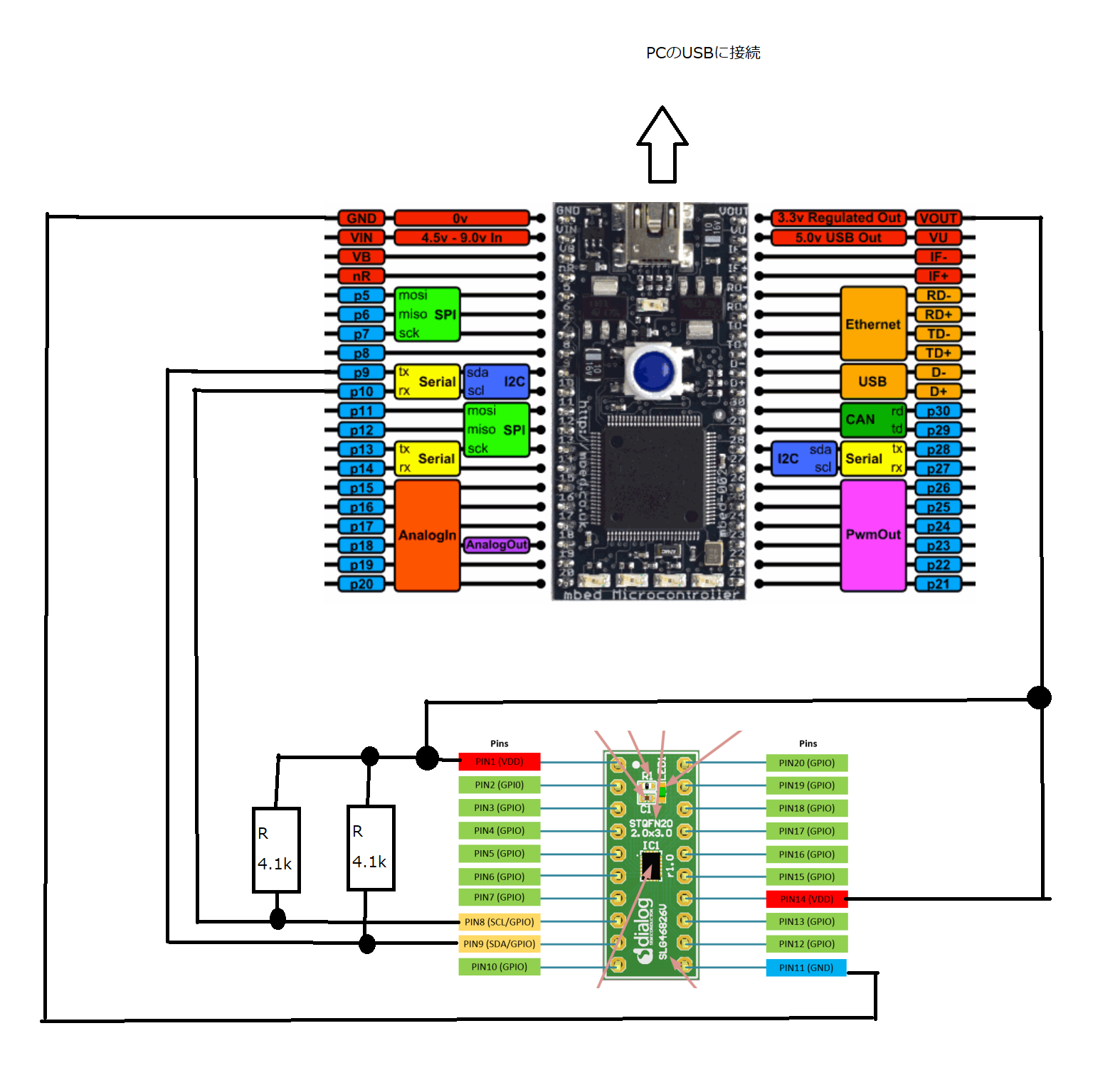
## 回路

mbed(LPC1768)とSLG46826V-DIPを以下のように接続します。

mbedに接続できるSLG46826V-DIPは１つです。

|  |  |
| --- | --- |
| mbed(LPC1768) | SLG46826V-DIP |
| VOUT(3.3v Regulated Out) | PIN1, PIN14 (VDD) |
| GND(0v) | PIN11(GND) |
| p9(sda) | PIN9(SDA)  Pull up抵抗として4.1[kΩ]をPIN1,PIN14に接続 (\*1) |
| p10(scl) | PIN8(SCL) ,  Pull up抵抗として4.1[kΩ]をPIN1,PIN14に接続 (\*1) |

\*1: 抵抗値は1[kΩ]～10[kΩ]程度でよい。



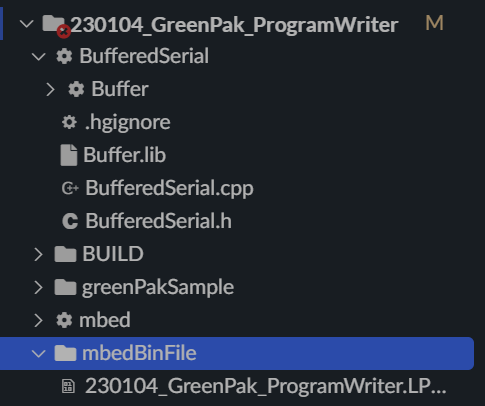
## mbedの実行ファイル

mbed(LPC1768)に実行ファイル(230104\_GreenPak\_ProgramWriter.LPC1768.bin)をコピーします。

PCにmbedを接続するとPCからUSBメモリとして認識されるので、このドライブにbinファイルをコピーします。

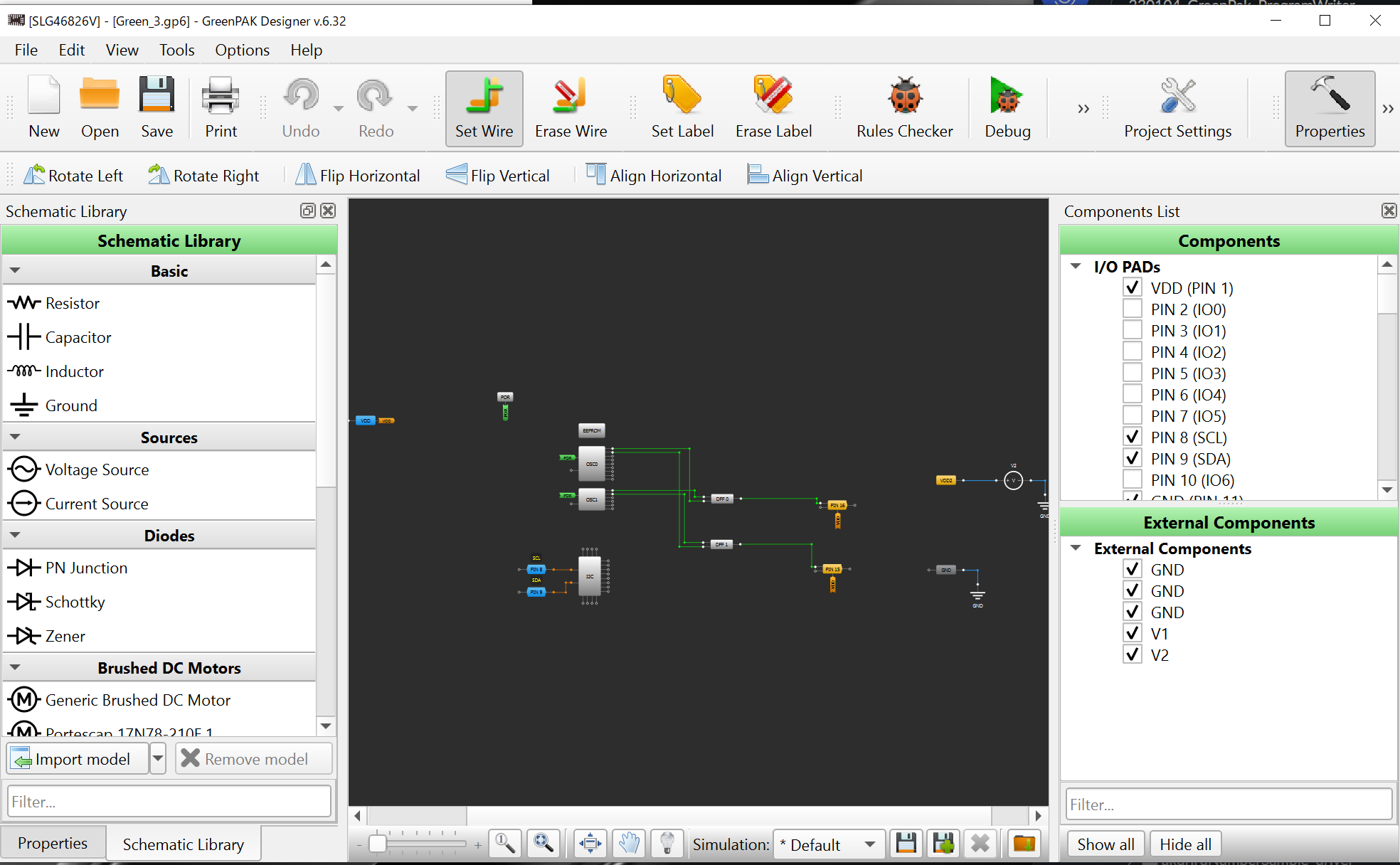
コピーした後、mbedのリセットボタンを押すとコピーしたファイルで動作を開始します。

“mbedBinFile”フォルダに実行ファイルが保管してあります。これを使えばコンパイルの必要はありません。



## SLG46826V-DIPのHEXファイルの準備

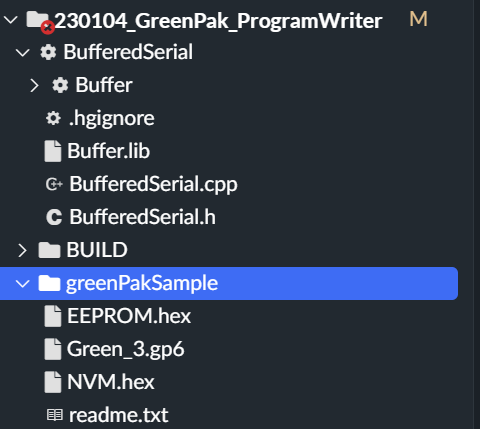
GreenPAK DesignerでHEXファイルを生成します。



動作確認用として”greenPakSample”フォルダに書き込み用のHEXファイル(NVM.hex, EEPROM.hex)を用意してあります。

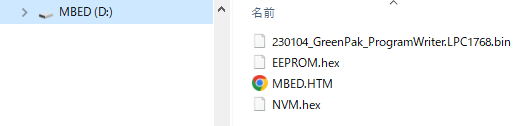
このファイルではSLG46826V-DIPの15Pin, 16Pinからランダムな波形を出力します。この端子にLEDを接続すれば点滅します。

EEPROM.hexにはアドレス0x00～0x03に0x00～0x03の値を設定してあります。（動作には影響しません。EEPROMのクリア、書き込みの動作確認用です）



このフォルダにあるNVM.hexとEEPROM.hexをmbedにコピーします。この２つのファイルの内容をmbedからSLG46826V-DIPに書き込みます。

ここまでmbedにファイルをコピーすると以下のようになります。



## PCからのmbedの操作

PCからmbedへの指示はターミナルソフトを使い指示や動作内容の表示を行います。ここではTera Termを使用しました。

Tera Term の設定 – シリアルポート　を選択して設定します。

シリアルポートの設定

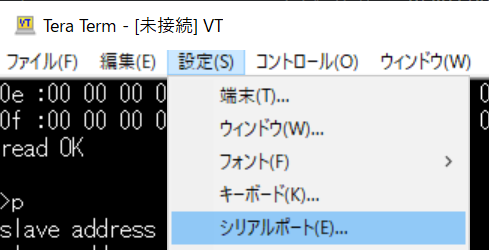
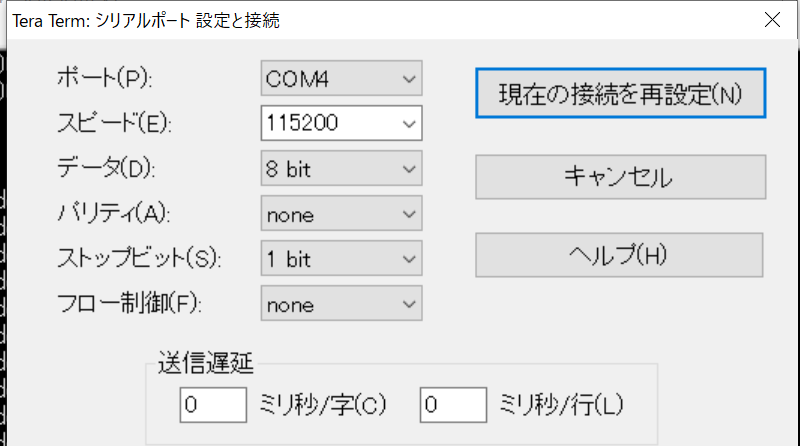
ポート: （これはPCとの関係で変わる）

スピード: 115200

データ: 8bit

パリティ: none

ストップビット: 1bit

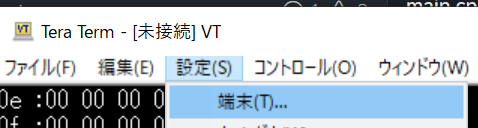
 

PCからのキーボード入力を表示させるためと、mbedからのデータ表示の改行の設定をします。

Tera Termの 設定－端末　を選択して設定します。

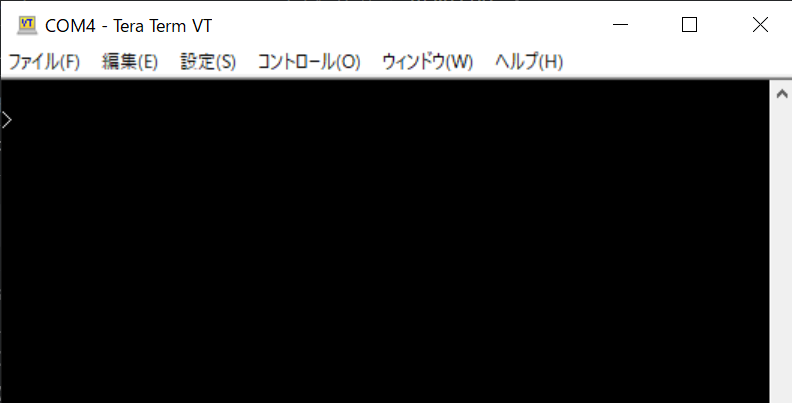
改行コードｰ受信：　AUTO

ローカルエコー　にチェックを入れる



# 使用方法

mbedをPCに接続して、PCからTera Tarmを立ち上げます。この後でmbedのリセットボタンを押すとTera Tarmの画面に”>”が表示されます。これでmbedからSLG46826V-DIPへの操作が可能になります。

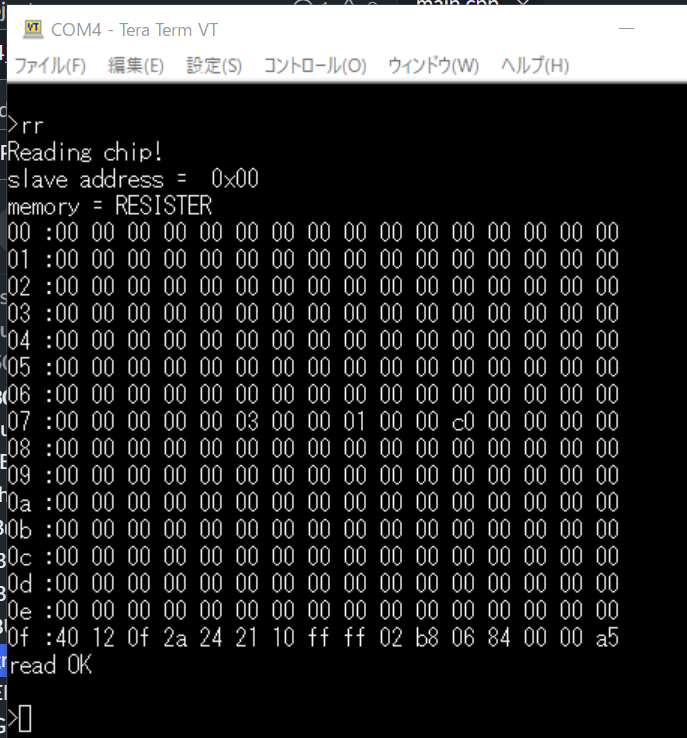


コマンドの入力には半角英数を使用します。アルファベットは大文字・小文字どちらでも構いません。

## 操作コマンドの説明

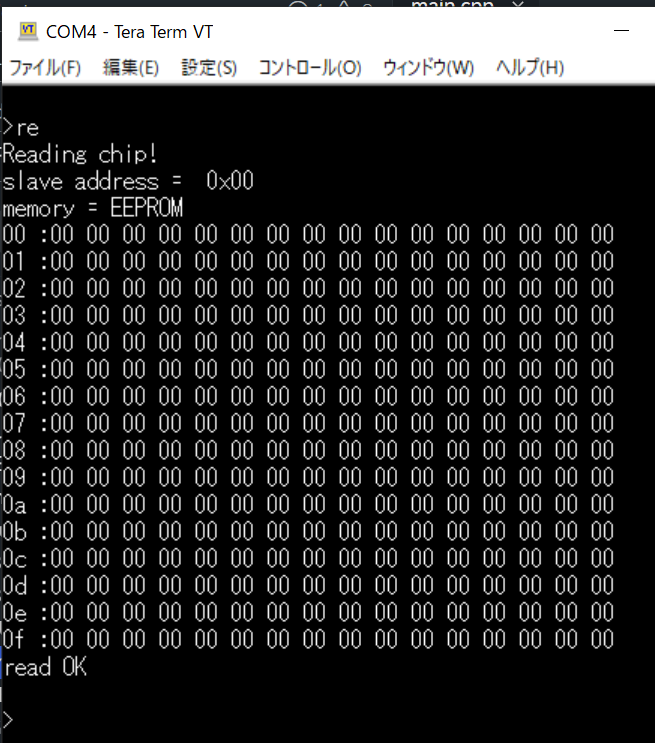
### レジスタ領域読み出し

“rr”を入力してenterを押します。



### EEPROM領域読み出し

“re”を入力してenterを押します。



### NVM領域読み出し

“rn”を入力してenterを押します。

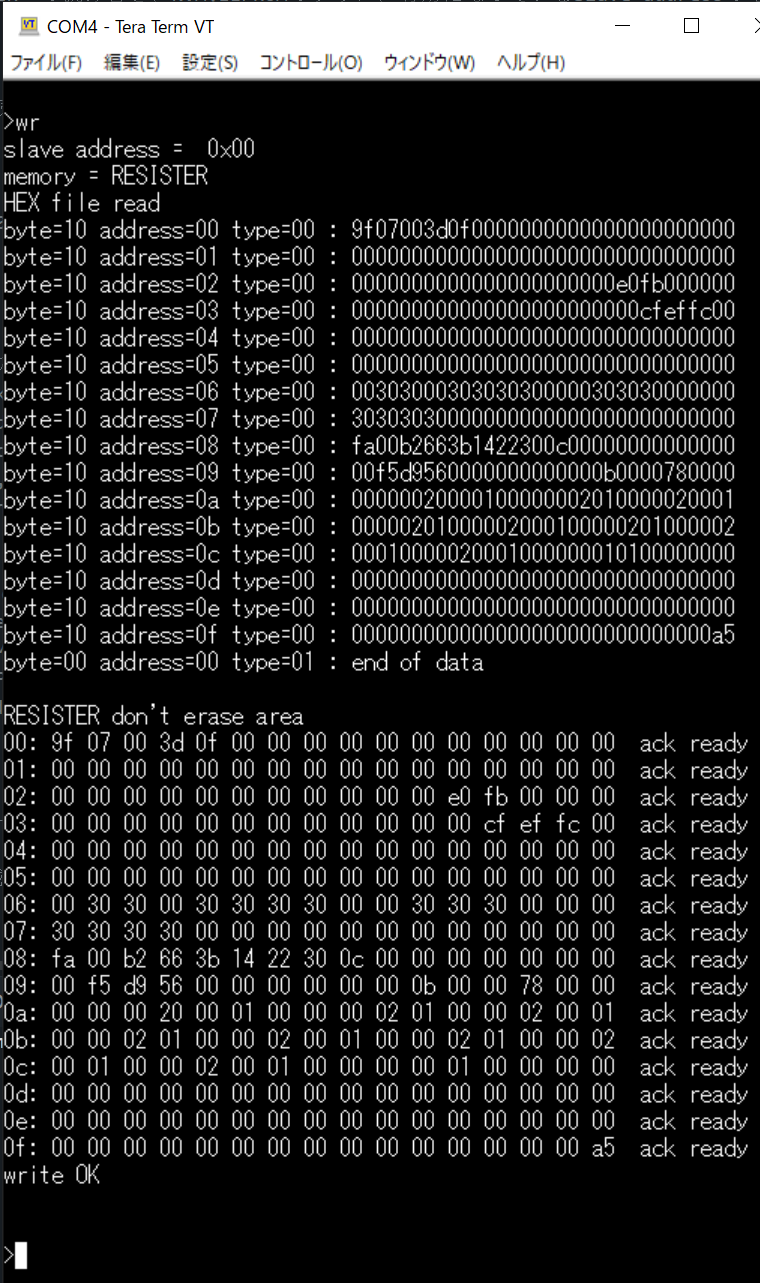


### レジスタ領域への書き込み

“wr”を入力してenterを押します。

書き込まれるデータはNVM.hexになります。

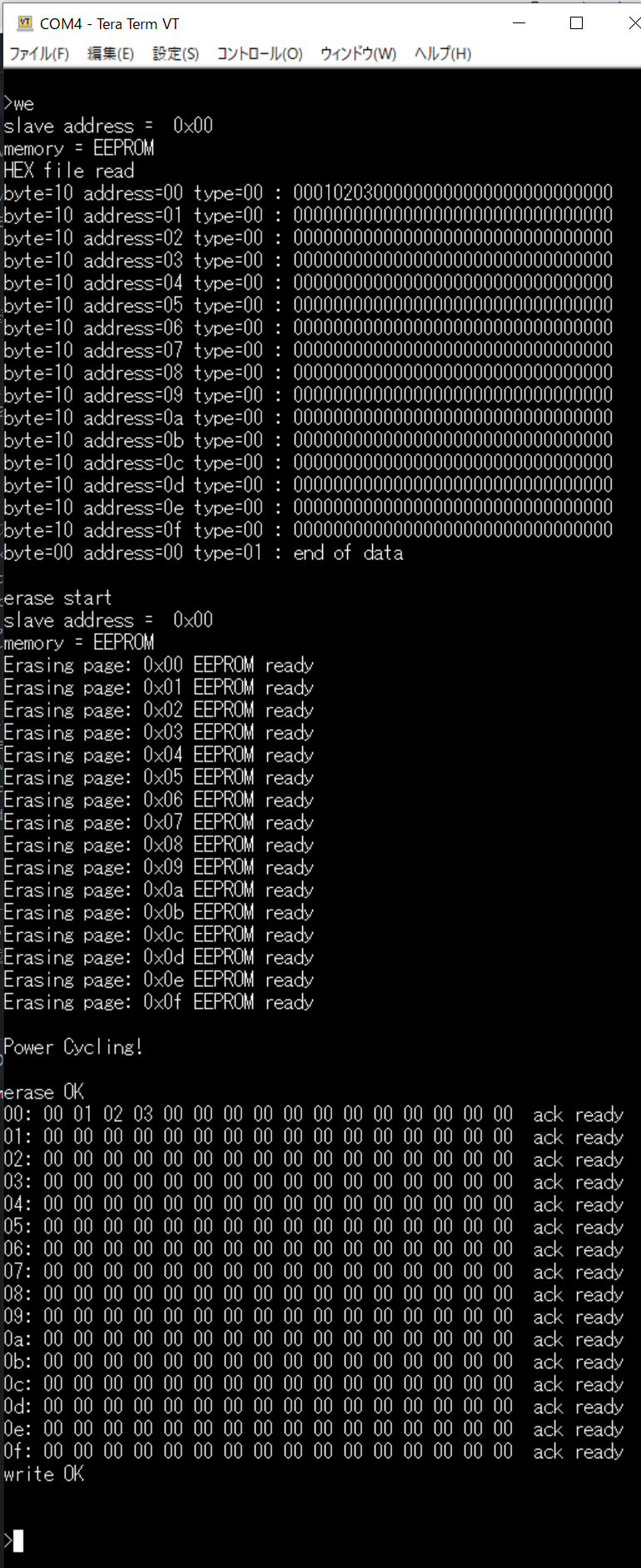
直接レジスタ領域に書き込んでもそのhexファイルの内容で動作します。NVMの寿命を気にしないで済むのでデバックの時に利用できます。ただし、レジスタ領域に書き込んだデータは電源を落とすと消えます。



### EEPROM領域への書き込み

“rn”を入力してenterを押します。

書き込まれるデータはEEPROM.hexになります。



### NVM領域への書き込み

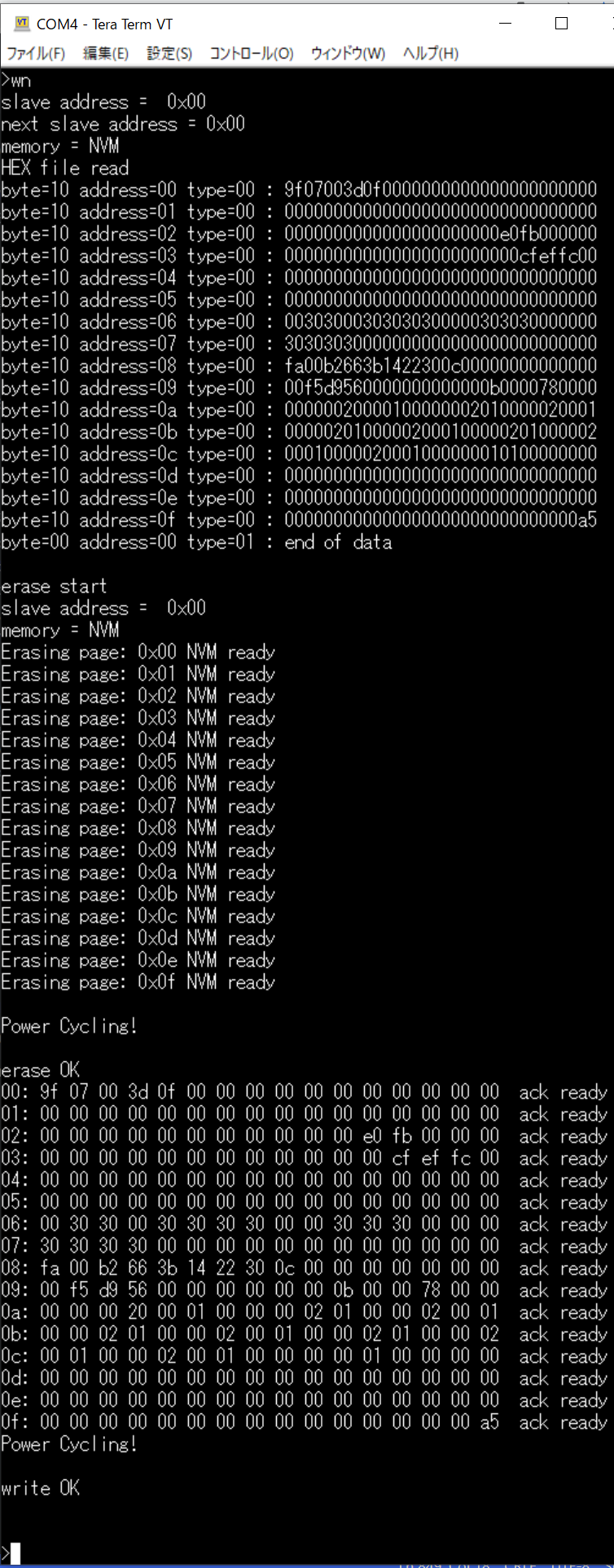
“rn?”を入力してenterを押します。

“?”には新たに設定するslave address(Contorl Code)を”0”～”9”,”a”～”f”の16進数を設定します。

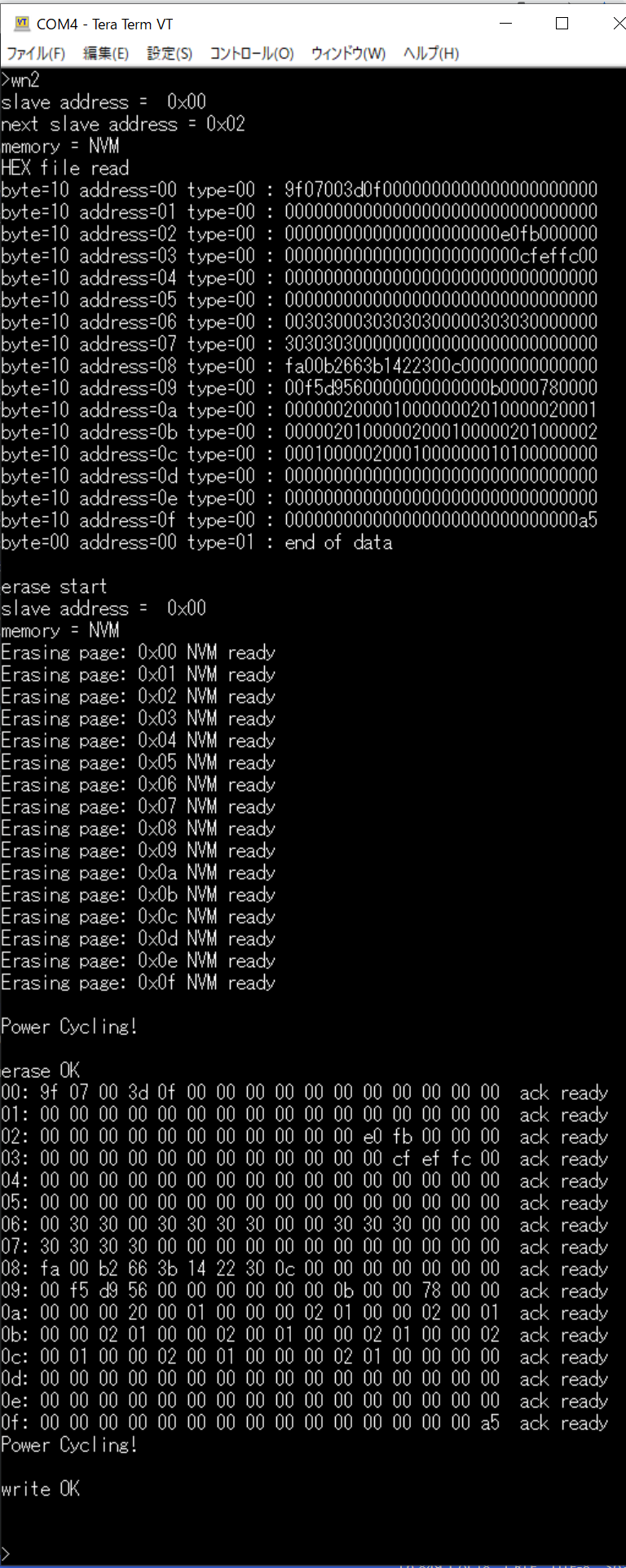
省略した場合は、現状のslave addressを継承します。

書き込まれるデータはNVM.hexになります。

“wn”として、現状のslave address(Contorl Code)を継承した場合

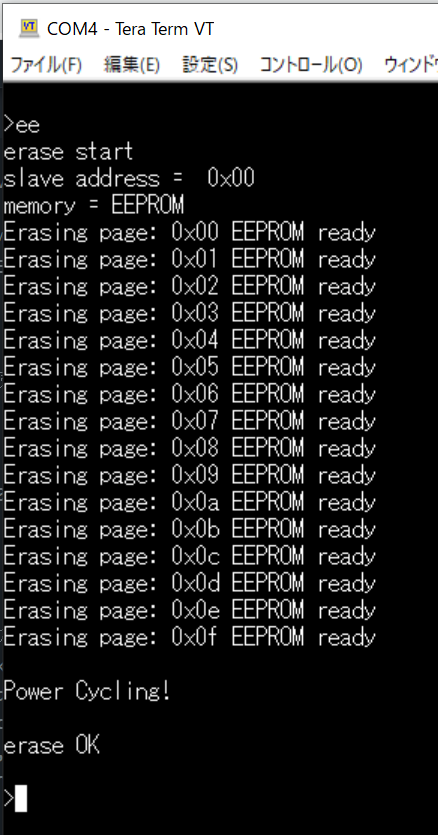


“wn2”としてslave addressが現状0x0から0x2に変更して書き込む場合



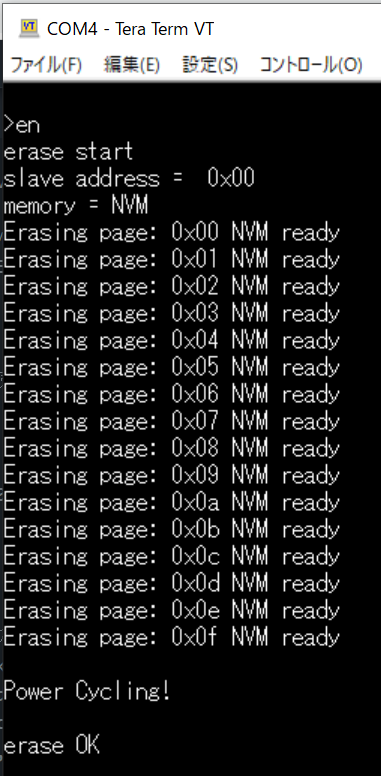
### EEPROM領域のクリア

“ee”を入力してenterを押します。



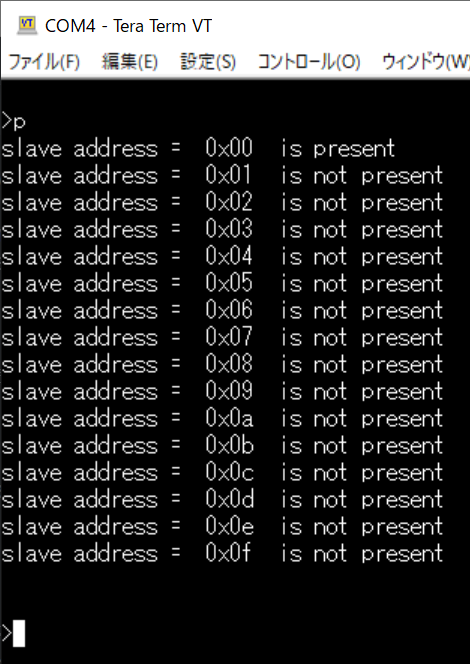
### NVM領域のクリア

“en”を入力してenterを押します。



### 現在のslave address(Control Code)の確認

“p”を入力してenterを押します。



# mbedの開発環境

mbedのプログラムはonline compler(studio.keil)で作成

フォルダ名 “230102\_GrennPak\_ProgramWriter”

mbedOS2 を使用している。

# 補足

mbedのプログラムは、下記のHPを参考にしました。

“M5StickCやM5Stackを使ってGreenPAKに設計データを焼いてみた“

<https://elchika.com/article/e0de0e0d-b41a-4d93-9d8a-2013f501c906/>