Cortex-M0 BLE Solution for AB1602 controller

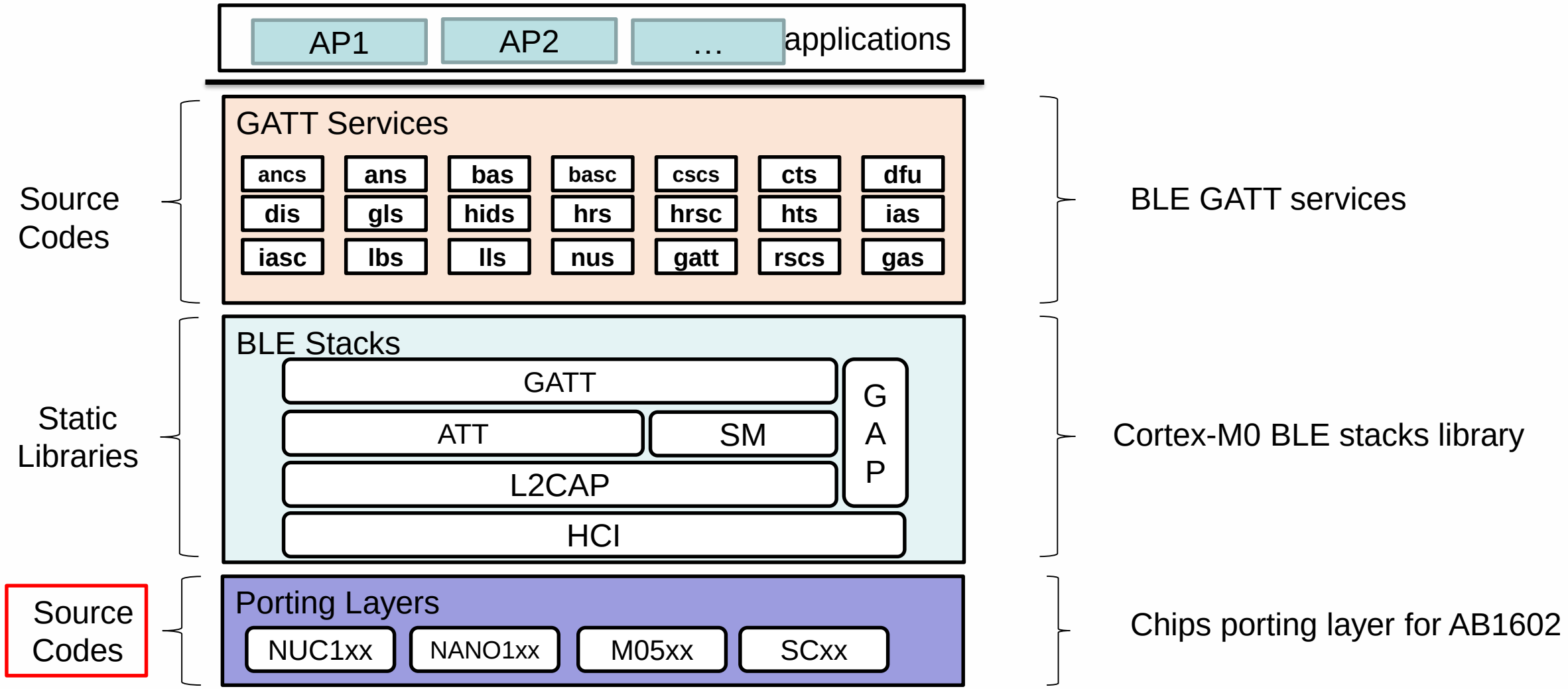
Tallplay

2017/07/11

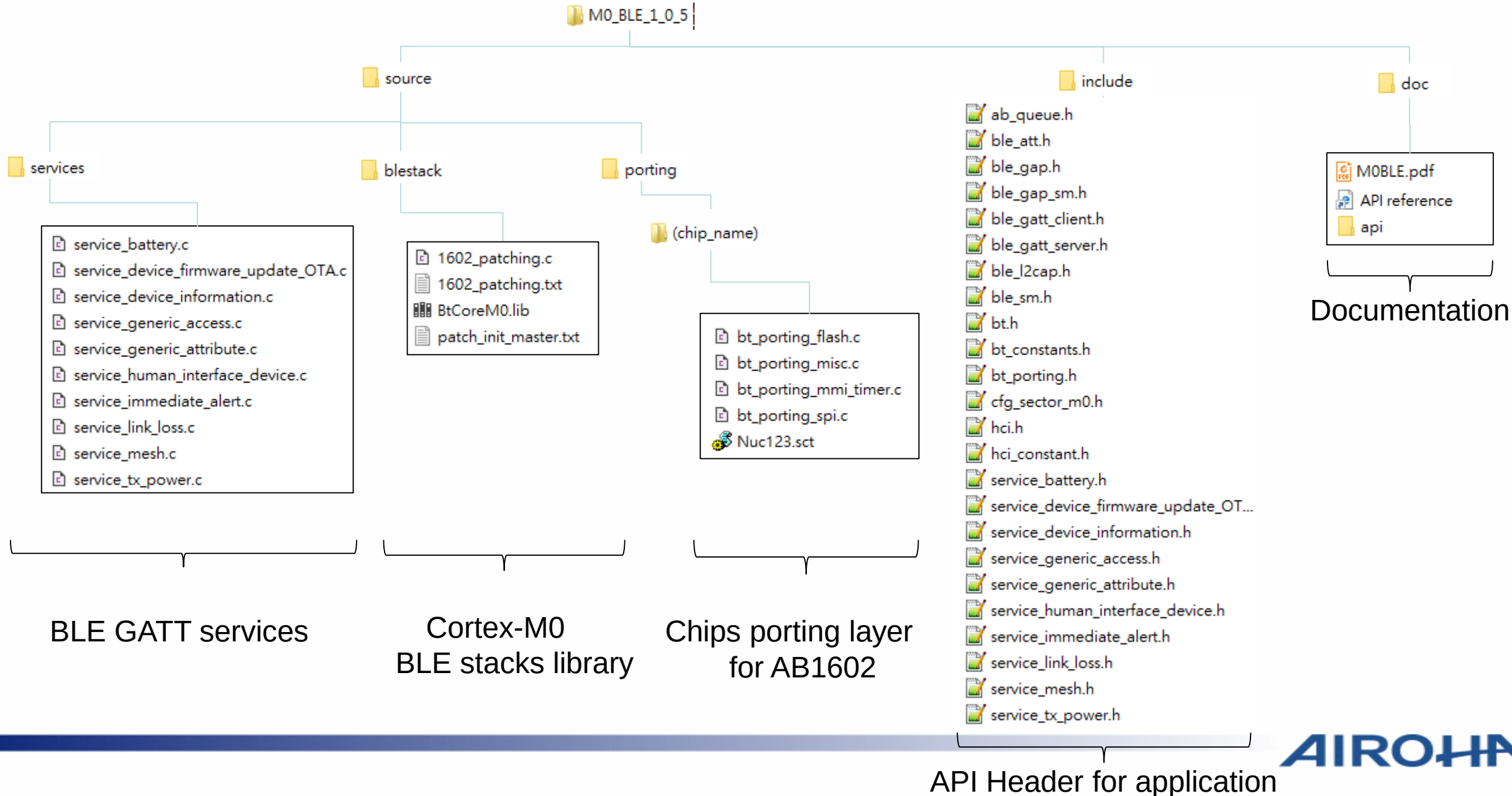
Agenda

- Introduction
  - Solution Architecture
  - File Hierarchy
  - Architecture vs Hierarchy
- Installation and Environment Setup

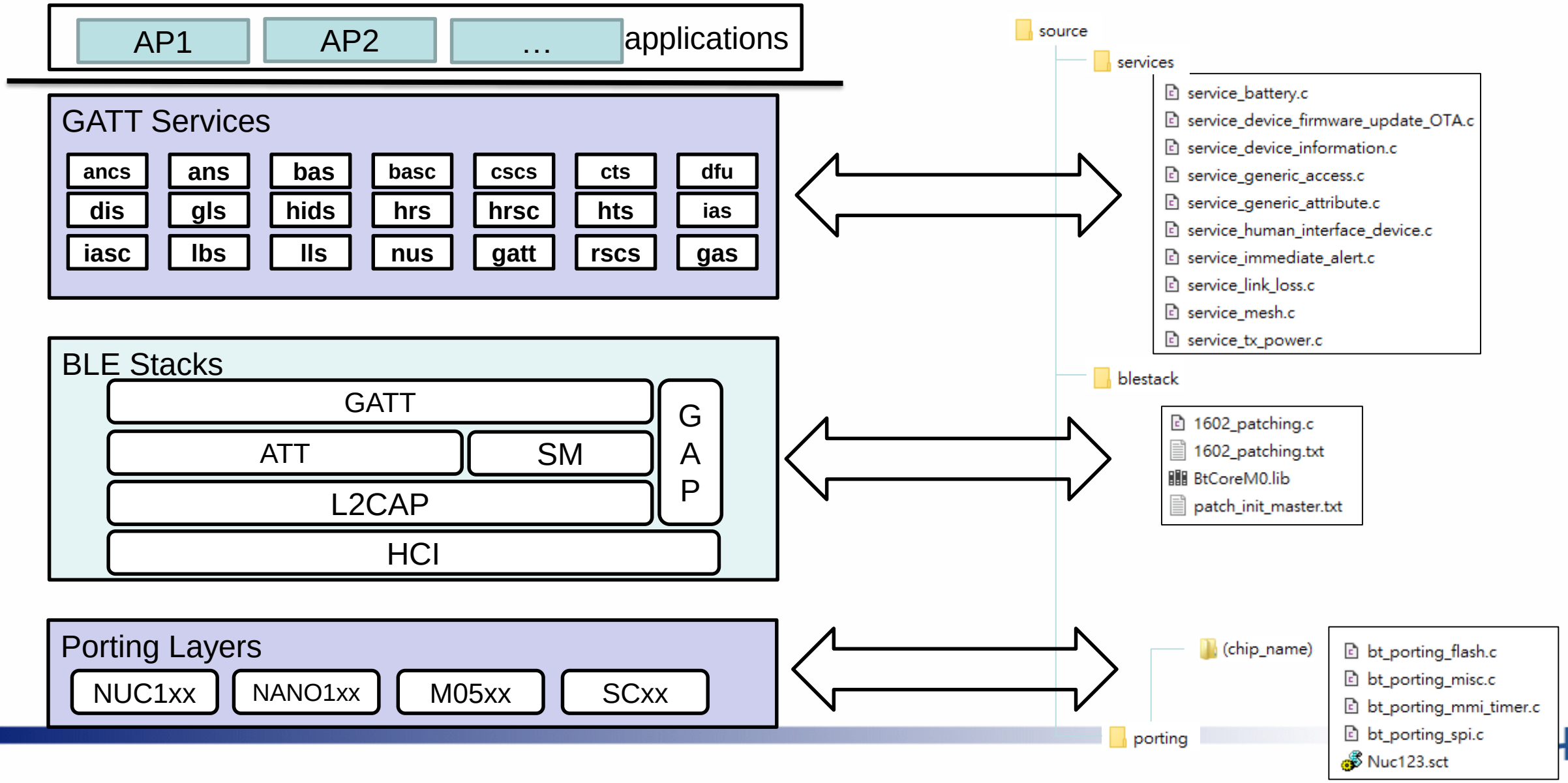
Solution Architecture



File Hierarchy

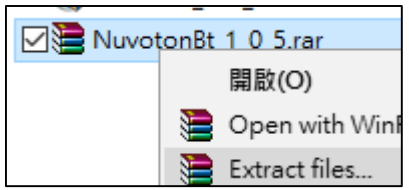


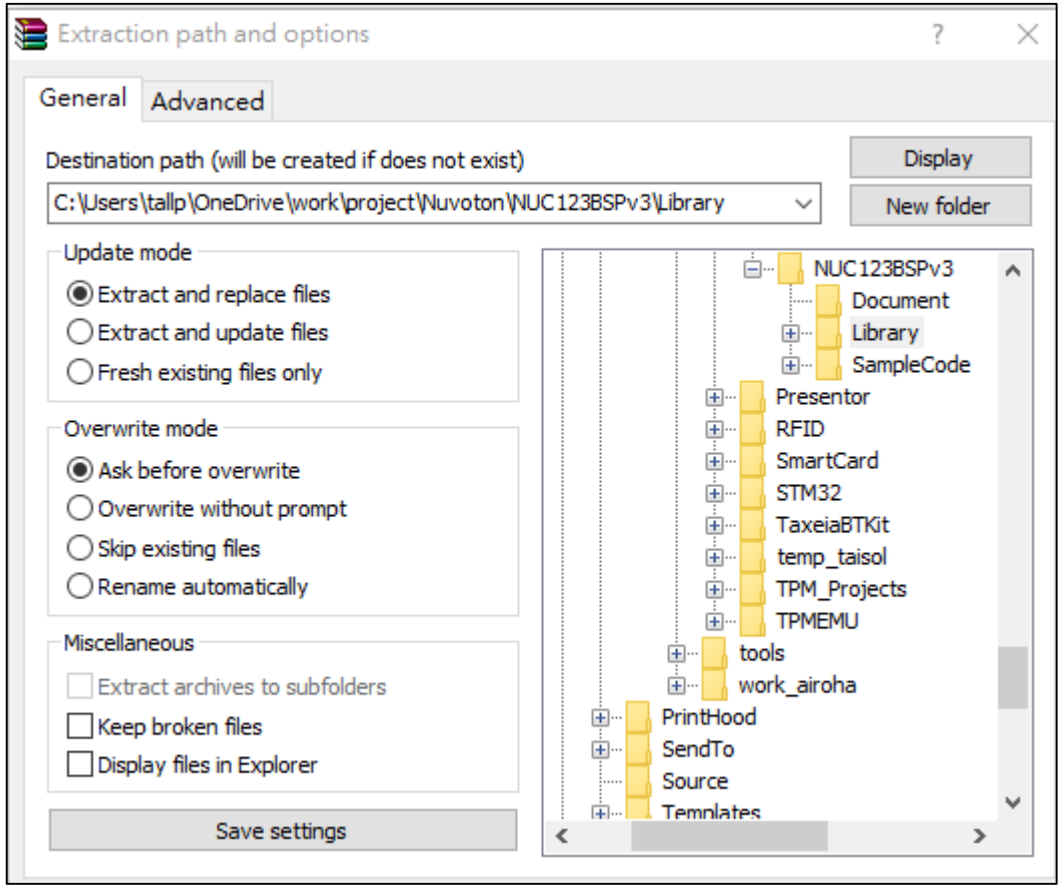
Architecture vs Hierarchy

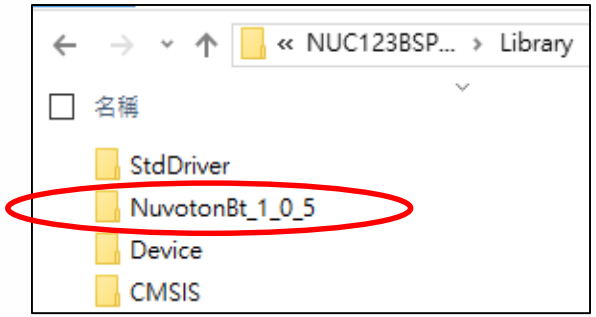


Installation (NUC123 as example)

- Extract *M0BLE_x_x_x.rar* to library folder

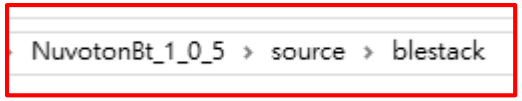


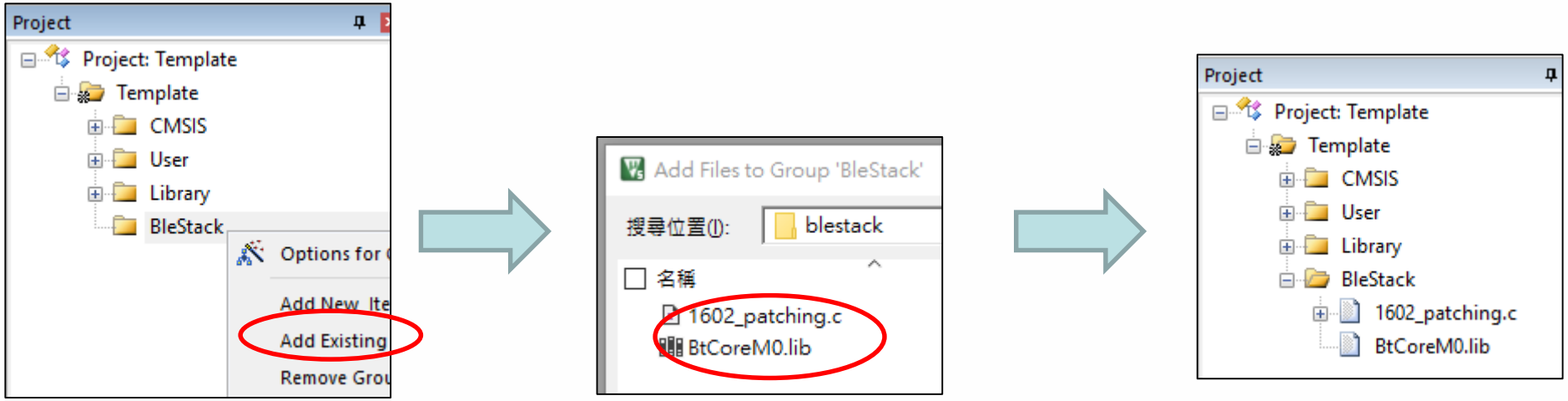




Setup(1) – Add BLE Core Library (NUC123 as example)

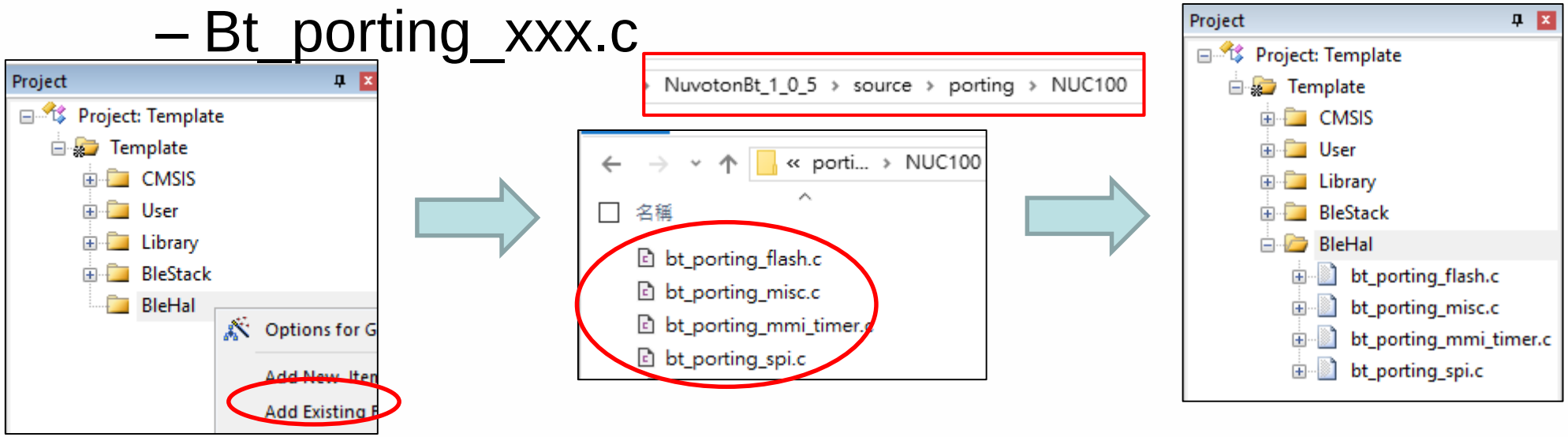
- Add BLE M0 stacks library to project.
  - BtCoreM0.lib
  - 1602_patching.c



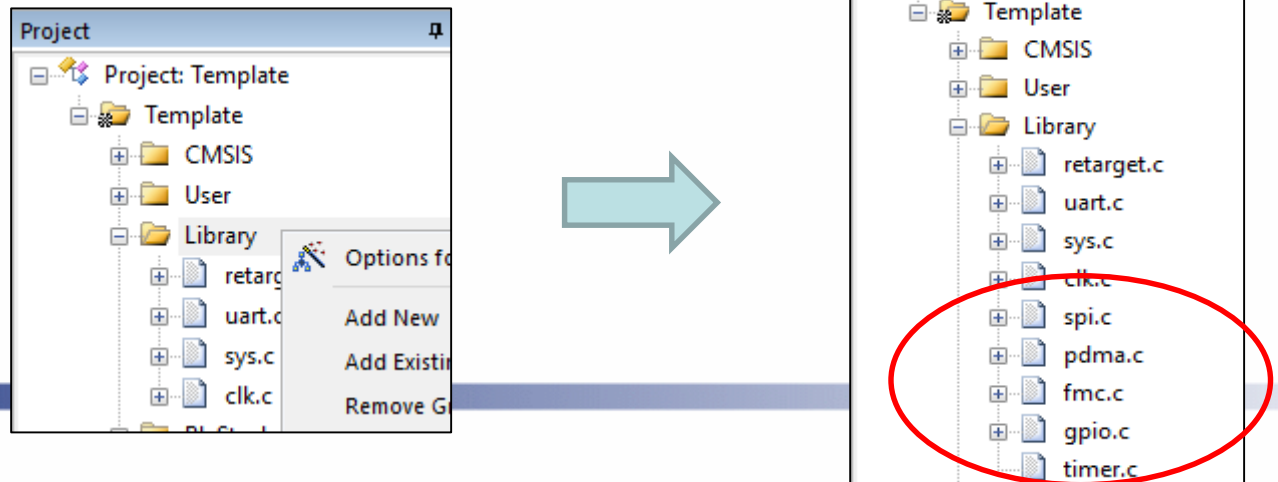


Setup(2) -- add porting layer (NUC123 as example)

- Add porting layer to project.
  - Bt_porting_xxx.c



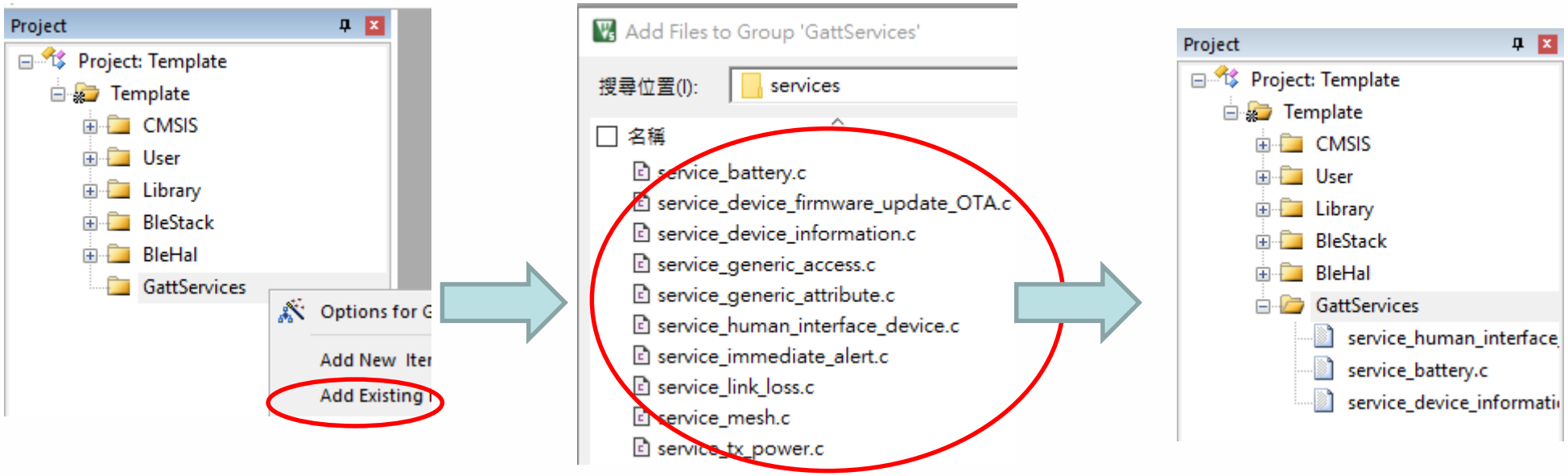
- Add related driver. (NUC123 as example)
  - Spi
  - Pdma.c
  - Fmc.c
  - Gpio.c
  - Timer.c



Setup(3) -- Add GATT Services (NUC123 as example)

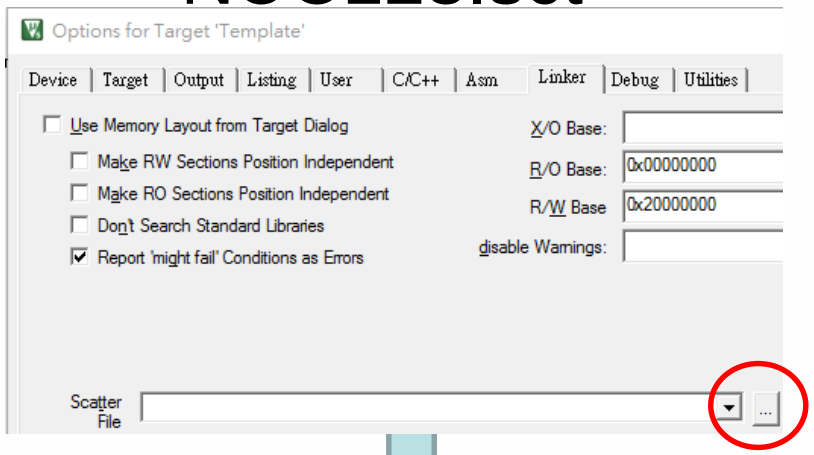
- Add needed GATT services to p
  - Service_xxxx.c

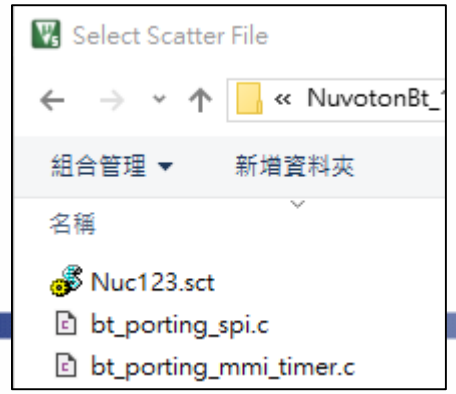
NuvotonBt_1_0_5 › source › services

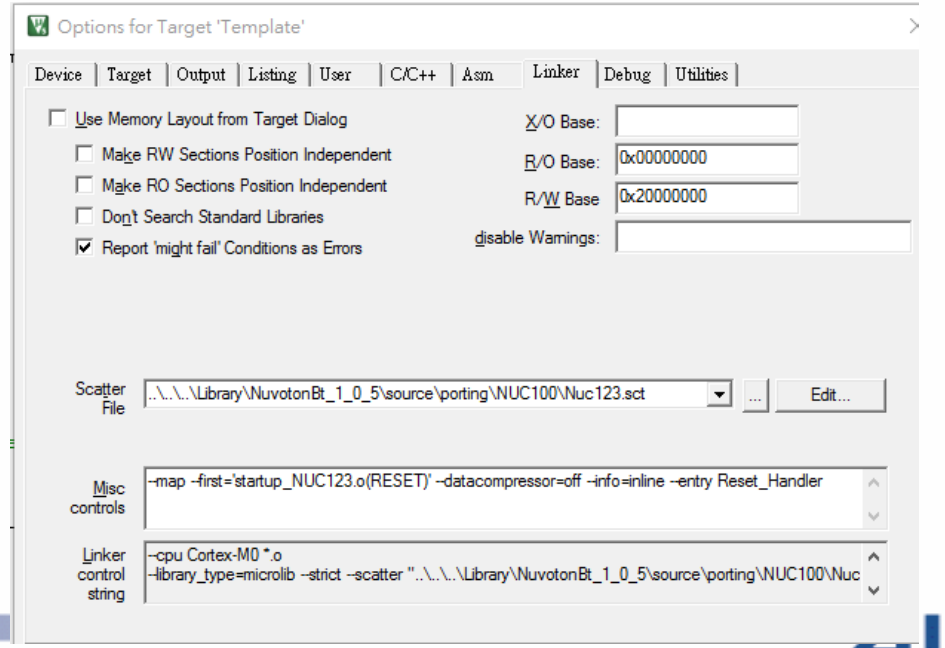


Setup(4) – Set scatter file(NUC123 as example)

- Setup scatter file
  - NUC123.sct

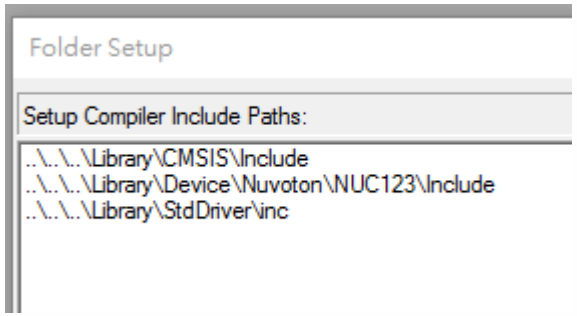


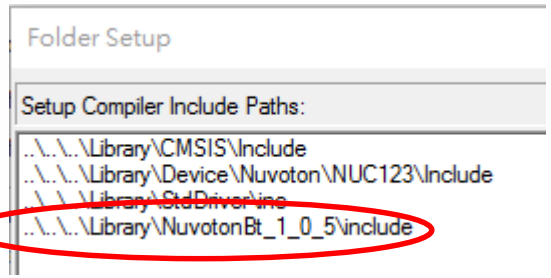




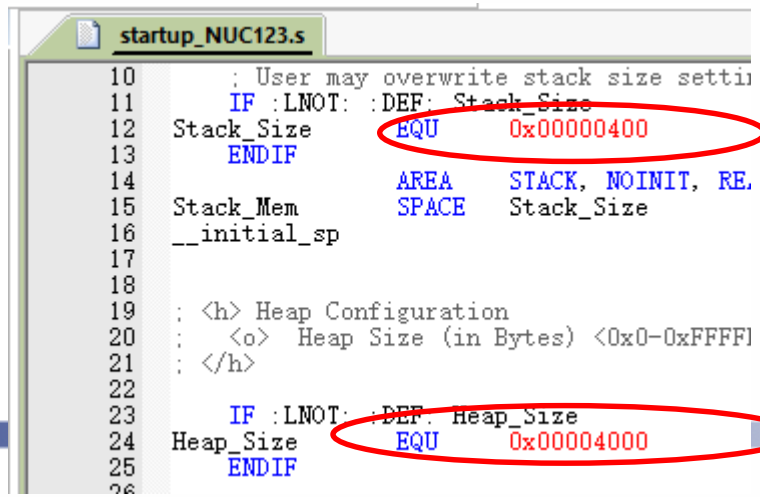
Setup(5) – Set Include Path and Size of HEAP/STACK

- Setup compiler include path:





- Setup size of HEAP/STACK
  - Stack 1024 bytes
  - Heap 16384 bytes



Setup(6) --- AB1602 Initialization

1. Calling initialization function before main loop.
   - *Airoha_PortingInit();*
   - *Bt_InitEx2(bt_evt_hdl, …);*
2. Add .h header file
   - *Bt.h* (for Bt_InitEx2(…) prototype)
   - *Cfg_sector_m0.h*
   - *Bt_porting.h* (For Airoha_PortingInit() prototype)
3. Add BT event callback function.
   - *bt_evt_hdl(bt_evt_t*)*

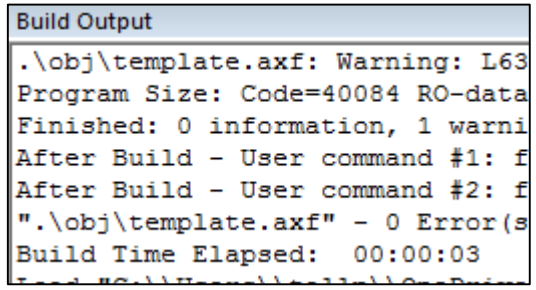
```
int main()
{
    int8_t ch;

    /* Unlock protected registers */
    SYS_UnlockReg();

    SYS_Init();

    /*airoha porting init*/
    Airoha_PortingInit();

    /* Lock protected registers */
    SYS_LockReg();

    /* Init UART0 to 115200-8n1 for print message */
    UART_Open(UART0, 115200);

    /* Bt Init*/
    BT_InitEx2(bt_evt_hdl, BT_INIT_DATA, BT_INIT_DATA_NUM,
        BT_LOG_HCI_CMD | BT_LOG_HCI_EVT);
```

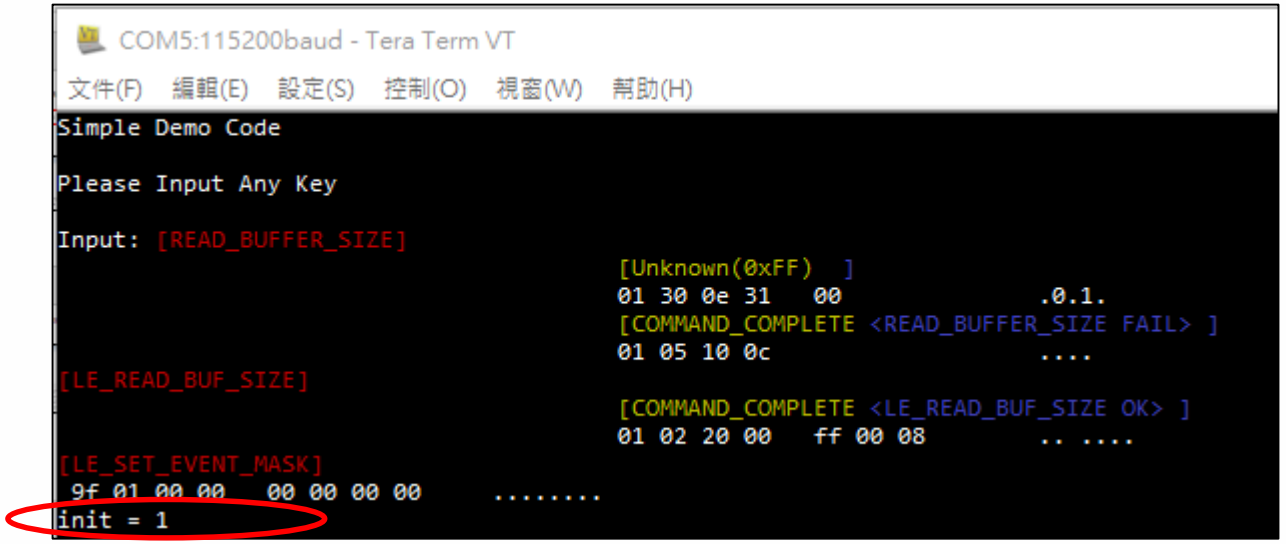
```
/**************************
#include <stdio.h>
#include "NUC123.h"
#include "bt.h"
#include "cfg_sector_m0.h"
#include "bt_porting.h"
```

```
void bt_evt_hdl(bt_evt_t *evt)
{
    switch(evt->evt_id)
    {
        case BT_MISC_EVENT_INITED:
            printf("init = %d\n", evt->evt.misc_evt.param.inited.is_success);
        break;
    }
}
```

Setup(7) – Test

- Check if AB1602 initialization success.
  - If success, then user could start develop application.





Q&A