



GR533x LCP开发者指南

版本： 1.0

发布日期： 2023-10-15

版权所有 © 2023 深圳市汇顶科技股份有限公司。保留一切权利。

非经本公司书面许可，任何单位和个人不得对本手册内的任何部分擅自摘抄、复制、修改、翻译、传播，或将其全部或部分用于商业用途。

商标声明

GOODIX 和其他汇顶商标均为深圳市汇顶科技股份有限公司的商标。本文档提及的其他所有商标或注册商标，由各自的所有人持有。

免责声明

本文档中所述的器件应用信息及其他类似内容仅为您提供便利，它们可能由更新之信息所替代。确保应用符合技术规范，是您自身应负的责任。

深圳市汇顶科技股份有限公司（以下简称“GOODIX”）对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保，包括但不限于针对其使用情况、质量、性能、适销性或特定用途的适用性的声明或担保。GOODIX对因这些信息及使用这些信息而引起的后果不承担任何责任。

未经GOODIX书面批准，不得将GOODIX的产品用作生命维持系统中的关键组件。在GOODIX知识产权保护下，不得暗或以其他方式转让任何许可证。

深圳市汇顶科技股份有限公司

总部地址：深圳市福田区腾飞工业大厦B座12-13层

电话：+86-755-33338828 邮编：518000

网址：www.goodix.com

前言

编写目的

本文档主要介绍GR533x LCP（Light Communication Protocol）协议（Goodix自定义的2.4 GHz私有协议）的数据包格式和应用模型等，旨在帮助用户快速开发基于LCP协议的通信应用。

读者对象

本文适用于以下读者：

- 芯片用户
- 软件开发人员
- 测试人员

版本说明

本文档为第1次发布，对应的软件为GR533x SDK。

修订记录

版本	日期	修订内容
1.0	2023-10-15	首次发布

目录

前言.....	I
1 简介.....	1
2 数据包格式.....	2
3 应用模型.....	3
3.1 通信模式.....	3
3.1.1 软件触发发送模式LCP_TRX_MODE_SW_TX.....	3
3.1.2 软件触发接收模式LCP_TRX_MODE_SW_RX.....	3
3.1.3 Timer触发发送模式LCP_TRX_MODE_TIMER_TX.....	3
3.1.4 Timer触发接收模式LCP_TRX_MODE_TIMER_RX.....	3
3.2 通信角色.....	3
3.2.1 PTX时序.....	4
3.2.2 PRX时序.....	4
3.3 LCP API.....	4
3.3.1 结构体.....	5
3.3.1.1 gdx_lcp_config_t.....	5
3.3.2 枚举.....	6
3.3.2.1 PROTOCOL_MODE.....	6
3.3.2.2 LCP_RATE.....	6
3.3.3 接口函数.....	7
3.3.3.1 gdx_lcp_init.....	7
3.3.3.2 gdx_lcp_deinit.....	7
3.3.3.3 gdx_lcp_timer_binding.....	7
3.3.3.4 gdx_lcp_tx_power_set.....	8
3.3.3.5 gdx_lcp_tx_power_get.....	8
3.3.3.6 gdx_lcp_channel_set.....	8
3.3.3.7 gdx_lcp_channel_get.....	8
3.3.3.8 gdx_lcp_data_tx.....	9
3.3.3.9 gdx_lcp_auto_txrx_mode_set.....	9
3.3.3.10 gdx_lcp_rx_oneshot_mode_set.....	9
3.3.3.11 gdx_lcp_rx_window_size_set.....	9
3.3.3.12 gdx_lcp_rx_start.....	10
3.3.3.13 gdx_lcp_rx_stop.....	10
3.3.3.14 gdx_lcp_whitening_seed_set.....	10
3.3.3.15 gdx_lcp_t2r_turn_around_time_adjust.....	10
3.3.3.16 gdx_lcp_r2t_turn_around_time_adjust.....	11
3.4 创建LCP应用.....	11
3.4.1 软件触发收发模式的应用.....	11
3.4.2 Timer触发收发模式的应用.....	11

3.4.3 脉冲定检示例工程.....12

1 简介

GR533x系列芯片集成2.4 GHz射频（RF）收发器，可支持低功耗蓝牙协议与Goodix自定义的2.4 GHz私有协议—LCP（Light Communication Protocol）协议。

LCP协议主要特性：

- 工作频点：2360 MHz～2520 MHz频段内的偶数频点
- 传输速率：GFSK 1 Mbps/2 Mbps
- 支持循环冗余校验（CRC）
- 支持白化（Whitening）
- 收发模式
 - 软件触发收发（单次）
 - Timer触发收发（周期性）

2 数据包格式

LCP协议的数据包格式如下图所示：

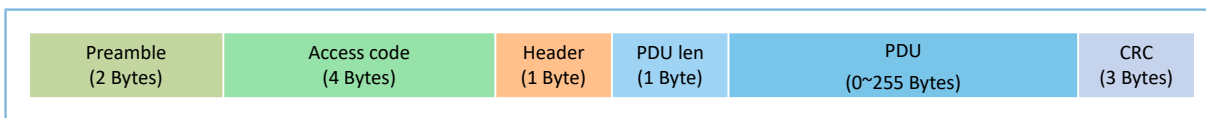


图 2-1 LCP协议数据包格式

- **Preamble:** 前导码，2字节，由SDK包自动确定的0、1交替序列，其最后一个bit与Access code的第一个bit相异。
- **Access code:** 访问码，4字节，用于标识设备，可自定义配置。接收端只能接收指定Access code的数据，GR533x仅支持同一时刻检查1个Access code。
- **Header:** 包头，1字节，由用户自定义。
- **PDU len:** 数据包长度，1字节。
- **PDU:** 用户数据，0~255字节。实际长度由PDU len决定。
- **CRC:** 循环冗余校验码，3字节，CRC校验算法与蓝牙SPEC指定的算法相同。用户可配置使能或禁用CRC校验。

上述参数，除PDU之外，其他位域的长度均固定，不可配置。


3 应用模型

本章主要描述LCP协议的通信模式、角色、API接口，以及LCP应用的实现流程和参考示例。

3.1 通信模式

根据RF行为触发源类型，可将LCP收发模式分为以下四类：

- 软件触发接收模式（LCP_TRX_MODE_SW_RX）：由软件显示调用LCP RX接口，接收数据包。
- 软件触发发送模式（LCP_TRX_MODE_SW_TX）：由软件显示调用LCP TX接口，发送数据包。
- Timer触发接收模式（LCP_TRX_MODE_TIMER_RX）：由Timer自动触发调用LCP RX接口，接收数据包。
- Timer触发发送模式（LCP_TRX_MODE_TIMER_TX）：由Timer自动触发调用LCP TX接口，发送数据包。

 说明:

LCP协议支持四种Timer：APP_DUAL_TIM_ID_0、APP_DUAL_TIM_ID_1、APP_TIM_ID_0、APP_TIM_ID_1。

3.1.1 软件触发发送模式LCP_TRX_MODE_SW_TX

在该模式下，当应用程序调用发包接口gdx_lcp_data_tx后，立即（ASAP）发送数据包。

3.1.2 软件触发接收模式LCP_TRX_MODE_SW_RX

在该模式下，当应用程序调用收包接口gdx_lcp_rx_start后，立即（ASAP）启动RX。

3.1.3 Timer触发发送模式LCP_TRX_MODE_TIMER_TX

在该模式下，当应用程序调用发包接口gdx_lcp_data_tx后，在指定的Timer时刻触发发送数据包，并且在指定的间隔后，自动转换为RX模式。

3.1.4 Timer触发接收模式LCP_TRX_MODE_TIMER_RX

在该模式下，当应用程序调用收包接口gdx_lcp_rx_start后，在指定的Timer时刻触发接收数据包，并且在指定的间隔后，自动转换为TX模式。

3.2 通信角色

根据LCP应用中的通信模式，可将设备分为以下两类角色：

- PTX：发送设备，支持LCP_TRX_MODE_SW_TX和LCP_TRX_MODE_TIMER_TX两种通信模式。
- PRX：接收设备，支持LCP_TRX_MODE_SW_RX和LCP_TRX_MODE_TIMER_RX两种通信模式。

3.2.1 PTX时序

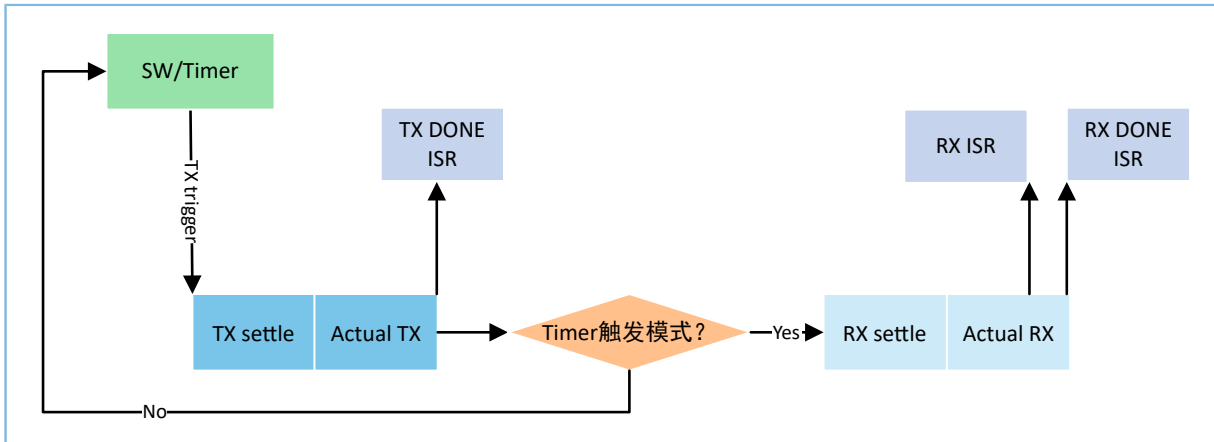


图 3-1 PTX时序

说明:

仅Timer触发模式支持T2R（TX to RX）的自动转换功能，且该功能默认使能。

3.2.2 PRX时序

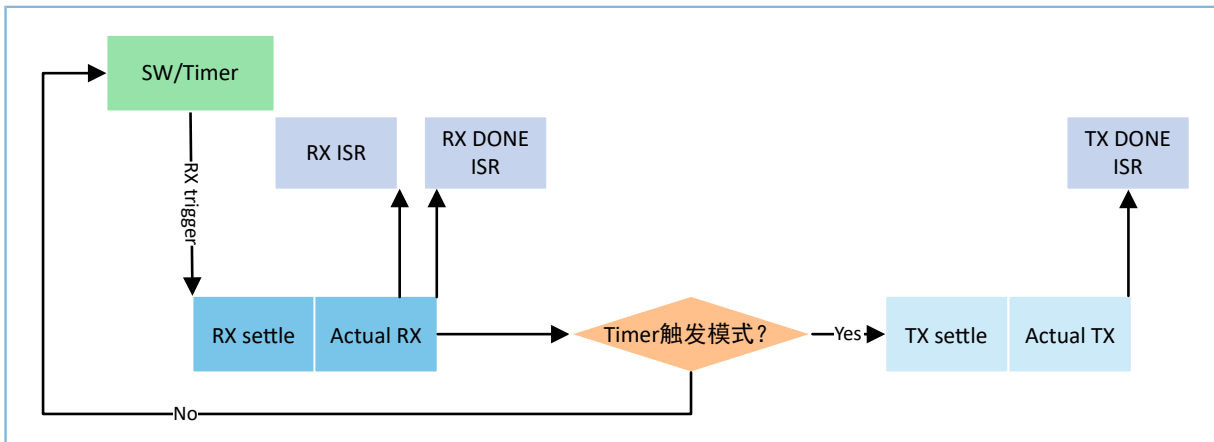


图 3-2 PRX时序

说明:

仅Timer触发模式支持R2T（RX to TX）的自动转换功能，且该功能默认使能。

3.3 LCP API

GR533x SDK提供了丰富的LCP API，可方便用户使用LCP协议。

提示:

关于LCP API的详细定义，可参考SDK_Foler\components\sdk\ble_lcp.h文件，其中SDK_Foler为GR533x SDK包根目录。

3.3.1 结构体

3.3.1.1 gdx_lcp_config_t

gdx_lcp_config_t结构体用于定义LCP通信的环境变量，例如收发模式、Access code、回调注册等信息。

- 结构体定义

```
typedef struct {
    uint8_t  trx_mode;
    int8_t   txpwr_dbm;
    uint32_t freq_mhz;
    uint32_t access_address;
    uint32_t crc_init;
    uint32_t rx_window_size_us;
    uint8_t  rate;
    bool     whiten_en;
    bool     b_disable_rx_oneshot_mode;
    uint32_t trx_timer_period_us;
    uint32_t trx_timer_trigger_trx_time_us;
    tx_done_cb_t tx_done_cb;
    rx_done_cb_t rx_done_cb;
    rx_handler_cb_t rx_handler_cb;
} gdx_lcp_config_t;
```

- 成员说明

表 3-1 gdx_lcp_config_t结构体成员

成员	类型	描述
trx_mode	uint8_t	LCP收发模式，具体定义参考 3.3.2.1 PROTOCOL_MODE 。
txpwr_dbm	int8_t	发射功率，取值范围： - 20 dBm~4 dBm
freq_mhz	uint32_t	工作频点，取值： 2360 MHz~2520 MHz频段内的偶数频点。
access_address	uint32_t	设备地址
crc_init	uint32_t	CRC初始值。 0：禁用CRC，不进行校验。
rx_window_size_us	uint32_t	RX开窗宽度，取值范围0~10239687。 0：开无限窗
rate	uint8_t	LCP传输速率，支持1 Mbps、2 Mbps。
whiten_en	bool	是否使能白化。
b_disable_rx_oneshot_mode	bool	是否禁用 RX单次接收模式（Timer触发模式下，仅支持单次接收模式）。 <ul style="list-style-type: none"> ◦ true：禁用RX单次接收模式，即使用RX连续接收模式。当收包时间超过rx_window_size时，退出RX模式。 ◦ false：使用RX单次接收模式，即接收到一个数据包后就退出RX模式。

成员	类型	描述
trx_timer_period_us	uint32_t	Timer触发模式下的收发周期时间。
time_before_trx_timer_expired_us	uint32_t	当使用Timer触发模式时，Timer计时直至触发TX/RX行为。 说明：该计时时间必须小于Timer周期。
tx_done_cb	tx_done_cb_t	TX完成中断回调
rx_done_cb	rx_done_cb_t	RX完成中断回调
rx_handler_cb	rx_handler_cb_t	RX处理中断回调

3.3.2 枚举

3.3.2.1 PROTOCOL_MODE

PROTOCOL_MODE枚举用于定义LCP收发模式。

- 枚举定义

```
enum PROTOCOL_MODE {
    LCP_TRX_MODE_SW_TX = 2,      /*TX by software, just once */
    LCP_TRX_MODE_SW_RX = 3,      /*RX by software, just once */
    LCP_TRX_MODE_TIMER_TX = 4,    /*TX by timer with specified period, will stop when
the timer stop */
    LCP_TRX_MODE_TIMER_RX = 5,    /*RX by timer with specified period, will stop when
the timer stop */
};
```

- 成员说明

表 3-2 PROTOCOL_MODE枚举成员

成员	值	描述
LCP_TRX_MODE_SW_TX	2	软件触发TX模式（单次）
LCP_TRX_MODE_SW_RX	3	软件触发RX模式（单次）
LCP_TRX_MODE_TIMER_TX	4	Timer触发TX模式（周期性）
LCP_TRX_MODE_TIMER_RX	5	Timer触发RX模式（周期性）

3.3.2.2 LCP_RATE

LCP_RATE枚举用于定义LCP传输速率。

- 枚举定义

```
enum LCP_RATE {
    LCP_RATE_1MBPS = 0,
    LCP_RATE_2MBPS,
};
```

- 成员说明

表 3-3 LCP_RATE枚举成员

成员	值	描述
LCP_RATE_1MBPS	0	传输速率为1 Mbps。
LCP_RATE_2MBPS	1	传输速率为2 Mbps。

3.3.3 接口函数

3.3.3.1 gdx_lcp_init

表 3-4 gdx_lcp_init函数

函数原型	uint16_t gdx_lcp_init(gdx_lcp_config_t *gdx_lcp_config)
功能说明	初始化LCP模块。
输入参数	gdx_lcp_config: 指向gdx_lcp_config_t 结构体的指针，该结构体包含LCP通信的环境变量，例如收发模式、传输速率、频点等配置信息。
返回值	<ul style="list-style-type: none">• 0: 成功• 其他: 失败
备注	需在Bluetooth LE协议栈初始化完成后才能调用该接口函数。

3.3.3.2 gdx_lcp_deinit

表 3-5 gdx_lcp_deinit函数

函数原型	void gdx_lcp_deinit(void)
功能说明	反初始化LCP模块。
输入参数	无
返回值	无
备注	无

3.3.3.3 gdx_lcp_timer_binding

表 3-6 gdx_lcp_timer_binding函数

函数原型	void gdx_lcp_timer_binding(bool b_dual_timer, uint8_t timer_id)
功能说明	绑定LCP通信行为和Timer，以实现Timer自动触发收发行为。
输入参数	<ul style="list-style-type: none">• b_dual_timer: 指定timer类型。<ul style="list-style-type: none">◦ true: app dual timer◦ false: app timer• timer_id: 0或1
返回值	无
备注	<ul style="list-style-type: none">• 支持APP_DUAL_TIM_ID_0/1和APP_TIM_ID_0/1四种Timer。• 仅当使用Timer触发收发模式时，才需要调用该接口函数。

3.3.3.4 gdx_lcp_tx_power_set

表 3-7 gdx_lcp_tx_power_set函数

函数原型	uint16_t gdx_lcp_tx_power_set(int8_t txpwr_dbm)
功能说明	设置数据包的发射功率。
输入参数	txpwr_dbm: 发射功率, 取值范围 - 20 dBm~4 dBm
返回值	<ul style="list-style-type: none"> 0: 成功 其他: 失败
备注	

3.3.3.5 gdx_lcp_tx_power_get

表 3-8 gdx_lcp_tx_power_get函数

函数原型	uint16_t gdx_lcp_tx_power_get(int8_t *txpwr_dbm)
功能说明	获取数据包的发射功率。
输入参数	*txpwr_dbm: 指向发射功率的指针。
返回值	<ul style="list-style-type: none"> 0: 成功 其他: 失败
备注	

3.3.3.6 gdx_lcp_channel_set

表 3-9 gdx_lcp_channel_set函数

函数原型	uint16_t gdx_lcp_channel_set(uint32_t freq_mhz)
功能说明	设置工作频点。
输入参数	freq_mhz: 2360 MHz~2520 MHz频段内的偶数频点。
返回值	<ul style="list-style-type: none"> 0: 成功 其他: 失败
备注	LCP模块初始化完成后, 才能调用该接口函数。

3.3.3.7 gdx_lcp_channel_get

表 3-10 gdx_lcp_channel_get函数

函数原型	uint16_t gdx_lcp_channel_get(uint32_t *freq_mhz)
功能说明	获取工作频点。
输入参数	freq_mhz: 指向工作频点的指针。
返回值	<ul style="list-style-type: none"> 0: 成功 其他: 失败
备注	

3.3.3.8 gdx_lcp_data_tx

表 3-11 gdx_lcp_data_tx函数

函数原型	uint16_t gdx_lcp_data_tx(uint8_t header, uint8_t length, uint8_t *p_payload)
功能说明	设置待发送数据。
输入参数	<ul style="list-style-type: none"> • header: LCP协议数据包中的Header信息。 • length: 待发送数据长度, 对应LCP协议数据包中的PDU len。 • p_payload: 待发送数据, 对应 LCP协议数据包中的PDU。
返回值	<ul style="list-style-type: none"> • 0: 成功 • 其他: 失败
备注	<ul style="list-style-type: none"> • 当使用软件触发发送模式时, 调用该接口函数后, 将立即发送数据。 • 当使用Timer触发发送模式时, 调用该接口函数后, 需等待Timer触发发送行为后才发送数据。

3.3.3.9 gdx_lcp_auto_txrx_mode_set

表 3-12 gdx_lcp_auto_txrx_mode_set函数

函数原型	void gdx_lcp_auto_txrx_mode_set(bool enable)
功能说明	设置TX/RX自动转换模式。
输入参数	enable: <ul style="list-style-type: none"> • false: 禁用TX/RX自动转换（使用软件触发接收/发送模式）。 • true: 使能TX/RX自动转换（使用Timer触发接收/发送模式）。
返回值	无
备注	

3.3.3.10 gdx_lcp_rx_oneshot_mode_set

表 3-13 gdx_lcp_rx_oneshot_mode_set函数

函数原型	void gdx_lcp_rx_oneshot_mode_set(bool enable)
功能说明	设置RX收包模式。
输入参数	enable: <ul style="list-style-type: none"> • false: 禁用RX单次接收模式, 即使用RX连续接收模式。当收包时间超过rx_window_size时, 退出RX模式。 • true: 使用RX单次接收模式, 即接收到一个数据包后就退出RX模式。
返回值	无
备注	使用Timer触发模式时, 仅支持RX单次接收模式。

3.3.3.11 gdx_lcp_rx_window_size_set

表 3-14 gdx_lcp_rx_window_size_set函数

函数原型	void gdx_lcp_rx_window_size_set(uint32_t time_us)
功能说明	设置RX开窗宽度。

输入参数	time_us: 取值范围0~10239687。0: 表示RX开窗宽度为无限长。
返回值	无
备注	

3.3.3.12 gdx_lcp_rx_start

表 3-15 gdx_lcp_rx_start函数

函数原型	uint16_t gdx_lcp_rx_start(void)
功能说明	启动接收数据包。
输入参数	无
返回值	<ul style="list-style-type: none">0: 成功其他: 上一个RX还未结束, 启动失败。
备注	<ul style="list-style-type: none">当使用软件触发接收模式时, 调用该接口函数后, 将立即启动接收数据。当使用Timer触发接收模式时, 调用该接口函数后, 需等待Timer触发RX机制, 才启动接收数据。

3.3.3.13 gdx_lcp_rx_stop

表 3-16 gdx_lcp_rx_stop函数

函数原型	void gdx_lcp_rx_stop(void)
功能说明	停止接收数据包。
输入参数	无
返回值	无
备注	

3.3.3.14 gdx_lcp_whitening_seed_set

表 3-17 gdx_lcp_whitening_seed_set函数

函数原型	uint16_t gdx_lcp_whitening_seed_set(uint8_t whitening_seed)
功能说明	设置当前通信的白化种子。
输入参数	whitening_seed: 白化种子
返回值	无
备注	默认使用标准Bluetooth LE白化算法, 具体可参考Core_v5.3中的Vol6/PartB/3.2 Data Whitening。 该算法与Channel Number相关。SDK内部使用的Channel Number与频点 (Freq_mhz) 之间的关系为: $\text{Channel Number} = (\text{Freq_mhz} - 2360)/2$

3.3.3.15 gdx_lcp_t2r_turn_around_time_adjust

表 3-18 gdx_lcp_t2r_turn_around_time_adjust函数

函数原型	void gdx_lcp_t2r_turn_around_time_adjust(uint8_t added_us)
------	--

功能说明	调整TX转换为RX的时间间隔，默认值为48 μs。
输入参数	added_us: 在默认值基础上增加的时间
返回值	无
备注	仅Timer触发TX模式下可调用该接口函数。

3.3.3.16 gdx_lcp_r2t_turn_around_time_adjust

表 3-19 gdx_lcp_r2t_turn_around_time_adjust函数

函数原型	void gdx_lcp_r2t_turn_around_time_adjust(uint8_t added_us)
功能说明	调整RX转换为TX的时间间隔，默认值为52 μs。
输入参数	added_us: 在默认值基础上增加的时间
返回值	无
备注	仅Timer触发RX模式下可调用该接口函数。

3.4 创建LCP应用

本节详细介绍各种LCP应用的实现流程，以帮助用户快速创建自定义LCP应用。

3.4.1 软件触发收发模式的应用

创建软件触发接收/发送模式的应用，可按以下流程实现：

- 1. 调用gdx_lcp_init函数初始化LCP模块。
- 2. 若发送数据包，则调用gdx_lcp_data_tx函数发送数据；若接收数据包，则调用gdx_lcp_rx_start函数启动RX，开始接收数据。
- 3. 调用rx_handler_cb函数处理接收的数据。
- 4. 数据发送/接收完成后，调用tx_done_cb/rx_done_cb回调函数，准备下一次通信。

3.4.2 Timer触发收发模式的应用

创建Timer触发接收/发送模式的应用，可按以下流程实现：

- 1. 初始化Timer。
- 2. 调用gdx_lcp_init函数初始化LCP模块
- 3. 调用gdx_lcp_timer_binding函数绑定LCP通信行为与指定Timer。
- 4. 若发送数据包，则调用gdx_lcp_data_tx函数发送数据；若接收数据包，则调用gdx_lcp_rx_start函数启动RX，开始接收数据。
- 5. 调用rx_handler_cb函数处理接收的数据。
- 6. 数据发送/接收完成后，调用tx_done_cb/rx_done_cb回调函数，准备下一次通信。

3.4.3 脉冲定检示例工程

- 功能描述:

在PTX端的APP_IO_PIN_0输入方波、PRX端的APP_IO_PIN_0输出恢复波，在空口不丢包的情况下，恢复精度 $< 0.5 \mu s$ 。

- 示例工程参考:

- PTX角色示例工程: SDK_Folder\projects\ble\ble_peripheral\ble_app_lcp。
- PRX角色示例工程: SDK_Folder\projects\ble\ble_central\ble_app_lcp_c。