

将 NFC 库移植到 KW36

在实际应用过程中，经常有客户提出需要 ble+NFC 这样组合的功能。目前我们的 IOT-DK006 是唯一拥有 NFC 模块的开发板，但是开发板的例程的 NFC 功能并不全面，所以我们移植 PN7150 这个 NFC reader 的库提供给 KW 系列单片机，从而使得 KW 系列可以处理 ble+NFC 功能组合的需求。

下面将介绍 nfc reader—PN7150 库移植到 kw36 的方法，使得 kw 具有 nfc 功能

一、PN7150 简介

PN7150 是 PN7120 的高性能版本，PN7120 是即插即用 NFC 解决方案，可轻松集成到任何操作系统环境中，减少物料清单 (BOM) 尺寸和成本。PN71xx 控制器是网关等家庭自动化应用的理想选择，可与 NFC 连接标签无缝协作。

平台：支持 Linux、Android、RTOS 和 NullOS

支持所有 NFC 操作模式

- Read/Write Mode: ISO/IEC 14443 A&B up to 848 kbit/s, FeliCa at 212 & 424 kbit/s, MIFARE Classic with 1K, 4K, NFC Forum type 1, 2, 3, 4, 5 tags, ISO/IEC 15693
- All peer-to-peer modes
- Card Emulation Mode (from host): NFC Forum T4T (ISO/IEC 14443 A&B) at 106 kbit/s, NFC Forum T3T (FeliCa, PN7150 only)

二、开发环境

硬件：FRDM-KW36 开发板，PN7150 开发板，以及若干枚杜邦线。

软件：mcuxpresso11.3

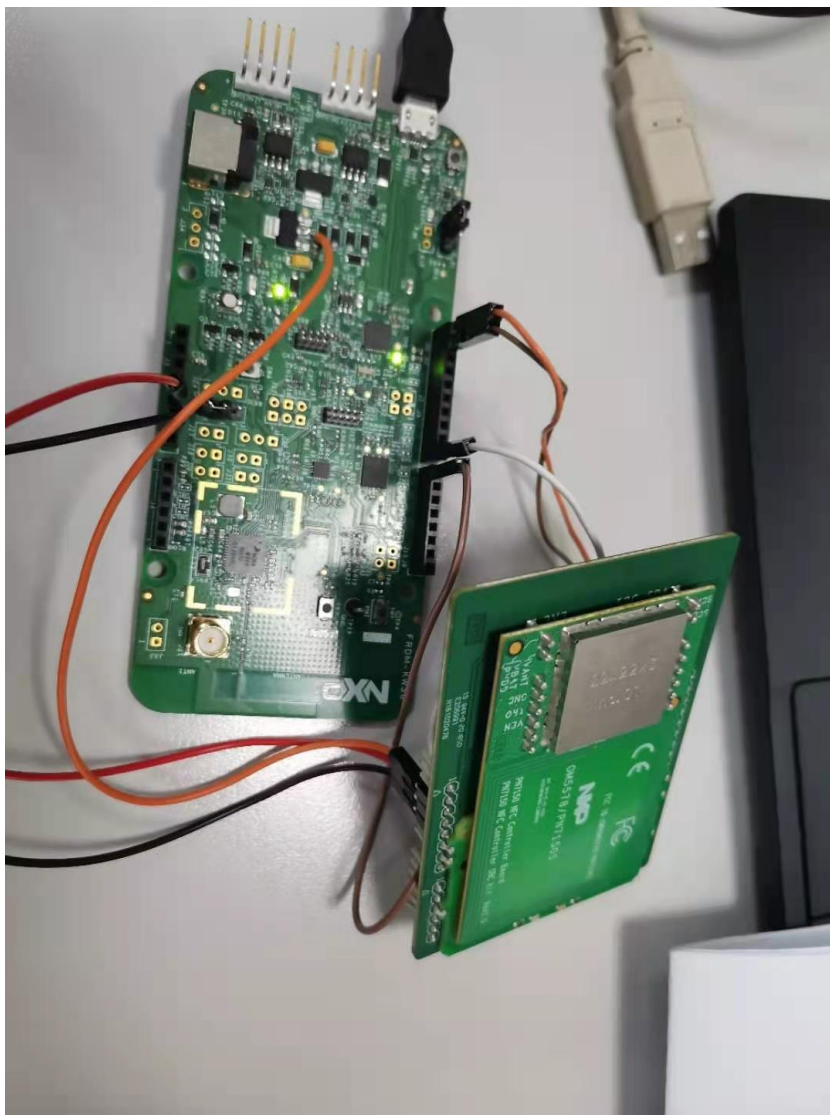
开发包：[NXP-NCI MCUXpresso example Project](#) 里面有若干例程，我们需要里面的库文件。移植参照的例程是 NXPNCI-K64F_example。Kw36 的 sdk 版本是 2.2.8，例程使用的是 frdmkw36_rtos_examples_freertos_i2c

三、移植步骤

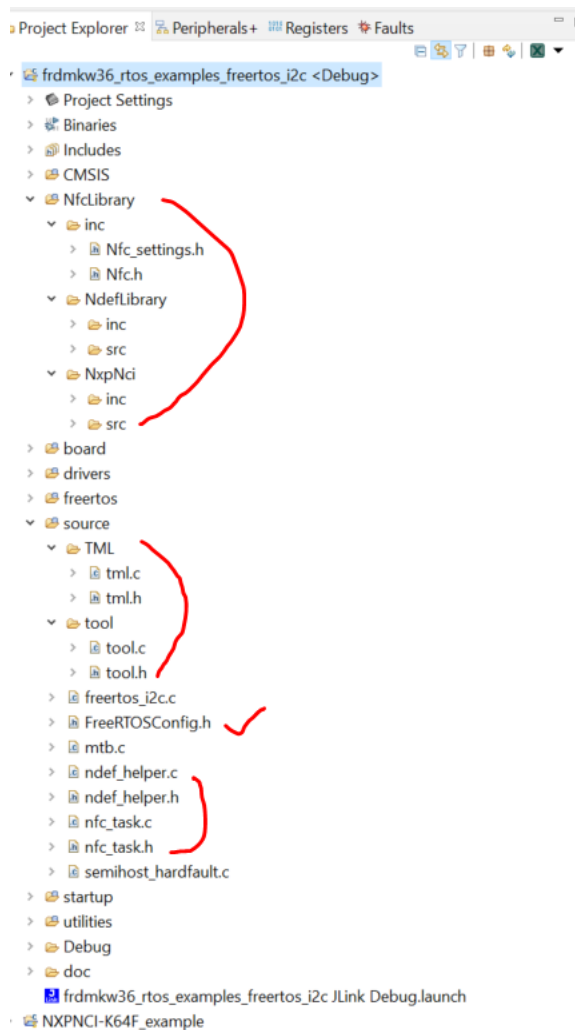
硬件部分：PN7150 我们需要如下连接，注意虽然 PN7150 可以直接插进 arduino 接口，但是由于插上后，arduino 接口提供的电压(1.8V)不够，不足以带动 PN7150，所以我们使用杜邦线连接，并且我们从 U1 这个 LDO 上引出一个 3.3V

PN7150	FRDM-KW36
VBAT/PVDD	3.3V
VANT	5V
GND	GND
IRQ	PTA16
VEN	PTC15
SCL	PTB0, I2C0
SDA	PTB1, I2C0

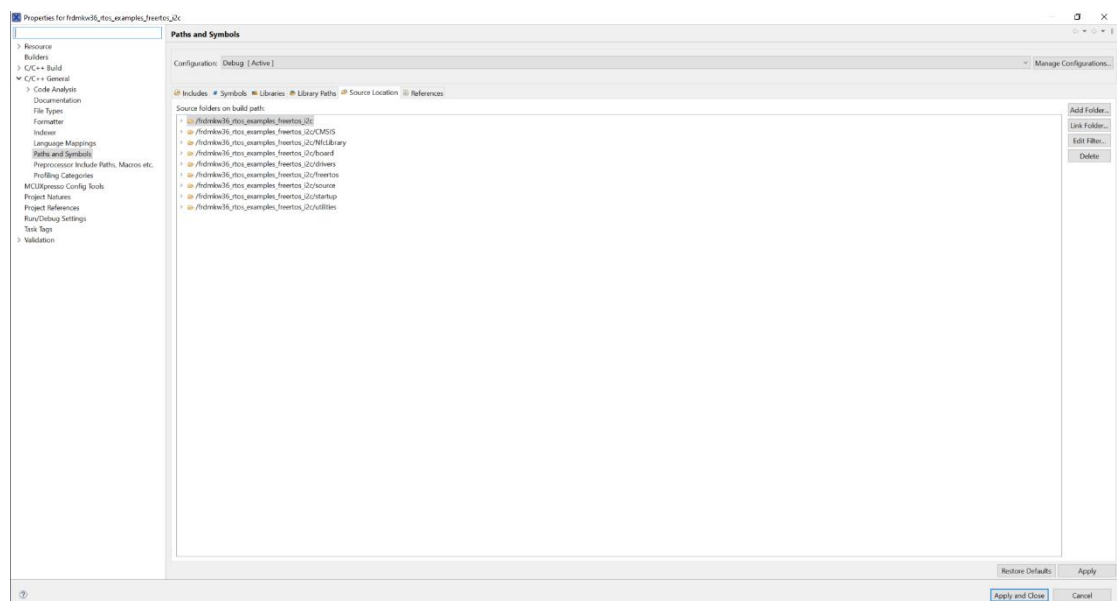
连接如图：



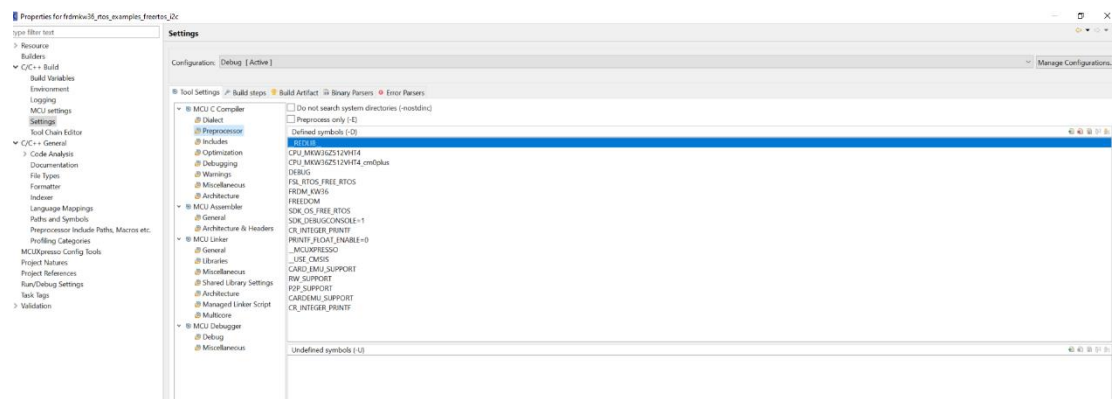
软件部分：首先我们需要移植必要的 nfc 库文件以及文件夹加入路径。这包括如图，具体如何添加文件夹到目前工程可以参考这篇[链接](#)
如图，红色标记的就是需要的文件



注意，添加完文件夹路径除了要包括头文件路径以外，还需要再 path and symbols 里添加文件夹，具体可以参见刚才链接



我们还需要添加一些宏定义在预编译器里



文件放入以后,我们将 freertos_i2c.c 文件里的内容替换为 NXPNCI-K64F_example 例程里的 main.c 的内容。下面我们需要配置修改的是 pin_mux.c, tml.c, board.h 文件。

在 board.h 里，加入如下宏定义

```

/* NXPNCI NFC related declaration */
#define BOARD_NXPNCI_I2C_INSTANCE I2C0
#define BOARD_NXPNCI_I2C_BAUDRATE (100000)
#define BOARD_NXPNCI_I2C_ADDR (0x28)
#define BOARD_NXPNCI_IRQ_PORTIRQn PORTA_IRQn
#define BOARD_NXPNCI_IRQ_GPIO (GPIOA)
#define BOARD_NXPNCI_IRQ_PORT (PORTA)
#define BOARD_NXPNCI_IRQ_PIN (16U)
#define BOARD_NXPNCI_VEN_GPIO (GPIOC)
#define BOARD_NXPNCI_VEN_PORT (PORTC)
#define NXPNCI_VEN_PIN (5U)

```

在 pin_mux.c 文件里添加 board.h 头文件，再添加代码，配置 i2c 接口和 VEN，IRQ，i2c0 例程里已经提前配置好了，它同时配置了 I2C1 可以注释掉。

```

/* Initialize NXPNCI GPIO pins below */
/* IRQ and VEN PIN_MUX Configuration */
PORT_SetPinMux(BOARD_NXPNCI_IRQ_PORT, BOARD_NXPNCI_IRQ_PIN,
kPORT_MuxAsGpio);
PORT_SetPinMux(BOARD_NXPNCI_VEN_PORT, NXPNCI_VEN_PIN, kPORT_MuxAsGpio);
/* IRQ interrupt Configuration */
NVIC_SetPriority(BOARD_NXPNCI_IRQ_PORTIRQn, 6);
EnableIRQ(BOARD_NXPNCI_IRQ_PORTIRQn);
    PORT_SetPinInterruptConfig(BOARD_NXPNCI_IRQ_PORT, BOARD_NXPNCI_IRQ_PIN,
kPORT_InterruptionRisingEdge);

```

最后到 `tm1.c` 文件里把 `PORTC_IRQHandler` 改成 `PORTA_IRQHandler` 修改完这些就可以了。很简单吧。

四、实验结果

我们使用 ntag 来测试读取和写入操作。



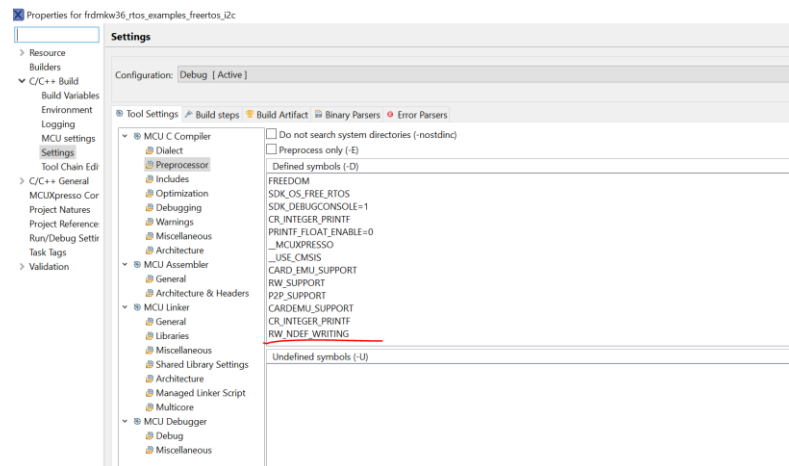
当把 tag 靠近以后，就可以获得如下信息。

```
COM11 - PuTTY
Running the NXP-NCI project.
                                WAITING FOR DEVICE DISCOVERY
card activated                    - POLL MODE: Remote T2T
                                SENS_RES = 0x44 0x0
                                NFCID = 4 f7 f d2 9c 39 80
                                SEL_RES
= 0x0
--- NDEF record received:      Text record: VER=03
                                --- NDEF Record sent
                                CARD REMOVED
                                WAITING FOR DEVICE DISCOVERY
```

tag 内记录的文字是 VER=03

下面我们来修改这段文字。

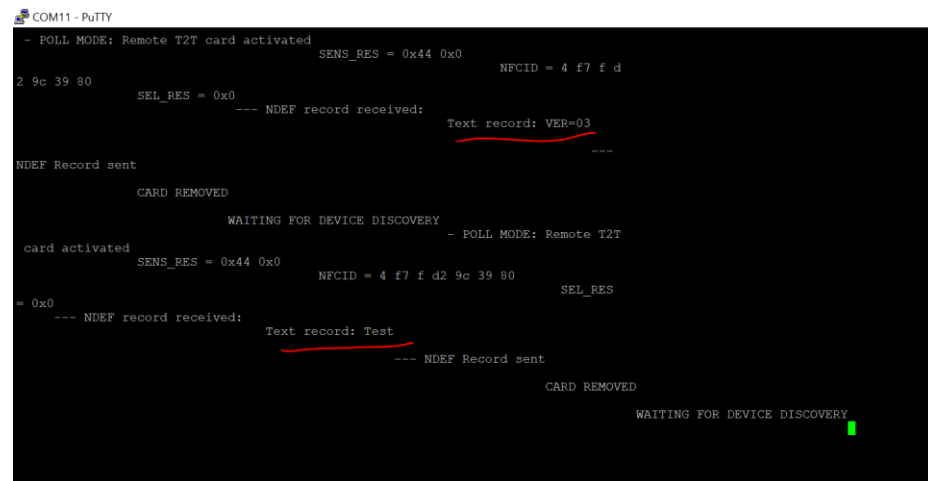
在预处理器中添加新的宏定义 `RW_NDEF_WRITING`



在函数 `task_nfc_reader` 里通过修改 `NDEF_MESSAGE`

```
#if defined P2P_SUPPORT || defined CARDEMU_SUPPORT || defined RW_NDEF_WRITING
#define ADD 0 // Enlarge NDEF message by adding dummy content
const char NDEF_MESSAGE[14 + ADD] = { 0xC1, // MB/ME/CF/1/IL/TNF
    0x01, // TYPE LENGTH
    (0x07 + ADD) >> 24, // PAYLOAD LENGTH MSB
    (0x07 + ADD) >> 16, // PAYLOAD LENGTH
    (0x07 + ADD) >> 8, // PAYLOAD LENGTH
    (0x07 + ADD) & 0xFF, // PAYLOAD LENGTH LSB
    'T', // TYPE
    // PAYLOAD
    0x02, // Status
    'e', 'n', // Language
    'T', 'e', 's', 't'};
```

来编辑我们要写入的文字，代码中写入的文字是 Test，我们重新下载代码，然后可以看到现象，第一次读取的是原来字符，同时写入了新的信息。然后再次读取 tag，文字变成了 Test



下面我们再写一个长一点数据，此时就需要修改 ADD。这个 message，只发送 4 个字符时候 ADD 为 0，每多发一个字符就要加一，我们修改 tag 的文字为 Ver=03，多了两个字符，所以 ADD 要定义为 2

```

// #11 defined F4F_SUPPORT || defined CARDLESS_SUPPORT || defined RW_NDEF_Writing
#define ADD 2 // Enlarge NDEF message by adding dummy content
const char NDEF_MESSAGE[14 + ADD] = { 0xC1, // MB/ME/CF/1/IL/TNF
    0x01, // TYPE LENGTH
    (0x07 + ADD) >> 24, // PAYLOAD LENGTH MSB
    (0x07 + ADD) >> 16, // PAYLOAD LENGTH
    (0x07 + ADD) >> 8, // PAYLOAD LENGTH
    (0x07 + ADD) & 0xFF, // PAYLOAD LENGTH LSB
    'T', // TYPE
    // PAYLOAD
    0x02, // Status
    'e', 'n', // Language
    'V', 'e', 'r', '=', '0', '3'};

```

这样一开始是 Test 后来就变成了新字符串

COM11 - PuTTY

```

Running the NXP-NCI project.

                                WAITING FOR DEVICE DISCOVERY
                                - POLL MODE: Remote T2T

card activated
    SENS_RES = 0x44 0x0
                                NFCID = 4 f7 f d2 9c 39 80
                                SEL_RES
= 0x0
--- NDEF record received:
                                Text record: Test
                                --- NDEF Record sent

                                CARD REMOV
ED

                                WAITING FOR DEVICE DISCOVERY
                                - POLL MODE: Remote T2T card activated
                                SENS_RES
= 0x44 0x0
                                NFCID = 4 f7 f d2 9c 39 80
                                SEL_RES = 0x0
                                --- NDEF record rec
eived:
                                Text record: Ver=03
                                --- NDEF Record sent

                                CARD REMOVED

                                WAITING FOR DEVICE D
ISCOVERY

```

其他的卡读写，都可以定义宏来使能这些功能。默认下，是不使用的。