# Dialog SDK 5.x.x 培训教程4–睡眠模式

2016.5

…personal
…portable
…connected

# BLE 睡眠模式

睡眠模式概览

扩展睡眠模式

深度睡眠模式

对单个非易失性存储单元进行掉电操作

结论

# BLE 睡眠模式

## 睡眠模式概览

# BLE 睡眠模式

**The DA1458x 有两种睡眠:**

- 扩展睡眠模式 (请看下一页PPT框图):
  - ➢ 只有42KB的系统RAM和非易失性RAM仍保持上电状态.

- 深度睡眠模式 (请看下一页PPT框图):
  - ➢ 只有非易失性RAM保持上电状态.

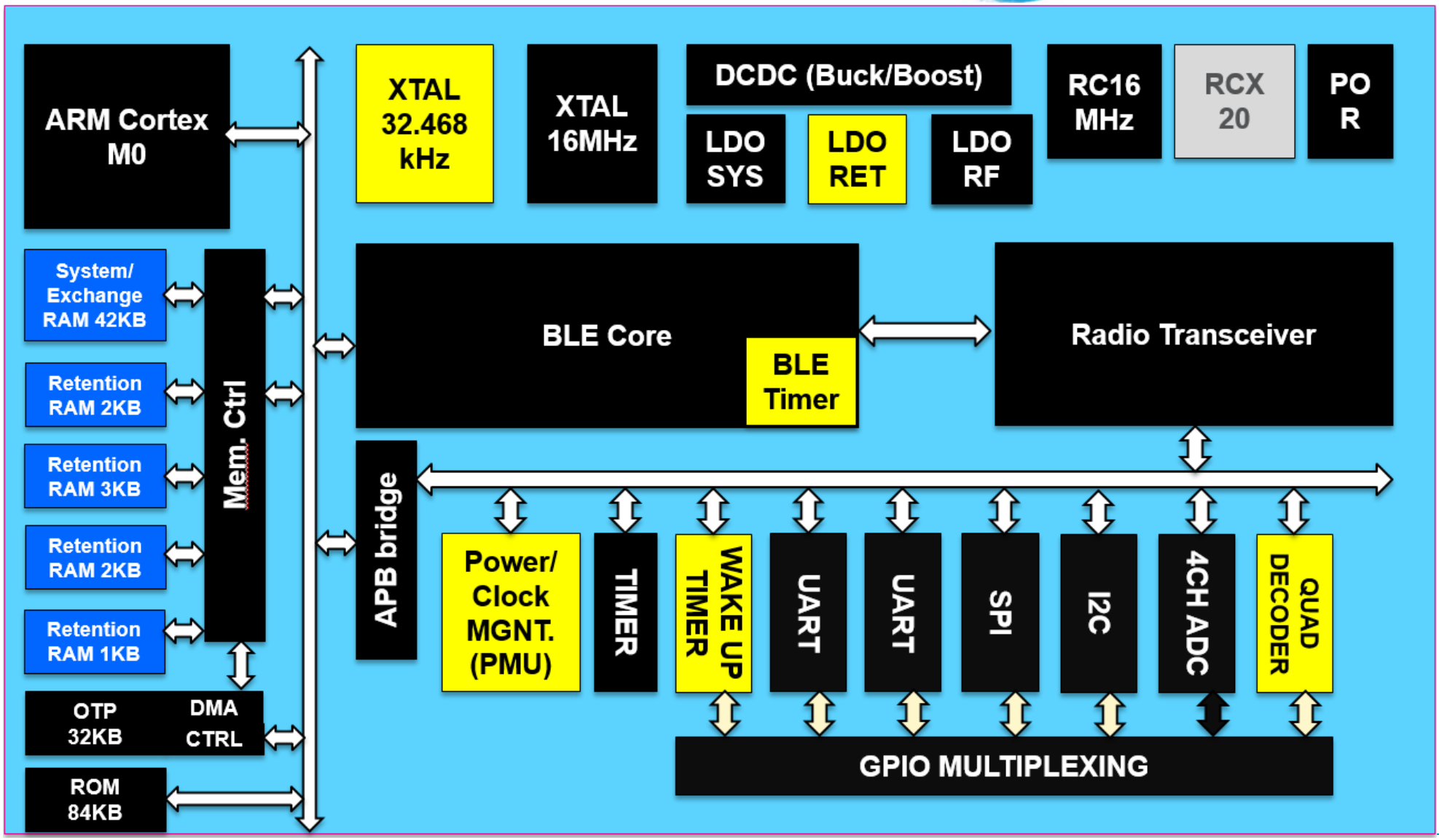
  **注意:** 只有在OTP被烧录程序的状态下，才能检测深度睡眠模式下的电流.

**无论选择哪种睡眠模式，DA1458X都可以被以下两种方式唤醒:**

- 同步模式，BLE定时器可以被编程用来唤醒系统
- 异步模式，通过外部输入中断来唤醒.

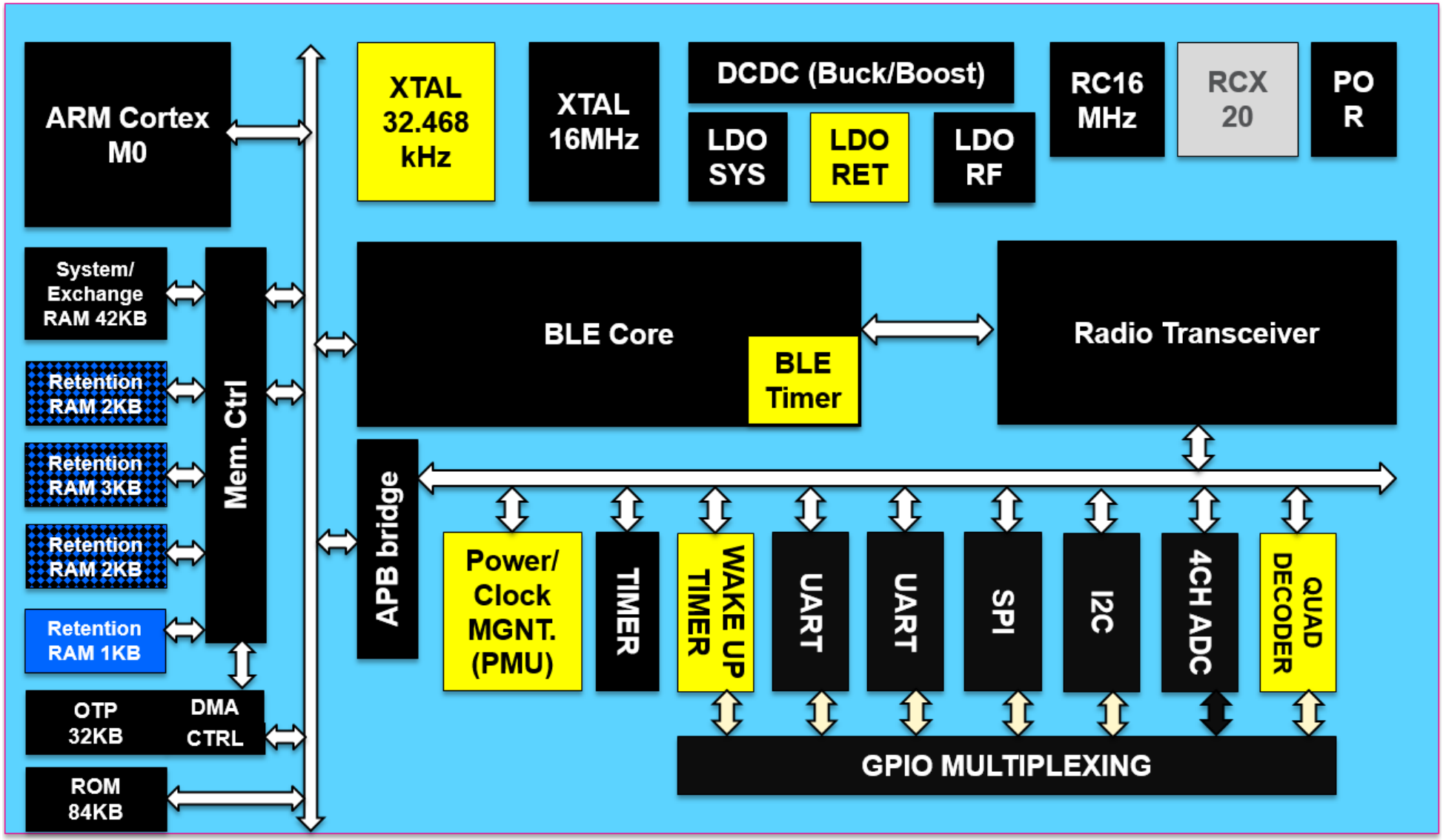
# BLE 睡眠模式

**扩展睡眠模式:**



# BLE 睡眠模式

**深度睡眠模式:**



# BLE 睡眠模式

## 睡眠模式特点:

**1) 外部处理器模式（通过GTL接口）:**

在一段时间内，周期性唤醒去查询GTL接口的流控状态，可以通过如下定义：#define:

```
#define CFG_MAX_SLEEP_DURATION_PERIODIC_WAKEUP_MS 500  // 500 msec
```

默认设定500ms用来唤醒，这个时间被认为对UART接口操作是最优的选择。

最长可以设定的时间为23.3小时，因为定时器变量是27bit，所以最长时间为= 2^27 * 0.625ms(BLE 单个心跳持续时间)

最短可以设定的时间为10ms (The DA1458x 需要5.7ms来唤醒)，这不是一个比较好的选择,只是用来说明周期性唤醒条件下最短可以设定的时间

# BLE 睡眠模式

## 睡眠模式特点:

**2) 内部处理器模式:**

一段用于唤醒DA1458X的周期可以通过以下定义来实现:

```
#define CFG_MAX_SLEEP_DURATION_EXTERNAL_WAKEUP_MS 10000  // 10s
```

The DA1458x 在没有BLE和定时器事件需要处理的情况下，会在之前提到的时间间隔里面醒来（在我们的例子里面是10S）.

最长可以设定的时间为23.3小时。这是因为定时器变量是27bit，所以最长时间为= 2^27 * 0.625ms(BLE 单个心跳持续时间)

最短可以设定的时间为10ms (The DA1458x 需要5.7ms时间来唤醒)，这不是一个比较好的选择,只是用来说明周期性唤醒条件下最短可以设定的时间

在进入睡眠前可以关闭这个唤醒功能，只需要调用API接口: app_ble_ext_wakeup_on();

➔ **这个会关闭所有的BLE事件和周期性事件**

当58x从冬眠模式唤醒后，这个API接口会被调用: app_ble_ext_wakeup_off();

这个流程在Proximity Tag的参考设计里面已经实现，可以参考以下链接:

http://support.dialog-semiconductor.com/connectivity/reference-design/proximity-tag

# BLE 睡眠模式

## 扩展睡眠模式

# BLE 睡眠模式

## 设置扩展睡眠模式

**TODO 1 – 通过以下链接打开proximty reporter工程：**

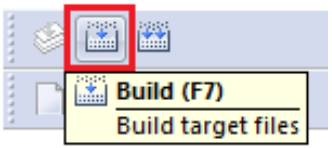
*projects\target_apps\ble_examples\prox_reporter\Keil_5*

**TODO 2 – 在user_config文件夹下打开文件/* @file user_config.h */.**

**TODO 3 – 设定app_default_sleep_mode变量为ARCH_EXT_SLEEP_ON：**

```
const static sleep_state_t app_default_sleep_mode = ARCH_EXT_SLEEP_ON;
```

**TODO 4 – 通过点击Build按钮，编译工程 ：**



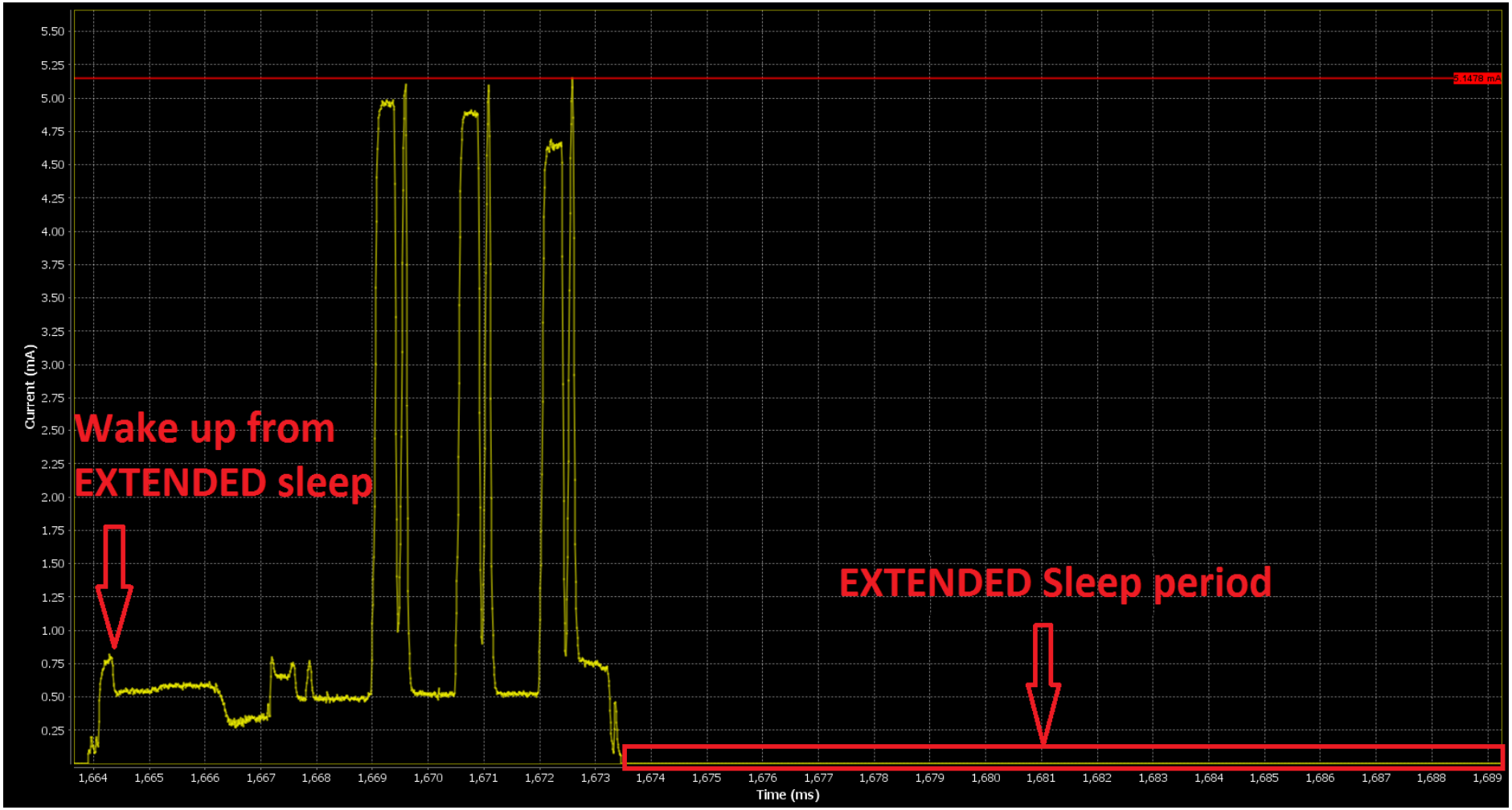

**TODO 5 – 将PRO或者BASIC开发板连接PC.**

**TODO 6 – 点击“开始调试”按钮.** 如果再次点击，则会退出调试状态，并让DA1458X开始运行。

# BLE 睡眠模式

**TODO 6** - 打开SmartSnippets工具(可以在如下链接下载到: http://support.dialog-semiconductor.com/

你会看到:



# BLE 睡眠模式

## 测量扩展睡眠模式

**TODO 1** – 在user_config文件下打开文件 /* @file **user_config.h** */ .

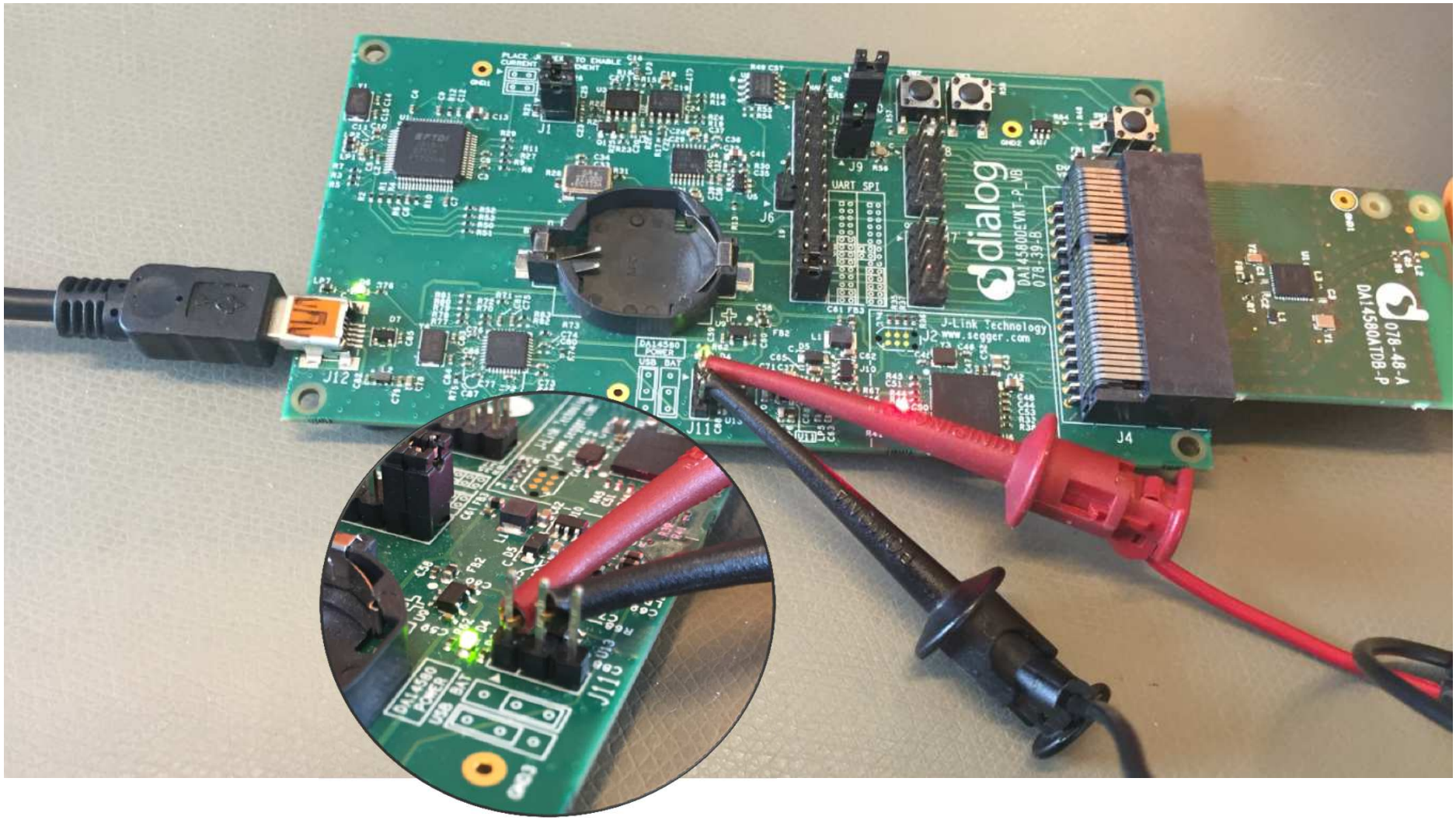
**TODO 2** – 修改结构**user_undirected_advertise_conf**里的变量**.intv**的值到10000(=6.2 sec), 用来设定较大的广播周期.
这样会有足够的时间来测量扩展睡眠条件下的电流.

```
static const struct advertise_configuration user_undirected_advertise_conf = {
    /// Advertise operation type.
    .advertise_operation = ADV_UNDIRECT,
    /// Own BD address source of the device:
    .address_src = GAPM_PUBLIC_ADDR,
    /// Advertise interval
           .intv = 10000, // EXTENDED SLEEP CURRENT = 10000*0.625 = 6.2 sec
    ///Advertising channel map
    .channel_map = 0x7,
};
```

**TODO 3** – 重复前页PPT的操作4到6.

# BLE 睡眠模式

**TODO 4 – 参照如下图连接安培表用来测量扩展睡眠条件下的电流.**



# BLE 睡眠模式

**TODO 5** – 测量扩展睡眠模式下的电流.

电流值应该为**1.4 μA左右**. 在我们的测试里,测量得到1.35 μA.



不建议通过SmartSnippets工具来检测100uA以下的电流.

# BLE 睡眠模式

## 深度睡眠模式

# BLE 睡眠模式

## 设定睡眠模式

**TODO 1** – 从以下目录打开 proximity reporter工程

*projects\target_apps\ble_examples\prox_reporter\Keil_5*

**TODO 2 – 在user_config文件夹下打开文件**/* @file **user_config.h** */.

**TODO 3 – 设定变量app_default_sleep_mode为ARCH_DEEP_SLEEP_ON**

```
const static sleep_state_t app_default_sleep_mode = ARCH_DEEP_SLEEP_ON;
```

**TODO 4 – 在user_config文件夹下打开文件**/* @file **da1458x_config_advanced.h** */

**TODO 5** – 定义CFG_BOOT_FROM_OTP

**TODO 6 – 在user_config文件夹下打开文件** /* @file **da1458x_config_basic.h** */.

**TODO 7** – 取消定义CFG_MEM_MAP_EXT_SLEEP 参量

- 取消定义CFG_DEVELOPMENT_DEBUG 参量
- 定义CFG_MEM_MAP_DEEP_SLEEP 参量

# BLE 睡眠模式

## 设定深度睡眠模式

**TODO 8 – 修改结构user_undirected_advertise_conf里的变量.intv的值到**10000(=6.2 sec), 用来设定较大的广播周期.

这样会有足够的时间来测量扩展睡眠条件下的电流.

```
static const struct advertise_configuration user_undirected_advertise_conf = {
    /// Advertise operation type.
    .advertise_operation = ADV_UNDIRECT,
    /// Own BD address source of the device:
    .address_src = GAPM_PUBLIC_ADDR,
    /// Advertise interval
           .intv = 10000, // EXTENDED SLEEP CURRENT = 10000*0.625 = 6.2 sec
    ///Advertising channel map
    .channel_map = 0x7,
};
```

**TODO 9 – 连接PRO或者BASIC板到PC.**

**TODO 10 –** 用SmartSnippets工具烧录OTP.

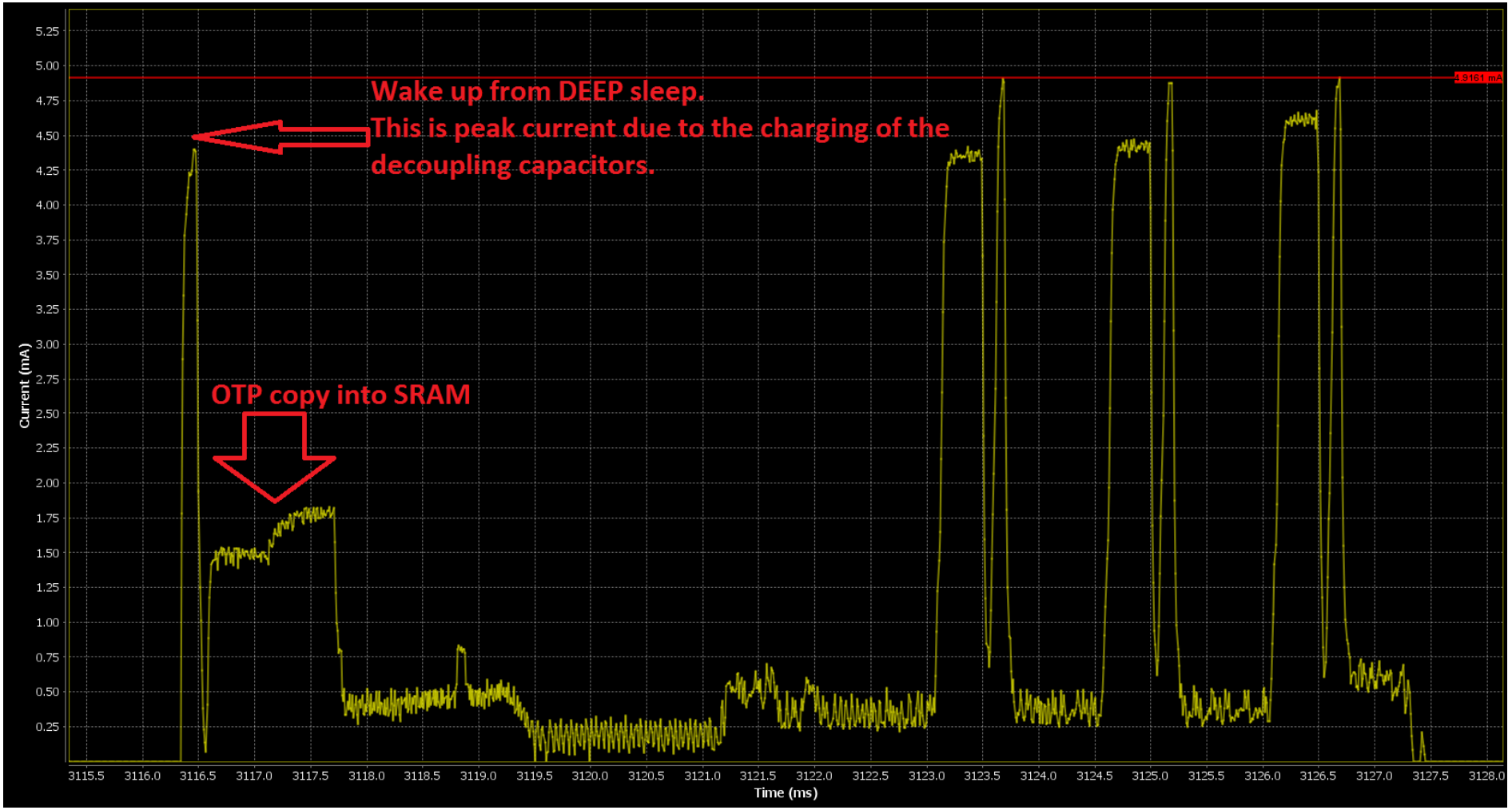
可以通过查询用户手册里的帮助信息, 来了解如何烧录OTP.

# BLE 睡眠模式

## 设定深度睡眠模式

**TODO 6** - 打开SmartSnippet工具 (可以在链接找到: http://support.dialog-semiconductor.com/

你会看到:

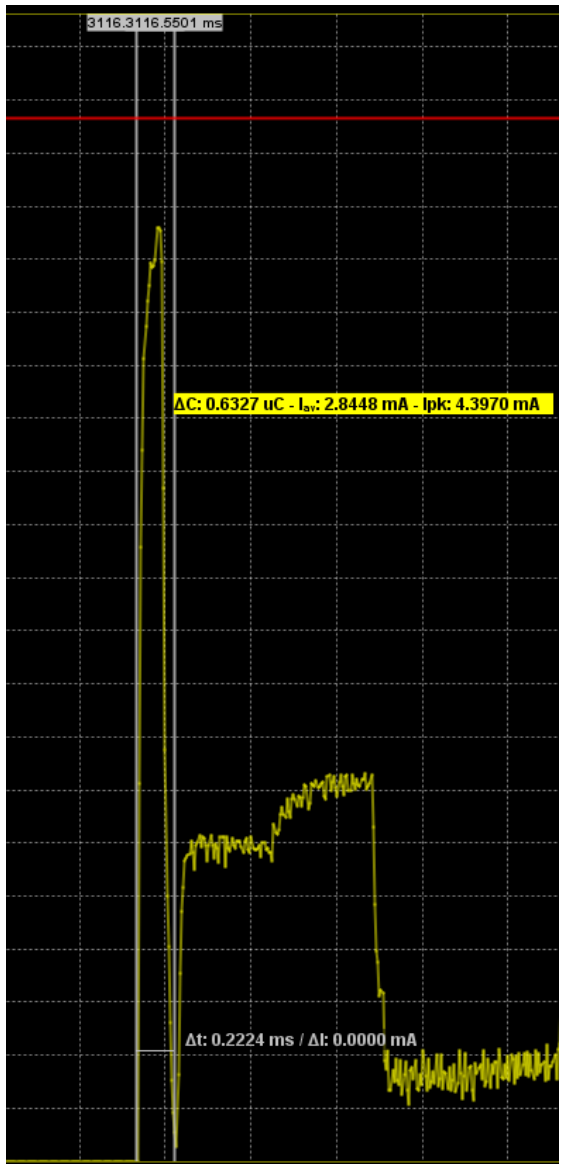


# BLE 睡眠模式

## 设定深度睡眠模式

**测量：**

**由于充电的电容影响，峰值电流时的电量消耗大概是 0.6 µC**

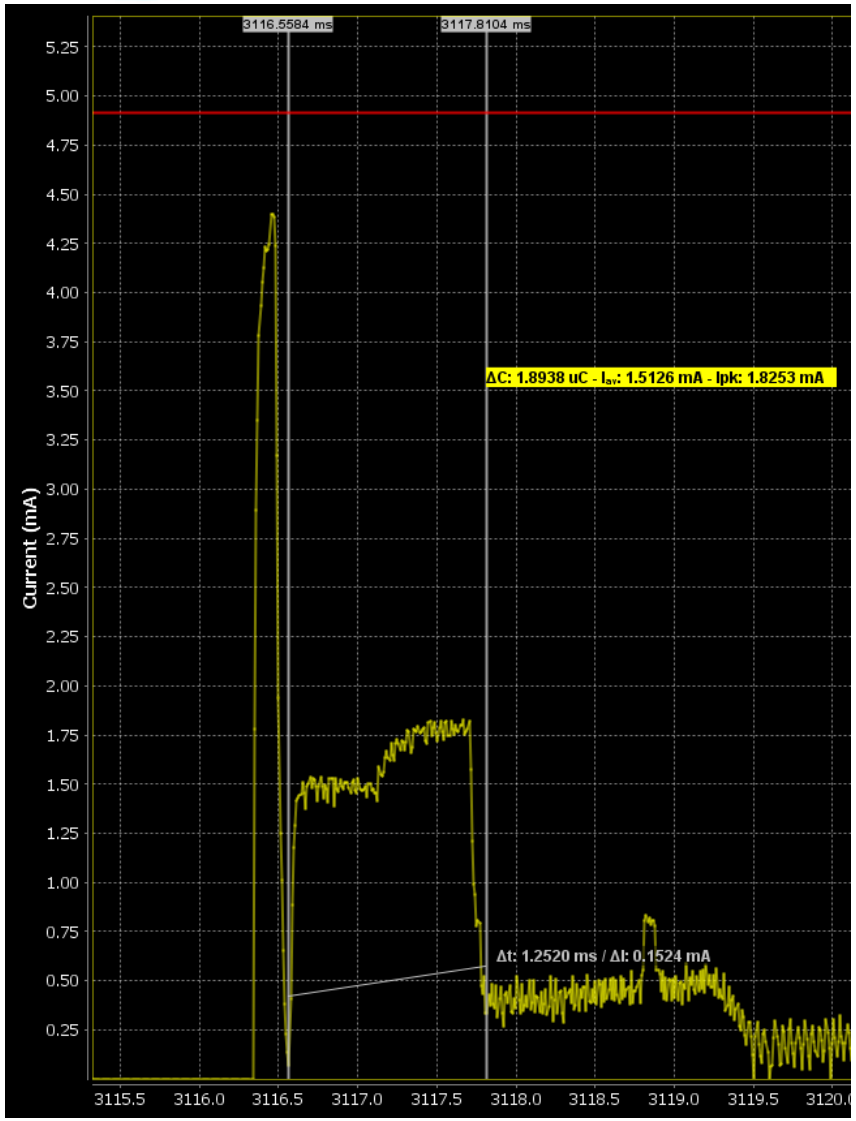


# BLE 睡眠模式

## 设定深度睡眠模式

**测量结果：**

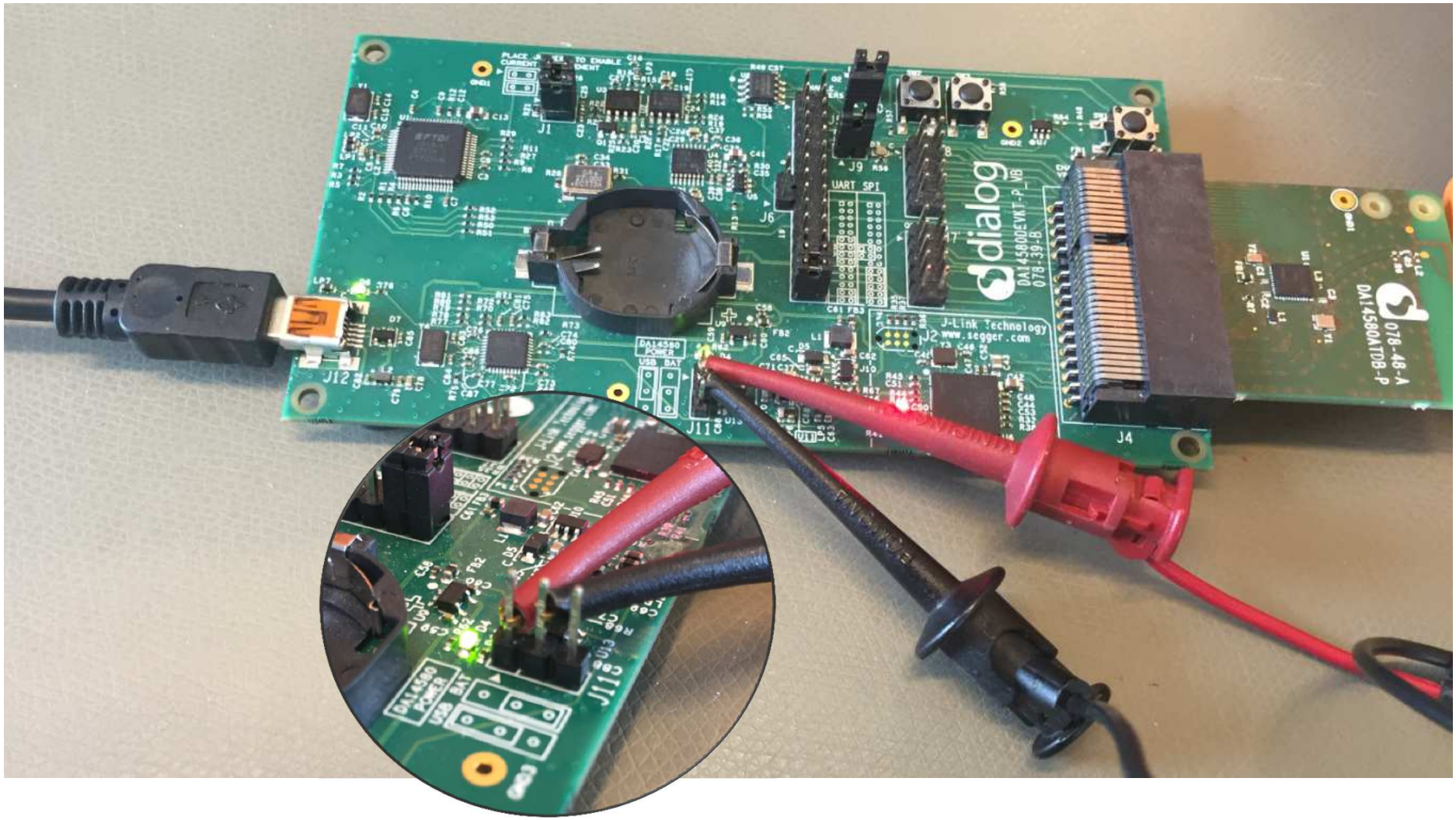
**OTP拷贝，约需耗费2μC电量**



# BLE 睡眠模式

## 测量深度睡眠模式

**TODO 1** – 连接电流表用于测量深度睡眠模式下的电流.



# BLE 睡眠模式

**TODO 5** – 测量深度睡眠模式下的电流.

一般都是 **800 nA**. 在我们的测试里，结果为810 nA.



不建议通过SmartSnippets工具来检测100uA以下的电流.

# BLE睡眠模式

对单个非易失性存储单元进行掉电操作

# BLE 睡眠模式


**对单个非易失性存储单元进行掉电操作只能用于扩展睡眠模式.**

TODO 1 – 从以下目录打开 proximity reporter工程
projects\target_apps\ble_examples\prox_reporter\Keil_5

TODO 2 – 找到void SystemInit (void) 函数流程

TODO 3 – 修改SetBits16(PMU_CTRL_REG, RETENTION_MODE, 0xF);
到SetBits16(PMU_CTRL_REG, RETENTION_MODE, 0x3);

TODO 4 – 找到常量static const struct advertise_configuration user_undirected_advertise_conf

TODO 5 – 修改* .intv = 1100, * ------------> * .intv = 11000, *

TODO 6 – 修改 sleep_state_t app_default_sleep_mode=ARCH_EXT_SLEEP_ON;

TODO 7 – 编译代码，并下载binary文件到设备.

# BLE 睡眠模式

像Agilent 34461A 6 1/2 数字万用表这种精密仪器会被拿来测试电流


**结果：**

**依据如下配置的不同，会省一些功耗：**

`SetBits16(PMU_CTRL_REG, RETENTION_MODE, 0xF);` Extended sleep mode current consumption: **2,037 µA**
`SetBits16(PMU_CTRL_REG, RETENTION_MODE, 0x3);` Extended sleep mode current consumption: **1,957 µA**
**差值大概在80 nA**

# BLE 睡眠模式

## 结论

# BLE 睡眠模式

## 结论：扩展和深度两种睡眠之间的差异

| | 扩展睡眠模式 | 深度睡眠 |
|---|---|---|
| 内存保持上电状态 | 系统RAM 42 kB<br>+<br>8 kB 扩展RAM | 8 kB 非易失性RAM |
| 电流消耗<br>(降压模式, 8 kB 非易失性RAM保持Active状态，使用外部32KHz晶体) | ≈ 1.4 µA | ≈ 800 nA |
| OTP内容是否需要copy? | 从扩展睡眠模式下醒来，OTP内容不需要拷贝到SRAM里面。（对功耗没有影响） | 从深度睡眠模式下醒来，OTP内容需要拷贝到SRAM里面（额外功耗: **2.6 µC电量!**) |

对一般的应用来说，如果广播周期和连接周期小于2s, **那么一般推荐扩展睡眠模式**.

内部**RCX20晶体 (<500 ppm)**, **只有在降压模式下才能被使用**:

- 两种睡眠模式下都会计数
- 在连接状态下，或者不限时间的广播状态下，最多可以记到2s时间

# BLE 睡眠模式

## 参考文件

- **在Dialog官网注册可以获得更好的开发支持**
  - http://support.dialog-semiconductor.com/user/register
  - UM-B-006_DA14580_581 Sleep mode configuration

# The Power To Be…

…personal
…portable
…connected