



# PAN211x 开发套件使用手册

发布 1.0.0



磐启微电子 PAN211x 项目组  
2024 年 09 月 05 日

# Table of contents

<b>1 快速入门</b>	<b>1</b>
1.1 SDK 开发环境说明	1
1.1.1 1. SDK 开发 IDE	1
1.1.2 2. Keil Flash 下载程序	2
1.2 SDK 快速开发指南	2
1.2.1 1. 概述	2
1.2.2 2 PAN211x EVB 介绍	3
1.2.3 3 PAN211x EVB 开发环境确认	3
1.2.4 4 PAN211x KEIL 工程介绍	4
<b>2 硬件资料</b>	<b>9</b>
2.1 PAN211x EVB 评估板说明	9
2.1.1 1. 简介	9
2.1.2 2. 底板原理图	9
2.1.3 3. 主要功能	11
2.1.4 3. 评估板接口说明	11
2.1.5 4. 电源管理	12
2.1.6 5. 开发环境与调试	12
2.1.7 6. 适用范围	12
2.1.8 7. 附件与资源	12
<b>3 演示例程</b>	<b>13</b>
3.1 例程介绍	13
3.2 例程列表	13
3.2.1 00_normal 普通型收发例程	13
3.2.2 01_normal_setup 普通型收发配置例程	15
3.2.3 02_enhance 增强型发射例程	17
3.2.4 03_normal_with_ack 普通型带自动应答收发例程	19
3.2.5 04_carrier 载波模式例程	20
3.2.6 05_sleep_mode 休眠模式例程	20
3.2.7 06_multipipe 多通道地址例程	22
3.2.8 07_ble_beacon 蓝牙广播例程	24
3.2.9 08_ble_filter 蓝牙过滤例程	24
3.2.10 09_ble_longrange 蓝牙长距离模式例程	25
3.2.11 10_oled OLED 评估例程	26
<b>4 开发指南</b>	<b>37</b>
4.1 PAN211x 例程移植指南	37
4.1.1 1. 驱动文件移植	37
4.1.2 2. EASY_RF 使用	37
4.1.3 QA	38
4.2 PAN211x FIFO 和帧格式使用指南	38
4.2.1 1. FIFO 操作	38
4.2.2 2. 工作模式 (WorkMode) 与协议帧格式 (ChipMode)	38
4.2.3 3. 私有协议帧格式	39
4.2.4 4. 蓝牙帧格式	42
4.3 常见问题 (FAQs)	43

4.3.1	Q1: 为什么 PAN211x EVB 开发板 SWD 下载固件老是失败? . . . . .	44
4.3.2	Q2: PAN211x 收发两端为什么不能正常通信? . . . . .	44
4.3.3	Q3: PAN211x 通信距离能达到多远? . . . . .	44
4.3.4	Q4: PAN211x 可以设置多大的功率? . . . . .	44
4.3.5	Q5: PAN211x 休眠功耗为多大? . . . . .	44
<b>5</b>	<b>开发工具</b> . . . . .	<b>45</b>
5.1	PAN211x 代码生成工具 . . . . .	45
5.1.1	1. 简介 . . . . .	45
5.1.2	2. 参数配置界面说明 . . . . .	45
5.1.3	3. 代码导出方法说明 . . . . .	46
<b>6</b>	<b>其他文档</b> . . . . .	<b>49</b>
<b>7</b>	<b>ChangeLog</b> . . . . .	<b>51</b>
7.1	PAN211x SDK V1.0.0 Release Notes . . . . .	51
7.1.1	1. SDK . . . . .	51
7.1.2	2. HDK . . . . .	52
7.1.3	DOC . . . . .	52
7.1.4	3. TOOLS . . . . .	52

# Chapter 1

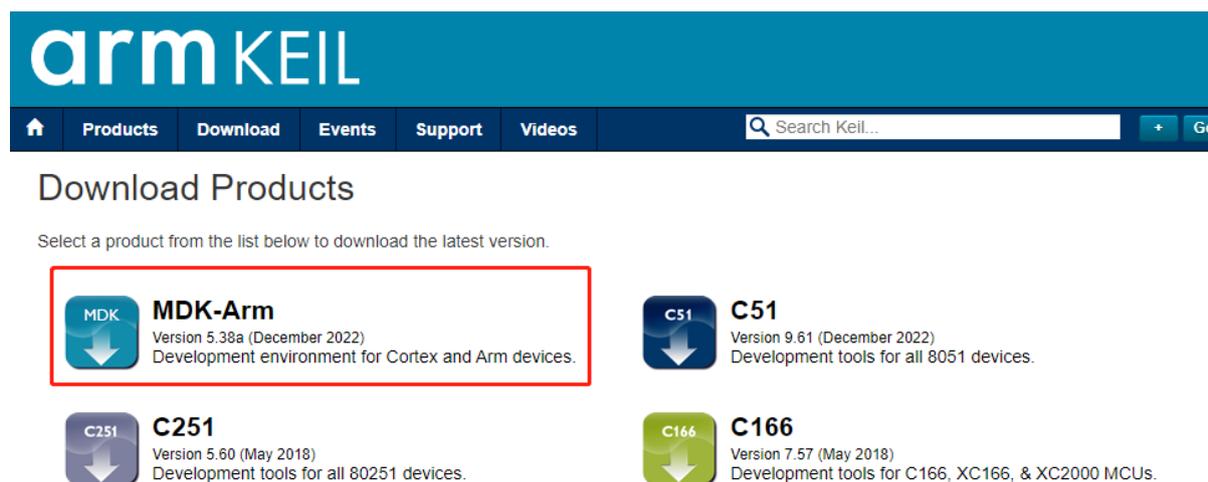
## 快速入门

### 1.1 SDK 开发环境说明

#### 1.1.1 1. SDK 开发 IDE

KEIL 官方下载连接:

<https://www.keil.com/download/product/>



Keil 下载 MDK-Arm 版本

当前 PAN211x EVB 开发使用的工具为 KEIL，开发中使用的版本为 5.25，所以建议使用该版本及以上版本。

Keil 使用版本如下：



Keil 版本

## 1.1.2 2. Keil Flash 下载程序

为使用 Keil + Jlink(SWD) 烧录代码, 请将此目录下的 FLM 文件, 拷贝到 Keil MDK 安装目录下, 如:

- C:\Keil\_v5\ARM\Flash

相关 FLM 文件默认在 SDK 中路径为 pan211x-dk\04\_TOOLS\PAN108\_PACK

- PAN1080\_1020KB\_FLASH\_KEIL.FLM

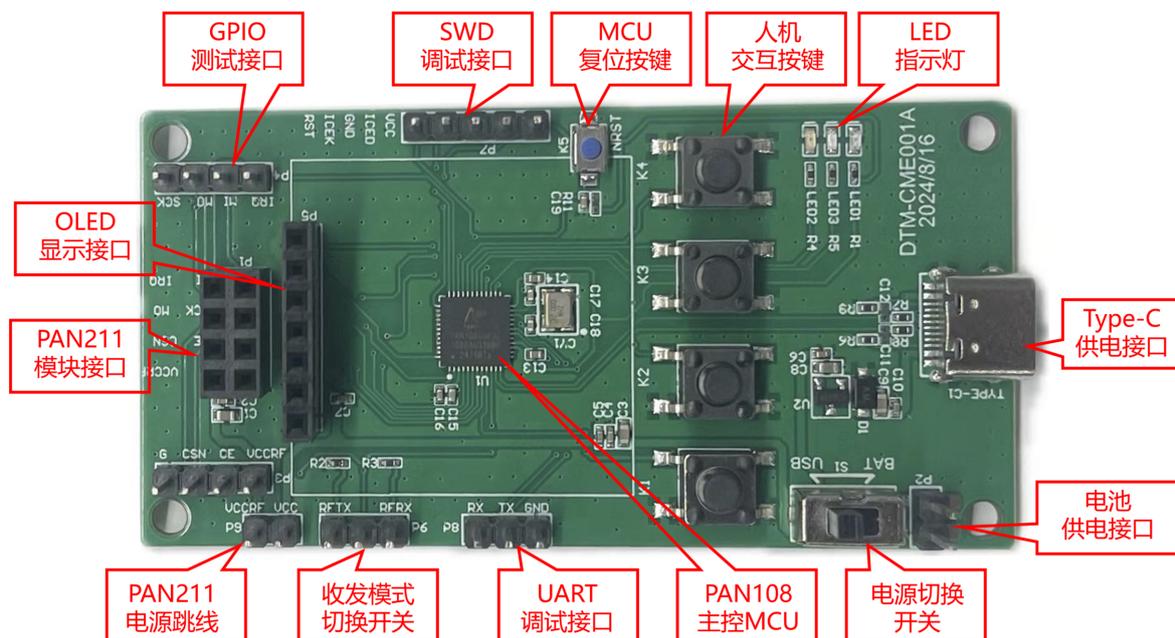
## 1.2 SDK 快速开发指南

### 1.2.1 1. 概述

本文是 PAN211x 代码开发的快速入门指引, 旨在帮助使用者快速入门 PAN211x 驱动及应用的相关开发。

## 1.2.2 2 PAN211x EVB 介绍

PAN211x 评估板采用我司 PAN1080 为主控 MCU，集成了 PAN211x RF 模块和必要的外围接口，支持用户通过多种方式进行 RF 功能评估和开发。评估板提供了丰富的外设接口和配置选项，支持连接 OLED 显示屏、不同供电模式切换以及 RF 模块的收发模式选择。



关

于 PAN211x EVB 开发板硬件的详细介绍，请参考 [PAN211x EVB 硬件资源介绍](#)。

## 1.2.3 3 PAN211x EVB 开发环境确认

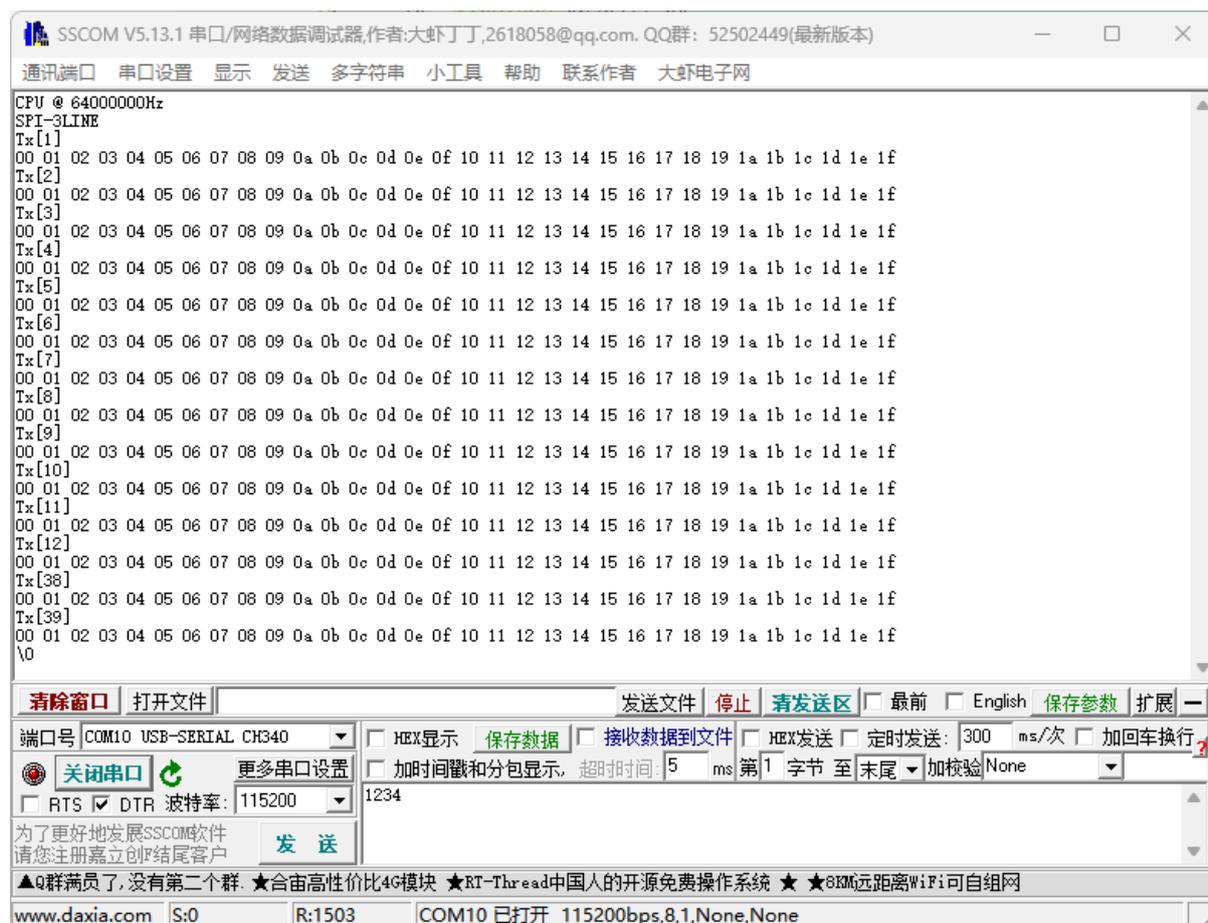
### 3.1 PC 环境检查

请确认 KEIL(推荐 5.25 版本以上), PAN108x 下载依赖的 flm 文件, Jlink 设备等准备就绪。

### 3.2 快速编译运行一个简单的例程

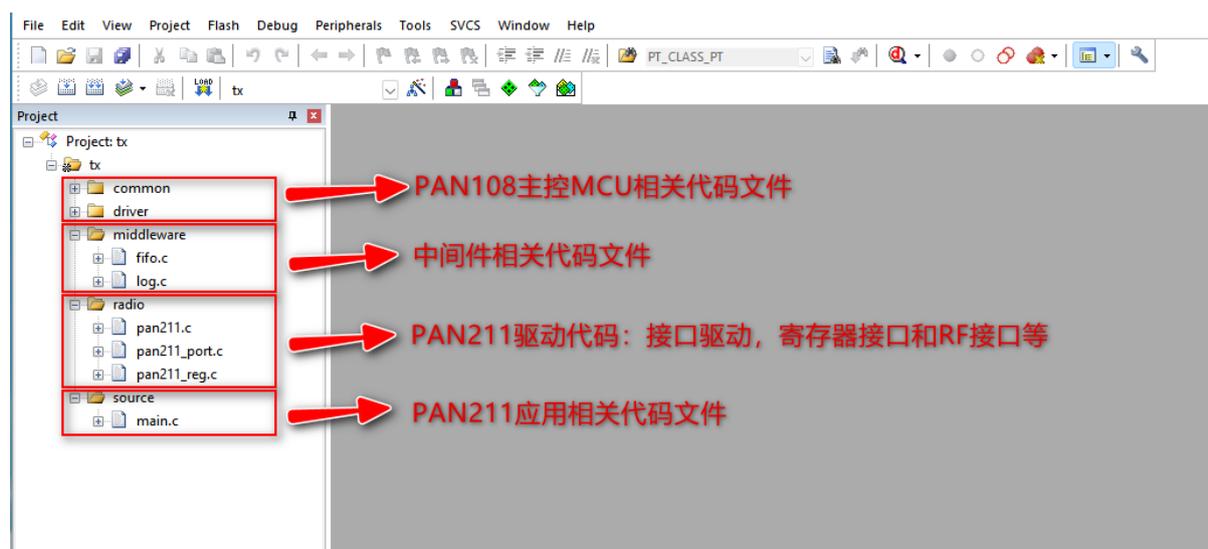
硬件接线准备，请确认您已经将 PAN108x EVB 板的：

1. SWD (P46: SWD\_CLK, P47: SWD\_DAT, GND: SWD\_GND) 接口通过 JLink 连接至 PC。
2. UART1 接口通过 USB 转串口模块 (采用 TTL 电平) 连接至 PC, UART1-Tx: P14, UART1-Rx: P15。
3. 打开一个 sample 工程，例如 01\_SDK\example\00\_normal\tx\keil 下的 keil 目录下的工程文件 tx.uvprojx。
4. 点击 Build 编译按钮，然后点击 Download 按钮进行下载 (如果缺少 FLM 文件，KEIL 会弹出提示框进行提示)。
5. 下载完成后会通过串口观察 log 输出 (**串口波特率: 115200**)



## 1.2.4 4 PAN211x KEIL 工程介绍

### 4.1 KEIL 工程结构介绍



### 4.2 简易例程使用说明

#### 4.2.1 配置文件导出

- 下图为 PAN211x 常用的典型参数配置列表, 用户可通过鼠标单击的方式选择对应的参数配置

Configuration List																
index	ChipMode	TrxMode	WorkMode	DataRate	Channel	EnWhiten	TxPower	Crc	ENDPL	TxNoAck	Endian	SPI CLK	TxLength	RxLength	ScrSkipAddr	
1	297L	TX	normal	1M	2402MHz	enable	3	2bytes	disable	1	1	1	32	32	0	
2	297L	RX	normal	1M	2402MHz	enable	3	2bytes	disable	1	1	1	32	32	0	
3	297L	TX	enhance	1M	2402MHz	enable	3	2bytes	enable	0	1	1	32	32	0	
4	297L	RX	enhance	1M	2402MHz	enable	3	2bytes	enable	0	1	1	32	32	0	

- 根据实际需求修改参数配置，配置界面如下图

Chip Parameters

BasicSetting    Address

TrxMode	TX	ChipMode	297L	WorkMode	normal	Channel	2402	MHz
DataRate	1M	TxPower	3	Crc	2bytes	ENDPL	disable	
EnWhiten	enable	TxNoAck	enable	Endian	big	CrcSkipAddr	disable	
TxLength	32	RxLength	32	SPI CLK	1MHz	TxDelayTimeUs	0	us
RxTimeoutUs	0	AutoDelayUs	250	AutoMaxCnt	3			

- 点击”导出配置文件“，跳转到代码预览界面后，点击导出头文件，即可生成配置文件

FrameFormat

Preamble	Address	Payload	CRC
3bytes	5bytes	32bytes	2bytes

CRC/Whiten (Exclude Addr)

CRC/Whiten

导出配置文件

导出工程代码

- 配置头文件 easy\_reg.h 的预览界面如下图，点击右上方的 ‘ ’ 导出头文件“按钮即可生成 easy\_reg.h。

以 01\_SDK\example\00\_normal\tx 例程为例，如果用户要更改 RF 参数配置，只要将生成的 easy\_reg.h 头文件替代 01\_SDK\example\00\_normal\tx\src 目录下的 easy\_reg.h 后，再编译 KEIL 代码工程即可实现参数更改。

PAN211代码生成工具V1.0.1

代码预览

导出头文件

```

#ifndef _EASY_REG_H
#define _EASY_REG_H

const uint8_t gPN211Page0Table[][2] = {
    {0x05, 0xC8},
    {0x07, 0x89},
    {0x08, 0xA3},
    {0x09, 0x20},
    {0x0A, 0x20},
    {0x2A, 0x41},
    {0x2B, 0x00},
    {0x2C, 0x00},
    {0x39, 0x02},
    {0x43, 0x3A},
    {0x44, 0x72},
    {0x5A, 0x09},
    {0x5B, 0xF6},
    {0x5C, 0xE2},
    {0x66, 0x34},
};

const uint8_t gPN211Page1Table[][2] = {
    {0x27, 0xCA},
    {0x37, 0x15},
    {0x3A, 0x14},
    {0x3D, 0x84},
    {0x43, 0x11},
    {0x46, 0xBC},
    {0x48, 0xB6},
    {0x49, 0xA4},
    {0x4C, 0x58},
};

#endif

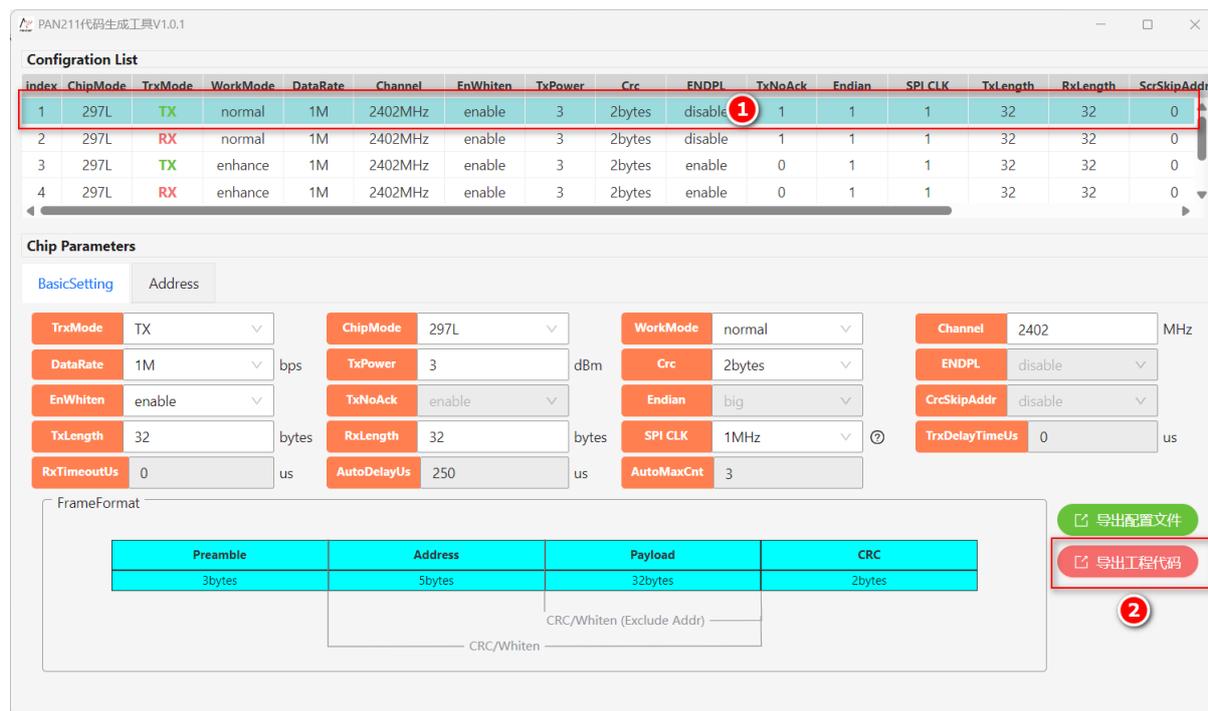
```

注意：使用 easy\_reg.h 头文件的前提是将 pan\_211.h 中 EASY\_RF 定义为 1：

```
#define EASY_RF 1 //使用 easy rf 配置方式
```

#### 4.2.2 代码工程导出

- 选中典型参数配置列表配置 1，再点击右下角的”导出工程代码“



- 选择代码工程要保存的位置后并确定，即可生成” PAN211\_Xn297NormalTx\_134329 “类似格式的 PAN211x 代码工程, 打开 PAN211\_Xn297NormalTx\_134329\example\00\_normal\tx\keil\tx.uvprojx，再进行编译下载

即可快速进行 PAN211x 的发送功能演示。

- 编译运行效果 (通过串口输出的 log 信息)

```
CPU @ 64000000Hz
```

```
SPI-3LINE
```

```
Tx[1]
```

```
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c
1d 1e 1f
```

```
Tx[2]
```

```
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c
1d 1e 1f
```

```
Tx[3]
```

```
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c
1d 1e 1f
```

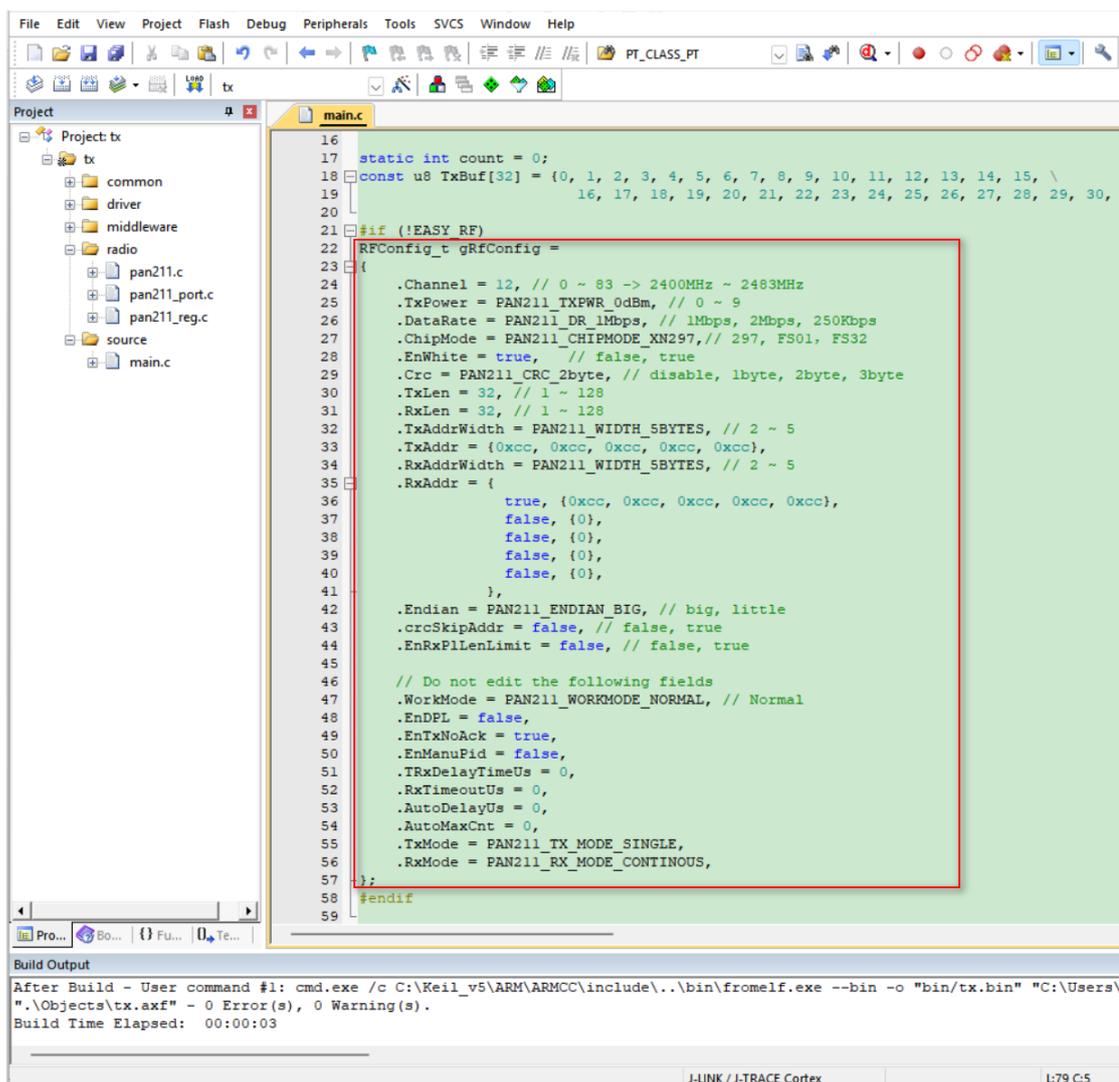
### 4.3 通用例程使用说明

4.3.1 编译开关配置 以 01\_SDK\example\01\_normal\_setup\tx\keil 为示例进行说明，将 pan211.h 中的 EASY\_RF 定义为 0 后，将使用用户自定义的配置参数进行代码编译。

```
//pan211.h
```

```
#define EASY_RF 0 //不使用 easy rf 配置方式
```

此种方式对于用户浏览代码比较友好，用户只要通过修改 gRfConfig 结构体中的各配置参数，然后再调用 PAN211\_SetUpConfig 即可实现对 PAN211x 的参数配置，缺点是编译出的固件会比较大。



#### 4.3.2 gRfConfig 结构体

4.3.3 编译运行效果 编译 01\_SDK\example\01\_normal\_setup\tx\keil 后再下载到 PAN211x EVB 板, 运行串口日志如下:

CPU @ 64000000Hz

SPI-3LINE

Tx[1]

00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e  
1f

Tx[2]

00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e  
1f

Tx[3]

00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e  
1f



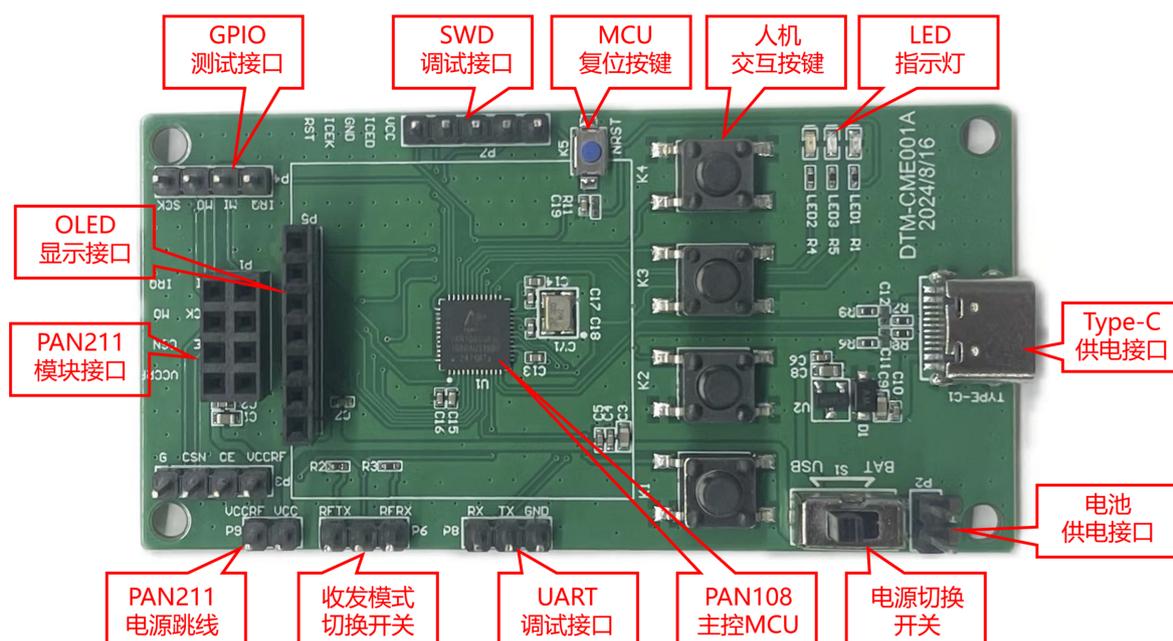
## Chapter 2

# 硬件资料

## 2.1 PAN211x EVB 评估板说明

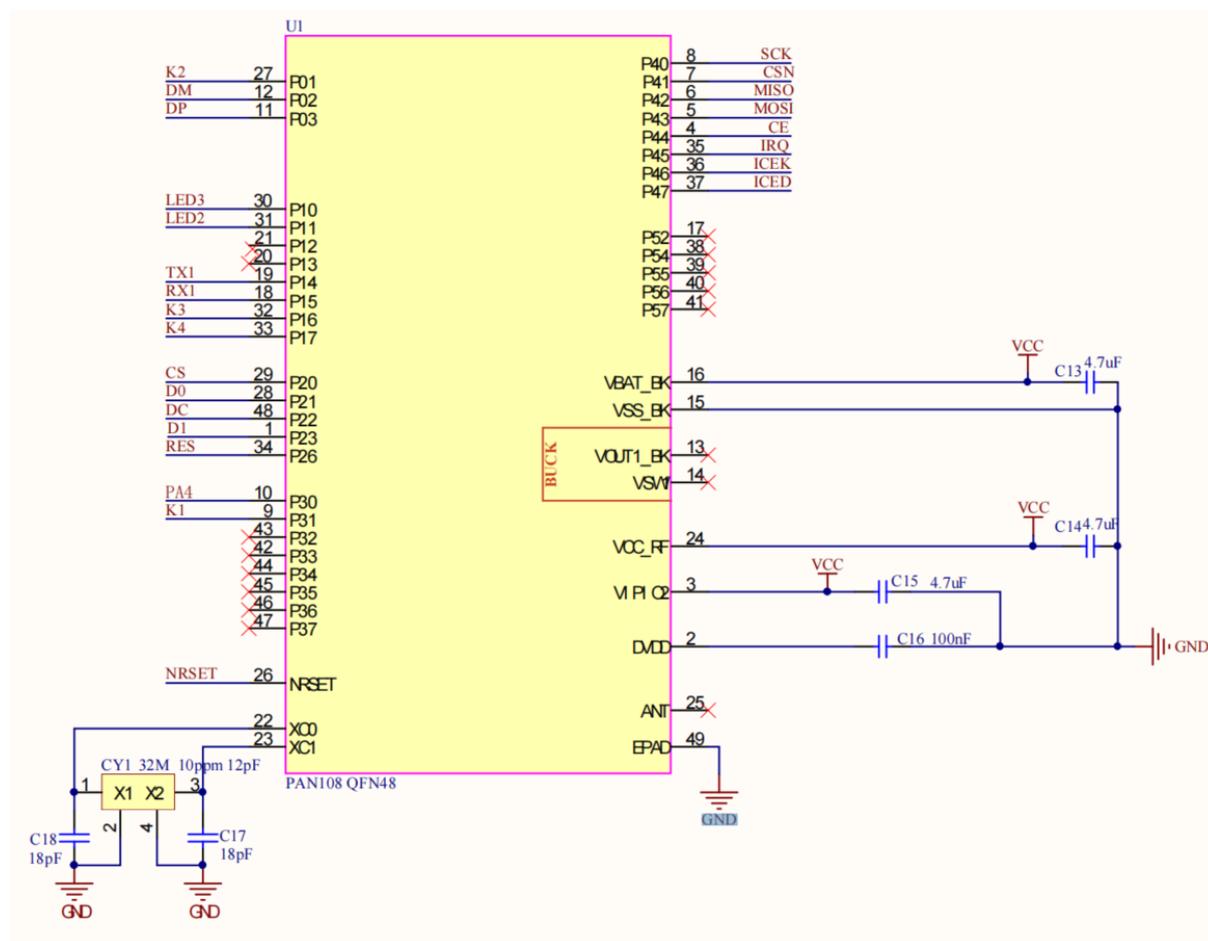
### 2.1.1 1. 简介

PAN211x RF 芯片评估板是一款为 2.4GHz 频段的无线通信应用而设计的硬件开发平台。该评估板采用我司 PAN1080 为主控 MCU，集成了 PAN211x RF 模块和必要的外围接口，支持用户通过多种方式进行 RF 功能评估和开发。评估板提供了丰富的外设接口和配置选项，支持连接 OLED 显示屏、不同供电模式切换以及 RF 模块的收发模式选择。

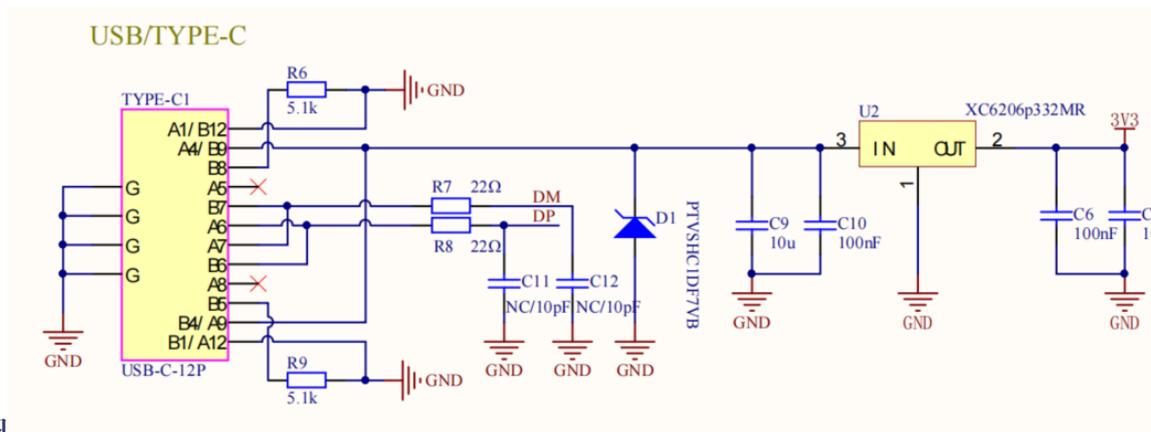


### 2.1.2 2. 底板原理图

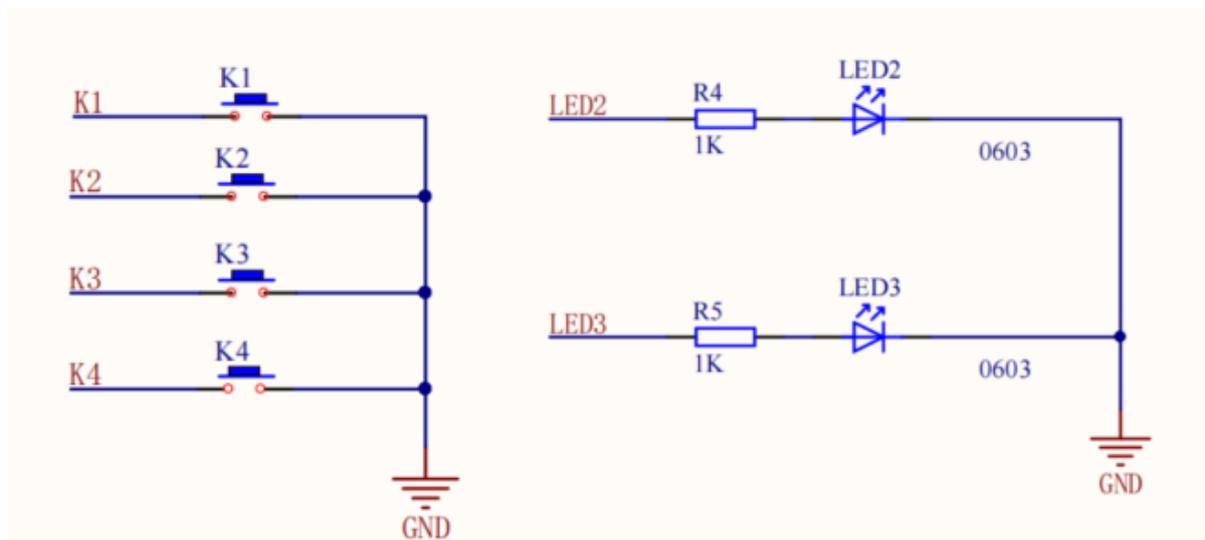
## 2.1 PAN108 主控 MCU 原理图



## 2.2 USB 及供电原理图



## 2.3 按键和指示灯原理图



### 2.1.3 3. 主要功能

- **PAN211x 模块接口:** 评估板上提供了专用的 PAN211x 模块插槽, 支持与外部射频模块的快速连接和配置。
- **128x64 OLED 显示接口:** 评估板集成了标准的 OLED 显示屏接口, 支持 128x64 分辨率的显示, 便于进行信息展示和调试。
- **多种供电方式:** 支持通过 Type-C 接口和电池接口供电, 供电方式可通过电源切换开关选择。
- **收发模式切换:** 通过板载跳线可以方便地在 PAN211x 模块的发送和接收模式之间切换, 以满足不同的测试需求。
- **人机交互按钮:** 板载 4 个交互按钮, 用户可以通过按钮进行操作与调试。
- **GPIO 测试接口:** 提供了标准的 GPIO 接口, 方便用户进行额外的功能扩展和测试。
- **SWD 调试接口:** 支持通过 SWD 接口进行固件下载和调试。
- **UART 调试接口:** 支持通过 UART 进行功能调试和 log 输出。
- **LED 指示灯:** 板载多个 LED 指示灯, 用于显示电源状态、数据传输等工作状态。

### 2.1.4 3. 评估板接口说明

- **PAN211x 模块接口:** 用于连接 PAN211x RF 模块。
- **OLED 显示接口:** 用于连接 128x64 分辨率的 OLED 显示屏。
- **GPIO 测试接口:** 提供 GPIO 信号的调试端口。
- **SWD 调试接口:** 用于连接 SWD 调试器进行固件调试。
- **MCU 复位按钮:** 用于复位板载 MCU。
- **人机交互按钮:** 4 个按钮, 用于与板载 MCU 交互。
- **LED 指示灯:** 显示不同工作状态。
- **Type-C 供电接口:** 支持通过 Type-C 接口供电。
- **电池供电接口:** 支持外部电池供电。
- **PAN211x 电源跳线:** 用于监测 PAN211x 模块的功耗电流。

- **收发模式切换开关**: 用于在 PAN211x 模块的发送和接收模式之间切换。
- **UART 调试接口**: 用于连接 UART 调试器进行串口调试。

#### 2.1.5 4. 电源管理

评估板支持两种供电方式: Type-C 接口供电和电池供电。用户可以通过电源切换开关选择当前的供电方式。建议在进行无线通信测试时, 确保电源稳定, 以保证通信效果。

#### 2.1.6 5. 开发环境与调试

PAN211x 评估板支持通过 SWD 接口进行调试, 用户可以使用 KEIL 进行固件开发。评估板上的 UART 接口也支持串口调试, 便于实时监控 RF 模块的工作状态。

#### 2.1.7 6. 适用范围

该评估板适用于物联网设备、无线传感器网络、低功耗无线通信模块等领域的开发和测试, 特别适合需要评估 2.4GHz 频段 RF 性能的应用场景。

#### 2.1.8 7. 附件与资源

- PAN211x 底板原理图

## Chapter 3

# 演示例程

### 3.1 例程介绍

源码路径: PAN211x\_DK/01\_SDK/example。

- **00\_normal**: 演示 PAN211x 普通型单向通信功能, 使用 EASY\_RF 方式配置参数, 主要支持对代码大小有要求的应用场景。
- **01\_normal\_setup**: 在 00\_normal 例程的基础上进一步补充了非 EASY\_RF 的配置功能, 支持更丰富、自由的参数配置功能。
- **02\_enhance**: 演示 PAN211x 增强型 (双向) 通信功能。
- **03\_normal\_with\_ack**: 演示普通型带 ACK, 也称 M1 型通信功能。
- **04\_carrier\_mode**: 演示载波模式的发射与接收。
- **05\_sleep\_mode**: 演示休眠唤醒功能在普通型单向通信中的应用, 并对基于 SDK 函数作为转移条件的休眠唤醒状态机作出说明。
- **06\_multipipe**: 演示多通道地址接收功能在普通型单向通信中的应用。
- **07\_ble\_beacon**: 演示发射 BLE 广播包的功能。
- **08\_ble\_filter**: 演示 BLE 模式接收功能以及白名单过滤, 长度过滤的功能。
- **09\_ble\_longrange**: 演示 BLE 模式的 S=2, S=8 编码功能。
- **10\_oled**: 基于按键和 OLED 屏幕的人机交互界面。

### 3.2 例程列表

#### 3.2.1 00\_normal 普通型收发例程

##### 1. 简介

本例程演示 PAN211x 普通 (单向) 通信功能, Tx 发射数据, Rx 接收数据。

##### 2. 应用代码说明

**2.1 发射应用代码** TX 端设置为发射模式, 间隔 500ms 发送一次数据, 流程如下:

1. PAN211\_Init() 初始化芯片, 从 DeepSleep 切换到 STB3 状态, 期间写入应用参数并校准。
2. 在发射之前清除中断 PAN211\_ClearIRQFlags

3. 调用 PAN211\_SendPacket 发送数据
4. PAN211\_GetIRQFlags() 获取中断标志, 如果发送成功则会有 RF\_IT\_TX\_IRQ 标志
5. 在发送结束后使用 PAN211\_ClearIRQFlags 清除中断

**2.2 接收应用代码** RX 端设置为接收模式, 轮询是否有数据接收到, 流程如下:

1. PAN211\_Init() 初始化, 从 DeepSleep 切换到 STB3 状态, 期间写入应用参数并校准。
2. 在接收之前清除中断 PAN211\_ClearIRQFlags
3. 调用 PAN211\_RxStart 开始接收
4. 轮询 PAN211\_GetIRQFlags() 获取中断标志, 如果接收成功则会有 RF\_IT\_RX\_IRQ 标志
5. 使用 PAN211\_ReadFifo 读取数据
6. 在接收结束后使用 PAN211\_ClearIRQFlags 清除中断

### 3. 如何修改应用参数

应用参数比如频率、发射功率、速率等以参数表的形式存储在 easy\_reg.h 中, 用户可以根据自己的需求, 使用配置导出工具生成对应的参数表, 使用方法见《PAN211x 参数配置导出工具使用说明》。

PAN211\_Init() 会在初始化过程中将 easy\_reg.h 中的参数表写入到 PAN211 中。

### 4. 如何切换 3 线 SPI, 4 线 SPI, IIC 接口

用户可以通过修改 pan211\_port.h 中的 INTERFACE\_MODE 宏定义来切换接口模式。

3 线 SPI 模式:

```
#define INTERFACE_MODE USE_SPI_3LINE
```

4 线 SPI 模式:

```
#define INTERFACE_MODE USE_SPI_4LINE
```

IIC 模式:

```
#define INTERFACE_MODE USE_I2C
```

### 5. 例程演示

例程位置: 01\_SDK/example/00\_normal。

确保 pan211\_port.h 中的 EASY\_RF 宏定义为 1, 即 #define EASY\_RF 1。

分别打开 tx 和 rx 工程, 编译下载到两个 PAN211x 开发板上。

**TX 端日志:**

```
CPU @ 64000000Hz
SPI-3LINE
Tx[1]
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
Tx[2]
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
Tx[3]
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
```

**RX 端日志:**

```
CPU @ 64000000Hz
SPI-3LINE
Rx[1]:
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
Rx[2]:
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
Rx[3]:
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
```

### 3.2.2 01\_normal\_setup 普通型收发配置例程

#### 1. 简介

本例程在 00\_normal 例程的基础上进一步补充了非 EASY\_RF 的配置功能。另外对 IOMUX 功能也进行了说明。

#### 2. 应用代码说明

2.1. 发射和接收流程 见上一个例程 00\_normal 的说明。

2.2. EASY\_RF 与非 EASY\_RF 配置方式 与 00\_normal 例程相比, 本例程增加了 EASY\_RF 配置选项, 用户可以修改 pan211.h 中的 EASY\_RF 定义来选择不同的配置方式。

##### 2.2.1. EASY\_RF 方式

```
//pan211.h
#define EASY_RF 1
```

这种情况下, 采用与 00\_normal 例程相同的方式进行配置, 上电以后, PAN211\_Init() 函数会自动根据 easy\_reg.h 中的参数表进行初始化, 用户无需再进行额外的参数配置操作。

##### 2.2.2. 非 EASY\_RF 方式

```
//pan211.h
#define EASY_RF 0
```

这种情况下, 用户需要手动配置参数表, 即 gRfConfig 结构体中的参数, 然后调用 PAN211\_SetUpConfig, 此函数会在 PAN211x 已经完成上电初始化的情况下, 重新将参数表写入到 PAN211x 中, 并进行校准, 这一过程的时间会比较长, 可达数十毫秒, 不建议在频繁的操作中使用。

可修改的配置如下:

```
RFConfig_t gRfConfig =
{
    .Channel = 12, // 0 ~ 83 -> 2400MHz ~ 2483MHz
    .TxPower = PAN211_TXPWR_0dBm, // 0 ~ 9
    .DataRate = PAN211_DR_1Mbps, // 1Mbps, 2Mbps, 250Kbps
    .ChipMode = PAN211_CHIPMODE_XN297, // 297, FS01, FS32
    .EnWhite = true, // false, true
    .Crc = PAN211_CRC_2byte, // disable, 1byte, 2byte, 3byte
    .TxLen = 32, // 1 ~ 128
    .RxLen = 32, // 1 ~ 128
    .TxAddrWidth = PAN211_WIDTH_5BYTES, // 2 ~ 5
    .TxAddr = {0xcc, 0xcc, 0xcc, 0xcc, 0xcc},
    .RxAddrWidth = PAN211_WIDTH_5BYTES, // 2 ~ 5
    .RxAddr = {
        true, {0xcc, 0xcc, 0xcc, 0xcc, 0xcc},
```

(下页继续)

```

        false, {0},
        false, {0},
        false, {0},
        false, {0},
    },
    .Endian = PAN211_ENDIAN_BIG, // big, little
    .crcSkipAddr = false, // false, true
    .EnRxPlLenLimit = false, // false, true

    .WorkMode = PAN211_WORKMODE_NORMAL, // Normal
    .EnDPL = false,
    .EnTxNoAck = true,
    .EnManuPid = false,
    .TRxDelayTimeUs = 0,
    .RxTimeoutUs = 0,
    .AutoDelayUs = 0,
    .AutoMaxCnt = 0,
    .TxMode = PAN211_TX_MODE_SINGLE,
    .RxMode = PAN211_RX_MODE_CONTINUOUS,
};

```

注意：EASY\_RF 的两种配置方式本质上一样，这不属于芯片本身的属性，只是为了方便用户使用而提供的两种软件配置方式！

**2.3. IOMUX 功能** 本例程还增加了 IOMUX 功能，IOMUX 用于 3 线 SPI 和 IIC 接口模式下，将 PAN211x 的 SDA 引脚复用为中断输出引脚。

PAN211\_EnableInterfaceMuxIRQ(true); 使能 IOMUX 功能。

PAN211\_ConfigIT(RF\_IT\_TX\_IRQ, true); 将 TX 中断输出到 SDA 引脚，即 SDA 引脚在 TX 中断触发时会输出低电平，否则为高电平。

while(!IRQ\_DETECTED); 表示等待 SDA 引脚输出低电平，即 TX 中断触发。

### 3. 例程演示

例程位置：01\_SDK/example/01\_normal\_setup。

pan211\_port.h 中的 EASY\_RF 宏定义为 0 时，可通过修改 gRfConfig 结构体来配置参数。

pan211\_port.h 中的 EASY\_RF 宏定义为 1 时，可使用配置导出工具替换对应工程的 ./src/easy\_reg.h 文件来配置参数。

分别打开 tx 和 rx 工程，编译后下载到两个 PAN211x 开发板上。

**TX 端日志：**

```

CPU @ 64000000Hz
SPI-3LINE
Tx[1]
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
Tx[2]
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
Tx[3]
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f

```

**RX 端日志：**

```

CPU @ 64000000Hz
SPI-3LINE
Rx[1]:

```

(下页继续)

(续上页)

```

00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
Rx[2]:
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
Rx[3]:
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f

```

### 3.2.3 02\_enhance 增强型发射例程

#### 1. 简介

本例程演示 PAN211x 增强 (双向) 通信功能。

TX 首先发射数据, 然后芯片自动切换至接收状态, 等待接收 ACK 数据包。RX 首先处于接收, 收到数据后会自动切换到发射状态, 发射 ACK 数据包。Tx 发射之后收到 ACK 数据包可视为增强型模式的一次通信成功。

增强模式的优势在于做双向通信时切换的时间比手动快, 同样, 因为芯片是自动切换发射/接收状态, 一定要了解增强模式下的时序切换。

#### 2. 应用代码说明

**2.1. 增强型参数说明与中断处理** 通过与普通型例程的对比, 可以看到增强型的配置方式与普通型的配置方式有所不同。

WorkMode 设置为增强型 (必选)

TxMode 设置为单次发射模式 (必选)

RxMode 设置为单次接收模式 (必选)

**Tx 端:**

TRxDelayTimeUs 设置为 0, 目的是为了在发射数据之后尽快切换到接收状态, 在发射完成后将触发 TX\_IRQ 中断。

RxTimeoutUs 设置为 32000, 目的是为了切换到接收数据之后等待一段时间, 如果没有接收到 ACK 数据包, 自动开始重新发射数据包。如果收到了 ACK 数据包则触发 RX\_IRQ 中断。

AutoMaxCnt 设置为 3, AutoDelayUs 设置为 250, 目的是为了在接收超时之后, 自动重新发射数据包, 最多重复 3 次, 每次超时之后等待 250us 再进入发射状态, 如果重复次数超过 3 次则触发 MAX\_RT\_IRQ 中断。

**Rx 端:**

TRxDelayTimeUs 设置为 16000, 目的是为了在接收数据之后延时一段时间切换到发射状态, 主要是给用户端足够的时间向 FIFO 写入 ACK 数据包的内容, 另一方面也给 Tx 端足够的时间切换到接收状态。

更多参数的说明请见《PAN211x FIFO 和帧格式使用指南》

**2.2. 增强型的动态负载 (Dynamic Payload) 功能** 与普通型不同的是, 增强型可以开启 EnDPL 功能。此通常用于发射端需要发送不同长度数据包的场景。

发送端的数据长度将填入增强型帧格式中的长度字段。

接收端自动解析增强型帧格式的长度字段, 用户通过读取对应的寄存器获取数据包长度。

相关代码如下:

```

u8 RecvLen = PAN211_GetRecvLen();
printf("+RecvLen=%d\n", RecvLen);
if(RecvLen > 0)
{

```

(下页继续)

(续上页)

```
PAN211_ReadFifo(TRX_FIFO, RxBuf, RecvLen);
}
```

当关闭动态数据包长度设置时, 使用固定长度接收数据, 数据包长度由 RxLen 决定。

**2.3. TxNoAck 功能** 与普通型不同的是, 增强型可以开启 TxNoAck 功能。启用无应答发送设置后, 增强型帧格式中的 TxNoAck 字段将置 1, 这要求接收端不发送应答包, 发射端也不会自动重传。

**2.4. RSSI 功能** 普通型和增强型都可以开启 RSSI 功能, 在收到包之后, 可以通过读取 RSSI 寄存器获取收到包的信号强度。

### 3. 例程演示

例程位置: 01\_SDK/example/02\_enhance。

pan211\_port.h 中的 EASY\_RF 宏定义为 0 时, 可通过修改 gRfConfig 结构体来配置参数。

pan211\_port.h 中的 EASY\_RF 宏定义为 1 时, 可使用配置导出工具替换对应工程的 ./src/easy\_reg.h 文件来配置参数。

分别打开 tx 和 rx 工程, 编译后下载到两个 PAN211x 开发板上。

**TX 端:**

```
CPU @ 64000000Hz
SPI-3LINE
>>>RF_IT_TX_IRQ[0x80].1
+RecvLen=32
<<<RF_IT_RX_IRQ[0x01].
RSSI=-20dBm
>>>RF_IT_TX_IRQ[0x80].2
+RecvLen=32
<<<RF_IT_RX_IRQ[0x01].
RSSI=-20dBm
>>>RF_IT_TX_IRQ[0x80].3
+RecvLen=32
<<<RF_IT_RX_IRQ[0x01].
RSSI=-19dBm
```

**RX 端:**

```
CPU @ 64000000Hz
SPI-3LINE
+RecvLen = 32, cnt = 1
+RF_RXMODE=00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b
↔1c 1d 1e 1f
RSSI=-20dBm
>>>RF_IT_TX_IRQ[0x80].
+RecvLen = 32, cnt = 2
+RF_RXMODE=00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b
↔1c 1d 1e 1f
RSSI=-21dBm
>>>RF_IT_TX_IRQ[0x80].
+RecvLen = 32, cnt = 3
+RF_RXMODE=00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b
↔1c 1d 1e 1f
RSSI=-21dBm
>>>RF_IT_TX_IRQ[0x80].
```

(下页继续)

### 3.2.4 03\_normal\_with\_ack 普通型带自动应答收发例程

#### 1. 简介

本例程演示 PAN211x 普通型带 ACK 通信功能。

在上一个 02\_enhance 例程中, 我们演示了 PAN211x 的增强型双向通信功能。PAN211x 也支持普通型带 ACK 的双向通信, 也称为 normal\_m1 型模式。

时序与增强型相同: TX 首先发射数据, 然后芯片自动切换至接收状态, 等待接收 ACK 数据包。RX 首先处于接收, 收到数据后会自动切换到发射状态, 发射 ACK 数据包。Tx 发射之后收到 ACK 数据包可视为增强型模式的一次通信成功。

与增强型不同的地方在于, 帧格式与普通型相同, 因此不支持 Dynamic Payload, PID 等功能。

#### 2. 应用代码说明

2.1. 普通型带 Ack 参数说明与中断处理 通过与增强型例程的对比, 可以看到部分参数需要调整。

WorkMode 设置为普通型 (必选)

TxNoAck 设置为 false (必选)

更多参数的说明请见《PAN211x FIFO 和帧格式使用指南》

2.3. TxNoAck 功能 需要注意, 在增强型的帧格式中, 有一个 TxNoAck 位, 用于控制是否需要 ACK。在普通型带 Ack 通信中, TxNoAck 参数设置为 false 时, 帧格式中并不真实存在 TxNoAck 位, 但是芯片会自动处理 ACK 的发送和接收。

#### 3. 例程演示

例程位置: 01\_SDK/example/03\_normal\_with\_ack。

pan211\_port.h 中的 EASY\_RF 宏定义为 0 时, 可通过修改 gRfConfig 结构体来配置参数。

pan211\_port.h 中的 EASY\_RF 宏定义为 1 时, 可使用配置导出工具替换对应工程的 ./src/easy\_reg.h 文件来配置参数。

分别打开 tx 和 rx 工程, 编译后下载到两个 PAN211x 开发板上。

**TX 端:**

```
CPU @ 64000000Hz
SPI-3LINE
>>>RF_IT_TX_IRQ[0x80].1
<<<RF_IT_RX_IRQ[0x01].
RSSI=-27dBm
>>>RF_IT_TX_IRQ[0x80].2
<<<RF_IT_RX_IRQ[0x01].
RSSI=-26dBm
>>>RF_IT_TX_IRQ[0x80].3
<<<RF_IT_RX_IRQ[0x01].
RSSI=-26dBm
```

**RX 端:**

```

CPU @ 64000000Hz
SPI-3LINE
cnt 1
+RF_RXMODE=01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c
↔1d 1e 1f 20
RSSI=-26dBm
>>>RF_IT_TX_IRQ[0x80].
cnt 2
+RF_RXMODE=01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c
↔1d 1e 1f 20
RSSI=-23dBm
>>>RF_IT_TX_IRQ[0x80].
cnt 3
+RF_RXMODE=01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c
↔1d 1e 1f 20
RSSI=-22dBm
>>>RF_IT_TX_IRQ[0x80].

```

### 3.2.5 04\_carrier 载波模式例程

#### 1. 简介

本例程演示 PAN211x 进入载波模式，该模式常用来测试 RF 频偏。

#### 2. 应用代码说明

芯片上电初始化后，进入载波模式，DEFAULT\_CHANNEL 为载波的频点。

PAN211\_TxCarrierWave() 进入载波模式。

PAN211\_ExitCarrierWave(); 退出载波模式。

#### 3. 例程演示

例程位置：01\_SDK/example/04\_carrier\_mode。

确保 pan211\_port.h 中的 EASY\_RF 宏定义为 0，即 #define EASY\_RF 0。

打开工程，编译后下载到 PAN211 开发板上。

```

CPU @ 64000000Hz
PAN211_Init success.
PAN211_TxCarrierWave success.

```

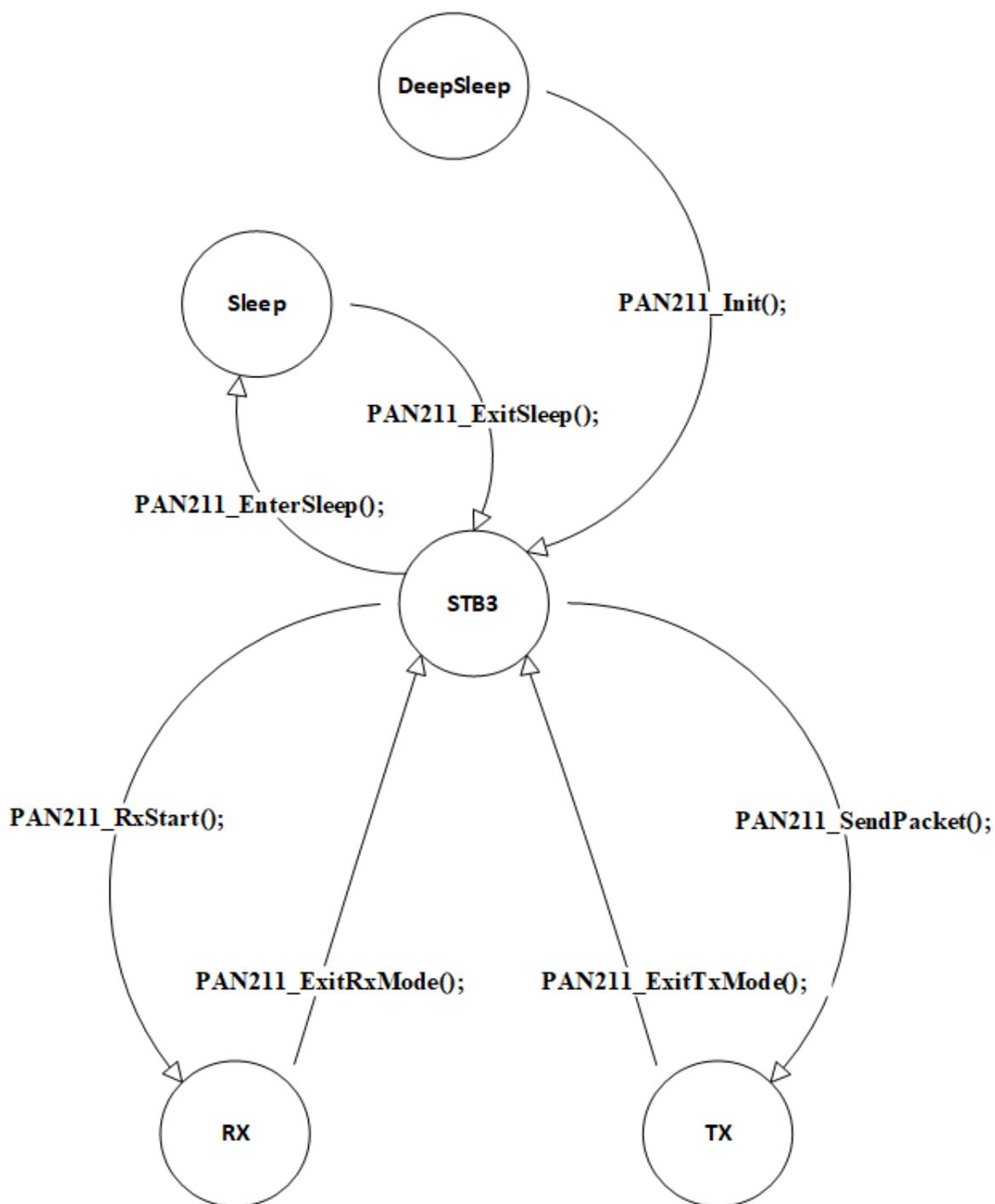
### 3.2.6 05\_sleep\_mode 休眠模式例程

#### 1. 简介

本例程在 00\_normal 例程的基础上，增加了休眠唤醒功能。

#### 2. 应用代码说明

基于 SDK 函数库的状态切换简化流程图如下：



Tx 端休眠唤醒流程:

1. 初始化 PAN211x 芯片: PAN211\_Init()
2. 发射数据: PAN211\_SendPacket()
3. 退出发射模式: PAN211\_ExitTxMode()
4. 进入休眠模式: PAN211\_EnterSleep()
5. 退出休眠模式: PAN211\_ExitSleep()
6. 重复步骤 2

Rx 端休眠唤醒流程:

1. 初始化 PAN211x 芯片: PAN211\_Init()

2. 进入接收模式: PAN211\_RxStart()
3. 退出接收模式: PAN211\_ExitRxMode()
4. 进入休眠模式: PAN211\_EnterSleep()
5. 退出休眠模式: PAN211\_ExitSleep()
6. 重复步骤 2

### 3. 例程演示

例程位置: 01\_SDK/example/05\_sleep\_mode。

pan211\_port.h 中的 EASY\_RF 宏定义为 0 时, 可通过修改 gRfConfig 结构体来配置参数。

pan211\_port.h 中的 EASY\_RF 宏定义为 1 时, 可使用配置导出工具替换对应工程的 ./src/easy\_reg.h 文件来配置参数。

分别打开 tx 和 rx 工程, 编译后下载到两个 PAN211x 开发板上。

TX 端日志:

```
CPU @ 64000000Hz
SPI-3LINE
Tx[1]
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
Tx[2]
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
Tx[3]
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
```

RX 端日志:

```
CPU @ 64000000Hz
SPI-3LINE
Rx[1]:
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
Rx[2]:
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
Rx[3]:
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
```

## 3.2.7 06\_multipipe 多通道地址例程

### 1. 简介

本例程在 00\_normal 例程的基础上, 增加了多通道地址功能。PAN211x 共有 6 个通道可以使用, 每个通道可以绑定不同的地址。其中通道 0,1 可以任意设置 (5bytes)。通道 2,3,4,5 的高 4 bytes 默认与通道 1 相同, 仅最低字节可以更改。接收端收到数据后, 可以通过读取对应的寄存器读出哪个通道接收到。

### 2. 应用代码说明

2.1. 地址填写说明 需要注意: 由于 PAN211x 的地址寄存器排列顺序为 LSB->MSB。在 Tx 端的代码中

```
uint8_t AddressList[6][5] = {
    {0xC0,0xCC,0xCC,0xCC,0xCC},
    {0xC1,0xCC,0xCC,0xCC,0xCC},
    {0xC2,0xCC,0xCC,0xCC,0xCC},
    {0xC3,0xCC,0xCC,0xCC,0xCC},
```

(下页继续)

(续上页)

```
{0xC4,0xCC,0xCC,0xCC,0xCC},
{0xC5,0xCC,0xCC,0xCC,0xCC},
};
```

实际上对应的各通道的地址 (MSB->LSB) 为:

```
Pipe0: 0xCC,0xCC,0xCC,0xCC,0xC0
Pipe1: 0xCC,0xCC,0xCC,0xCC,0xC1
Pipe2: 0xCC,0xCC,0xCC,0xCC,0xC2
Pipe3: 0xCC,0xCC,0xCC,0xCC,0xC3
Pipe4: 0xCC,0xCC,0xCC,0xCC,0xC4
Pipe5: 0xCC,0xCC,0xCC,0xCC,0xC5
```

用户可以通过 PAN211\_SetTxAddr 接口来修改发射地址。

**2.2. 通道号读取** 在 Rx 端的代码中, 可以通过 PAN211\_GetRxPipeNum 接口来获取当前接收到的数据是哪个通道的。

```
u8 PipeNum = (u8)PAN211_GetRxPipeNum();
PAN211_ReadFifo(TRX_FIFO, RxBuf, 32);
printf("Rx[%d] on pipe %d\r\n", count++, PipeNum);
```

### 3. 例程演示

例程位置: 01\_SDK/example/06\_multipipe。

pan211\_port.h 中的 EASY\_RF 宏定义为 0 时, 可通过修改 gRfConfig 结构体来配置参数。

pan211\_port.h 中的 EASY\_RF 宏定义为 1 时, 可使用配置导出工具替换对应工程的 ./src/easy\_reg.h 文件来配置参数。

分别打开 tx 和 rx 工程, 编译后下载到两个 PAN211x 开发板上。

**TX 端日志:**

```
CPU @ 64000000Hz
SPI-3LINE
Tx[0] on addr 0
Tx[1] on addr 1
Tx[2] on addr 2
Tx[3] on addr 3
Tx[4] on addr 4
```

**RX 端日志:**

```
CPU @ 64000000Hz
SPI-3LINE
Rx[0] on pipe 0
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
Rx[1] on pipe 1
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
Rx[2] on pipe 2
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
Rx[3] on pipe 3
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
Rx[4] on pipe 4
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f
```

### 3.2.8 07\_ble\_beacon 蓝牙广播例程

#### 1. 简介

本例程演示了如何使用 PAN211x 模块实现 BLE beacon 功能。

#### 2. 应用代码说明

2.1. 参数配置说明 Channel 配置为广播频道 37。

DataRate 配置为 1Mbps, BLE 模式还支持 2Mbps 和 250Kbps

TxLen 为广播包长度 (不包括 header)

.BleHeadNum 配置为 2, 表示 Header 由 2 个字节组成, 分别是 BleHead0 和 length, 其中 length 由 TxLen 决定。

.BleHead0 配置为 0x42 表示 ADV\_NONCONN\_IND 类型广播包。

.BleHead1 配置为 0

.WhiteInit 配置为 PAN211\_BLE\_WH\_INIPHA\_CH37, 与广播频道对应。

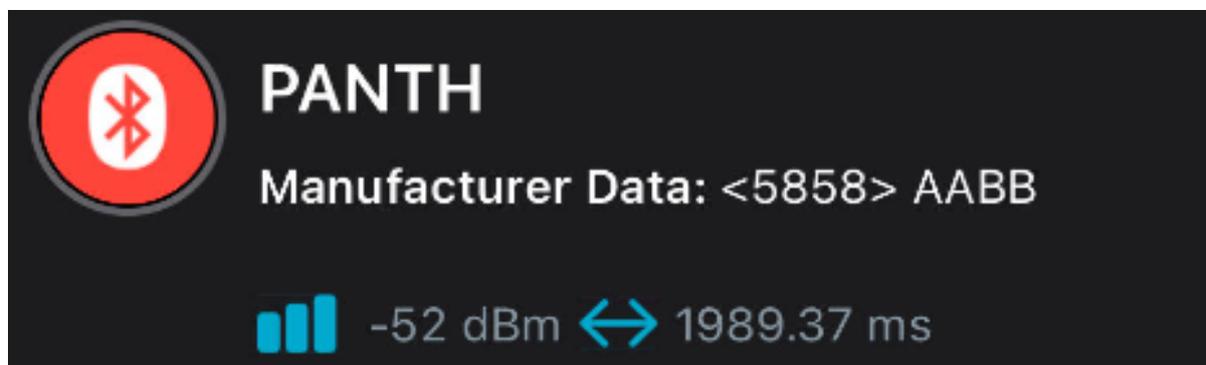
#### 3. 例程演示

例程位置: 01\_SDK/example/07\_ble\_beacon。

确保 pan211\_port.h 中的 EASY\_RF 宏定义为 0, 即 #define EASY\_RF 0。

打开工程, 编译后下载到 PAN211x 开发板上。

蓝牙抓包使用手机 APP 抓包, 可以看到对应的广播包。



日志:

```
CPU @ 64000000Hz
SPI-3LINE
Tx[1]
Tx[2]
Tx[3]
```

### 3.2.9 08\_ble\_filter 蓝牙过滤例程

#### 1. 简介

本例程演示了如何使用 PAN211x 模块实现 BLE 白名单过滤和长度过滤功能。

## 2. 应用代码说明

**2.1. 参数配置说明** 白名单配置如下: WhiteListMatchMode 配置为 5 字节。WhileListOffset 配置为 0, 表示从 Payload 第一个字节开始匹配 (不包含 header)。WhilteList 配置为 {0xcc, 0xcc, 0xcc, 0xcc, 0xcc}, 表示过滤其中的 ADVA 地址。WhilteListLen 配置为 5。长度过滤如下: LengthFilterMode 配置为 Disable, 也可以分别配置为 exceed 或 beneath, 长度过滤的参考值为 RxLen。

## 3. 例程演示

例程位置: 01\_SDK/example/08\_ble\_filter。

确保 pan211\_port.h 中的 EASY\_RF 宏定义为 0, 即 #define EASY\_RF 0。

打开工程, 编译后下载到 PAN211x 开发板上。

日志:

```
CPU @ 64000000Hz
irq flag:0x01
Rx done[1]
Header:0x42 RxLen:0x13 Payload:cc cc cc cc cc cc 06 09 50 41 4e 54 48 05 ff 58
58 aa bb
irq flag:0x01
Rx done[2]
Header:0x42 RxLen:0x13 Payload:cc cc cc cc cc cc 06 09 50 41 4e 54 48 05 ff 58
58 aa bb
irq flag:0x01
Rx done[3]
Header:0x42 RxLen:0x13 Payload:cc cc cc cc cc cc 06 09 50 41 4e 54 48 05 ff 58
58 aa bb
```

### 3.2.10 09\_ble\_longrange 蓝牙长距离模式例程

#### 1. 简介

本例程演示了如何使用 PAN211x 模块实现 BLE Long Range 模式。

#### 2. BLE Long Range 模式介绍

BLE Long Range 模式也称为 Coded PHY 模式, 在蓝牙 5 协议版本中引入, 其目的是为了满足蓝牙应用在长距离通信上的需求。

原始的蓝牙协议支持 1Mbps 和 2Mbps 通信速率。长距离模式在 1Mbps 的基础上经过编码, 每个数据位由 2 或 8 个符号表示, 也就是 S2 或 S8 编码模式。编码后的速率变成了 500Kbps 或 250kbps, 如下表:

PHY	Modulation scheme	Coding scheme		Data rate
		Access Header	Payload	
LE 1M	1 Msym/s modulation	Uncoded	Uncoded	1 Mb/s
LE 2M	2 Msym/s modulation	Uncoded	Uncoded	2 Mb/s
LE Coded	1 Msym/s modulation	S=8	S=8	125 kb/s
			S=2	500 kb/s

Table 3.1: Summary of PHYs, modulation schemes, and coding schemes

下图反映了这种编码模式在帧格式上如何体现：

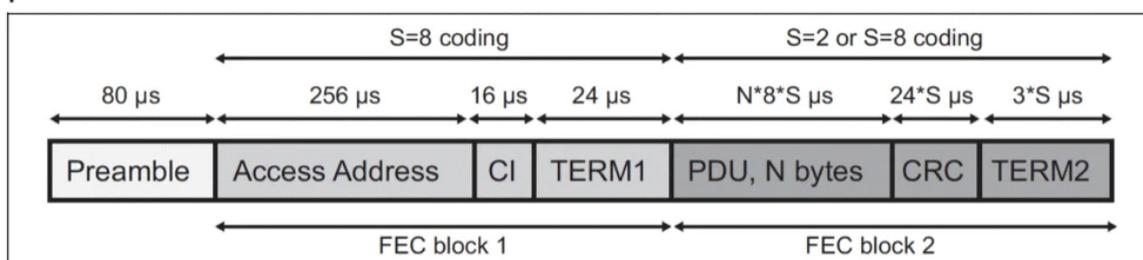


Figure 2.3: Link Layer packet format for the LE Coded PHY

图中，FEC Block 1 始终使用 S8 编码。CI（编码指示符）用于指示在 FEC 块 2 中使用哪种编码方案（S=2 或 S=8）。FEC Block 2 根据 CI 的配置使用 S2 或 S8 编码。

PAN211 在发射模式的 1Mbps 和 250kbps 下都支持 S2 和 S8 编码，在接受模式下，只要开启了相应的配置，硬件自动识别编码标识符，可以同时接收 S2 或 S8 的数据包。

### 3. 应用代码说明

基于 07\_ble\_beacon 例程和 08\_ble\_filter 例程，通过修改 .S2S8Mode 参数，我们可以很容易的实现 BLE Long Range 模式。

#### 4. 例程演示

例程位置：01\_SDK/example/09\_ble\_longrange。

确保 pan211\_port.h 中的 EASY\_RF 宏定义为 0，即 #define EASY\_RF 0。

打开工程，编译后分别下载到两个 PAN211x 开发板上。

**TX 端日志：**

```
CPU @ 64000000Hz
SPI-3LINE
Tx[1]
Tx[2]
Tx[3]
```

**RX 端日志：**

```
CPU @ 64000000Hz
```

```
CPU @ 64000000Hz irq flag:0x01 Rx done[1] Header:0x42 RxLen:0x13 Payload:cc cc cc cc cc cc 06 09
50 41 4e 54 48 05 ff 58 58 aa bb irq flag:0x01 Rx done[2] Header:0x42 RxLen:0x13 Payload:cc cc cc cc cc
cc 06 09 50 41 4e 54 48 05 ff 58 58 aa bb irq flag:0x01 Rx done[3] Header:0x42 RxLen:0x13 Payload:cc
cc cc cc cc cc 06 09 50 41 4e 54 48 05 ff 58 58 aa bb
```

### 3.2.11 10\_oled OLED 评估例程

#### 1. 简介

本例程演示了基于按键和 OLED 屏幕的人机交互界面。

- 支持单包发射、1000 包发射、连续发射、连续接收、载波模式、休眠模式一键测试，无需重复烧录代码。

- 支持 OLED 菜单的页面切换、参数和配置、收发包数统计、错误包数统计、测试参数预览等丰富的评估功能。
- 支持 TX/RX 模式上电自动选择。

## 2. 例程演示

例程位置: 01\_SDK/example/10\_oled。

确保 pan211\_port.h 中的 EASY\_RF 宏定义为 0, 即 `#define EASY_RF 0`。

打开工程, 编译后下载到 PAN211x 开发板上。

通过调整收发模式切换开关, 可以选择不同的 Tx/Rx 模式。参考《PAN211x EVB 快速开发指南》。

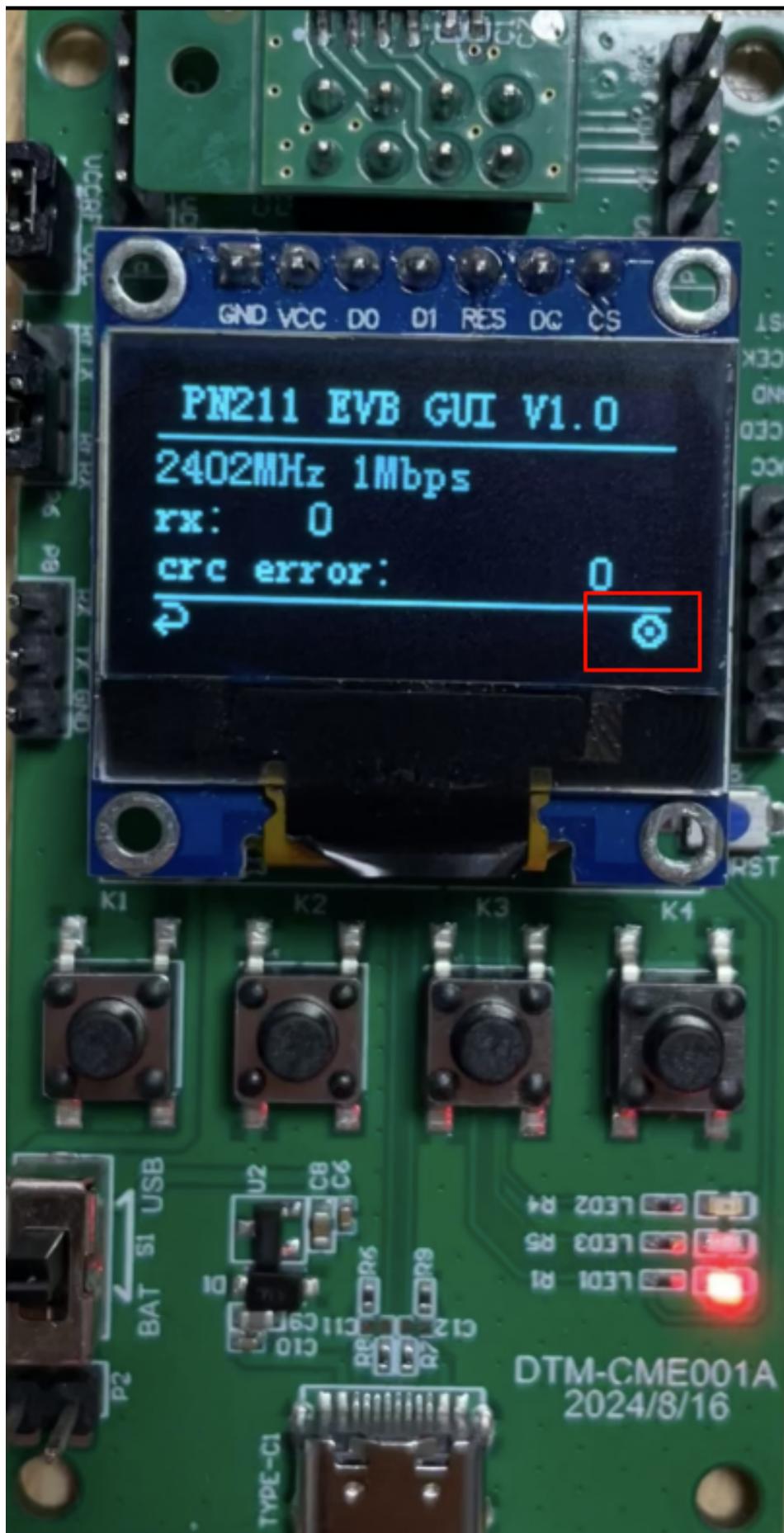
Tx 模式上电后, 默认为 1000 包-发射模式。OLED 屏幕显示如下: K1 为返回键, 按下 K4 为开始发射, 再次按下 K4 为停止发射。



Rx 模式上电后，默认为连续接收模式。OLED 屏幕显示如下：K1 为返回键，按下 K4 为开始接收，再次按下 K4 为停止接收。



按下 K4 后可以看到右边的图标中多了一个小圆点，表示正在发射或接收数据。



在 TX/RX 模式下，按下 K1 键可以返回到参数选择界面。参数选择界面可以设置发射功率、发射频率等参数。OLED 屏幕显示如下：K1 为返回键，K2 为上翻键，K3 为下翻键，K4 为确认键。按下返回键后，可以返回到发射或接收界面。按下上下翻键可以切换参数，按下确认键可以进入参数设置界面。



参数设置界面中，可以设置参数具体的值。K1 为返回键，K2 为上翻键，K3 为下翻键，K4 为确认键。按下返回键后，可以返回到参数设置界面。按下上下翻键可以切换参数，按下确认键同样返回到参数选择界面。



## Chapter 4

# 开发指南

### 4.1 PAN211x 例程移植指南

#### 4.1.1 1. 驱动文件移植

用户需要移植时关心以下驱动文件：

1. `pan211.c` & `pan211.h` 包含 PAN211x 初始化、参数配置、FIFO 操作以及状态的定义。
2. `pan211_reg.c` & `pan211_reg.h` 包含 PAN211x 寄存器操作和 FIFO 操作的定义。
3. `pan211_port.c` & `pan211_port.h` 包含 3 线 SPI/4 线 SP/IIC 的底层软件模拟实现方式。以使用 3 线 SPI 为例，使用者需要根据所用的 MCU 实现相关接口。

Step 1: 实现 `pan211_port.c` 中下列接口：

- `rf_delay_us` 微秒级延时函数
- `rf_delay_ms` 毫秒级延时函数
- `RF_BusInit` SPI 接口相关寄存器初始化操作

Step 2: 实现 `pan211_port.h` 中下列接口：

- `SPI_DATA_OUTPUT` : SPI 数据线输出模式
- `SPI_DATA_INPUT`: SPI 数据线输入模式
- `SPI_SCK_LOW`: SPI 时钟线拉低
- `SPI_SCK_HIGH`: SPI 时钟线拉高
- `SPI_DATA_HIGH`: SPI 数据线拉高
- `SPI_DATA_LOW`: SPI 数据线拉高
- `IRQ_DETECTED`: SPI 数据线与 IRQ 分时复用时，检测 SPI 数据线是否为低电平

使用者可以根据需求，删除不需要的代码。

#### 4.1.2 2. EASY\_RF 使用

##### 2.1 简介

EASY\_RF 驱动提供了一套极简的配置方法。使用预先导出的配置表，代码极简，能够满足大部分简单的应用场景的需求。

## 2.2 EASY\_RF 在例程中的使用方法

**参数预配置** 为了达到极致的性能, PAN211 提供了非常丰富的参数配置选项, 以 297 模式为例, PAN211 提供了频点、速率、CRC、白化、地址、数据长度等二十余个参数可配置。用户可以使用配置导出工具导出配置, 在经过调校的参数表基础上, 实现极简应用代码。

**参数预写入** 修改 pan211.h, 使能 EASY\_RF, 如:

```
#define EASY_RF 1
```

开启后, 在 PAN211\_Init 函数中, 会写入 pan211\_reg.h 中给出的寄存器表。

使用者也可以通过配置导出工具修改参数, 导出并替换掉工程中原有的 easy\_reg.h 头文件。

## 2.3 配置导出工具使用方法

见《PAN211 配置生成工具使用指南》。

## 2.4 EASY\_RF 与一般配置方法的区别

### 4.1.3 QA

**QA1: 如果只想要最简单的代码应该改?**

将所有以下格式的代码删除即可:

```
#if (!EASY_RF)
...
#endif
```

## 4.2 PAN211x FIFO 和帧格式使用指南

### 4.2.1 1. FIFO 操作

PAN211x 有两个 64 字节的 FIFO。在普通型模式下, 只存在收或发单向通信, 可以拼接为 128 字节使用。在增强型或 M1 模式下仅支持 64bytes, 分别为发送和接收 FIFO。

FIFO 仅能存储和读取单个数据包的信息。如果 FIFO 中已存有一个数据包, 在写入新数据前应先读取完毕, 否则前一个数据包将被覆盖。FIFO 不支持清空操作, 写入新的数据时会覆盖原有 FIFO 的内容。

### 4.2.2 2. 工作模式 (WorkMode) 与协议帧格式 (ChipMode)

PAN211x 支持各种不同的帧格式和工作模式。

工作模式 (WorkMode) 分为不带自动重传和 ACK 的通信模式 (后简称为 Normal 型)、带自动重传和 ACK 通信模式 (后简称为 Enhance 模式) 以及基于 Normal 型帧格式带自动重传和 ACK 的通信模式 (后简称为 Normal\_M1 模式)。

为了兼容不同的私有协议帧格式, PAN211x 帧格式 (ChipMode) 分为 297 帧格式, FS01 帧格式, FS32 帧格式和 BLE 帧格式。

#### 2.1. XN297 帧格式

普通型 297 帧格式:

增强型 297 帧格式:

## 2.2. 扩展协议帧一 FS01

普通型 FS01 帧格式:

## 2.3. 扩展协议帧二 FS32

普通型 FS32 帧格式:

增强型 FS32 帧格式:

## 2.4. BLE 帧格式

BLE 兼容帧格式:

下面我们将分别介绍在使用不同的私有协议帧格式 (297L, FS01, FS32) 和蓝牙帧格式时应当如何配置参数。

### 4.2.3 3. 私有协议帧格式

**频道 (Channel)**: 通道号, 指定无线通信所用的频率通道。范围: 2400MHz ~ 2483MHz

**功率 (TxPower)**: 发射功率设置, 较高的功率可以提高通信范围, 但也可能增加功耗和干扰。范围: 0dBm ~ 9dBm

**速率 (DataRate)**: 数据传输速率设置, 较高的数据速率可以提高通信速度, 但可能降低传输距离或增加错误率。范围: 1Mbps, 2Mbps, 250Kbps

#### 3.1. 前导码

前导码设置, 与特定的协议帧格式有关, 不能由寄存器配置。

前导码用于使接收器解调器与输入比特流同步。

#### 3.2. 地址

**发射地址长度 (TxAddrWidth)**: 发射地址宽度设置, 指定发射地址的字节数。范围: 3 ~ 5

**发射地址 (TxAddr)**: 发射地址设置, 指定发射地址。范围: [0x00 ~ 0xFF, 0x00 ~ 0xFF, 0x00 ~ 0xFF, 0x00 ~ 0xFF]

**接收地址长度 (RxAddrWidth)**: 接收地址宽度设置, 指定接收地址的字节数。范围: 3 ~ 5

**接收地址 (RxAddr)**: 接收地址设置, 指定接收地址。范围: [0x00 ~ 0xFF, 0x00 ~ 0xFF, 0x00 ~ 0xFF, 0x00 ~ 0xFF]

#### 3.3. 数据

**发射数据长度 (TxLen)**: 发射数据长度设置, 指定发射数据的字节数。范围: 0 ~ 128 (工作模式为普通型); 0 ~ 64 (工作模式为增强型)

**接收数据长度 (RxLen)**: 接收数据长度设置, 指定接收数据的字节数。范围: 0 ~ 128 (工作模式为普通型); 0 ~ 64 (工作模式为增强型)

### 3.4. CRC

**CRC 校验 (CRC):** CRC 校验设置可以提高通信的可靠性。范围: Disable, 1byte, 2byte, 3byte

**CRC 跳过地址 (CRCSkipAddr):** 指定 CRC 校验跳过地址, 只有在 FS32 模式才可配。范围: false, true (FS32 模式); false (297L, FS01 模式)

### 3.5. 白化扰码

**白化 (EnWhite):** 启用白化可以提高通信抗干扰能力。范围: true, false

**白化跳过地址 (ScrSkipAddr):** 指定白化扰码跳过地址, 各芯片模式使用固定配置。范围: false (297L, FS01 模式); true (FS32 模式)

### 3.6. 大小端

**大小端 (Endian):** 指定字节序, 只有在 FS32 模式才可配。范围: Little, Big (FS32 模式); Big (297L, FS01 模式)

### 3.7. 长度标识

长度标识参数在增强型帧格式中用于标识数据包的长度, 发射端的数据长度将填入增强型帧格式中的长度字段。接收端自动解析增强型帧格式的长度字段, 用户可通过读取对应的寄存器获取数据包长度。

**使能动态数据长度 (EnDPL):**

仅增强型模式可用, 默认使能。当关闭此功能时, 数据包实际收发长度为固定长度, 由 TxLen 和 RxLen 决定。启用此功能时, 通常用于发射端需要发送不同长度数据包的场景, 接收端可以根据长度标识动态解析数据长度。范围: true, false

**使能长度过滤限制 (EnRxPILenLimit):**

仅增强型模式可用。针对增强模式自动匹配包长的情况下增加最大包长限制, 如果自动匹配的包长大于最大限制包长, 则退出此模式。该功能在 rx\_goon 打开的情况下, 接收包长错误时, 按照最大长度接收后, 可以重新进入接收, 直到接收到正确的数据帧或者超时退出。范围: true, false

### 3.8. PID 标识

PID 参数用于确定发射端数据的唯一性, 发射端的 PID 将填入增强型帧格式中的 PID 字段。接收端根据需要自动解析增强型帧格式的 PID 字段, 用户通过读取对应的寄存器获取数据包 PID。

**使能手动 PID (EnManuPid):**

仅增强型模式可用, 默认不使能。启用手动 PID 设置, 用户可以通过设置寄存器手动设置 PID 值, 否则 PID 值由芯片自动设置。范围: true, false

### 3.9. 无应答标识

**使能无应答 (EnTxNoAck):**

普通型和增强型都可用。启用无应答发送设置后, 要求接收端不发送应答包, 发射端也不会自动重传。增强型模式默认应该关闭此功能。普通型模式关闭此功能之后, 芯片在接收到数据后会自动切换到发射状态, 发送应答包, 应答包帧格式不包含标识符。这种模式也称为 normal M1 模式。范围: true, false

### 3.10. 时间参数

#### 接收超时 (RxTimeoutUs):

普通型和增强型都可用。增强型模式中，指发射端的应答包接收超时时间设置，单位为 us。发射端在发送数据包之后，如果在 RxTimeoutUs 时间内没有收到应答包，会自动重发上一次的数据包。普通型模式中，如果采用单次带超时的接受模式，该配置指接收端的接收超时时间设置，单位为 us。接收端如果在 RxTimeoutUs 时间内没有收到数据包，会触发超时中断。范围：0 ~ 65535

#### 收发切换延时 (TRxDelayTimeUs):

仅增强型模式可用。发射接收切换延时时间设置，指定发射接收切换延时时间，单位为 us。范围：0 ~ 32767

#### 自动重传延时 (AutoDelayUs):

仅增强型模式可用。自动应答发射延时时间设置，指定发射端在没有收到应答包之后，自动发射应答包的延时时间，单位为 us。范围：250, 500, 750, 1000..., 3750, 4000

#### 自动重传最大次数 (AutoMaxCnt):

仅增强型模式可用。自动重传的最大次数设置，指定发射端在没有收到应答包之后，自动发射应答包的最大次数。如果设置为 0，则不会自动重传。范围：0 ~ 15

### 3.11. 收发模式

#### 发射模式 (TxMode):

普通型模式可选：单次发送，连续发送。增强型模式可选：单次发送。

**接收模式 (RxMode):** 普通型模式可选：单次接收，连续接收，单次带超时的接收。增强型模式可选：单次接收。

### 3.11. 软件示例代码

```
RFConfig_t gRfConfig =
{
    .Channel = 12,
    .TxPower = PAN211_TXPWR_0dBm,
    .DataRate = PAN211_DR_1Mbps,
    .ChipMode = PAN211_CHIPMODE_XN297,
    .EnWhite = true,
    .Crc = PAN211_CRC_2byte,
    .TxLen = 32,
    .RxLen = 32,
    .TxAddrWidth = PAN211_WIDTH_5BYTES,
    .TxAddr = {0xcc, 0xcc, 0xcc, 0xcc, 0xcc},
    .RxAddrWidth = PAN211_WIDTH_5BYTES,
    .RxAddr = {
        true, {0xcc, 0xcc, 0xcc, 0xcc, 0xcc},
        false, {0},
        false, {0},
        false, {0},
        false, {0},
    },
    .Endian = PAN211_ENDIAN_BIG,
    .crcSkipAddr = false,
    .EnRxPllLenLimit = false,

    .WorkMode = PAN211_WORKMODE_NORMAL,
    .EnDPL = false,
    .EnTxNoAck = true,
}
```

(下页继续)

```

.EnManuPid = false,
.TRxDelayTimeUs = 0,
.RxTimeoutUs = 0,
.AutoDelayUs = 0,
.AutoMaxCnt = 0,
.TxMode = PAN211_TX_MODE_SINGLE,
.RxMode = PAN211_RX_MODE_CONTINUOUS,
};

```

#### 4.2.4 4. 蓝牙帧格式

**频道 (Channel)** : 见上一章

**功率 (TxPower)** : 见上一章

**速率 (DataRate)** : 数据传输速率设置, 较高的数据速率可以提高通信速度, 但可能降低传输距离或增加错误率。范围: 1Mbps, 2Mbps, 250Kbps (普通 BLE 模式); 1Mbps, 250Kbps (S2, S8 模式)

#### 4.1. 数据

**发射数据长度 (TxLen)**: 发射数据长度设置, 指定发射数据的字节数。范围: 0 ~ 128 (工作模式为普通型); 0 ~ 64 (工作模式为增强型)

**接收数据长度 (RxLen)**: 接收数据长度设置, 指定接收数据的字节数。范围: 0 ~ 128 (工作模式为普通型); 0 ~ 64 (工作模式为增强型)

#### 4.2. 白化初始值

**白化初始值 (WhiteInit)**:

指定白化扰码初始化值。在特定的广播频道上需要使用特定的白化扰码初始化值。37 广播频道对应白化扰码初始化值为 0x53, 38 广播频道对应白化扰码初始化值为 0x33, 39 广播频道对应白化扰码初始化值为 0x73。范围: 0x00 ~ 0xFF

#### 4.3. BLE 帧头

**帧头字节数 (BLEHeadNum)**:

BLE 帧头字节数设置, 指定 BLE 帧头字节数。范围: 0 ~ 3

**帧头标识符 0 (BleHead0)**:

BLE 帧头 header 第一个字节设置, 指定 BLE 帧头第一个字节。范围: 0x00 ~ 0xFF

**帧头标识符 1 (BleHead1)**:

BLE 帧头 header 第二个字节设置, 指定 BLE 帧头第二个字节。范围: 0x00 ~ 0xFF

#### 4.4. 白名单过滤

**白名单匹配模式 (WhilteListMatchMode)**: 范围: 0 ~ 6 (0: 不匹配; 1 ~ 6: 匹配 1~6 个字节)

**白名单匹配偏移 (WhilteListOffset)**: 范围: 0 ~ 127

**白名单 (WhilteList)**: 范围: 0x00 ~ 0xFF

**白名单长度 (WhilteListLen)**: 范围: 1 ~ 6

#### 4.5. 长度过滤

##### 长度过滤模式 (LengthFilterMode):

接受模式下的长度过滤, 长度过滤的比较值为 RxLen。范围: 0 ~ 3(0: 不过滤; 1: 仅接受长度等于 RxLen 的数据包; 2: 仅接受长度大于 RxLen 的数据包; 3: 仅接受长度小于 RxLen 的数据包)

#### 4.6. 扩频模式

##### 扩频模式 (S2S8Mode):

S2S8 模式设置, 指定 S2S8 模式。范围: disable, S2, S8

#### 4.7. 其他参数

BLE 模式中, 其他参数为固定值, 在参数配置时不需要额外关注。

Parameter	Value
ChipMode	BLE
WorkMode	普通型
TxAddrWidth	4
TxAddr	0x8e89bed6
RxAddrWidth	4
RxAddr	0x8e89bed6
EnWhite	true
CRCSkipAddr	true
ScrSkipAddr	true
Endian	Little
CRC	3byte
EnDPL	true

#### 4.8. 软件示例代码

```
BLEConfig_t Config =
{
    .Channel = BLE_CHANNEL_37,
    .DataRate = PAN211_DR_1Mbps,
    .TxPower = PAN211_TXPWR_3dBm,
    .TxLen = sizeof(BleTxPacket)/sizeof(u8),
    .RxLen = 0,

    .BleHeadNum = 2,
    .BleHead0 = BLE_ADV_HEADER,
    .BleHead1 = 0,

    .S2S8Mode = PAN211_PRIMODE_DIS,
    .WhiteInit = PAN211_BLE_WH_INIPHA_CH37,
    .WhiteListMatchMode = PAN211_BLE_WhiteList_DISABLE,
    .WhiteListOffset = 0,
    .WhiteList = {0},
    .WhiteListLen = 0,
    .LengthFilterMode = PAN211_BLE_LEN_FILTER_DISABLE,
};
```

### 4.3 常见问题 (FAQs)

#### 4.3.1 Q1: 为什么 PAN211x EVB 开发板 SWD 下载固件老是失败?

导致这种问题的原因可能性比较多:

1. 检查 SWC, SWD, ,RESET, GND 连接是否正常
2. 如果烧写线比较长, 请将烧写速率配置在 500K 以下
3. PAN1080\_1020KB\_FLASH\_KEIL.FLM 文件是否正确放入 keil 安装目录

#### 4.3.2 Q2: PAN211x 收发两端为什么不能正常通信?

RF 收发通信需要多个参数匹配才能正常通信, 具体包括以下几点:

1. 工作模式: 保证收发两端处于相同的工作模式
2. 通信速率: 保证收发两端处于相同的通信速率
3. 通信地址: 保证收发两端通信地址长度和内容都相同
4. 数据长度: 普通模式时(固定包长)保证收发两端通信数据长度要相同
5. CRC 校验: CRC 校验的开启/关闭两端要匹配
6. 距离较远时, 选择高功率, 低速率方式通信

#### 4.3.3 Q3: PAN211x 通信距离能达到多远?

距离跟板子的 layout, 发射端使用的功率和通信的速率都有比较大的关系。

1M 速率下, 设置 9dbm 功率, 空旷距离可达 100 米以上。

#### 4.3.4 Q4: PAN211x 可以设置多大的功率?

芯片最低功率可以设置为-20dbm, 最高可以设置 8dbm。中间有多个功率档位可供选择, 具体请根据实际应用场景设置发射功率, 通常建议使用 8dbm 以下。

#### 4.3.5 Q5: PAN211x 休眠功耗为多大?

3.3V 供电情况下, shutdown(deepsleep) 模式下最低功耗电流可达 0.3uA; sleep 模式下最低功耗电流可达 0.8uA。

## Chapter 5

# 开发工具

### 5.1 PAN211x 代码生成工具

#### 5.1.1 1. 简介

PAN211x 代码生成工具是 Shanghai Panchip Microelectronics Co.,Ltd. 为 PAN211x 提供生成配置文件和代码工程的功能。用户可根据自定义配置生成特定的配置头文件 (.h) 或者 KEIL 收发代码工程，用户无需关心 sdk 中具体寄存器定义，节省用户开发时间，提升效率。

PAN211x 代码生成工具 (PAN211x\_ES\_TOOL) 下载链接

#### 5.1.2 2. 参数配置界面说明

##### 2.1 基本参数配置界面

如下图所示，此处包含芯片基本参数为：

- **芯片类型**：可选：297L、24101、52832
- **速率可选**：off, 1byte, 2bytes,3bytes
- **通信频点**：支持 2400~2485
- **功率档位**：支持 0~9dBm
- **工作模式**：可选普通型 (normal)、增强型 (enhance)。当选择普通型时，动态 payload 和大小端不可配。

Configuration List															
Index	ChipMode	TrxMode	WorkMode	DataRate	Channel	EnWhiten	TxPower	Crc	ENDPL	TxNoAck	Endian	SPI CLK	TxLength	RxLength	ScrSkipAddr
1	297L	TX	normal	1M	2402MHz	enable	3	2bytes	disable	1	1	1	32	32	0
2	297L	RX	normal	1M	2402MHz	enable	3	2bytes	disable	1	1	1	32	32	0
3	297L	TX	enhance	1M	2402MHz	enable	3	2bytes	enable	0	1	1	32	32	0
4	297L	RX	enhance	1M	2402MHz	enable	3	2bytes	enable	0	1	1	32	32	0

##### 2.2 地址参数配置界面

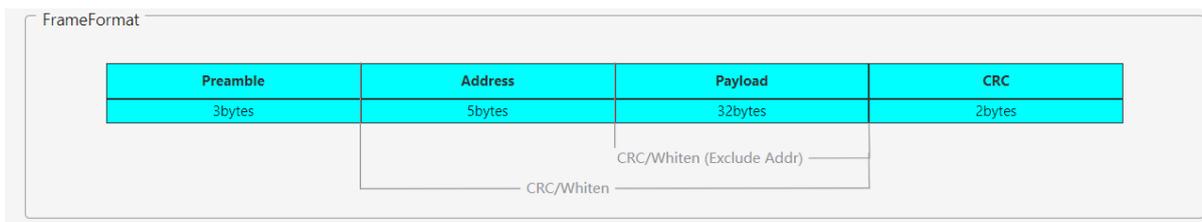
- TX 地址直接在 Tx 地址区域配置，分别填写 5 个输入框中值（16 进制）。
- RX 地址如果使用单通道（即在上方的“多通道地址”选择框中，选择关闭），直接在 Rx 地址区域配置，分别填写 5 个输入框中值（16 进制）。
- RX 地址如果使用多通道（即在上方的“多通道地址”选择框中，选择打开），需要在下方设置多通道地址。

- 五个通道的地址 [39:32]、地址 [31:24]、地址 [23:16]、地址 [15:8] 一致，地址 [7:0] 可不一致。



### 2.3 帧格式显示界面

根据用户在参数区的配置，如选择 ChipMode、WorkMode、地址长度、crc 长度、TxLen/RxLen(Payload) 后，下图的帧格式实时刷新。



## 5.1.3 3. 代码导出方法说明

### 3.1 参数配置选择

1. 点击上方列表中的其中一个典型推荐配置，下方的参数区及帧格式区会自动载入相应的典型配置参数，用户可按照实际需求修改部分参数。

### 3.2 配置头文件导出

- 参照 3.1 选择一个典型配置，并按需求更改配置参数



- 点击“导出配置文件”按钮，即可预览用户配置生成的 easy\_reg.h 头文件内容，具体如下图所示。

Code Preview window showing the generated header file content:

```

PAN211代码生成工具V1.0.1
< 代码预览 导出头文件

* EndPL = off
* TRxDelayTimeUs = 0us
* RxTimeoutUs = 0us
* AutoDelayUs = 250us
* AutoMaxCnt = 3
* TxMode = Single
* RxMode = Continuous
*****/

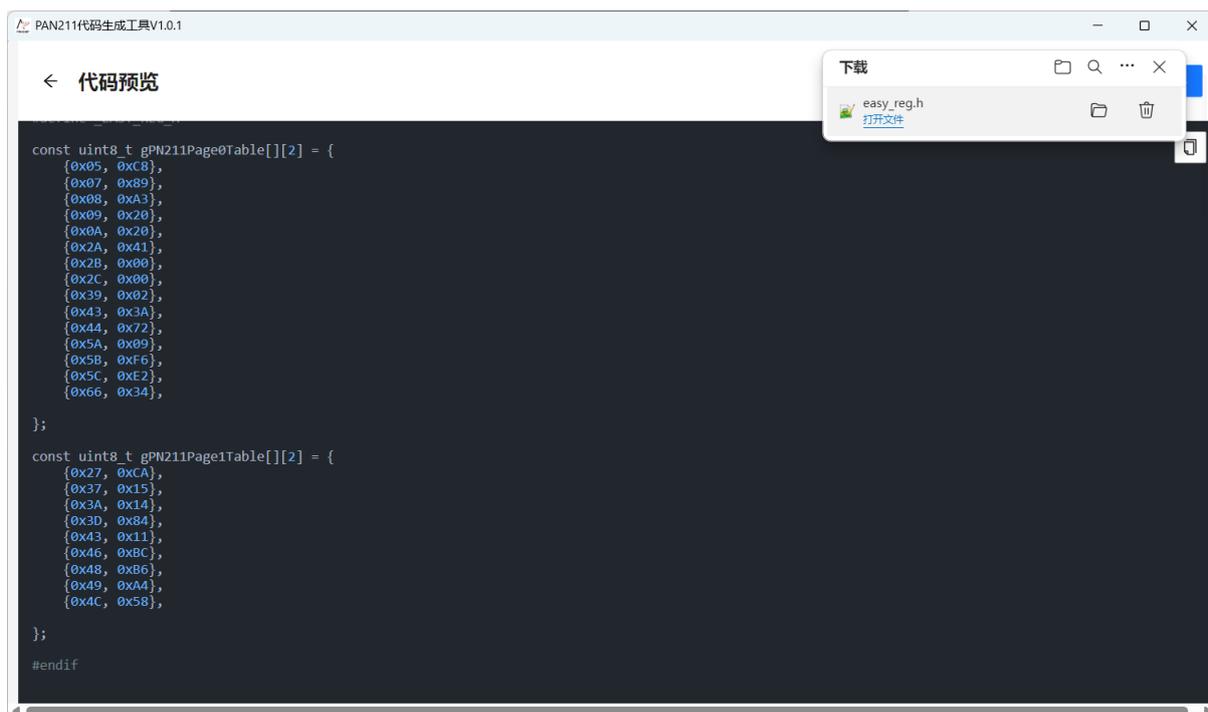
#ifndef _EASY_REG_H
#define _EASY_REG_H

const uint8_t gPN211Page0Table[][2] = {
    {0x05, 0xc8},
    {0x07, 0x89},
    {0x08, 0xa3},
    {0x09, 0x20},
    {0x0a, 0x20},
    {0x2a, 0x41},
    {0x2b, 0x00},
    {0x2c, 0x00},
    {0x39, 0x02},
    {0x43, 0x3a},
    {0x44, 0x72},
    {0x5a, 0x09},
    {0x5b, 0xf6},
    {0x5c, 0xe2},
    {0x66, 0x34},
};

const uint8_t gPN211Page1Table[][2] = {
    {0x27, 0xca},
    {0x37, 0x15},
    {0x3a, 0x14},
    {0x3d, 0x84},
};

```

- 点击右上角导出按钮，自动下载头文件到本地。



### 3.3 KEIL 代码工程导出

- 点击“导出工程代码”按钮，弹出导出路径选择，选择好路径后，点击保存。



- 工具根据参数区中的 TrxMode 和 WorkMode 参数，生成指定的工程代码。
  - 普通型对应的工程代码路径：
    - TX: example/00\_normal/tx
    - RX: example/00\_normal/rx
  - 增强型对应的工程代码：
    - TX: example/02\_enhance/tx
    - RX: example/02\_enhance/rx

## Chapter 6

# 其他文档

PAN211x 相关的其他文档请参考：

- PAN211x 产品说明书



# Chapter 7

## ChangeLog

### 7.1 PAN211x SDK V1.0.0 Release Notes

#### 7.1.1 1. SDK

##### 改动:

- 将原有 SDK 代码迁移到 pan1080 评估板 v1.1。
- 引入 EASY\_RF 配置方法。
  - EASY\_RF 驱动提供了一套极简的配置方法，使用预先导出的配置表，代码极简，能够满足大部分简单的应用场景的需求。
  - 用户可以配合 TOOLS 目录下的配置导出工具修改并生成参数，替换掉工程中原有的 easy\_reg.h 头文件。
  - 原有的直接通过函数接口配置参数的方法也保留在函数库中，支持更丰富的参数配置功能。

##### 新增:

- 00\_normal 例程。演示 PAN211x 普通型单向通信功能，使用 EASY\_RF 方式配置参数，主要支持对代码大小有要求的应用场景。
- 01\_normal\_setup 例程。在 00\_normal 例程的基础上进一步补充了非 EASY\_RF 的配置功能，支持更丰富、自由的参数配置功能。
- 02\_enhance 例程。演示 PAN211x 增强型 (双向) 通信功能。
- 03\_normal\_with\_ack 例程。演示普通型带 ACK，也称 M1 型通信功能。
- 04\_carrier\_mode 例程。演示载波模式的发射。
- 05\_sleep\_mode 例程。演示休眠唤醒功能在普通型单向通信中的应用，并对基于 SDK 函数作为转移条件的休眠唤醒状态机作出说明。
- 06\_multipipe 例程。演示多通道地址接收功能在普通型单向通信中的应用。
- 07\_ble\_beacon 例程。演示发射蓝牙广播包的功能。
- 08\_ble\_filter 例程。演示蓝牙接收功能以及白名单过滤，长度过滤的功能。
- 09\_ble\_longrange 例程。演示蓝牙长距离模式的 S=2，S=8 编码功能。
- 10\_oled 例程。基于按键和 OLED 屏幕的人机交互界面。
  - 支持单包发射、1000 包发射、连续发射、连续接收、载波模式、休眠模式一键测试，无需重复烧录代码。
  - 支持 OLED 菜单的页面切换、参数和配置、收发包数统计、错误包数统计、测试参数预览等丰富的评估功能。

- 支持 TX/RX 模式上电自动选择。

各例程对 EASY\_RF 的支持如下:

例程	支持 EASY_RF	支持非 EASY_RF
00_normal	Y	N
01_normal_setup	Y	Y
02_enhance	Y	Y
03_normal_with_ack	Y	Y
04_carrier_mode	N	Y
05_sleep_mode	Y	Y
06_multipipe	Y	Y
07_ble_beacon	TBD	Y
08_ble_filter	TBD	Y
09_ble_longrange	TBD	Y
10_oled	N	Y

### 7.1.2 2. HDK

- 上传了 pan211x 评估底板的原理图。

### 7.1.3 DOC

#### 新增:

- 新增《PAN211x FIFO 和帧格式使用指南》
- 新增《PAN211x 配置生成工具使用指南》
- 新增《PAN211x 例程移植指南》

### 7.1.4 3. TOOLS

- 更新了 pan211x\_es\_tool.exe, 使用方法见《PAN211x 配置生成工具使用指南》