MBL Series Evaluation Kit User Manual

Next generation package-compatible S ub-1G wireless module

E220-400MBL-01



Chengdu Ebyte Electronic Technology Co.,Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Contents

DISCLAIMER 2

CHAPTER 1 PRODUCT OVERVIEW 3

1.1 PRODUCT INTRODUCTION 3
1.2 SIZE AND INTERFACE DESCRIPTION 3
1.3 SUPPORT MATRIX 4

CHAPTER 2 INTRODUCTION TO SOFTWARE 5

2.1 DIRECTORY STRUCTURE 5
2.2 IAR ENGINEERING 7
2.3 MAIN FUNCTIONS 8
2.4 SENDING AND RECEIVING TIMING 8
2.5 PROGRAMMING 10

CHAPTER 3 QUICK PRESENTATION 12

3.1 SIGNAL LINE CONNECTION 12
3.2 SERIAL PORT ASSISTANT 13

CHAPTER IV FREQUENTLY ASKED QUESTIONS 14

4.1 THE TRANSMISSION DISTANCE IS NOT IDEAL 14
4.2 MODULES ARE EASILY DAMAGED 14
4.3 THE BIT ERROR RATE IS TOO HIGH 14

REVISION HISTORY 15

ABOUT US 15

Copyright ©2012–2021，Chengdu Ebyte Electronic Technology Co.,Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Disclaimer

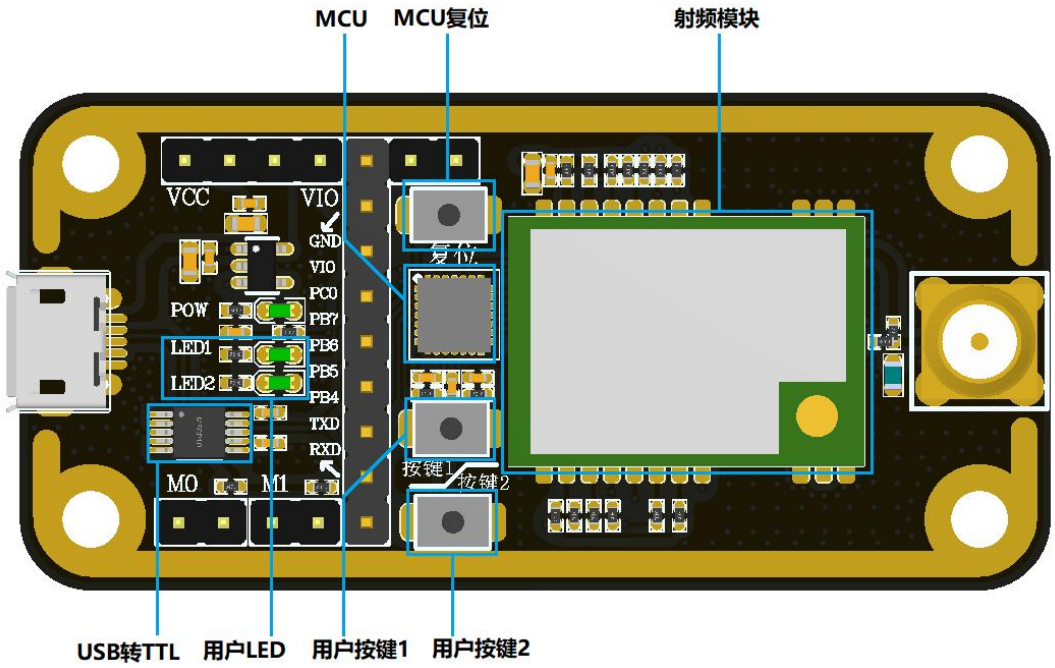
EBYTE reserves all rights to this document and the information contained herein. Products, names, logos and designs described herein may in whole or in part be subject to intellectual property rights. Reproduction, use, modification or disclosure to third parties of this document or any part thereof without the express permission of EBYTE is strictly prohibited.

The information contained herein is provided “as is” and EBYTE assumes no liability for the use of the information. No warranty, either express or implied, is given, including but not limited, with respect to the accuracy, correctness, reliability and fitness for a particular purpose of the information. This document may be revised by EBYTE at any time. For most recent documents, visit www.ebyte.com.

Copyright ©2012–2021，Chengdu Ebyte Electronic Technology Co.,Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Chapter 1 Product Overview



1.1 Product Introduction

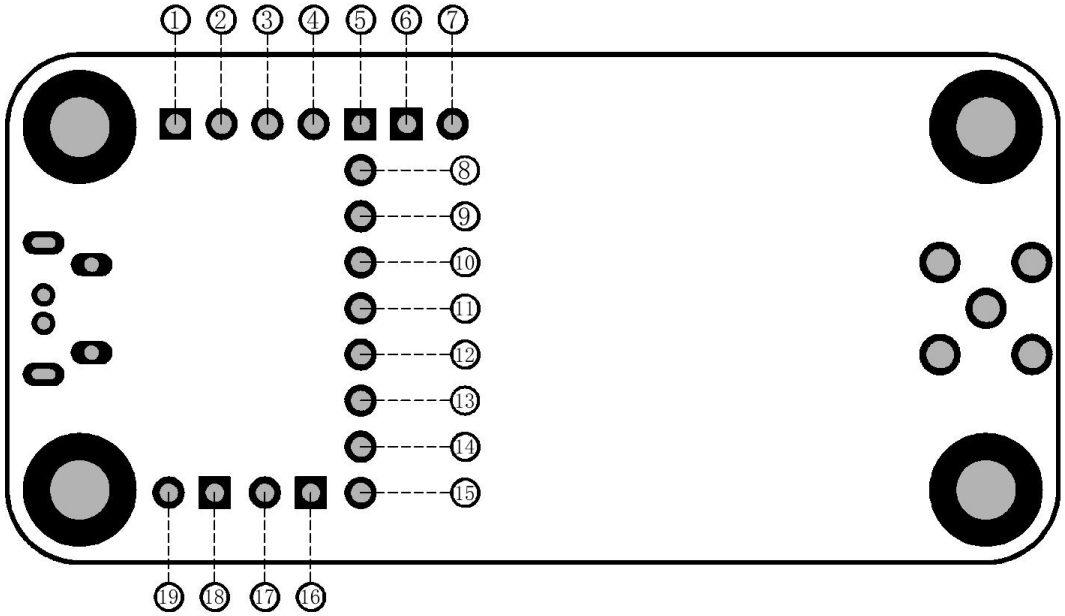
The MBL series evaluation kits are designed to help users quickly evaluate Ebait's next-generation package-compatible wireless modules. Most of the pins on the board are led out to the pin headers on both sides, making it easy for developers to connect a variety of peripherals via jumpers to their needs.

The kit provides complete software application examples to help customers get started quickly with wireless data communication development. Different types of Sub-1G wireless modules can be loaded on board according to customer needs. Supported modules are available in pin-compatible packages for quick replacement.

1.2 Size and interface description

Copyright ©2012–2021，Chengdu Ebyte Electronic Technology Co.,Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000



Pin serial number	definition	Function description
1	VCC	The module power supply pin needs to be shorted to pin 2 to power the module
2	3.3V	3.3V electrical lead pin
3	3.3V	3.3V electrical lead pin
4	SAW	The MCU power supply pin needs to be shorted to pin 3 to power the MCU
5	GND	Bottom plate reference
6	RIS	MCU external reset pin
7	SWIM	The SWIM pin of the MCU
8	SAW	MCU supply pin
9	PC0	Module reset pin
10	PB7	Module MISO pin
11	PB6	Module MOSI pin
12	PB5	Module SCLK pin
13	PB4	Module NSS pin
14	TXD	MCU serial port TXD
15	RXD	MCU serial port RXD
16	M1	Module mode switch pin (see module data sheet for details)
17	GND	Bottom plate reference
18	M0	Module mode switch pin (see module data sheet for details)
19	GND	Bottom plate reference

1.3 Support Matrix

Copyright ©2012–2021，Chengdu Ebyte Electronic Technology Co.,Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	RF solutions	manufacturer	Module model
1	CC1101	Texas Instruments	E07-400M10S
2	CC1101	Texas Instruments	E07-900M10S
3	SI4438	Silicon Labs	E30-400M20S
4	SI4463	Silicon Labs	E30-900M20S
5	LLCC68	Semtech	E220-400M22S
6	LLCC68	Semtech	E220-900M22S
7	SX1278	Semtech	E32-400M20S
8	SX1276	Semtech	E32-900M20S
9	SX1268	Semtech	E22-400M22S
10	SX1262	Semtech	E22-900M22S
11	AX5243	ON Semiconductor	E31-400M17S
12	LLCC68	Semtech	E220-400MM22S
13	LLCC68	Semtech	E220-900MM22S

Chapter 2 Introduction to Software

2.1 Directory Structure

	Matters	illustrate
1	File directory	You can download the sample project from the official website and open the directory as shown in the following figure 
2	Table of contents description	You can use the IARFor STM8 development environment to locate the entry file to open the project

Copyright ©2012–2021，Chengdu Ebyte Electronic Technology Co.,Ltd.

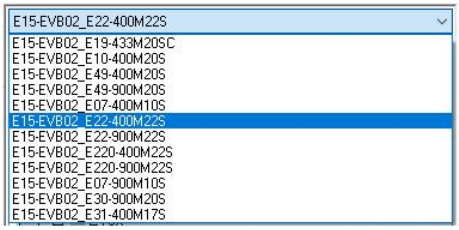
Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

		├─ **E15-EVB02 Demo** //主文件夹 │ │ │ ├─ **0_Project** │ │ └─ IAR_for_Stm8 //工程文件夹 使用 IAR 打开工程 │ │ │ ├─ **1_Middleware** │ │ ├─ Kfifo //通用数据队列 │ │ └─ Produce //PC测试 │ │ │ ├─ **2_Ebyte_Board_Support** │ │ └─ E15-EVB02 //板载资源初始化 │ │ │ ├─ **3_Ebyte_WirelessModule_Drivers** │ │ ├─ E07xMx //E07模块驱动 │ │ ├─ E10xMx //E10模块驱动 │ │ ├─ E19xMx //E19模块驱动 │ │ ├─ E22xMx //E22模块驱动 │ │ ├─ E30xMx //E30模块驱动 │ │ ├─ E31xMx //E31模块驱动 │ │ ├─ E49xMx //E49模块驱动 │ │ └─ E220xMx //E220模块驱动 │ │ │ └─ **4_STM8_L15x_StdPeriph_Drivers**

Copyright ©2012–2021，Chengdu Ebyte Electronic Technology Co.,Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

2.2 IAR engineering

	Matters	illustrate
	Engineering structure	Open the project using the IAR For STM8 development environment to see the basic structure 
	Switch workspaces	The C/C++ Compiler option defines global macro definitions and file paths to distinguish driver files from different modules. When switching workspaces, different macro definitions are used, switching the driver files for different modules  Changed the Exclude from build property of Drivers/Ebyte/RF to select the target module driver folder to participate in the compilation process. Changed the additional include in the project C/C++ Compiler, which specifies the path to the module driver file. Changed the Defined symbols in the project C/C++ Compiler, which defines global macro definitions to help configure module-driven properties. 

Copyright ©2012–2021，Chengdu Ebyte Electronic Technology Co.,Ltd.

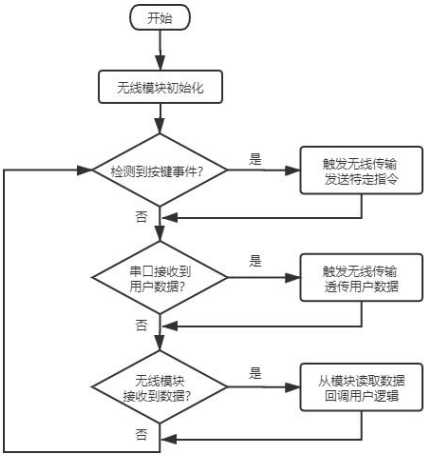
Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

2.3 Main functions

main.c is the main function entry. The demo feature flow simplifies the process as follows:

	Matters	illustrate
1	Key function	If a button is pressed, the command data will be sent wirelessly. Essentially, sending a specific string "ping" and expecting to receive a response "pong"
2	Serial port data to wireless sending	After the serial port receives the data, it automatically starts wireless transparent transmission of data, of course, it contains some special instruction responses, which are mainly used for special tests and can be ignored by users. After the sending is complete, the user function is automatically called, so that the sending logic is handled by itself.
3	Receive data wirelessly	Generally, the internal state ID of the module is read to determine whether there is data, and the underlying driver will copy the data and pass it to the user callback function, so as to handle the receiving logic by itself

The software process simplification is shown in the following diagram:



2.4 Sending and Receiving Timing

Wireless modules have multiple operating states and can only complete specific functions in the corresponding state. From the simplest sending and receiving data, only the sending mode and receiving mode are considered.

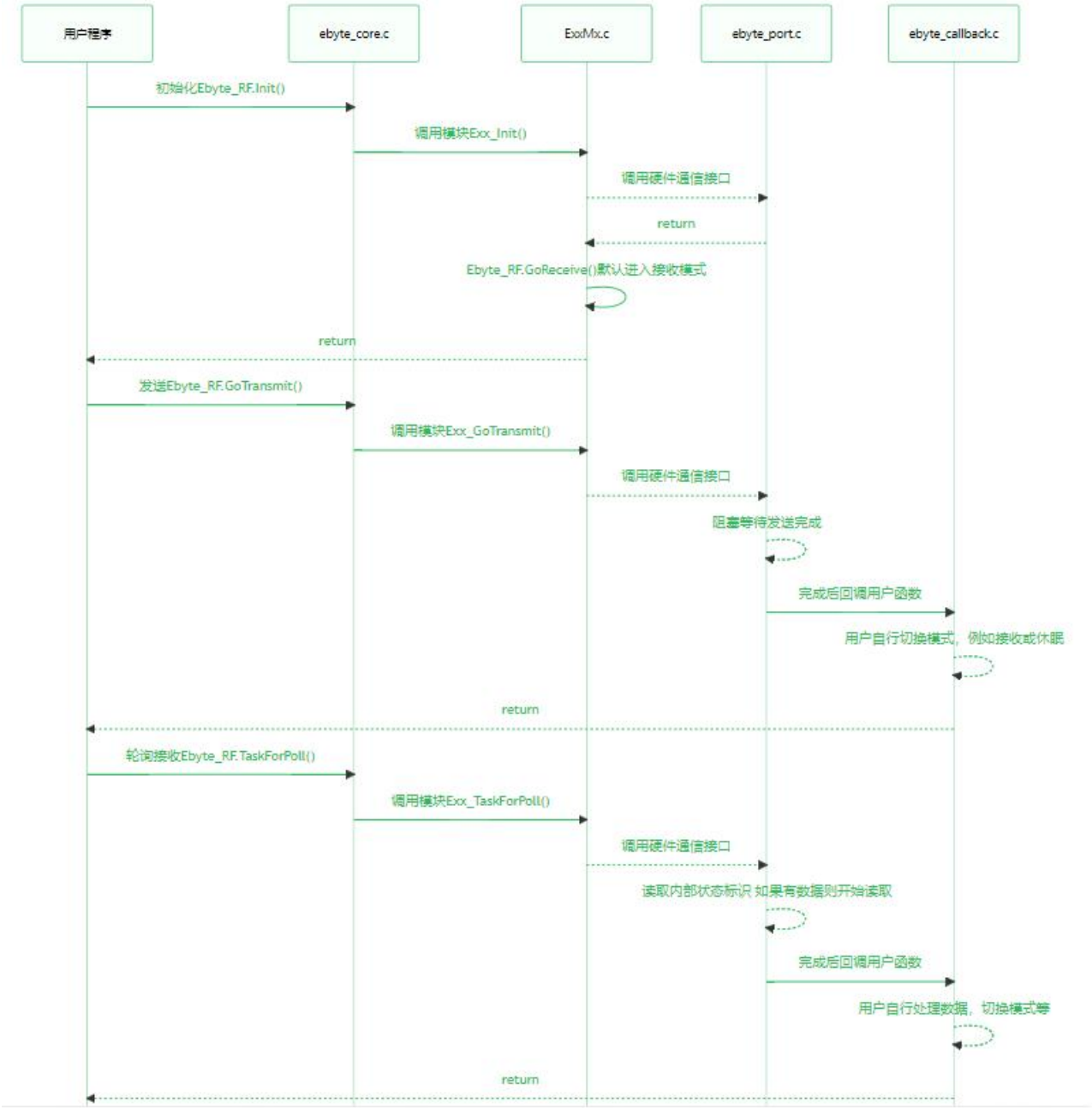
	Matters	illustrate
1	Receive mode	After the default initialization is completed, it automatically enters receive mode. Essentially, the receive function is called in initialization, thus entering receive mode. If you need to consider entering other modes after initialization, such as sleep, you can simply replace it with a function of the same type Go_xxxxx().
2	Send mode	When calling the transmit function, the underlying driver actually switches the module into standby mode first, usually in this mode to complete the configuration of modulation parameters, such as frequency, power, frequency offset, and so on. After

Copyright ©2012–2021，Chengdu Ebyte Electronic Technology Co.,Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

		the parameters are configured correctly, gradually enter some intermediate modes, turn on internal FIFO, PA, external XTAL, etc., and the current consumption also gradually increases. Finally, it switches into transmit mode, triggering wireless data transmission. After completion, the module enters standby mode, and cannot continue to send and receive in this state, and the user needs to handle the next mode by himself in the callback function. When the function is complex, continuous reception or continuous transmission is required, please further switch other modes according to the characteristics of the chip.

The timing diagram looks like this:



Copyright ©2012–2021，Chengdu Ebyte Electronic Technology Co.,Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

2.5 Programming

	file	Key notes

1	ebyte_core.h	A module structure is defined, abstracting the basic functionality to which the functions of the underlying module will be bound. When used for simple sending and receiving applications, there is no need to understand the low-level working details of each module, and you can directly call the abstract function to start sending and receiving data. If you need to customize some functionality, you can also consider integrating it into the structure. If you know enough about the functional functions of the underlying module, you can also directly remove the ebyte_core.c/h file, and there is no strong coupling between layers.
		`typedef struct`
		`{`
		`  uint8_t  ( *Init )( void ); //初始化`
		`  uint8_t  ( *GoTransmit )( uint8_t *buffer, uint8_t size ); //切换发送模式 开始传输数据`
		`  uint8_t  ( *GoSleep )( void );  //切换到休眠模式 低功耗用到`
		`  uint8_t  ( *GoReceive )( void );  //切换到接收模式 开始监听数据`
		`  uint8_t  ( *TaskForPoll)( void ); //轮询函数 可以主循环周期调用 也可以视情况放入中断`
		`  void  ( *TaskForIRQ )( void ); //暂时保留，不必使用。用于将来扩展中断收/发`
		`  uint8_t  ( *GetStatus )( void );  //获取模块状态(软件状态机)`
		`  uint8_t * ( *GetName ) (void );  //获取模块识别码 为字符串 例如 "E22-400M22S"`
		`  uint8_t  ( *GetDriver ) (void );  //获取软件版本号`
		`}Ebyte_RF_t;`

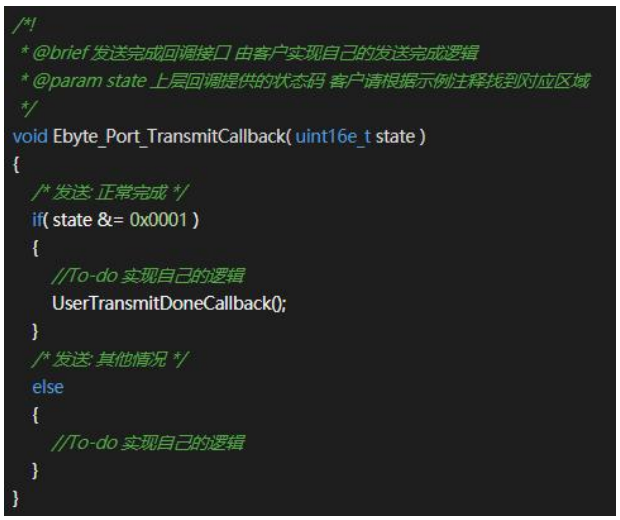
2	ebyte_exx.c	It is a specific module driver file, which is generally encapsulated and does not require users to change, only needs to consider how to input and output data from this "box".

3	ebyte_port.c	It is specially designed to bind SPI and GPIO under different hardware platforms, abstracted as "box" inputs. Users need to fill in the communication interfaces in their hardware platform to fixed locations according to comments. In general, it provides the transceiver function of the SPI and the level control of the pins. Some modules are slightly special, such as E49 uses half-duplex SPI, if you are too lazy to write the communication driver, then directly bind the IO to a fixed position, and the rest is left to the module driver to simulate IO to achieve communication. As shown in the following figure, the SPI interface location is required to fill in the specific transceiver function in the comment, and the data is sent by send to SPI and the result returns to SPI Receive data.
		`/*!`
		` * @brief 配置目标硬件平台SPI接口收发函数`
		` * @param send EBYTE驱动库上层调用需要传输的数据 1 Byte`
		` * @return SPI 接收的数据 1 Byte`
		` */`
		`uint8_t Ebyte_Port_SpiTransmitAndReceivce( uint8_t send )`
		`{`
		`  uint8_t result = 0;`
		`  /* 必须提供: SPI接口 */`
		`  result = Ebyte_BSP_SpiTransAndRecv( send ); //用户填充函数`
		`  return result;`
		`}`

	ebyte_callback.c	It is specially used to bind the user's own sending and receiving logic, abstracted into the output of the "box". Essentially, a module driver is to directly call the user's callback function after determining that the send or receive is complete. As shown in the following

Copyright ©2012–2021，Chengdu Ebyte Electronic Technology Co.,Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

		figure, the user's logical function can be filled in the To-do prompt position. state is driven out by the module, which is actually handled by the Exx_GoTransmit() function, and can be considered modified to support more cases when the functionality is complex.
		
		/*!
		* @brief 发送完成回调接口 由客户实现自己的发送完成逻辑
		* @param state 上层回调提供的状态码 客户请根据示例注释找到对应区域
		*/
		void Ebyte_Port_TransmitCallback(uint16e_t state)
		{
		/* 发送: 正常完成 */
		if(state &= 0x0001)
		{
		//To-do 实现自己的逻辑
		UserTransmitDoneCallback();
		}
		/* 发送: 其他情况 */
		else
		{
		//To-do 实现自己的逻辑
		}
		}

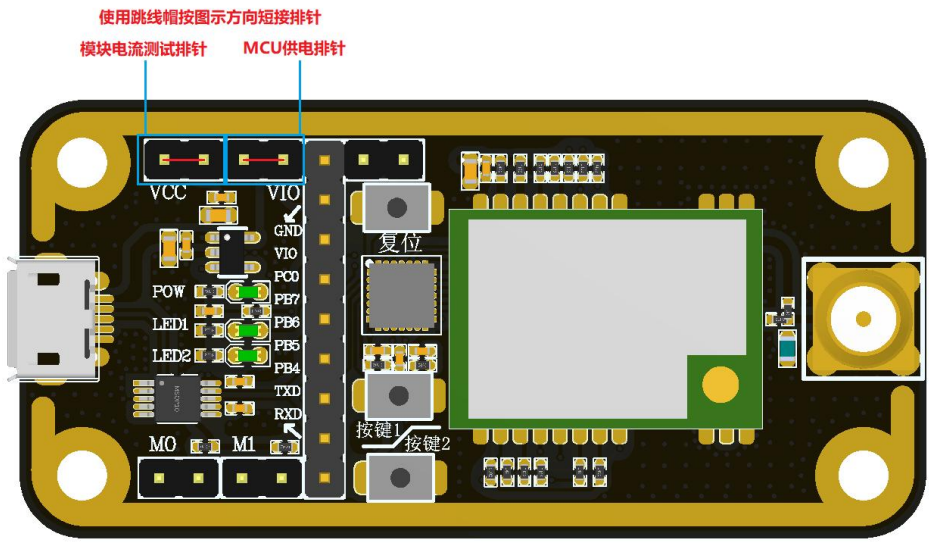
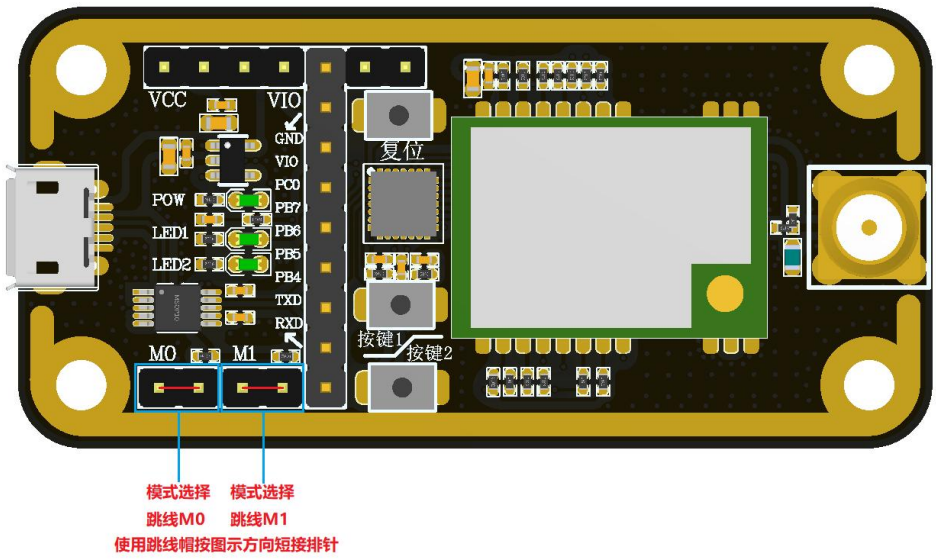
	ebyte_exx.h	Some general modulation parameters are defined, which generally do not need to be modified and can be adjusted by themselves. Note that when modifying, please understand the instructions in the comments, there is a range check for parameters in the module driver, and incorrect modulation parameters will cause initialization failure. As shown in the figure below, the FSK modulation parameters are example:  #define E07_DATA_RATE 1200 //空速 1.2 KBps #define E07_FREQUENCY_DEVIATION 14300 //频偏 14.3 K #define E07_BANDWIDTH 58000 //接收频宽 58 K #define E07_OUTPUT_POWER 10 //功率 [10 7 5 0 -10 -15 -20 -30] #define E07_PREAMBLE_SIZE 4 //前导码长度 [0:2 1:3 2:4 3:6 4:8 5:12 6:16 7:24] #define E07_SYNC_WORD 0x2DD4 //同步字 #define E07_IS_CRC 1 //CRC开关 [0:关闭 1:开启]
	board.c	STM8 peripheral initialization, involving SPI, TIMER, GPIO, etc., is strongly coupled to the hardware used.
	board_button.c	The key event queue is a FIFO in terms of data structure. After the timer detects the keystroke, it will save the corresponding event to the queue and wait for the main loop response.
	board_mini_printf.c	Simplified printf, although the function is shrunk, but the footprint is small. DEBUG macros in the project mainly rely on the mprintf provided by this file.
	ebyte_kfifo.c	For serial port data reception, optimized universal FIFO queue, suitable for cache.
	ebyte_debug.c	It is used to connect to a PC for some tests and is generally not required.
	stm8l15x_it.c	All interrupt function entries will be for serial ports, timers, button IO and other interrupt service functions are concentrated here

Copyright ©2012–2021，Chengdu Ebyte Electronic Technology Co.,Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Chapter 3 Quick Presentation

3.1 Signal line connection

	Matters	illustrate
1	Power jumper cap	
2	Mode selection jumper cap	
3	auxiliary	USB cable, antenna, PC, etc

Copyright ©2012–2021，Chengdu Ebyte Electronic Technology Co.,Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

3.2 Serial port assistant

	Matters	illustrate
1	Device Manager Check the serial port number	
2	Serial port software	
3	Example of keystroke communication	#RECV identifier, used only as a hint, to represent the data received by the wireless module. #SEND identifier, used only for hints, to represent the data sent by the wireless module 
4	Serial port data transparent transmission	Serial port data transmission directly transmits the required content through XCOM

Copyright ©2012–2021，Chengdu Ebyte Electronic Technology Co.,Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Chapter IV Frequently Asked Questions

4.1 The transmission distance is not ideal

- When there is a straight-line communication barrier, the communication distance will be attenuated accordingly;
- Temperature, humidity, and co-channel interference will lead to an increase in the communication packet loss rate;
- The ground absorbs and reflects radio waves, and the test effect near the ground is poor;
- Seawater has a strong ability to absorb radio waves, so the seaside test effect is poor;
- If there is a metal object near the antenna, or placed in a metal case, the signal attenuation will be very serious;
- the power register is set incorrectly, the air rate is set too high (the higher the air rate, the closer the distance);
- The low voltage of the power supply at room temperature is lower than the recommended value, and the lower the voltage, the smaller the power;
- The antenna used is poorly matched to the module or the quality of the antenna itself.

4.2 Modules are easily damaged

- Please check the power supply to ensure that between the recommended supply voltages, exceeding the maximum value will cause permanent damage to the module;
- Please check the stability of the power supply, the voltage cannot fluctuate greatly and frequently;
- Please ensure that the installation and use process of anti-static operation, high-frequency devices electrostatic sensitivity;
- Please ensure that the humidity during installation and use should not be too high, and some components are humidity sensitive devices;
- If there is no special need, it is not recommended to use it at too high or too low temperature.

4.3 The bit error rate is too high

- There is co-channel signal interference nearby, stay away from the interference source or modify the frequency and channel to avoid interference;
- Unsatisfactory power supply may also cause garbled characters, be sure to ensure the reliability of the power supply;
- Poor or long quality extension wires and feeders can also cause high bit error rates.

Copyright ©2012–2021，Chengdu Ebyte Electronic Technology Co.,Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Revision history

version	Revision date	Revision Instructions	Maintainers
1.0	2021-09-22	Initial version	CENTURY
1.1	2022-12-29	Modify the schematic diagram of the module and how to use it	HWJ

About us

Technical support: support@cdebyte.com
Tel：+86-28-61399028
Fax：028- 61543675
Web： https://www.cdebyte.com
Address: B5 Mould Park, 199# Xiqu Ave, High-tech District, Sichuan, China

Chengdu Ebyte Electronic Technology Co.,Ltd.

Copyright ©2012–2021，Chengdu Ebyte Electronic Technology Co.,Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000