



OMRON 形2JCIE-EVセンサ
評価ボード※2(ラズパイ対応)



※1 <https://www.omron.co.jp/ecb/product-detail?partId=5747>

※2 <https://www.omron.co.jp/ecb/sensor/evaluation-board/2jcie>

mbed CLIインストールからOMRONセンサーデータのデモまで
(Jun Takedaさんのデモをビルド)

※ mbed CLIを GCC_ARMコンパイラ環境でビルド

Mbed Studioは、純正ARMコンパイラのみサポートしているようです。
GCC ARMでビルドするために mbed CLIの環境構築をやってみました。

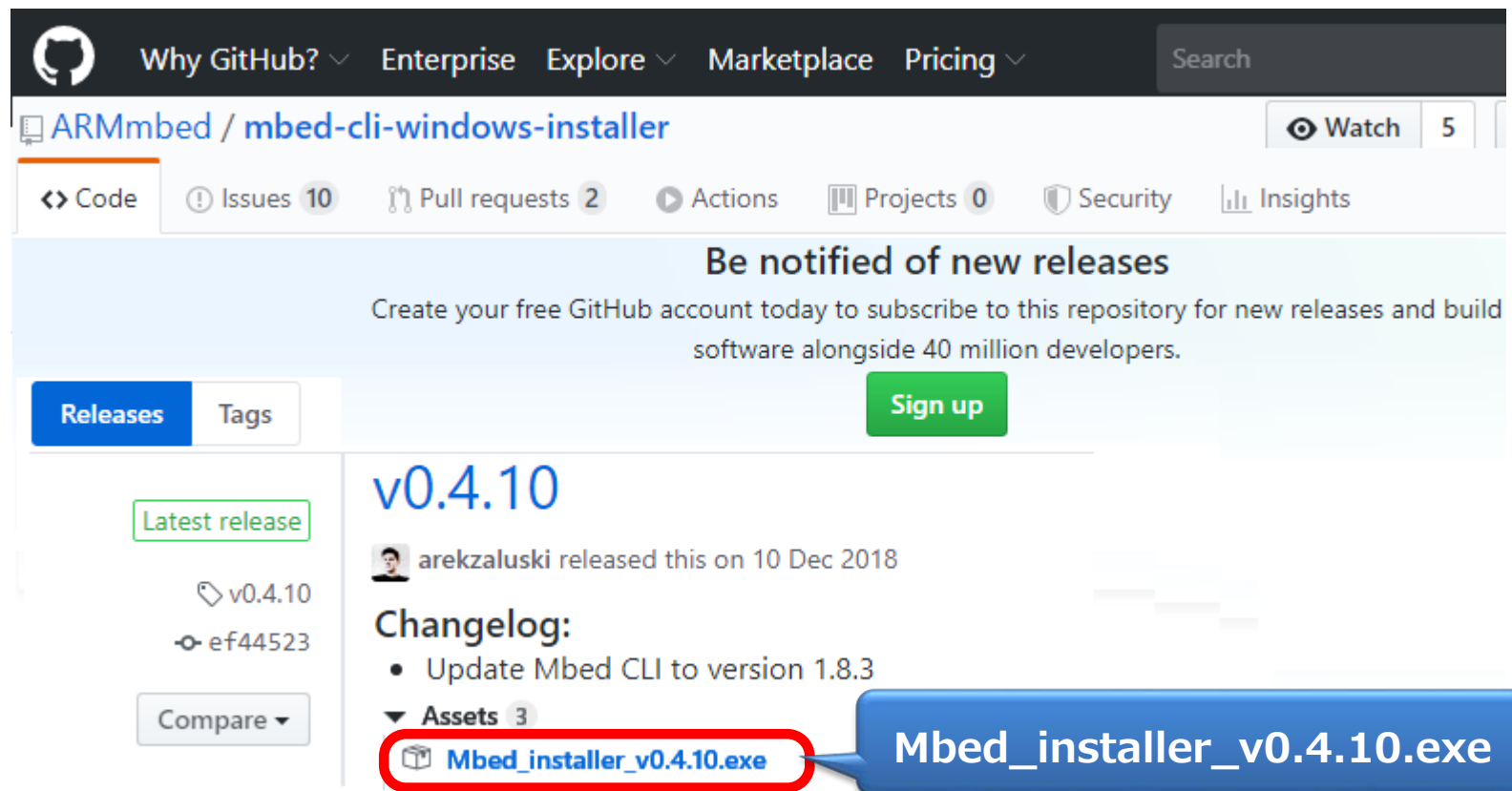
参考 mbed CLIをWindowsで試してみる（環境構築編）

<https://os.mbed.com/users/ytsuboi/notebook/ja-setup-mbed-cli-on-windows/>

Python 2.7系、Git、Mercurial、GNU Arm Embedded Toolchain

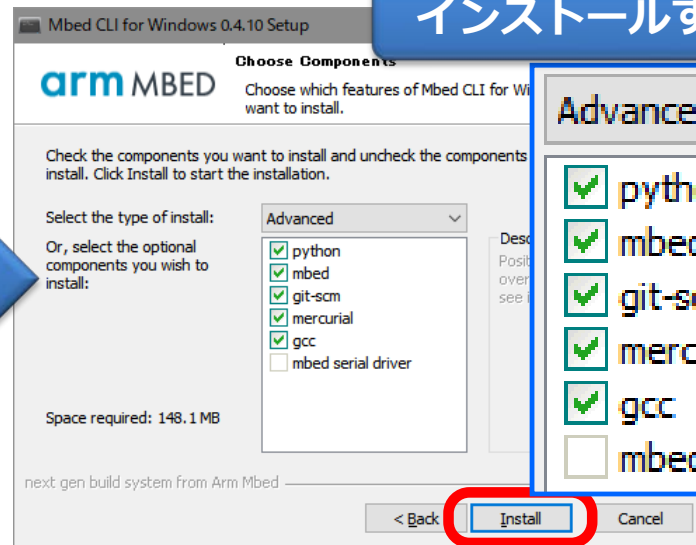
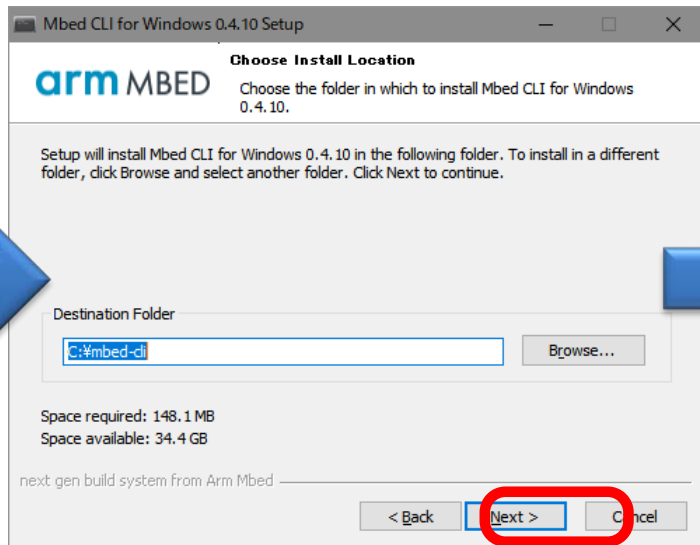
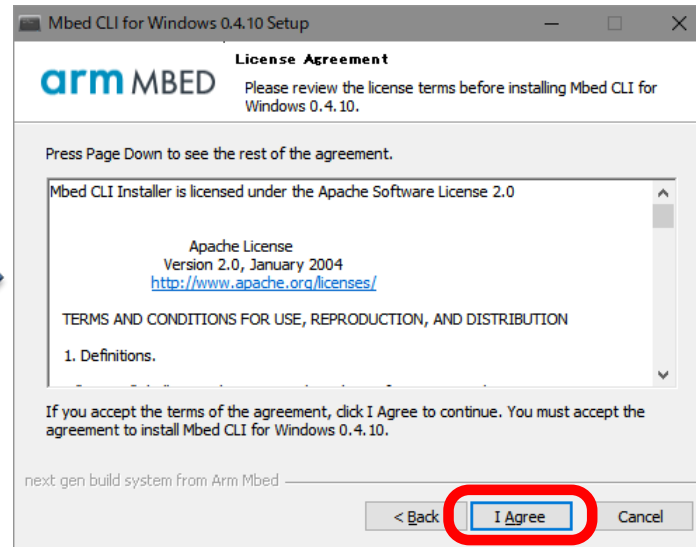
4つのツールをインストールする必要がありますがこれをダウンロードするだけでOKです。

<https://github.com/ARMmbed/mbed-cli-windows-installer/releases/>



The screenshot shows the GitHub repository page for `ARMmbed / mbed-cli-windows-installer`. The page features a navigation bar with links for "Why GitHub?", "Enterprise", "Explore", "Marketplace", and "Pricing", along with a search bar. Below the navigation bar, the repository name is displayed, followed by tabs for "Code", "Issues 10", "Pull requests 2", "Actions", "Projects 0", "Security", and "Insights". A prominent banner encourages users to "Be notified of new releases" by creating a free GitHub account. The "Releases" tab is selected, showing the latest release, `v0.4.10`, released by `arekzaluski` on 10 Dec 2018. The changelog for this release includes the update "Update Mbed CLI to version 1.8.3". Under the "Assets" section, the file `Mbed_installer_v0.4.10.exe` is listed and highlighted with a red circle. A blue callout box points to this asset with the text `Mbed_installer_v0.4.10.exe`.

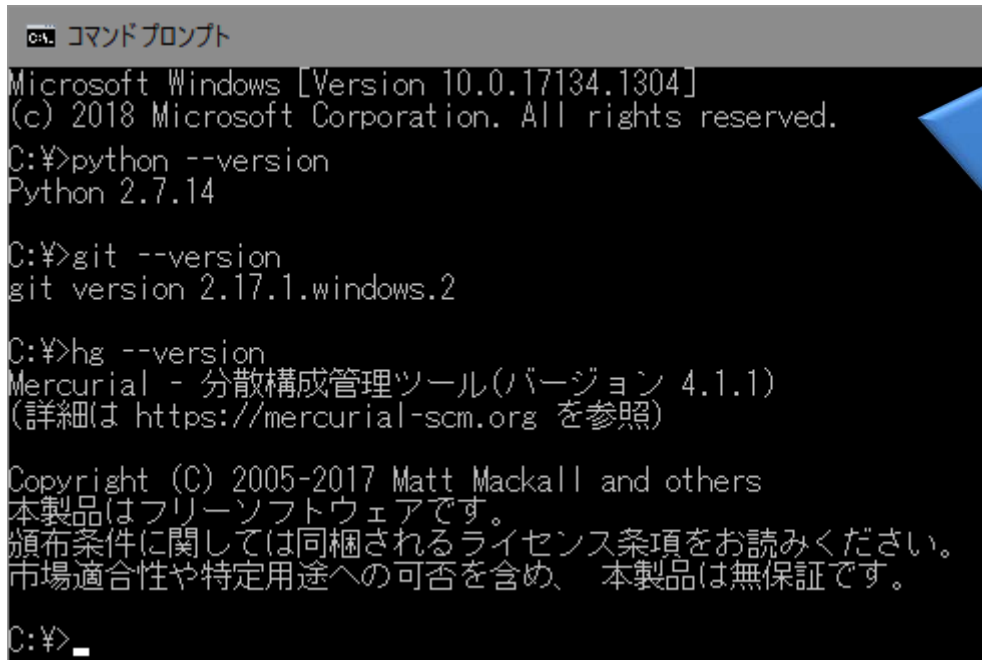
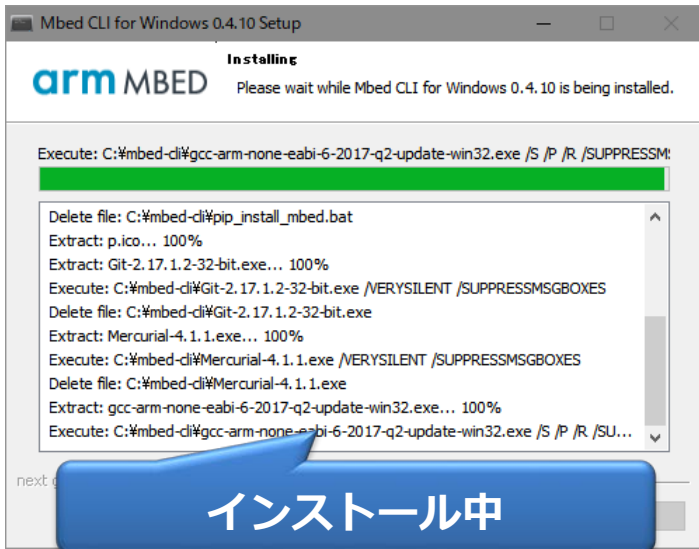
デフォルト設定でインストールを行う



インストールするツール群

- python
- mbed
- git-scm
- mercurial
- gcc
- mbed serial driver

インストール後、ツール群のバージョンを確認してみる



コマンドプロンプトでコマンド実行する

C:\> python --version

C:\> git --version

C:\> hg --version

バージョン表示されればOK

プロジェクトのセットアップ

コマンドプロンプトでプロジェクトフォルダを作る

```
C:¥> mkdir mtemp
```

```
C:¥> cd mtemp
```

JunTakedaさん作成のプロジェクトをimportする

```
C:¥mtemp> mbed import https://github.com/takjn/GRMOMronTest
```

```
C:¥> mkdir mtemp
```

```
C:¥> cd mtemp
```

```
C:¥mtemp> mbed import https://github.com/takjn/GRMOMronTest
```

```
[mbed] Working path "C:¥mtemp" (directory)
```

```
[mbed] Program path "C:¥mtemp"
```

プロジェクトのimportが開始される

```
[mbed] Importing program "GRMOMronTest" from "https://github.com/takjn/GRMOMronTest" at latest revision in the current branch
```

```
[mbed] Adding library "bootloader_d_n_d" from "https://github.com/d-kato/bootloader_d_n_d" at rev #34f943b0fae6
```

```
[mbed] Adding library "lvgl" from "https://github.com/littlevgl/lvgl" at rev #cde12976acc7
```

```
[mbed] Adding library "mbed-gr-libs" from "https://github.com/d-kato/mbed-gr-libs" at rev #3584be6f24f1
```

```
[mbed] Adding library "mbed-gr-libs¥components¥WIFI¥esp32-driver" from "https://github.com/d-kato/esp32-driver" at rev #0321d3cb4679
```

```
[mbed] Adding library "mbed-os" from "https://github.com/d-kato/mbed-os" at rev #82854c067198
```

```
[mbed] Auto-installing missing Python modules (colorama, urllib3, prettytable, junit_xml, pyyaml, jsonschema, future, six, mbed_cloud_sdk, requests, idna, pyserial, jinja2, intelhex, mbed_ls, mbed_host_tests, mbed_greentea, beautifulsoup4, pyelftools, manifest_tool, icetea, pycryptodome, pyusb, hidapi, cmsis_pack_manager, pywin32, wmi)...
```

```
C:¥mtemp> dir
```

```
C:¥mtemp のディレクトリ
```

```
2020/03/05 15:48 <DIR> .
```

```
2020/03/05 15:48 <DIR> ..
```

```
2020/03/05 15:51 <DIR> GRMOMronTest
```

dir コマンドで GRMOMron Test フォルダが作成されたことを確認

GRMOmron Testフォルダのmbed_settings.pyを編集してビルドを実行する

```
"""
mbed SDK
Copyright (c) 2016 ARM Limited

#####
# Build System Settings
#####
#BUILD_DIR = abspath(join(ROOT, "build"))

# ARM
#ARM_PATH = "C:/Program Files/ARM"

# GCC ARM
GCC_ARM_PATH = "C:/Program Files (x86)/GNU Tools ARM Embedded/6 2017-q2-update/bin"

# IAR
#IAR_PATH = "C:/Program Files (x86)/IAR Systems/Embedded Workbench 6.5/bin"

# Goanna static analyser
#GOANNA_PATH = "C:/Program Files (x86)/Reduzard/Goanna Central 0.2.0/bin"

#####
```

mbed_settings.py

省略

省略

GCCのフォルダパスを追加

GCC_ARM_PATH = "C:/Program Files (x86)/GNU Tools ARM Embedded/6 2017-q2-update/bin"

```
C:\¥mtemp¥GRMOmronTest> mbed compile -m GR_MANGO -t GCC_ARM --profile debug
```

```
[mbed] Working path "C:\¥mtemp¥GRMOmronTest" (program)
```

```
Building project GRMOmronTest (GR_MANGO, GCC_ARM)
```

```
Scan: GRMOmronTest
```

```
Using ROM regions bootloader, application in this build.
```

```
Region bootloader: size 0x11000, offset 0x50000000
```

-
-
-

ビルドを開始する

mbed compile -m GR_MANGO -t GCC_ARM --profile debug

ビルド完了後、GRMOmron Test.binを書き込めばOK

```

.
.
.
Compile [ 99.7%]: OperationListBase.cpp
Compile [ 99.8%]: PolledQueue.cpp
Compile [ 99.9%]: TaskBase.cpp
Compile [100.0%]: USBPhy_RZ_A2.cpp
Link: GRMOmronTest
Elf2Bin: GRMOmronTest
Merging Regions
  Filling region bootloader with C:\mtemp\GRMOmronTest\bootloader_d_n_d\GR_MANGO_boot.bin
  Filling region application with .\BUILD\GR_MANGO\GCC_ARM-DEBUG\GRMOmronTest_application.bin
Space used after regions merged: 0x154a38
Merging Regions
  Filling region application with .\BUILD\GR_MANGO\GCC_ARM-DEBUG\GRMOmronTest_application.bin
Space used after regions merged: 0x143a38
```

Module	.text	.data	.bss
2JCIE-EV01\BARO_2SMPB_02E.o	2264(+2264)	0(+0)	0(+0)

省略

[fill]	36(+36)	12(+12)	108(+108)
[lib]c.a	26712(+26712)	2472(+2472)	89(+89)
mbed-os\targets	26720(+26720)	54(+54)	93(+93)
Subtotals	449140(+449140)	3296(+3296)	1358396(+1358396)

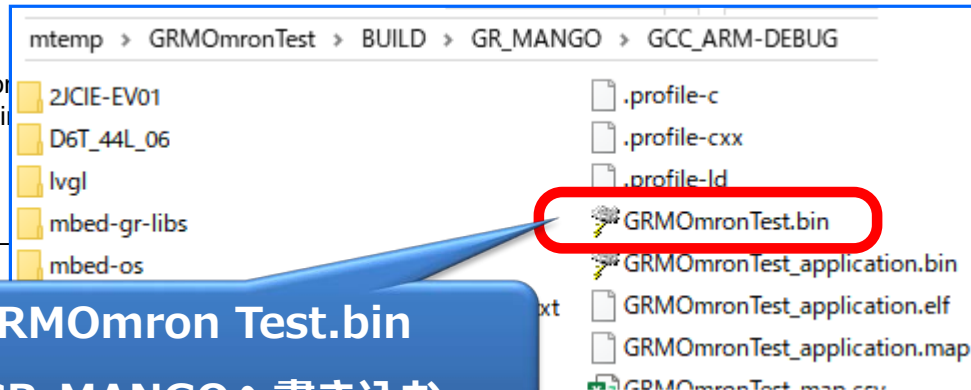
Total Static RAM memory (data + bss): 1361692(+1361692) bytes

Total Flash memory (text + data): 452436(+452436) bytes

```
Update Image: .\BUILD\GR_MANGO\GCC_ARM-DEBUG\GRMOmronTest_application.bin
Image: .\BUILD\GR_MANGO\GCC_ARM-DEBUG\GRMOmronTest.bin
```

```
C:\mtemp\GRMOmronTest>
```

ビルドが完了する



GRMOmron Test.bin
をGR-MANGOへ書き込む