

節子でWi-Fi!

 **SWITCHSCIENCE**

TSUBOI Yoshihiro (@ytsuboi)

会社紹介

SWITCHSCIENCE

- ・ 株式会社スイッチサイエンス
- ・ 2008年に設立
- ・ 社員だいたい15名



<https://www.switch-science.com/>
@ssci

自己紹介



坪井義浩 (つぼいよしひろ) @ytsuboi
薬屋、ときどきスイッチサイエンス
Make:は趣味



2009



2010-2015



2015-



節子とは…

ホーム 通知 メッセージ キーワード検索

クマーお (´°`)
@tedd_okano フォロー中

節子、ここアームちゃう。ロームや

19 リツイート 25 いいね

ARM [user avatars]

11:47 - 2013年9月15日

クマーお (´°`)
@tedd_okano フォローされています
シャケ・ドングリ・ハチミツ・ビール・マヨネーズ / 鮭漁師@天の川 / リットン餃子団員 / 日本マヨネーズ党(日マ党)党员 / RPN計算機万歳!
2010年1月に登録

アームとローム

ARMが入居しているビル

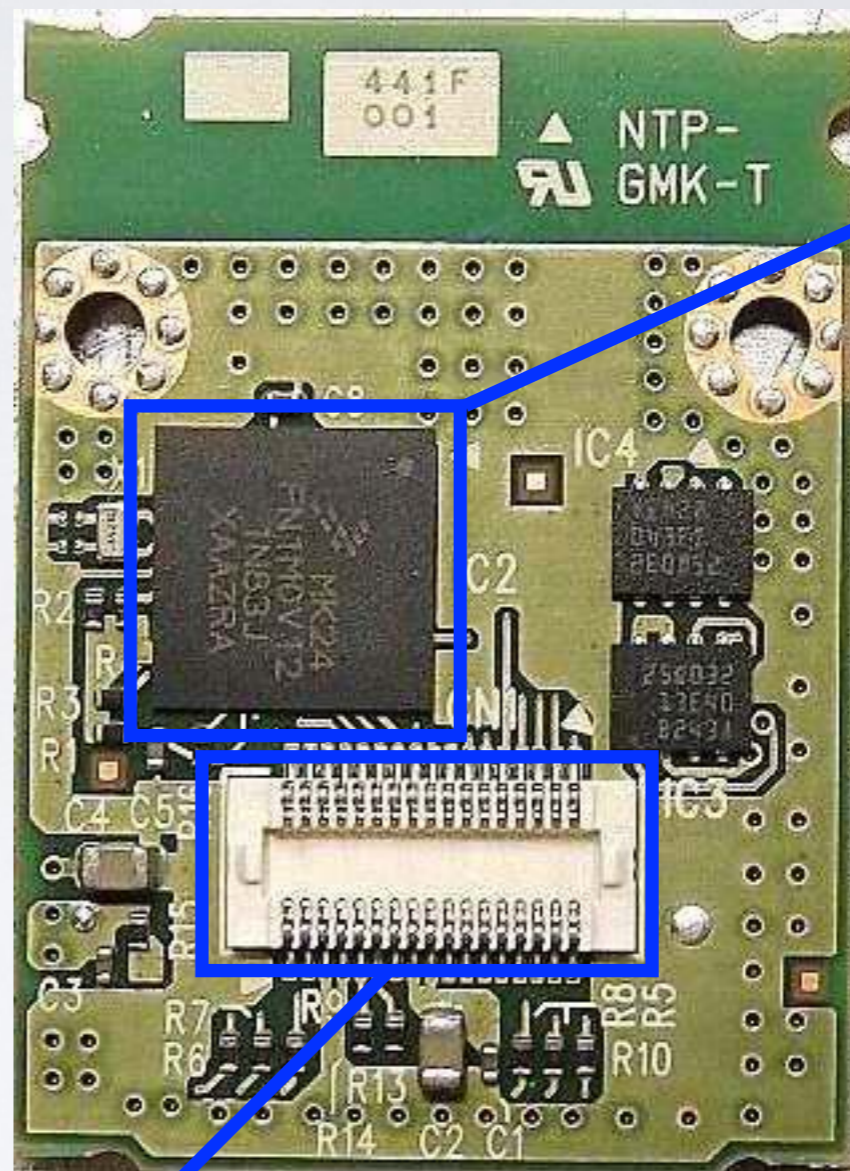


<https://goo.gl/maps/asbmnRscmPw>

BP359B



BP3580



BP359I 下位互換

!!!!!!?

mbed with Wi-Fi

Filter

mbed Enabled

mbed Enabled

mbed OS support

mbed OS 2

mbed OS 5

Target vendor

ARM

Atmel

Connectivity

Bluetooth Smart

CAN

Cellular

Ethernet

USB Device

USB Host

Wifi

Boards

Showing 1 of 103 ([Show all](#))



Delta DFCM-NNN40

- WiFi and Bluetooth Low Ener
- Cortex-M0, 16MHz
- 256KB Flash, 32KB RAM

mbed with Wi-Fi / mbed OS 5

Filter

mbed Enabled

mbed Enabled

mbed OS support

mbed OS 2

mbed OS 5

Target vendor

ARM

Atmel

Connectivity

Bluetooth Smart

CAN

Cellular

Ethernet

USB Device

USB Host

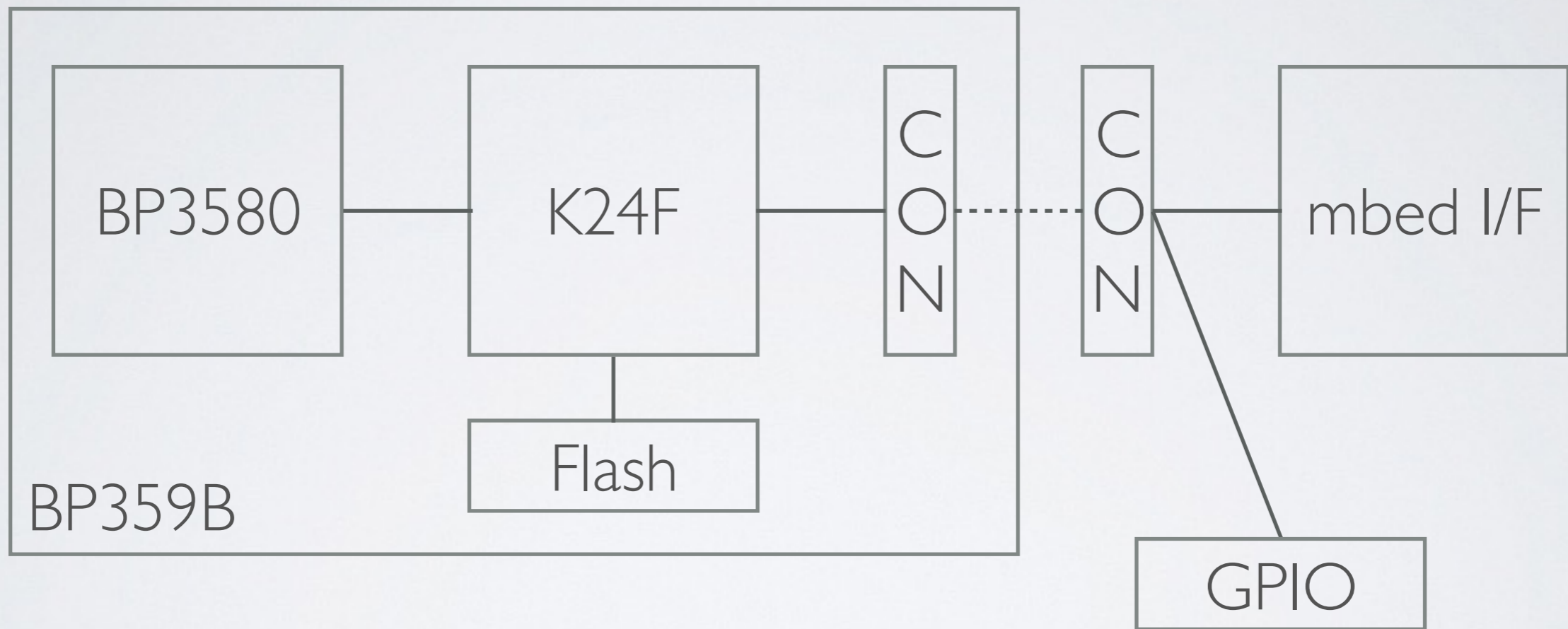
Wifi

Boards

Showing 0 of 103 ([Show all](#))



つまり...



いっすね、それ。

今度のmbed祭りでデモってくださいよ

@2016/9/5



This repository Search

Pull requests Issues Gist



ARMmbed / mbed-os

Watch 166

Unstar 765

Fork 758

Code

Issues 286

Pull requests 42

Projects 0

Wiki

Pulse

Graphs

Branch: master

Create new file

Upload files

Find file

History

mbed-os / hal / targets / hal / TARGET_Freescale / TARGET_KSDK2_MCUS /

mikaleppanen Corrected K66F and K64F drivers to make "or" operation instead of "an... Latest commit 2bad43d 13 days ago


..

TARGET_K22F	KSDK serial_api.c: Fix assertion error for ParityEven	14 days ago
TARGET_K66F	Corrected K66F and K64F drivers to make "or" operation instead of "an...	a day ago
TARGET_KL27Z	KSDK serial_api.c: Fix assertion error for ParityEven	14 days ago
TARGET_KL43Z	KSDK serial_api.c: Fix assertion error for ParityEven	14 days ago
TARGET_MCU_K64F	Corrected K66F and K64F drivers to make "or" operation instead of "an...	a day ago
api	HAL TRNG - add dummy variable to empty structs	2 days ago



NXP Kinetis Expert MK24FN1M0xxx12 (MK24FN1M0xxx12) English ytsuboi@switch-science.com

SDK Builder Tools Software Vault Configurations



MK24FN1M0xxx12 (device)

MK24FN1M0xxx12

Included Part Numbers
MK24FN1M0VDC12, MK24FN1M0VLL12, MK24FN1M0VLQ12, MK24FN1M0CAJ12

Board(s)
TWR-K64F120M, FRDM-K64F

Device
MK24F12

Core Type
Cortex-M4F

Memory Size
1024 KB Flash
256 KB RAM

Maximum CPU Frequency
120 MHz

Kinetis SDK

Kinetis SDK for the selected configuration will include:

- Kinetis MCU platform support
- Demo applications and driver examples
- Documentation - SDK API reference manual and user guides

The following optional items can be included:

Middleware

CMSIS DSP library FatFS lwIP USB stack

Operating systems

FreeRTOS µC/OS-II µC/OS-III

Your custom version of the Kinetis SDK is now ready to be packaged! Click the button below to complete the process.

Package name SDK_2.0_MK24FN1M0xxx12 **SDK version** SDK 2.0 **Supported toolchain(s)** Kinetis Design Studio **Host OS** Mac

[Build SDK Package](#) [SDK API Documentation v2.0](#)

Other Software Libraries

Additional Kinetis software products that may be of interest:

[Kinetis Design Studio \(KDS\)](#) [SDK Project Generator](#) [WolfSSL](#) [mbedTLS](#)

[Feedback](#)

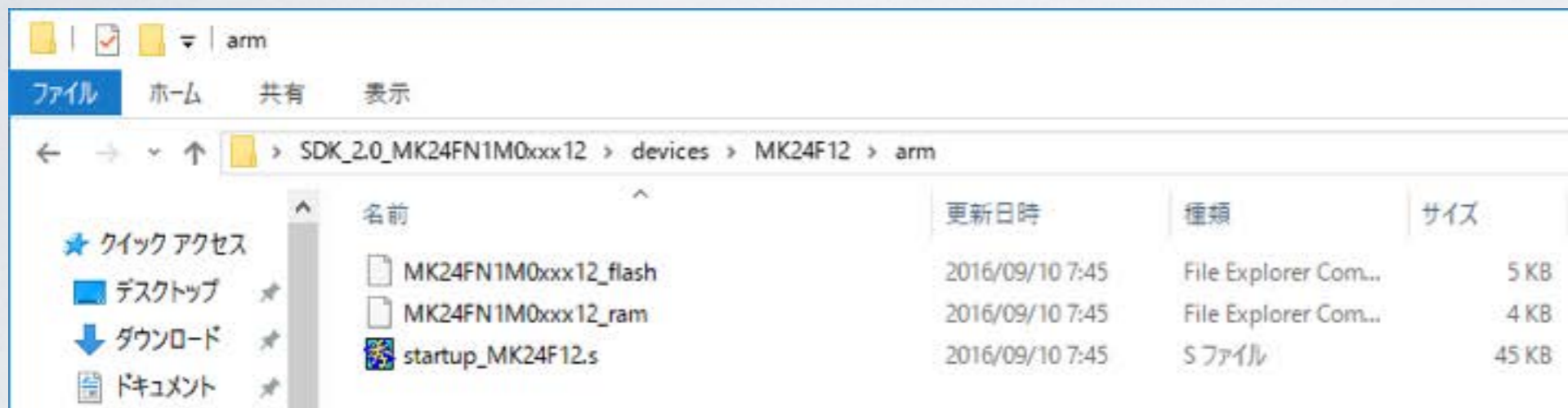
NXP Privacy Policy | Terms of Use | Contact © 2016 NXP Semiconductors. All rights reserved.

https://kex.nxp.com/

ハマったところ、その(1)

スキヤッタ…リンクに渡すイメージのメモリマップ

```
Error: L6218E: Undefined symbol Image$$RW_IRAM1$$ZI$$Limit (referred from sys.o).
Error: L6218E: Undefined symbol Image$$RW_m_data$$Base (referred from fsl_common.o).
Error: L6218E: Undefined symbol Image$$VECTOR_RAM$$Base (referred from fsl_common.o).
Error: L6218E: Undefined symbol Image$$VECTOR_ROM$$Base (referred from fsl_common.o).
```



SDK_2.0_MK24FN1M0xxx12\devices\MK24F12\arm のディレクトリ

2016/09/10	07:45	4,880	MK24FN1M0xxx12_flash.scf
2016/09/10	07:45	3,891	MK24FN1M0xxx12_ram.scf
2016/09/10	07:45	45,273	startup_MK24F12.s

「登録されている拡張子は表示しない」のチェックを外しても表示されない

ハマったところ、その(2)

無い！

NXP Kinetis Expert

Tools

- SDK Builder
- Tools
- Pin
- Clocks**
- Power
- Peripherals

Kinetis

Kinetis SDK

Kinetis SDK for the selected confi

- > Kinetis MCU platform support
- > Demo applications and driver ex
- > Documentation - SDK API refere

The following optional items can be included:

Middleware

- CMSIS DSP library
- FatFS

Operating systems

- FreeRTOS
- μ C/OS-II

MK24FN1M0xxx12 (device)

Included Part Numbers
MK24FN1M0VDC12, MK24FN1M0VLL12,
MK24FN1M0VLQ12, MK24FN1M0CAJ12

Board(s)
TWR-K64F120M, FRDM-K64F

Device
MK24F12

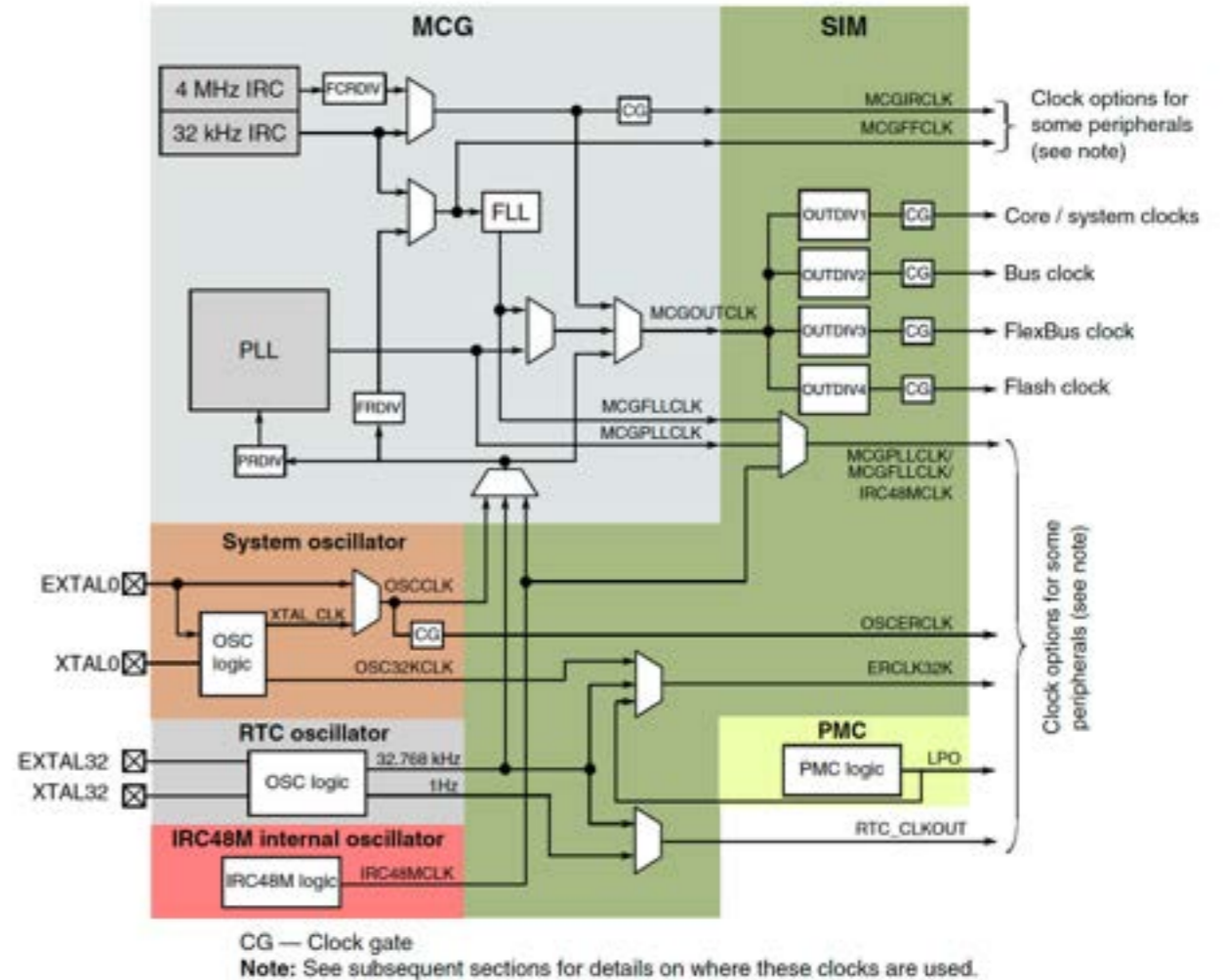
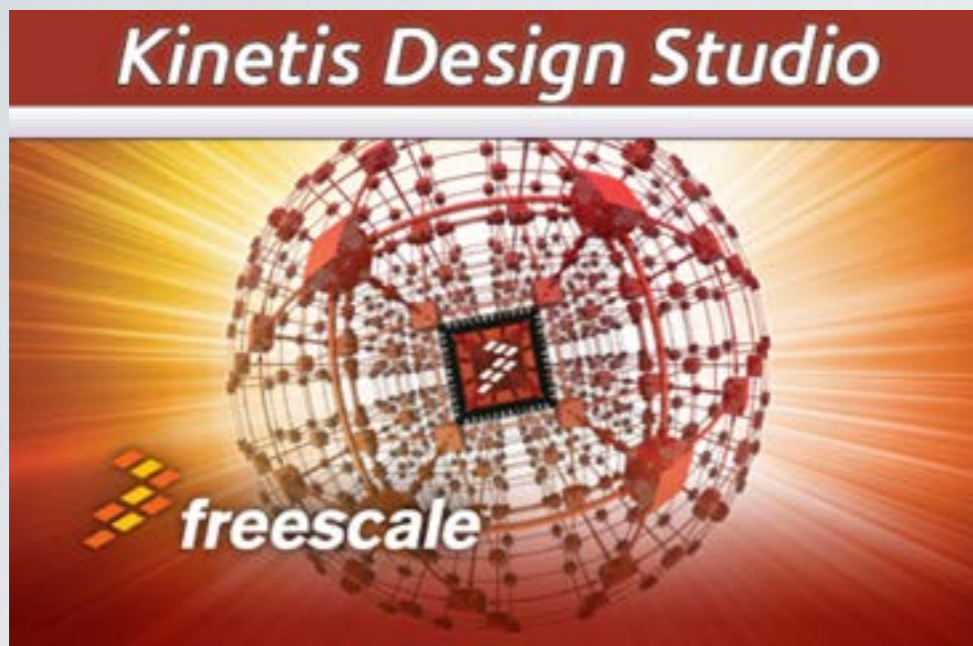


Figure 5-1. Clocking diagram

Chapter 5: Clock Distribution 34page
Chapter 25: MCG 36page
Chapter 12: SIM 38page

ハマったところ、その(2)



*Component Inspector - Cpu

Properties Methods Events Build options Resources

type filter text

- All
 - Clock settings
 - Clock sources
 - Clock configurations
 - Low power mode settings
 - Allowed power modes
 - Watchdog settings
 - Pins/Signals
 - Reset pin
 - Debug interface (JTAG)

Details for selected row:

Clock configuration 0

Configuration description Default part configuration

Internal reference clock External reference clock **MCG settings** Very low power mode System clocks

MCG mode FEI

MCG output clock	FLL clock	
MCG output [MHz]	95.977472	95.977472 MHz
MCG external ref. clock source	IRC 48MHz	
MCG external ref. clock [MHz]	48	48 MHz
Clock monitor	<input type="checkbox"/>	

FLL settings

FLL module	<input checked="" type="checkbox"/>	
FLL output [MHz]	95.977472	95.977472 MHz
MCGFFCLK clock [kHz]	32.768	32.768 kHz

Reference clock source Slow internal clock

FLL reference clock [kHz]	32.768	32.768 kHz
Multiplication factor	Auto select	2929

PLL 0 settings

PLL module	<input type="checkbox"/>	
PLL module in Stop	<input type="checkbox"/>	

KDS 1.3用のみ！

ハマったところ、その(2)

fsl_clock_config.c

```
const clock_config_t g_defaultClockConfigRun = {
    .mcgConfig =
    {
        mcgMode = kMCG_ModePEE,
        irclkEnableMode = kMCG_IrclkEnable,
        ircs = kMCG_IrcSlow,
        fcrdiv = 0U,
        frdiv = 7U,
        drs = kMCG_DrsLow,
        dmx32 = kMCG_Dmx32Default,
        oscsel = kMCG_Oscsel0sc,
        pll0Config =
        {
            .enableMode = 0U, .prdiv = 0x13U, .vdiv = 0x18U,
        },
    },
    .simConfig =
    {
        .pllFl1Sel = 1U,
        .er32kSrc = 2U,
        .clkdiv1 = 0x01140000U,
        .oscConfig = { .freq = BOARD_XTAL0_CLK_HZ,
            .capLoad = 0,
            .workMode = kOSC_ModeExt,
            .oscerConfig =
            {
                .enableMode = kOSC_ErClkEnable,
            }
        },
        #if (defined(FSL_FEATURE_OSC_HAS_EXT_REF_CLOCK_DIVIDER) &&
            FSL_FEATURE_OSC_HAS_EXT_REF_CLOCK_DIVIDER)
            .erclkDiv = 0U,
        #endif
    },
    .coreClock = 120000000U,
};
```

半分

system_LPC11Uxx.c

```
#define CLOCK_SETUP 1
#define SYSOSCCTRL_Val 0x00000000
#define WDTOSCCTRL_Val 0x00000000
#define SYSPLLCTRL_Val 0x00000023
#define SYSPLLCLKSEL_Val 0x00000001
#define MAINCLKSEL_Val 0x00000003
#define SYSAHBCLKDIV_Val 0x00000001
#define USBPLLCTRL_Val 0x00000023
#define USBPLLCLKSEL_Val 0x00000001
#define USBCLKSEL_Val 0x00000000
#define USBCLKDIV_Val 0x00000001
```

```
void BOARD_BootClockRUN(void)
{
```

```
    CLOCK_SetSimSafeDivs();
    CLOCK_InitOsc0(&g_defaultClockConfigRun.oscConfig);
    CLOCK_SetXtal0Freq(BOARD_XTAL0_CLK_HZ);
```

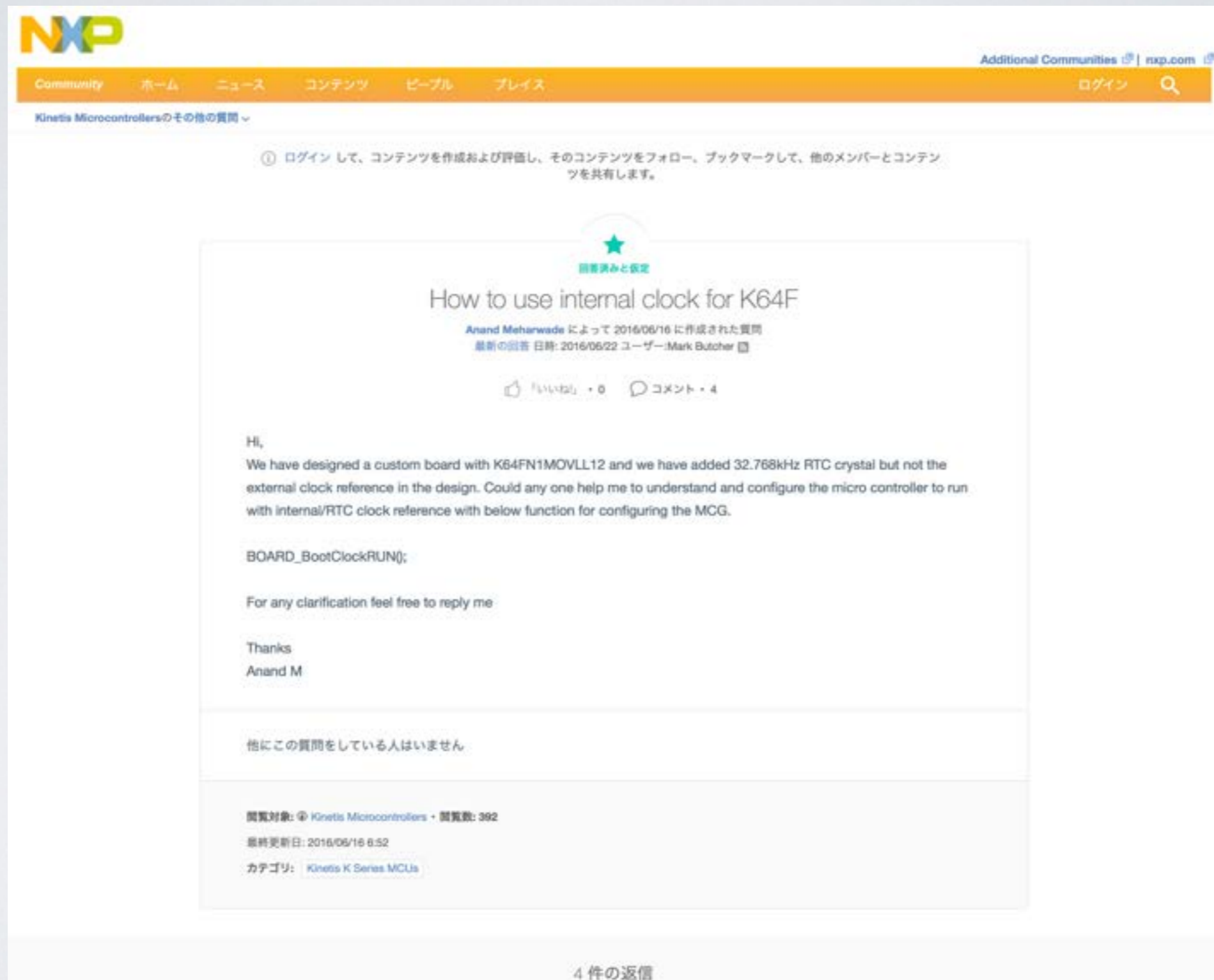
```
    CLOCK_BootToPeeMode(g_defaultClockConfigRun.mcgConfig.oscsel,
        kMCG_PllClkSelPll0,
        &g_defaultClockConfigRun.mcgConfig.pll0Config);
```

```
    CLOCK_SetInternalRefClkConfig(g_defaultClockConfigRun.mcgConfig.irclkEnableMode,
        g_defaultClockConfigRun.mcgConfig.ircs,
        g_defaultClockConfigRun.mcgConfig.fcrdiv);
```

```
    CLOCK_SetSimConfig(&g_defaultClockConfigRun.simConfig);
    SystemCoreClock = g_defaultClockConfigRun.coreClock;
```

```
}
```

ちょっとフォロー



The screenshot shows a forum post on the NXP community website. The post is titled "How to use internal clock for K64F" and was created by Anand Meharwade on 2016/06/16. It has 0 "いいね" (likes) and 4 comments. The post content asks for help in configuring the MCG of a K64FN1MOVLL12 microcontroller to use an internal/RTC clock reference instead of an external one. The code snippet provided is `BOARD_BootClockRUN();`. The post is categorized under "Kinetis K Series MCLUs" and has 392 views.

NXP Additional Communities | nxp.com

Community ホーム ニュース コンテンツ ビーブル プレイス ログイン

Kinetis Microcontrollersのその他の質問

① ログインして、コンテンツを作成および評価し、そのコンテンツをフォロー、ブックマークして、他のメンバーとコンテンツを共有します。

回答済みと仮定

How to use internal clock for K64F

Anand Meharwade によって 2016/06/16 に作成された質問
最新の回答 日時: 2016/06/22 ユーザー: Mark Butcher

いいね 0 コメント 4

Hi,
We have designed a custom board with K64FN1MOVLL12 and we have added 32.768kHz RTC crystal but not the external clock reference in the design. Could any one help me to understand and configure the micro controller to run with internal/RTC clock reference with below function for configuring the MCG.

```
BOARD_BootClockRUN();
```

For any clarification feel free to reply me

Thanks
Anand M

他にこの質問をしている人はいません

閲覧対象: Kinetis Microcontrollers • 閲覧数: 392
最終更新日: 2016/06/16 6:52
カテゴリ: Kinetis K Series MCLUs

4 件の返信

KEIL™
Tools by ARM

ULINK

RUN

COM

SB



C:\mbed\test-k64f\test-k64f\uvprojx - uVision

File Edit View Project Flash Debug Peripherals Tools SVCS Window Help

Registers Disassembly System and Thread Viewer

Register	Value
R0	0x07270E00
R1	0x20000074
R2	0x8000B010
R3	0x40047000
R4	0x00000000
R5	0x00000000
R6	0x00000000
R7	0x00000000
R8	0x00000000
R9	0x00000000
R10	0x00000000
R11	0x00000000
R12	0x00000000
R13 (SP)	0x20002100
R14 (LR)	0x00003467
R15 (PC)	0x00005640
xPSR	0x21000000

```

0x0000563E BD10 POP (r4,pc)
13: while (true) {
0x00005640 E016 B 0x00005670
14: led1 = !led1;
0x00005642 4D0C LDR r5,[pc,#48] ; @0x00005674
0x00005644 462C MOV r4,r5
117: return read();

```

```

1 #include "mbed.h"
2 #include "rtos.h"
3
4 //DigitalOut led1(PTB2);
5 DigitalOut led1(PTB2);
6 //Serial pc(USBTX, USBRX);
7
8 // main() runs in its own thread in the OS
9 // (note the calls to Thread::wait below for delays)
10 int main() {
11 // pc.baud(115200);
12 // pc.printf("hello, mbed OS\n");
13 while (true) {
14 led1 = !led1;
15 Thread::wait(500);
16 }
17 }
18

```

Property	Value
System	
Item	Value
Tick Timer:	1.000 mSec
Round Robin Timeout:	5.000 mSec
Default Thread Stack Size:	512
Thread Stack Overflow Check:	Yes
Thread Usage:	Available: 15, Used: 2 + os_...

ID	Name	Priority	State	Delay	Event Value	Event Mask	Stack Usage
1	os_thread_def_stack_osTimerThread	High	Wait_MBX				Overflow
2	thread_stack_main	Normal	Running				89%

Command: Load "C:\\mbed\\test-k64f\\build\\uvision5\\test-k64f.axf"

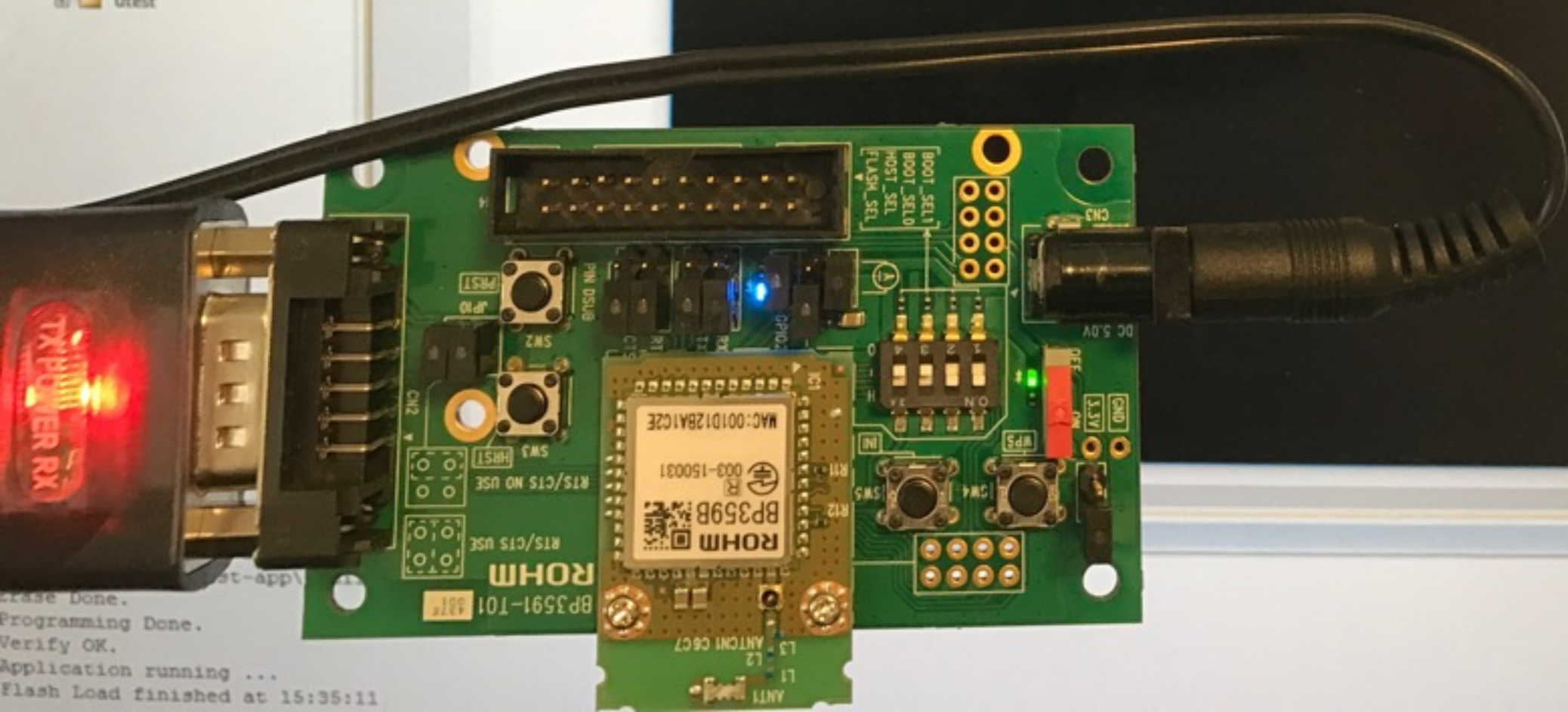
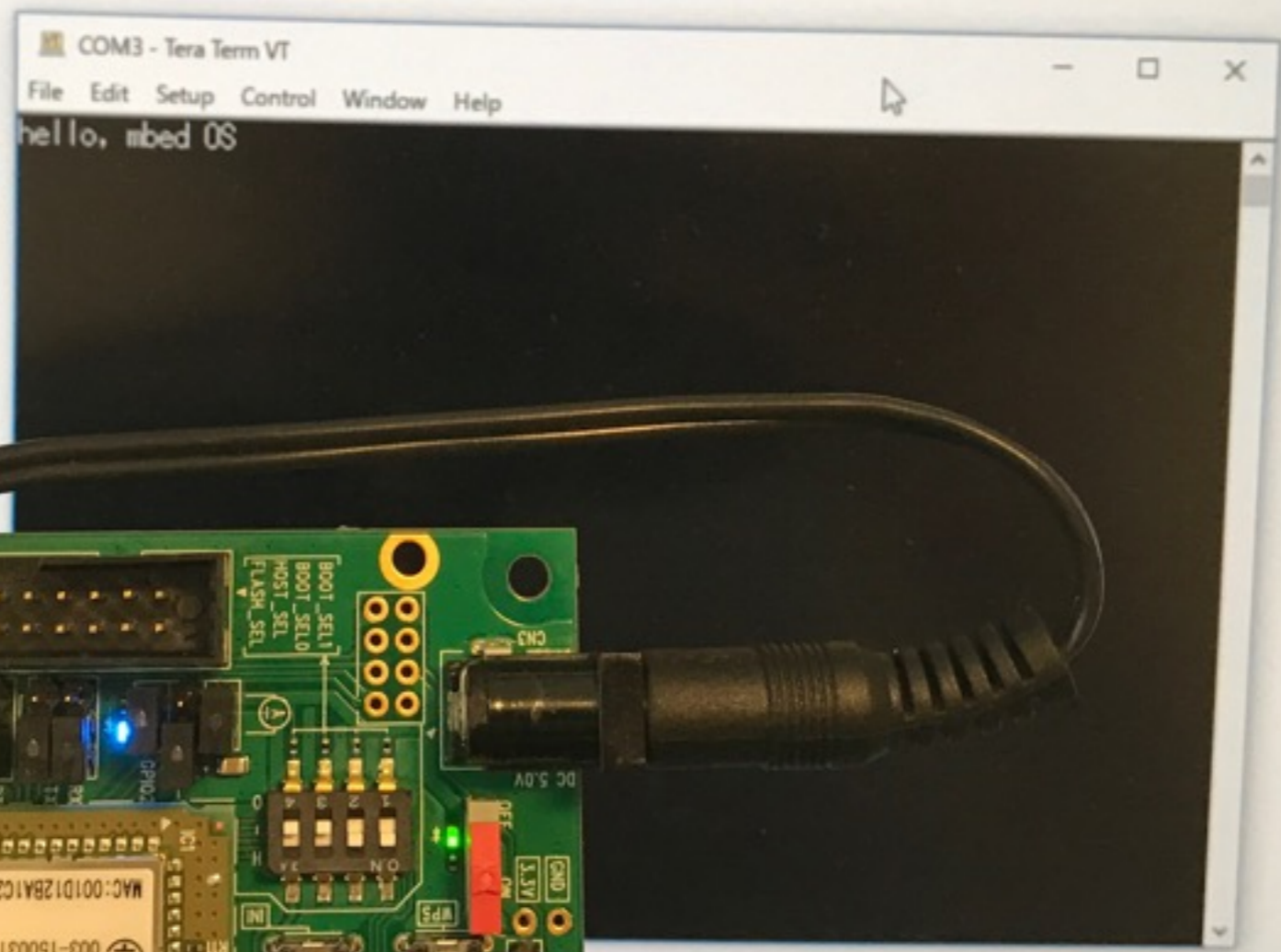
Name	Location/V...	Type
os_t...	0x200007D8	Task
thre...	0x20001188	Task
m...	0x00000000	int f()

ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVERAGE DEFINE DIR Display

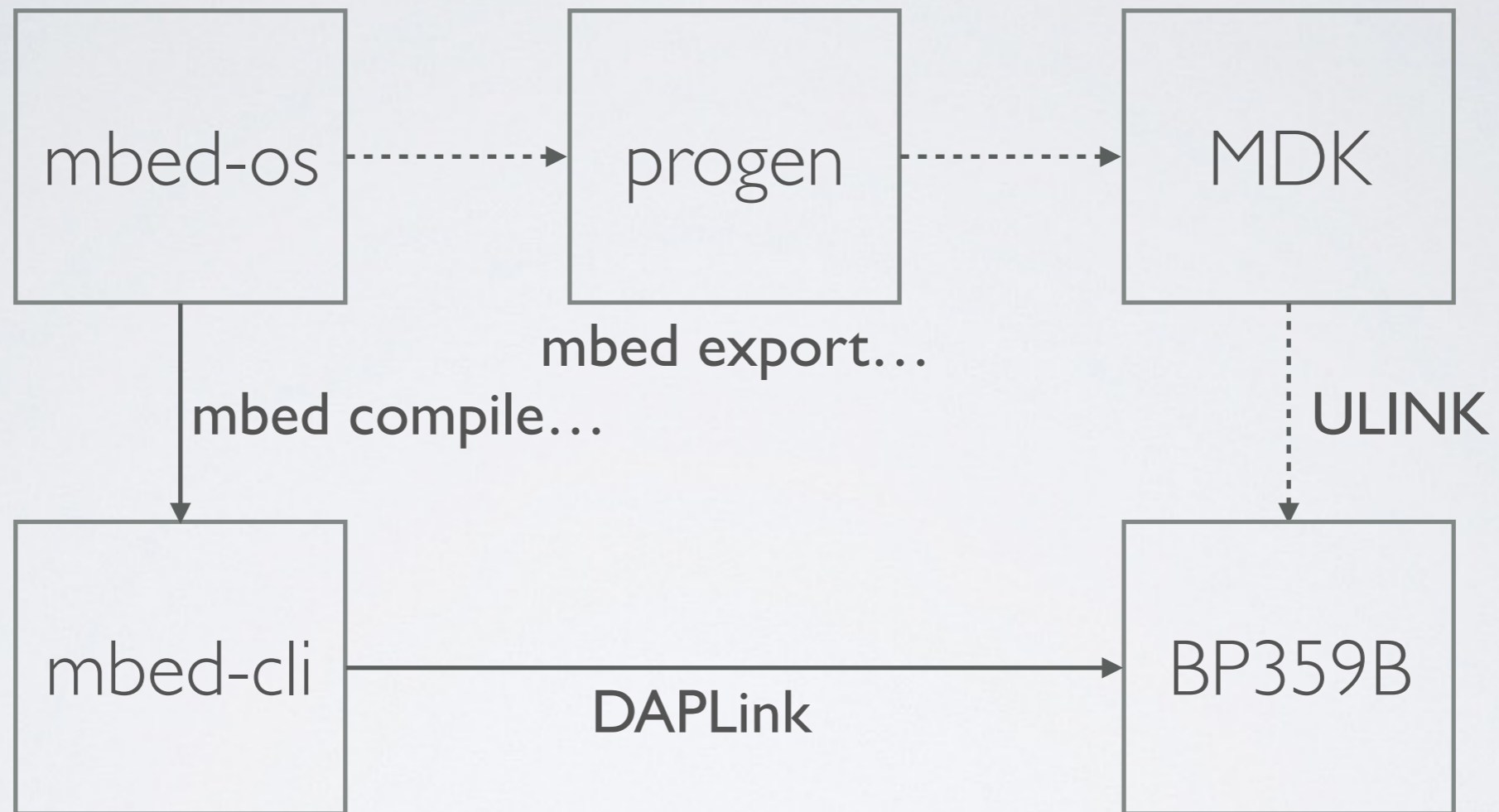
ULINK Pro Cortex Debugger t1: 0.00120534 sec L:13 C:1 CAP: NUM SCRL OVR: R/W

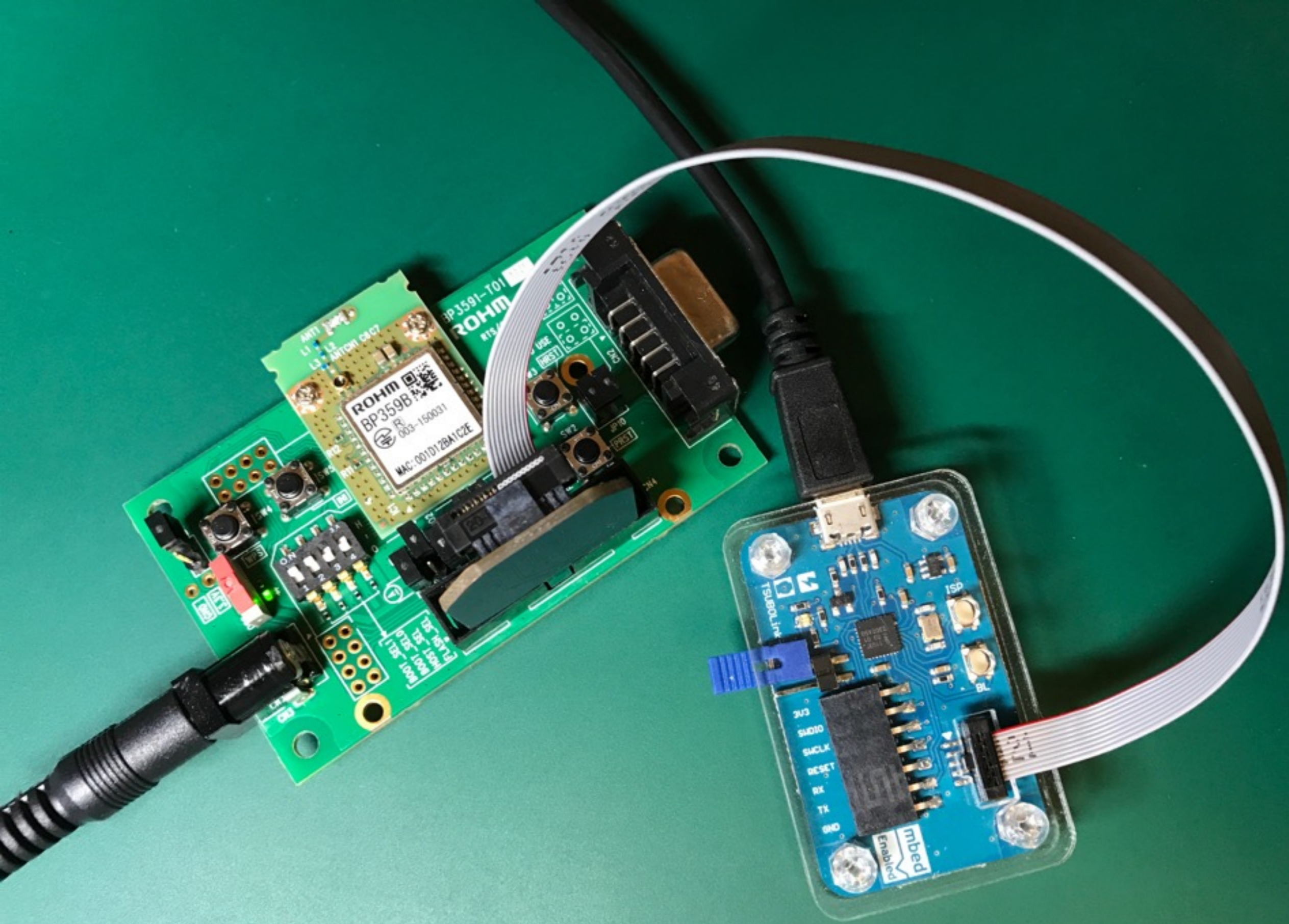
- Project: test-app
- test-app
 - TARGET_BP359B
 - TARGET_CORTEX_M
 - TARGET_MCU_K24F
 - TOOLCHAIN_ARM
 - TOOLCHAIN_ARM_STD
 - api
 - cmsis
 - common
 - drivers
 - greentea-client
 - hal
 - mbedtls
 - network-socket
 - rtos
 - source
 - src
 - storage_abstraction
 - unity
 - utest

```
main.cpp
1 #include "mbed.h"
2 #include "rtos.h"
3
4 //DigitalOut led1(PTB2);
5 DigitalOut led1(PTB2);
6 //Serial pc(USBTX, USBRX);
7 Serial pc(PTA14, PTA15);
8
9 // main() runs in its own thread in the OS
10 // (note the calls to Thread::wait below for delays)
11 int main() {
12     pc.baud(115200);
13     pc.printf("hello, mbed OS\r\n");
14     while (true) {
15         led1 = !led1;
16         Thread::wait(500);
17     }
18 }
19
```



Flash Load finished at 15:35:11
Application running ...
Verify OK.
Programming Done.
Please Done.





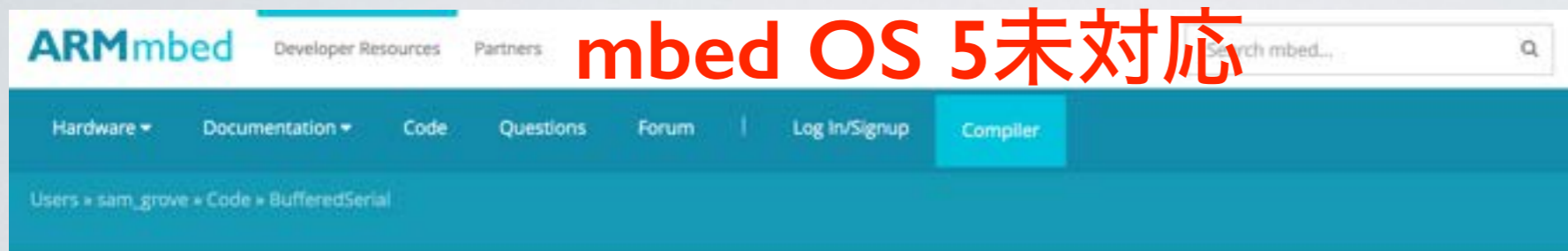
ROHM
BP359B
003-150031
MAC: 001D12BA1C2E

TSUBU.LiNk
Isp
BL
mbed
Enabled

3V3
SWDIO
SWCLK
RESET
RX
TX
GND

ハマったところ、その(3)

mbed OS 5未対応



ARMmbed Developer Resources Partners Search mbed... Hardware Documentation Code Questions Forum Log In/Signup Compiler

Users » sam_grove » Code » BufferedSerial



Sam Grove / BufferedSerial

Inherit from Serial and use software buffers for TX and RX. This allows the UART peripherals to operate. Overrides most (but not all) stdio functions as Serial did.

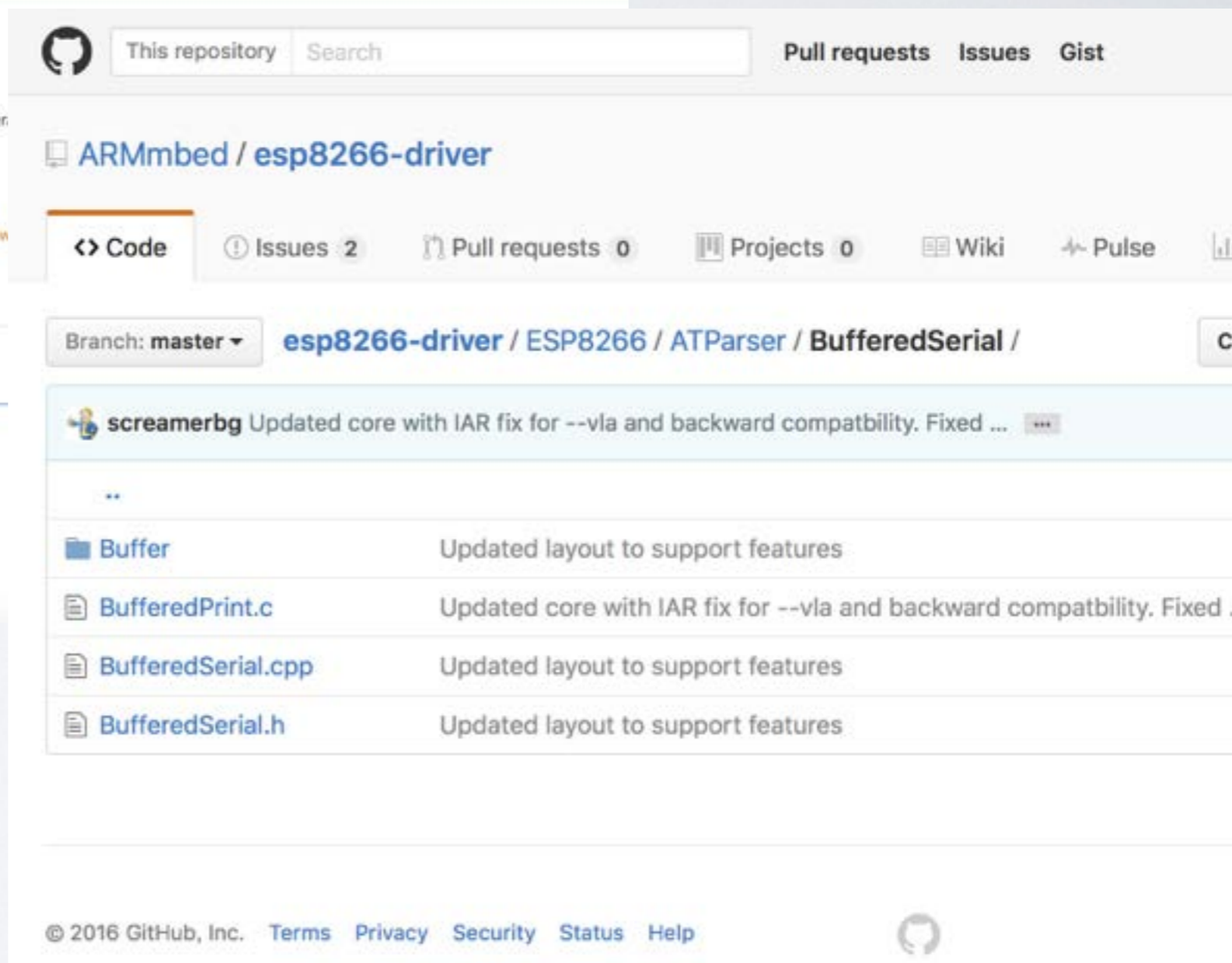
Dependencies: Buffer

Dependents: buffered_serial_test BLE_Police_HRM evena_BLE_Police_HRM df-2014-w... more

Home History Graph API Documentation Wiki Issues Pull Requests

Example

```
1 #include "mbed.h"
2 #include "BufferedSerial.h"
3
4 BufferedSerial pc(USBTX, USBRX);
5
6 int main()
7 {
8     pc.baud(115200);
9
10    while(1)
11    {
12        Timer ar;
13    }
```



This repository Search Pull requests Issues Gist

ARMmbed / esp8266-driver

Code Issues 2 Pull requests 0 Projects 0 Wiki Pulse

Branch: master esp8266-driver / ESP8266 / ATParser / BufferedSerial /

screamerbg Updated core with IAR fix for --vla and backward compatibility. Fixed ...

- Buffer Updated layout to support features
- BufferedPrint.c Updated core with IAR fix for --vla and backward compatibility. Fixed ..
- BufferedSerial.cpp Updated layout to support features
- BufferedSerial.h Updated layout to support features

© 2016 GitHub, Inc. Terms Privacy Security Status Help

Demo

当たります！



せんでん

SWITCHSCIENCE



SWITCHSCIENCE
OPEN SOURCE HARDWARE SHOP

送料0円~500円
3,000円または10,000円以上で送料無料
営業日14時までのお支払いで当日発送

VISA JCB MasterCard
PayPal 銀行振込
代引き

商品を探す 商品名、商品番号など

マイページ ログアウト カート

SWITCHSCIENCE
international store

基板製造サービス
スイッチサイエンス PCB

Category [すべて開く](#)

- › 新商品 (90)
- › SALE(37)
- スイッチサイエンス製品 (229)
- スイッチサイエンスマーケットプレイス (委託商品) (273)
- Rapiro(26)
- MakerBot(66)
- SoarkFun(386)

Keil USB-JTAG Adaptor

Keil ULINK2デバッグアダプタは、PCのUSBポートをターゲットシステムに（JTAG、SWD、またはOCDSを介して）接続して、ターゲットハードウェア上のエンベデッドプログラムのプログラミングおよびデバッグを可能にします。

2016年10月8日更新



名前	Keil USB-JTAG Adaptor
コード番号	ARM-ULINK2

Thanks!