

MicroPython で遊ぶ (2014/9)

- ・既存の組み込み Python と違い、マイコン上でスクリプトのコンパイルまでやってしまうらしいスグレモノ。

昨今話題

<http://www.slideshare.net/hktechno/micro-python-python>

メインサイトはここ

<http://micropython.org/>

ソースそのほかはここ

<https://github.com/micropython>

安価な Cortex-M4 評価基板 (STM32F4 Discovery) で動作する

<https://github.com/micropython/micropython/wiki/Board-STM32F407-Discovery>

必要な環境 (@Windows)

- 1.MinGW,msys をインストール (Unix 式 make のため。Cygwin や他の互換 make でもよいと思われる)
- 2.CPython をインストール (生成スクリプト用)
- 3.git をインストール (なくてもビルドできるが、バージョン生成に使っている)
- 4.arm ツールチェーンを インストール

推奨ツールチェーンはここ

<https://launchpad.net/gcc-arm-embedded>

STM32F4 Discovery 用にビルドする手順 (@Windows)

- 1.arm ツールチェーンの bin フォルダにある gccvar.bat を改変して、CPython(python.exe) と make にパスを通す
2. 改変した gccvar.bat でシェルを起動
- 3.micropython-master を展開したフォルダの下にある stmhal フォルダに移動
- 4.make BOARD=(ボード名) でビルドできる .

```
make BOARD=STM32F4DISC
```

で STM32F4 Discovery 用イメージができる

気づいた点 (2014/9/27 時点)

- 1.Windows で USB-CDC の応答が時折なくなる。Ctrl-D は効くので、片道つまる? Mac OS X では問題なく動作する。

おそらくこの問題

<https://github.com/micropython/micropython/issues/587>

とりあえず TTL-Serial をコンソールにする (STM32F4DISC, Ver1.3.3)

MicroPython サンプル集

<https://github.com/dhylands/upy-examples>

にある、boot.py を USB-MSC ドライブに上書きコピーして、

```
def init():
    if False: < - ここを False から True に
        uart = pyb.UART(6,115200)
        pyb.repl_uart(uart)
        print("REPL is also on UART 6 (Y1=Tx Y2=Rx)")
```

上記の箇所を False から True にしておくと、
UART6 がコンソールとして使えます。

STM32F4DISC では、UART6 は PC6(TX),PC7(RX) となっていて、
USB コネクタ (STM32 本体の) の右、GND のすぐそばに出ているので、
ここに TTL-Serial ドングルを接続するとコンソールとして使えます。

とりあえず builtin モジュールをでっちあげるには (ver 1.3.3)

まだまだダイナミックに変更があると思いますのでメモ程度に。

1) モジュールファイルを書いて別フォルダにまとめる

モジュールファイルのサンプルは、extmod のものがわかりやすいと思います。
とりあえず説明用のダミーモジュール を \$(TOP)/testmod に置きます。

2) ヘッダとシンボルとの結合を書く

< プラットフォーム非依存モジュールの場合 >

py/builtin.h のそれっぽい場所に

```
// extmod modules
extern const mp_obj_module_t mp_module_test;
```

py/builtintables.c のそれっぽい場所に

```
// extmod modules
#if MICROPY_PY_TESTMOD
    { MP_OBJ_NEW_QSTR(MP_QSTR_testmod), (mp_obj_t)&mp_module_test },
#endif
```

< プラットフォーム依存の場合 >

stmhal/mpconfigport.h のそれっぽい場所に

```
extern const struct _mp_obj_module_t mp_module_test;

#define MICROPY_PORT_BUILTIN_MODULES ¥
(中略)
{ MP_OBJ_NEW_QSTR(MP_QSTR_testmod), (mp_obj_t)&mp_module_test }, ¥
```

3) Makefile の修正

それっぽい場所を探して

シンボル、ハッシュ (qstr) 関連の挿入

```
# qstr definitions (must come before including py.mk)
QSTR_DEFS = qstrdefsport.h $(BUILD)/pins_qstr.h ../testmod/testqstr.h
```

モジュールのビルド設定を挿入

```
# for test module
MICROPY_PY_TESTMOD = 1   これはいんちき。
ifeq ($(MICROPY_PY_TESTMOD),1)
TESTMOD_DIR=testmod
INC += -I$(TOP)/testmod
CFLAGS_MOD += -DMICROPY_PY_TESTMOD=1
SRC_MOD += $(addprefix $(TESTMOD_DIR)/,¥
    modtest.c ¥
)
endif
```

4) 試してみる

```
Micro Python inchiki on 2014-10-05; F4DISC with STM32F407
Type "help()" for more information.
>>> import testmod
>>> dir(testmod)
['_name_', 'hello1']
>>> testmod.hello1([1,2,3])
'[1, 2, 3]'
>>>
```

TI の CC3000 WLAN モジュールを使ってみる (ver 1.3.3 , 2014/10/12)

・ビルド

```
make V=1 BOARD=STM32F4DISC MICROPY_PY_CC3K=1
```

・こんな感じです

```
>>> import network
>>> dir(network)
['_name_', 'CC3k', 'route']
>>> dir(network.CC3k)
['connect', 'disconnect', 'is_connected', 'ifconfig', 'patch_version', 'patch_program', 'WEP', 'WPA', 'WPA2']
### CC3k(spi, pin_cs, pin_en, pin_irq)
### B13=SCK, B14=MISO, B15=MOSI
>>> a = network.CC3k(pyb.SPI(2), pyb.Pin.cpu.A15, pyb.Pin.cpu.B10, pyb.Pin.cpu.B11)
```

```
# なぜか E15 ではダメ
#>>> a = network.CC3k(pyb.SPI(2), pyb.Pin.cpu.E15, pyb.Pin.cpu.B10, #pyb.Pin.cpu.B11)

>>> dir(a)
['connect', 'disconnect', 'is_connected', 'ifconfig', 'patch_version', 'patch_program', 'WEP',
'WPA', 'WPA2']

>>> a.patch_version()
(1, 24)

>>> a.is_connected()
False

>>> a.connect('SSID', 'password')

>>> a.is_connected()
True

>>> a.ifconfig()
IP:192.168.1.16
Mask:255.255.255.0
GW:192.168.1.1
DHCP:192.168.1.1
DNS:192.168.1.1
MAC:ho:ge:ho:ge:fu:ga
SSID: SSID

>>>
```