E50-900 NW20SX Product Specification

WM-BUS 868 /915 MHz SMD SoC module

成都亿佰特电子科技有限公司
Chengdu Ebyte Electronic Technology Co.,Ltd.

Table of contents

Disclaimer and Copyright Notice

The information in this article, including URL addresses for reference, is subject to change without notice. Documentation is provided "as is" without warranty of any kind, including any warranty of merchantability, fitness for a particular purpose, or non-infringement, and any warranty mentioned elsewhere in any proposal, specification or sample. This document disclaims all liability, including liability for infringement of any patent rights arising from the use of the information contained in this document. No license, express or implied, to the use of any intellectual property rights is granted herein by estoppel or otherwise.

* The test data obtained in this article are all obtained from Ebyte laboratory testing, and the actual results may be slightly different.
* All trade names, trademarks and registered trademarks mentioned in this article are the property of their respective owners and are hereby acknowledged.
* The final right of interpretation belongs to Chengdu Ebyte Electronic Technology Co., Ltd.

Notice:

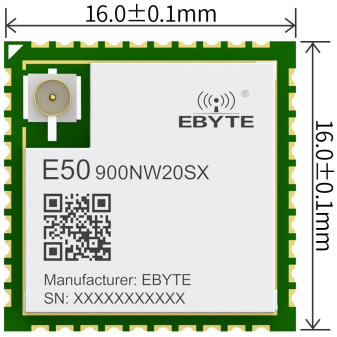
The contents of this manual may change due to product version upgrades or other reasons. Ebyte Electronic Technology Co., Ltd. reserves the right to modify the contents of this manual without any notice or prompt. This manual is only used as a guide. Chengdu Ebyte Electronic Technology Co., Ltd. makes every effort to provide accurate information in this manual. However, Chengdu Ebyte Electronic Technology Co., Ltd. does not ensure that the content of the manual is completely error-free. All statements in this manual , information and recommendations do not constitute any express or implied warranty.

Chapter 1 Overview

1.1 Introduction

The E50-900NW20SX wireless module is an ideal solution for sub-1 GHz IoT wireless connectivity for smart home, security, lighting, building automation and metering. It is equipped with Silicon Labs' EFR32FG23 series chips, which can achieve wireless broadcasting up to 0.3~2.5 kilometers with minimal data loss in dense urban and canyon environments. At the same time, the module also carries firmware that supports WM-BUS (Wireless M-BUS) bus technology. The E50-900NW20SX wireless module firmware is developed based on the EN13757-4/3 European standard, which can realize intelligent general instruments, data concentrators, and mobile computers. Seamless SUB-GHz RF communication protocol between meter devices and heat distribution meters.

The E50-900NW20SX wireless communication module can be used as an SoC (system on chip) development platform. It has the characteristics of small size, rich interface resources, and supports user secondary development, and can be widely used in the Internet of Things industry.



1.2 Module SoC Features

Low power wireless system on chip

- High-performance 32-bit 78 MHz ARM Cortex®-M33 with DSP instructions and floating-point unit for efficient signal processing.
- Up to 256 kB Flash program memory
- RAM data memory up to 32 kB
- Up to +20 dBm transmit power

Rich and diverse selection of MCU peripherals

- Exports up to 31 general-purpose I/O pins with output status retention and asynchronous interrupt capabilities
- Analog-to-digital converter (ADC)
- 12-bit@1 Msps
- 16-bit@76.9ksps
- 2 × Analog Comparator (ACMP)
- 2 × digital-to-analog converter (VDAC)
- Low Energy Sensor Interface (LESENSE)
- 8-channel DMA controller
- 12-Channel Peripheral Reflective System (PRS)
- 4*16-bit timers/counters with 3 compare/capture/PWM channels.
- 1*32-bit timer/counter with 3 compare/capture/PWM channels.
- 32-bit real-time counter
- 24-bit low-power timer for waveform generation
- 2 × watchdog timer
- 3 × upgraded universal synchronous/asynchronous receiver/transmitter (EUSART)
- 1× Universal synchronous/asynchronous receiver/transmitter (UART/SPI/SmartCard (ISO 7816)/IrDA/I2S)
- 2* I2C interfaces supporting SMBus
- Chip temperature sensor with ±2°C accuracy over entire temperature range

Wide working range

- 1.8 to 3.8 V power supply

- -40 to +85 degrees Celsius

Supported modulation formats

- 2/4 (G)FSK with fully configurable molding capabilities
- OQPSK DSSS
- (G)MSK
- OOK

Chip-level security features

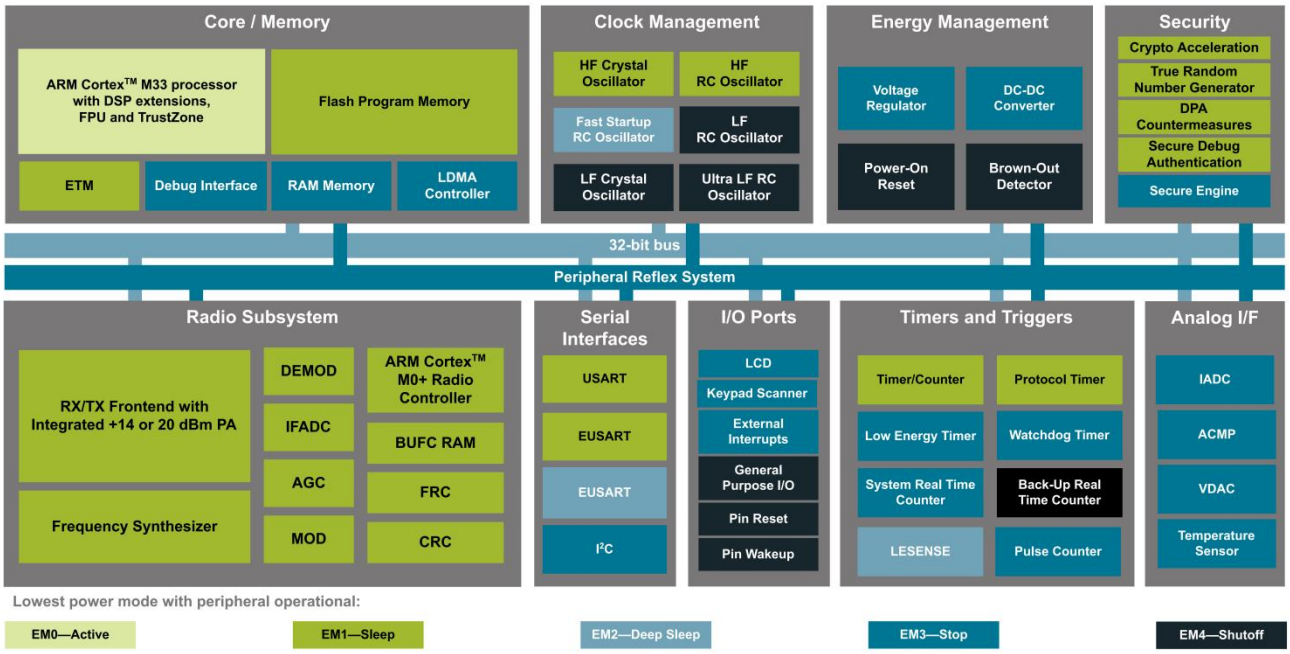
- AES128/192/256, ChaCha20-Poly1305, SHA-1, SHA-2/256/384/512, ECDSA+ECDH (P-192, P-256, P-384, P-521), Ed25519 and Curve25519, J -Hardware encryption acceleration of PAKE and PBKDF2
- True Random Number Generator (TRNG)
- ARM® TrustZone®
- Secure Boot (trust the root of the secure loader)
- Secure debug unlock
- Tamper proof

Protocol support

- Proprietary Product Private Agreement
- WM-BUS
- Sidewalk
- CONNECT

Encapsulation

- Patch stamp hole 16mm (length) x 16mm (width) x 3mm (height)



EFR32FG23 series chip internal functional block diagram

1.3 Module features

- Supports WM-BUS communication protocol and complies with EN 13757 and OMS organization protocol standards;
- Supports multiple WM-BUS working modes such as T(T1/T2), C(C1/C2), S(S1/S2) and R2;
- Supports 4-10 frequency points such as 868.3/868.95/869.525 MHz, and complies with WM-BUS communication protocol and European CE RED related RF specifications;
- Supports 4 airspeeds including 4.8/32.768/50/100 kbps;
- Supports multi-level transmit power settings, up to 20dBm;
- Support serial port AT commands;
- Support serial port hardware flow control;

- Supports 2400 ~ 921600 bps serial port baud rate;
- Supports long-distance communication, up to 2.5 km (clear and open sky, antenna gain 3.5 dBi, height 1.5 m, frequency 868.33MHz, airspeed 4.8 kbps);
- Supports low power consumption, transmit current ≤ 45 mA (@15 dBm), receive current ≤ 6 mA, sleep current ≤ 2 uA;
- Supports 28 common data structures of water, electricity, steam, heat and other table types;
- Supports M-BUS frame format output to facilitate users to parse data;

1.4 Application scenarios

WM-BUS (or Wireless Instrument Bus) is an open standard based on the star topology network (with master and backup devices) described in the EN 13757 standard and includes several different operating modes: S, T, R and C (868 MHz), F (433 MHz), and N (169 MHz). WM-Bus specifies a seamless SUB-GHz RF communication protocol between smart general meters, data concentrators, mobile meter reading devices and heat distribution meters. Water, gas and heat meters are wireless smart meter applications that require long battery life. , In order to meet this demand, the WM-BUS protocol stipulates that only a small amount of communication data between tables needs to be consumed, so that the battery life can be as long as 15-20 years.

WM-BUS is generally suitable for extremely energy-saving smart metering and advanced metering infrastructure (AMI) applications. It has been rapidly promoted in the field of automatic reading of electricity, gas, water and heating meter data in Europe, and also in the Middle East, India, Africa It is also widely used in countries and regions such as , Oceania and Brazil.

- Advanced Meter Infrastructure (AMI);
- Smart meters/smart metering;
- smart city/municipal infrastructure;
- Industrial applications/building automation/distribution automation;
- smart home;

Chapter 2 Specifications

2.1 Limit parameters

RF parameters	Parameter value	Notes
Working frequency	868MHz	Different working modes, working at different frequencies
Transmit power	20 dBm	Can be adjusted through instructions
Receive sensitivity (GFSK)	-125dBm	2.4 kbps, GFSK signal, Δf = ± 1.2 kHz, BER<0.1%
Measured distance	2.5 km	Clear and open sky, antenna gain 3.5 dBi, height 1.5 m, frequency 868.33MHz, airspeed 4.8 kbps

2.2 Electrical parameters

Electrical parameters	Minimum value	Typical value	Maximum value	Unit	Notes
Voltage	2.0	3.3	3.8	V	≥ 3.3V can guarantee the output power, exceeding 3.8V may damage the module
Communication level	-	3.3	-	V	It is recommended to add level conversion when using 5.0V TTL
Emission current	60	65	70	mA	≤ 70 mA @ 3.3V/20 dBm, instantaneous power consumption
Receive current	5.0	5.5	6.0	mA	≤ 6 mA @ 3.3V/20dBm/EM0
Sleep current	-	1.8	2	μA	≤ 2uA @ 3.3V/20dBm/EM2
Operating temperature	-40	-	85	℃	——
Working humidity	10	-	90	%	——
Storage temperature	-50	-	150	℃	——

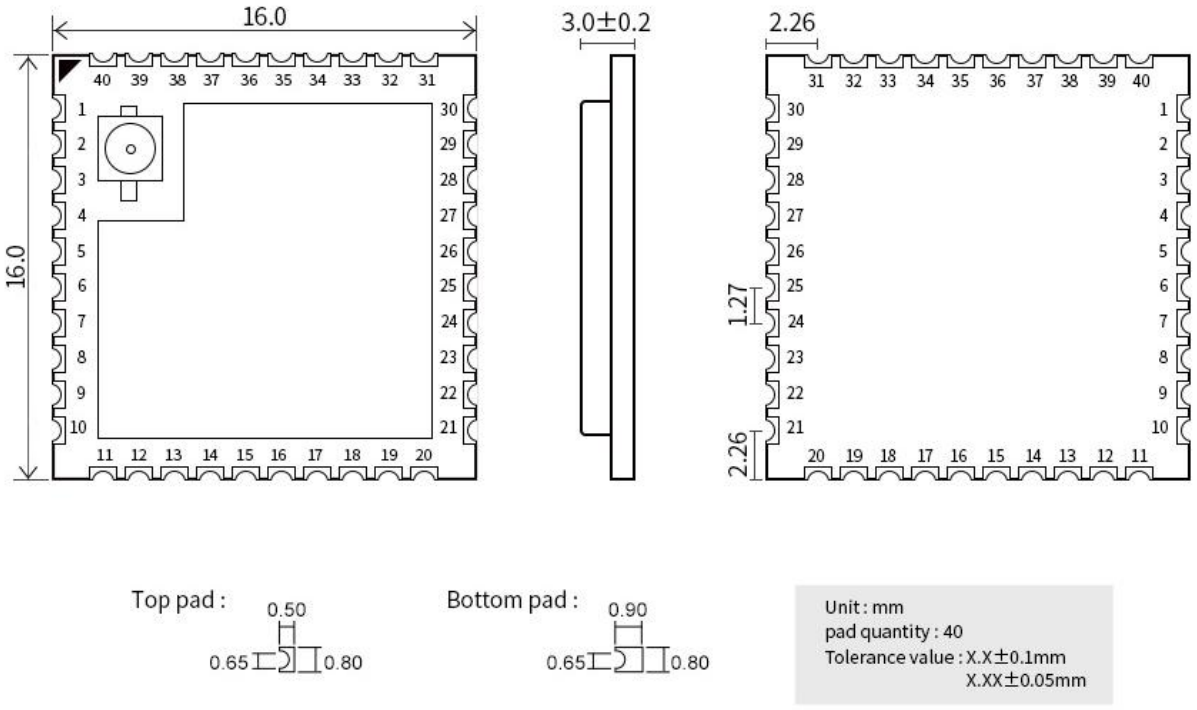
2.3 Hardware parameters

Hardware parameters	Parameter value	Notes
IC full name	EFR32FG23A020F256GM40-B	Can be replaced with other models of the same series, welcome to consult for customization
Kernel	ARM Cortex®-M33	Mainly clocked at 78MHz, high-performance 32-bit ARM processor with DSP instructions and floating point unit
FLASH	256 Kbytes	Can be replaced with other models of the same series, supporting up to 512 KByte. Welcome to consult for customization.
RAM	32 Kbytes	Can be replaced with other models of the same series, supporting up to 64 KByte. Welcome to consult for customization.
Crystal frequency	39MHz/32.768KHz	The module has been connected internally
Size	16(L) * 16(W) * 3(H) mm	——
Antenna form	IPEX/stamp hole	Equivalent impedance is about 50Ω
Communication	UART	Support hardware flow control RTS and CTS pins

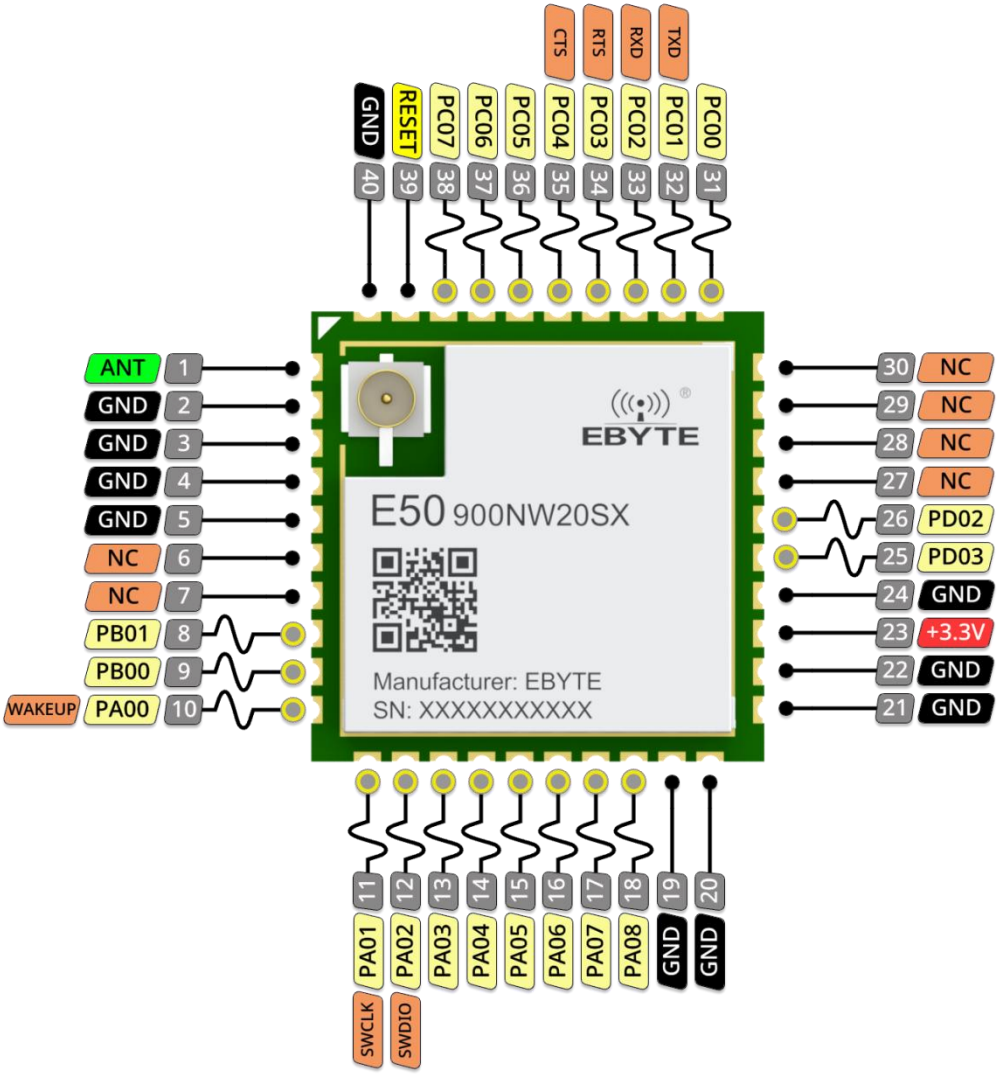
Interface		
Packaging method	Patch stamp hole	The distance between feet is 1.27mm
Weight	1.23g	——

Chapter 3 Mechanical Dimensions and Pin Definitions

3.1 Mechanical dimensions



3.2 Pin definition



Pin number	Pin name	Pin direction	Pin usage
1	ANT	-	RF interface
2	GND	-	power ground
3	GND	-	power ground
4	GND	-	power ground
5	GND	-	power ground
6	NC	-	Empty feet
7	NC	-	Empty feet
8	PB01	I/O	GPIO
9	PB00	I/O	GPIO
10	PA00	I	Wake-up pin, active low
11	PA01	I/O	SWCLK burning/programming pin

12	PA02	I/O	SWDIO burning/programming pin
13	PA03	I/O	GPIO
14	PA04	I/O	GPIO
15	PA05	I/O	GPIO
16	PA06	I/O	GPIO
17	PA07	I/O	GPIO
18	PA08	I/O	GPIO
19	GND	-	Power ground
20	GND	-	Power ground
21	GND	-	Power ground
22	GND	-	Power ground
23	VDD	-	Power supply
24	GND	-	Ground
25	PD03	I/O	GPIO
26	PD02	I/O	GPIO
27	NC	-	Empty feet
28	NC	-	Empty feet
29	NC	-	Empty feet
30	NC	-	Empty feet
31	PC00	I/O	GPIO
32	PC01	O	Module serial port sending end TXD
33	PC02	I	Module serial port receiving end RXD
34	PC03	O	The module serial port RTS and UART will take effect after they are configured to use hardware flow control. The default is not to use hardware flow control.
35	PC04	I	The module serial port CTS and UART will take effect after they are configured to use hardware flow control. The default is not to use hardware flow control.
36	PC05	I/O	GPIO
37	PC06	I/O	GPIO
38	PC07	I/O	GPIO
39	RESET	-	Module reset pin, active low level
40	GND	-	power ground

Chapter 4 Function Description

4.1 Role description

According to the WM-BUS protocol and EN 13757 specification, there are two roles in the WM-BUS

network: Meter (meter end) and Other (collection end). The E50-900NW20SX wireless communication module supports these two network roles.

- Meter (meter side) is divided into one-way meter and two-way meter;
- Other is bidirectional;
- One-way relay (only relays data from the table) ;
- The factory default value is Meter (meter side);

4.2 Transmission mode description

The module supports the following two transmission modes:

- Protocol specification standard mode: WM-BUS-Meter regularly sends standard data format (default timing time is 10s, minimum is 0.3s);
- WM-Bus master and slave stations automatically use the standard message reply function
- EBT mode: WM-BUS-Meter customizes sending data, two-way transparent transmission and instant sending and receiving;

4.2.1 Standard mode Meter terminal realizes simple data reporting

In this mode, the Meter end adopts SND-NR code to achieve timed reporting of sampling data on the Meter end. This feature defaults to using 2 DIF fields, with a maximum support of 4 DIF fields (no support for extended fields). VIF customization can be achieved through commands to quickly collect and report instrument data.

4. 2. 2 Master and slave station standard message automatic reply

In this mode, the master-slave role module will automatically reply to the matching ci code according to the standard. At the same time, users can set SND-UD , RSP-UD and other related ci codes to carry preset messages of user data through instructions to complete the interaction between master and slave stations. The master and slave stations only support a maximum of 15 preset messages. At the same time, the 15th message box, that is, a message with an index ID of 14 in AT+SETMSGMAIL, will be used as a template message.

4. 2. 3 EBT mode

In this mode, both the transceiver and transceiver modules use special device types and use SND-NR ci code to complete the transparent transmission of transceiver data. Users can use the AT+APPHEADER instruction to implement three frame formats, including data carrying no protocol header, short protocol header, and long protocol header.

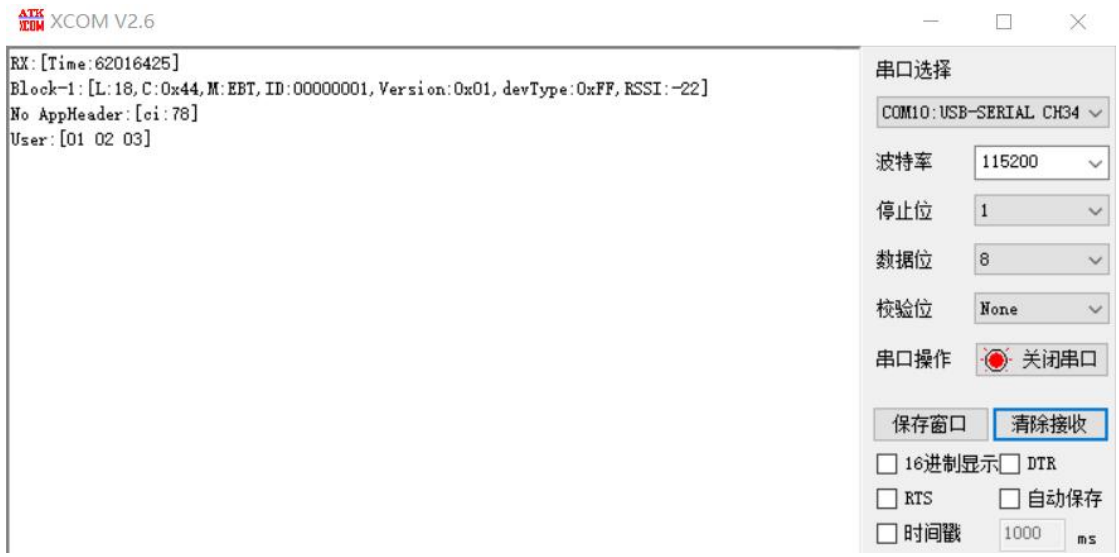
4.3 Data format description

Complies with EN-13737-4 standard .

4. 4 Module output mode description

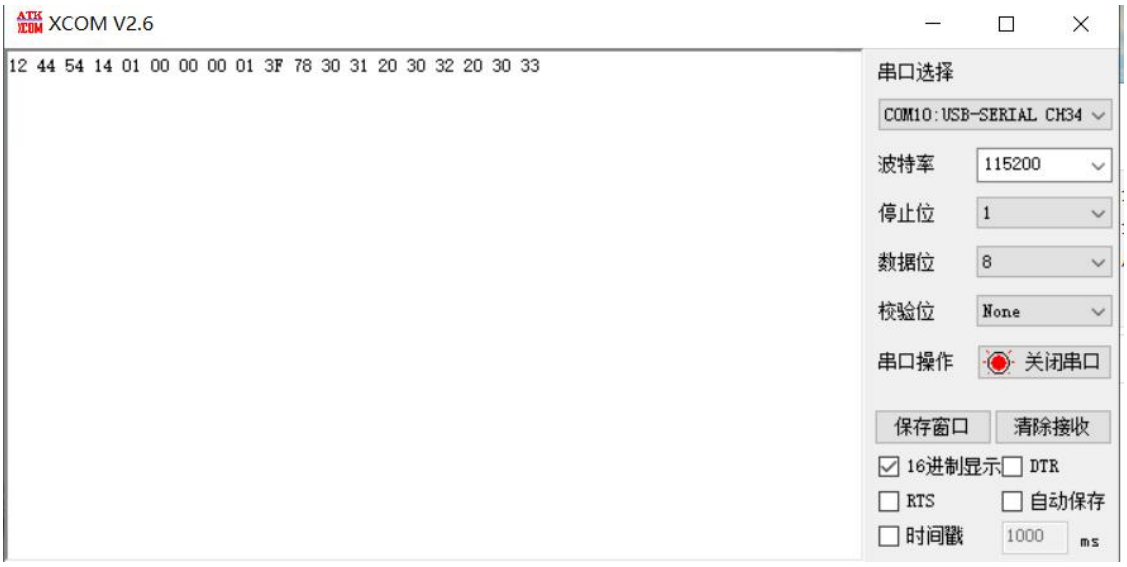
The module is compatible with three output modes

- A SCII code format output



- ASCII code, hex string.

- Hex code output directly



Chapter 5 AT Commands

5.1 Precautions when entering AT command configuration mode

- When sending ++++ to enter configuration mode, first ensure that the module is awake;
- Pin (13) PA06 is pulled low for more than 200ms, and the module enters the wake-up state;
- Enter ++++, the module returns to “enter at mode” and enters the configuration mode;
- In configuration mode, every time sending an AT command, you should wait for the serial port to return information before proceeding with the next command operation;

5.2 Instruction description

Command/return information type	Command format	Description
Query command	AT+[X]?	Parameters used to query corresponding settings
Setup instructions	AT+[X]=<param>	Used to set user-defined parameters
Execute instructions	AT+[X]	For instructions without parameters
Invalid command	——	return invalid cmd
Wrong parameters	——	Return error param
Precautions: ● Instruction parameters are all ASCII encoding; ● The instruction part is case-sensitive and must be capitalized; ● All instructions end with \r\n; ● Returns end with \r\n;		

5.3 Instruction list

5.3.1 AT - Test command

AT - test command	
Execution instructions: AT	response: +OK
	Parameter description: None
Example: Send: AT \r\n Response: +OK \r\n	

5.3.2 AT+UART? - Serial port parameter query command

AT+UART? - Serial port parameter query command	
Execution instructions: AT+UART?	response: +OK AT+UART=[PARAM1],[PARAM2],[PARAM3]
	Parameter Description: PARAM1: Range 1~12, factory default value 9

	PARAM2: Range (5~8) (N/E/O) (0~3), factory default value 8N1 PARAM3: Range 1/0, factory default value 0
Example: Send: AT+UART? \r\n Response: +OK AT+UART=9,8N1,1 \r\n	

5.3.3 AT+UART - Serial port parameter setting command

AT+UART - Serial port parameter query command [effective after restart]	
Execution instructions: AT+UART=[PARAM1],[PARAM2],[PARAM3]	response: +OK
	Parameter Description: PARAM1: baud rate, value range 1~12, default 9 1:2400 2:4800 3:9600 4:14400 5:19200 6:38400 7:43000 8:57600 9:115200 10:230400 11:460800 12:921600 PARAM2: Value range (5~8)/(N/E/O)/(0~3), factory default value 8N1 data bits 5: 5 bits 6: 6 bits 7: 7 digits 8: 8 bits Check Digit N: No verification E: Even parity O: odd parity Stop bit 0:0.5 bit 1:1 bit 2: 1.5 bits 3: 2 bits PARAM3: Hardware flow control, value range 0/1, factory default value 0 0: Turn off hardware flow control 1: Enable hardware flow control
example: Send: AT+UART=9,8N1,1 \r\n Response: +OK \r\n	

5.3. 4 AT+UART_ CUR - Serial port parameter setting command

AT+UART _CUR - Serial port parameter setting command [effective immediately, It will be not saved when power off]	
Execution instructions: AT+UART _CUR =[PARAM1],[PARAM2],[PARAM3]	response: +OK
	Parameter Description: PARAM1: baud rate, value range 1~12, default 9 1:2400

	2:4800 3:9600 4:14400 5:19200 6:38400 7:43000 8:57600 9:115200 10:230400 11:460800 12:921600 PARAM2: Value range (5~8)/(N/E/O)/(0~3), factory default value 8N1 data bits 5: 5 bits 6: 6 bits 7: 7 digits 8: 8 bits Check Digit N: No verification E: Even parity O: odd parity Stop bit 0:0.5 bit 1:1 bit 2: 1.5 bits 3: 2 bits PARAM3: Hardware flow control, value range 0/1, factory default value 0 0: Turn off hardware flow control 1: Enable hardware flow control
example: Send: AT+UART _CUR =9,8N1,1 \r\n Response: +OK \r\n	

5.3. 5 AT+WMBUSMODE? - Working mode query command

AT+WMBUSMODE? - Working mode query command	
Execution instructions: AT+WMBUSMODE?	response: +OK WMBUSMODE=[PARAM]
	Parameter Description: PARAM: range 1~7, factory default value 7
example: Send: AT+WMBUSMODE? \r\n Response: +OK WMBUSMODE=7 \r\n	

5.3. 6 AT+WMBUSMODE - Working mode setting command

AT+WMBUSMODE - Working mode setting command [effective after restart]	
Execution instructions: AT+WMBUSMODE=[PARAM]	response: +OK
	Parameter Description: PARAM: working mode, value range 1~7, factory default value 7 1:T1 2: T2 3:S1 4:S2 5:C1

	6:C2 7:R2
example: Send: AT+WMBUSMODE=7 \r\n Response: +OK \r\n	

5.3. 7 AT+WMBUSMODE_ CUR - Working mode setting command

AT+WMBUSMODE _CUR - Working mode setting command [effective immediately, not saved when power off]	
Execution instructions: AT+WMBUSMODE _CUR =[PARAM]	response: +OK
	Parameter Description: PARAM: working mode, value range 1~7, factory default value 7 1:T1 2: T2 3: S1 4:S2 5:C1 6:C2 7:R2
example: Send: AT+WMBUSMODE_ CUR =7 \r\n Response: +OK \r\n	

5.3. 8 AT+POWER? - Transmit power query command

AT+POWER? - Transmit power query command	
Execution instructions: AT+POWER?	response: +OK AT+POWER=[PARAM]
	Parameter Description: PARAM: range 0~8, factory default value 3
example: Send: AT+POWER? \r\n Response: +OK AT+POWER=3 \r\n	

5.3. 9 AT+POWER - Transmit power setting command

AT+POWER? - Transmit power setting command [effective after restart]	
Execution instructions: AT+POWER=[PARAM]	response: +OK
	Parameter Description: PARAM: value range 0~8, factory default value 3; 0: 0dBm 1: 3dBm 2: 5dBm 3: 8dBm 4:10dBm 5:12dBm 6：15dBm 7：18dBm 8：20dBm
example: Send: AT+POWER=3 \r\n Response: +OK \r\n	

5.3. 10 AT+POWER _CUR - Transmit power setting command

AT+POWER _CUR - Transmit power setting command [effective immediately, not saved when power off]

Execution instructions: AT+POWER_CUR = [PARAM]	response: +OK
	Parameter Description: PARAM: value range 0~8, factory default value 3; 0: 0dBm 1: 3dBm 2: 5dBm 3: 8dBm 4:10dBm 5:12dBm 6：15dBm 7：18dBm 8：20dBm
example: Send: AT+POWER_ CUR =3 \r\n Response: +OK \r\n	

5.3. 11 AT+DEVICEID? - Device ID query command

AT+DEVICEID? - Device ID query command	
Execution instructions: AT+DEVICEID?	response: +OK DEVICE ID=[PARAM]
	Parameter Description: PARAM: range 1~4294967295, factory default value 1
example: Send: AT+DEVICEID? \r\n Response: +OK DEVICE ID=1 \r\n	

5.3. 12 AT+DEVICEID - Device ID setting command

AT+DEVICEID - Device ID setting command [effective after restart]	
Execution instructions: AT+DEVICEID=[PARAM]	response: +OK
	Parameter Description: PARAM: Value range [decimal] 1 ~4294967295, factory default value 1
example: Send: AT+DEVICEID=1 \r\n Response: +OK \r\n	

5.3. 13 AT+DEVICEID _CUR - Device ID setting command

AT+DEVICEID _CUR - Device ID setting command [effective immediately, not saved when power off]	
Execution instructions: AT+DEVICEID _CUR =[PARAM]	response: +OK
	Parameter Description: PARAM: Value range [decimal] 1 ~4294967295, factory default value 1
example: Send: AT+DEVICEID _CUR =1 \r\n Response: +OK \r\n	

5.3.1 4 AT+DEVICETYPE? - Device type query command

AT+DEVICETYPE? - Device type query command	
Execution instructions: AT+DEVICETYPE?	response: +OK DEVICE TYPE=[PARAM]
	Parameter Description: PARAM: Please see [Table 5-1] for the range, the factory default value is 7
example: Send: AT+DEVICETYPE? \r\n	

Response: +OK DEVICE TYPE=7 \r\n illustrate: If the device role is a collector (host), the read return value will always be 0 without changing the device type.

5.3.1 5 AT+DEVICETYPE - Device type setting command

AT+DEVICETYPE - Device type setting command [effective after restart]	
Execution instructions: AT+DEVICETYPE=[PARAM]	response: +OK
	Parameter Description: PARAM: Please see [Table 5-1] for the value range, the factory default value is 7
example: Send: AT+DEVICETYPE=7 \r\n Response: +OK \r\n illustrate: This setting command is not applicable to the collection end (host), but other values can still be written. when sending data, the relevant sending data fields will be automatically overwritten to 0 (0x00). It is not recommended that the collector (host) use this command to avoid unnecessary confusion.	

5.3.1 6 AT+DEVICETYPE _CUR - Device type setting command

AT+DEVICETYPE _CUR - Device type setting command [effective immediately, not saved when power off]	
Execution instructions: AT+DEVICETYPE_CUR = [PARAM]	response: +OK
	Parameter Description: PARAM: Please see [Table 5-1] for the value range, the factory default value is 7
example: Send: AT+DEVICETYPE _CUR =7 \r\n Response: +OK \r\n illustrate: This setting command is not applicable to the collection end (host), but other values can still be written. When sending data, the relevant sending data fields will be automatically overwritten to 0 (0x00). It is not recommended that the collector (host) use this command to avoid unnecessary confusion.	

-Device type parameters-

According to the EN 13757-3 specification, the following table terminal types are supported	
Corresponding table end type (slave)	< *Oil meter* > 1: WMBUS_DEVICE_OIL_METER < *Electricity meter* > 2: WMBUS_DEVICE_ELECTRICITY_METER < *Gas meter* > 3: WMBUS_DEVICE_GAS_METER < *Heat meter* > 4: WMBUS_DEVICE_HEAT_METER < *Steam meter* > 5: WMBUS_DEVICE_STEAM_METER < *Warm water meter (30-90C)* > 6: WMBUS_DEVICE_WARM_WATER_METER < *Water meter* > 7: WMBUS_DEVICE_WATER_METER < *Heat cost allocator* > 8: WMBUS_DEVICE_HEAT_COST_ALLOCATOR < *Compressed air* > 9: WMBUS_DEVICE_COMPRESSED_AIR < *Cooling meter (volume measured at return temperature: outlet)* > 10: WMBUS_DEVICE_COOLING_METER_OUTLET

	< *Cooling meter (volume measured at flow temperature: inlet)* > 11: WMBUS_DEVICE_COOLING_METER_INLET < *Heat meter (volume measured at flow temperature: inlet)* > 12: WMBUS_DEVICE_HEAT_METER_INLET < *Combined heat/cooling meter* > 13: WMBUS_DEVICE_COMBINED_HEAT_COOLING_METER < *Bus/system component* > 14: WMBUS_DEVICE_BUS_SYSTEM_COMPONENT < *Calorific value* > 20: WMBUS_DEVICE_CALORIFIC_VALUE = 0x14 < *Hot water meter (>90C)* > 21: WMBUS_DEVICE_HOT_WATER_METER < *Cold water meter* > 22: WMBUS_DEVICE_COLD_WATER_METER < *Dual register (hot/cold) water meter* > 23: WMBUS_DEVICE_DUAL_WATER_METER < *Pressure meter* > 24: WMBUS_DEVICE_PRESSURE_METER <A/D converter> 25: WMBUS_DEVICE_AD_CONVERTER < *Smoke detector* > 26: WMBUS_DEVICE_SMOKE_DETECTOR < *Room sensor (eg temperature or humidity)* > 27: WMBUS_DEVICE_ROOM_SENSOR < *Gas detector* > 28: WMBUS_DEVICE_GAS_DETECTOR < *Breaker (electricity)* > 32: WMBUS_DEVICE_BREAKER < *Valve (gas or water)* > 33: WMBUS_DEVICE_VALVE < *Customer unit (display device)* > 37: WMBUS_DEVICE_CUSTOMER_UNIT < *Waste water meter* > 40: WMBUS_DEVICE_WASTE_WATER_METER <Garbage> 41:WMBUS_DEVICE_GARBAGE < *Ebyte Transmission* > 255: WMBUS_DEVICE_RADIO_EBYTE_OHTER

- Table 5-1 -

5.3.1 7 AT+DEVICE VERSION ? - Device VERSION (version field in WM-Bus) query command

AT+DEVIC EVERSION ? - Device VERSION (version field in WM-B us) query command	
Execution instructions: AT+DEVICE VERSION ?	response: +OK DEVICE VERSION =[PARAM]
	Parameter Description: PARAM: range 1 ~255 , factory default value 1
example: Send: AT+DEVICE VERSION ? \r\n Response: +OK DEVICE VERSION =1 \r\n	

5.3.1 8 AT+DEVICE VERSION - Device VERSION (version field in WM-B us) setting command

AT+DEVICE VERSION - Device VERSION (version field in WM-B us) setting command (valid after restart)	
Execution instructions: AT+DEVICE VERSION=[PARAM]	response: +OK
	Parameter Description: PARAM: range 1 ~255 , factory default value 1

example: Send: AT+DEVICE VERSION=231 \r\n Response: +OK\r\n

5.3.1 9 AT+DEVICE VERSION_CUR - Device VERSION (version field in WM-B us) query command

AT+DEVICEROLE? - Device VERSION (version field in WM-B us) query command (effective immediately, not saved when power off)	
Execution instructions: AT+DEVICE VERSION_CUR=[PARAM]	response: +OK
	Parameter Description: PARAM: range 1 ~255 , factory default value 1
example: Send: AT+DEVICE VERSION_CUR=231 \r\n Response: +OK\r\n	

5.3. 20 AT+DEVIC EMANUF ? - Equipment manufacturer id query command

AT+DEVIC EMANUF ? - Device manufacturer id setting instructions	
Execution instructions: AT+ DEVICEMANUF ?	response: +OK MANUF =[PARAM]
	Parameter Description: PARAM: 0~3 characters in length, factory default value E BT
example: Send: AT+ DEVIC EMANUF ? \r\n Response: +OK MANUF=EBT \r\n	

5.3. 21 AT+DEVICE MANUF - Device manufacturer id setting command

AT+ DEVIC EMANUF - Device manufacturer ID setting instructions (restart to take effect)	
Execution instructions: AT+DEVIC EMANUF=[PARAM]	response: +OK
	Parameter Description: PARAM: 0~3 characters in length, factory default value E BT
example: Send: AT+ DEVIC EMANUF=NI \r\n Response: +OK \r\n	

5.3. 22 AT+DEVICE MANUF_CUR - Device role query command

AT+ DEVIC EMANUF _ CUR - Device manufacturer ID setting instructions (effective immediately and not saved when power off)	
Execution instructions: AT+DEVIC EMANUF_CUR=[PARAM]	response: +OK
	Parameter Description: PARAM: 0~3 characters in length, factory default value E BT , note that the parameters must be uppercase letters
example: Send: AT+ DEVIC EMANUF_CUR=NI \r\n Response: +OK \r\n	

5.3. 23 AT+DEVICEROLE? - Device role query command

AT+DEVICEROLE? - Device role query command	
Execution instructions: AT+DEVICEROLE?	response: +OK DEVICE ROLE=[PARAM]
	Parameter Description: PARAM: Range 0/1/2 , factory default value 1
example: Send: AT+DEVICEROLE? \r\n	

Response: +OK DEVICE ROLE=1 \r\n

5.3. 24 AT+DEVICEROLE - Device role setting command

AT+DEVICEROLE - Device role setting command [effective after restart]	
Execution instructions: AT+DEVICEROLE=[PARAM]	response: +OK
	Parameter Description: PARAM: value range 0/2 , factory default value 1 0: Collection end (host) 1: Table side (slave) 2: Relay (one-way relay)
example: Send: AT+DEVICEROLE=1 \r\n Response: +OK \r\n	

5.3. 25 AT+DEVICEROLE _CUR - Device role setting command

AT+DEVICEROLE - Device role setting command [effective immediately, not saved when power off]	
Execution instructions: AT+DEVICEROLE _CUR =[PARAM]	response: +OK
	Parameter Description: PARAM: value range 0/1 /2 , factory default value 1 0: Collection end (host) 1: Table side (slave) 2: Relay (one-way relay)
example: Send: AT+DEVICEROLE _CUR =1 \r\n Response: +OK \r\n	

5.3. 26 AT+ ACCESS ? – Meter device access permission query command

AT+ ACCESS ? - Meter device access permission query command	
Execution instructions: AT+ ACCESS ?	response: +OK METER ACCESS =[PARAM]
	Parameter Description: PARAM: Range 0/1 /2/3 , factory default value 2
example: Send: AT+ ACCESS ? \r\n Response: +OK METER ACCESS=2 \r\n	

5.3. 27 AT+ ACCESS - Meter device access permission setting command

AT+ ACCESS - Meter device access permission setting command (restart to take effect)	
Execution instructions: AT+ ACCESS=[PARAM]	response: +OK
	Parameter Description: PARAM: value range 0/1 /2/3 , factory default value 1 0: No access allowed 1: Access is temporarily not allowed 2: Restrict access (there is a short receive window after the send window) 3: Unlimited
example: Send: AT+ ACCESS=3 \r\n Response: +OK \r\n	

5.3. 28 AT+ ACCESS_CUR - Meter device access permission setting command

AT+ ACCESS_CUR - Meter device access permission setting command (effective immediately, not saved when

power off)	
Execution instructions: AT+ ACCESS_CUR=[PARAM]	response: +OK
	Parameter Description: PARAM: value range 0/1 /2/3 , factory default value 1 0: No access allowed 1: Access is temporarily not allowed 2: Restrict access (there is a short receive window after the send window) 3: Unlimited
example: Send: AT+ ACCESS_CUR=3 \r\n Response: +OK \r\n	

5.3. 29 AT+ENCRYPT? - Encryption mode query command

AT+ENCRYPT? - Encryption mode query command	
Execution instructions: AT+ENCRYPT?	response: +OK ENCRYPT=[PARAM]
	Parameter Description: PARAM: range 0/1, factory default value 0
example: Send: AT+ENCRYPT? \r\n Response: +OK ENCRYPT=0 \r\n	

5.3. 30 AT+ENCRYPT - Encryption mode setting command

AT+ENCRYPT - Encryption mode setting command (restart to take effect)	
Execution instructions: AT+ENCRYPT=[PARAM]	response: +OK ENCRYPT=[PARAM]
	Parameter Description: PARAM: range 0/1, factory default value 0 0: No encryption, do not use encryption for data transmission 1: Encryption, use encryption method for data transmission
example: Send: AT+ENCRYPT=0 \r\n Response: +OK \r\n	

5.3. 31 AT+ENCRYPT _CUR - Encryption mode setting command

AT+ENCRYPT - Encryption mode setting command (effective immediately, not saved when power off)	
Execution instructions: AT+ENCRYPT_CUR = [PARAM]	response: +OK \r\n
	Parameter Description: PARAM: range 0/1, factory default value 0 0: No encryption, do not use encryption for data transmission 1: Encryption, use encryption method for data transmission
example: Send: AT+ENCRYPT _CUR =0 \r\n Response: +OK \r\n	

5.3. 32 AT+PERIODTIME? - Send time query command regularly

AT+PERIODTIME? - Send time query command regularly	
Execution instructions: AT+PERIODTIME?	response: +OK PERIOD TIME=[PARAM]
	Parameter Description: PARAM: Range 300~4 000000 , factory default value 10000 (10s), unit: ms (milliseconds)

example: Send: AT+PERIODTIME? \r\n Response: +OK PERIOD TIME=10000000 \r\n

5.3. 33 AT+PERIODTIME - Send time setting command regularly

AT+PERIODTIME - Send time setting instructions regularly (restart to take effect)	
Execution instructions: AT+PERIODTIME=[PARAM]	response: +OK
	Parameter Description: PARAM: Range 300000~4294967295, factory default value 10000000 (10s), unit: us (microseconds)
example: Send: AT+PERIODTIME=10000000 \r\n Response: +OK \r\n	

5.3. 34 AT+PERIODTIME _CUR - Send time setting command regularly

AT+PERIODTIME - Send time setting instructions regularly (effective immediately, not saved when power off)	
Execution instructions: AT+PERIODTIME_CUR =[PARAM]	response: +OK
	Parameter Description: PARAM: range 300000~4294967295, factory default value 10000000 (10s), unit: us (microseconds)
example: Send: AT+PERIODTIME _CUR =10000000 \r\n Response: +OK \r\n	

5.3. 35 AT+APLDATABLOCKCOUNT? - Data block number query command

AT+APLDATABLOCKCOUNT? - data block number query command	
Execution instructions: AT+APLDATABLOCKCOUNT?	response: +OK WMBUS APL USED BLOCK COUNT [PARAM]
	Parameter Description: PARAM: range 1~4, factory default value 2
example: Send: AT+APLDATABLOCKCOUNT? \r\n Response: +OK WMBUS APL USED BLOCK COUNT 2 \r\n	

5.3. 36 AT+APLDATABLOCKCOUNT - Data block number setting command

AT+APLDATABLOCKCOUNT - data block number setting command [effective after restart]	
Execution instructions: AT+APLDATABLOCKCOUNT=[PARAM]	response: +OK
	Parameter Description: PARAM: value range 1~4, factory default value 2
example: Send: AT+APLDATABLOCKCOUNT=2 \r\n Response: +OK \r\n illustrate: This command parameter is used in standard transmission. The set value represents how much data should be input during data transmission. However, the data type, value size and unit precision are related to the data block parameter setting instructions.	

5.3. 37 AT+APLDATABLOCK1? - Data block 1 parameter query command

AT+APLDATABLOCK1? - Data block 1 parameter query command	
Execution instructions: AT+APLDATABLOCK1?	response: +OK WMBUS APL BLOCK1=[PARAM1],[PARAM2],[PARAM3]

	Parameter Description: PARAM1: Range 0~2, factory default value 0 PARAM2: Range 0~2, factory default value 2 PARAM3: range, factory default value 19
example: Send: AT+APLDATABLOCK1? \r\n Response: +OK WMBUS APL BLOCK1=0,0,0 \r\n	

5.3. 38 AT+APLDATABLOCK1 - Data block 1 parameter setting command

AT+APLDATABLOCK1 - Data block 1 parameter setting command [effective after restart]	
Execution instructions: AT+APLDATABLOCK1=[PARAM1],[PARAM2],[PARAM3]	response: +OK
	Parameter Description: PARAM1: value type, value range 0~2, factory default value 0 0: Instantaneous value 1: minimum value 2: Maximum value PARAM2: value size, value range 0~2, factory default value 2 0: 8 bits (1 Byte) 1: 16 bits (2 Byte) 2: 32 bits (4 Byte) PARAM3: Please see [Table 5-2] for the value range, the factory default value is 19
example: Send: AT+APLDATABLOCK1=0,0,0 \r\n Response: +OK \r\n	

5.3. 39 AT+APLDATABLOCK2? - Data block 2 parameter query command

AT+APLDATABLOCK2? - Data block 2 parameter query command	
Execution instructions: AT+APLDATABLOCK2?	response: +OK WMBUS APL BLOCK2=[PARAM1],[PARAM2],[PARAM3]
	Parameter Description: PARAM1: Range 0~2, factory default value 0 PARAM2: Range 0~2, factory default value 2 PARAM3: range, factory default value 19
example: Send: AT+APLDATABLOCK2? \r\n Response: +OK WMBUS APL BLOCK2=0,0,0 \r\n	

5.3. 40 AT+APLDATABLOCK2 - Data block 2 parameter setting command

AT+APLDATABLOCK2 - Data block 2 parameter setting command [effective after restart]	
Execution instructions: AT+APLDATABLOCK2=[PARAM1],[PARAM2],[PARAM3]	response: +OK
	Parameter Description: PARAM1: value type, value range 0~2, factory default value 0 0: Instantaneous value 1: minimum value 2: Maximum value PARAM2: value size, value range 0~2, factory default value 2 0: 8 bits (1 Byte) 1: 16 bits (2 Byte) 2: 32 bits (4 Byte)

	PARAM3: Please see [Table 5-2] for the value range, the factory default value is 19
example: Send: AT+APLDATABLOCK2=0,0,0 \r\n Response: +OK \r\n	

5.3. 41 AT+APLDATABLOCK3? - Data block 3 parameter query command

AT+APLDATABLOCK3? - Data block 3 parameter query command	
Execution instructions: AT+APLDATABLOCK3?	response: +OK WMBUS APL BLOCK3=[PARAM1],[PARAM2],[PARAM3]
	Parameter Description: PARAM1: Range 0~2, factory default value 0 PARAM2: Range 0~2, factory default value 2 PARAM3: range, factory default value 19
example: Send: AT+APLDATABLOCK3? \r\n Response: +OK WMBUS APL BLOCK3=0,0,0 \r\n	

5.3. 42 AT+APLDATABLOCK3 - Data block 3 parameter setting command

AT+APLDATABLOCK3 - Data block 3 parameter setting command [effective after restart]	
Execution instructions: AT+APLDATABLOCK3=[PARAM1],[PARAM2],[PARAM3]	response: +OK
	Parameter Description: PARAM1: value type, value range 0~2, factory default value 0 0: Instantaneous value 1: minimum value 2: Maximum value PARAM2: value size, value range 0~2, factory default value 2 0: 8 bits (1 Byte) 1: 16 bits (2 Byte) 2: 32 bits (4 Byte) PARAM3: Please see [Table 5-2] for the value range, the factory default value is 19
example: Send: AT+APLDATABLOCK3=0,0,0 \r\n Response: +OK \r\n	

5.3. 43 AT+APLDATABLOCK4? - Data block 4 parameter query command

AT+APLDATABLOCK4? - Data block 4 parameter query command	
Execution instructions: AT+APLDATABLOCK4?	response: +OK WMBUS APL BLOCK4=[PARAM1],[PARAM2],[PARAM3]
	Parameter Description: PARAM1: Range 0~2, factory default value 0 PARAM2: Range 0~2, factory default value 2 PARAM3: range, factory default value 19
example: Send: AT+APLDATABLOCK4? \r\n Response: +OK WMBUS APL BLOCK4=0,0,0 \r\n	

5.3. 44 AT+APLDATABLOCK4 - Data block 4 parameter setting command

AT+APLDATABLOCK4 - Data block 4 parameter setting command [effective after restart]	
Execution instructions: AT+APLDATABLOCK4=[PARAM1],[PARAM2],[PARAM3]	response: +OK

	Parameter Description: PARAM1: value type, value range 0~2, factory default value 0 0: Instantaneous value 1: minimum value 2: Maximum value PARAM2: value size, value range 0~2, factory default value 2 0: 8 bits (1 Byte) 1: 16 bits (2 Byte) 2: 32 bits (4 Byte) PARAM3: Please see [Table 5-2] for the value range, the factory default value is 19
example: Send: AT+APLDATABLOCK4=0,0,0 \r\n Response: +OK \r\n	

- Data block Vif value (unit, precision) parameter table -

List some data types and precision according to EN 13757-3 specification. Special note: This parameter is only an identifier of the data type and precision. In actual applications, users still need to parse the data in the data block themselves.	
Vif value (unit, precision)	ENERGY_WH_MIN_UNIT = 0x00, //1e-3 ENERGY_WH_MAX_UNIT = 0x07, //1e4 ENERGY_J_MIN_UNIT = 0x08, //1 ENERGY_J_MAX_UNIT = 0x0F, //1e7 VOLUME_M3_MIN_UNIT = 0x10, //1e-6 VOLUME_M3_MAX_UNIT = 0x17, //1e-1 MASS_KG_MIN_UNIT = 0X18, //1e-3 MASS_KG_MAX_UNIT = 0X1F, //1e4 ONTIME_SS_UNIT = 0x20, ONTIME_MM_UNIT = 0x21, ONTIME_HH_UNIT = 0x22, ONTIME_DD_UNIT = 0x23, OPERATINGTIME_SS_UNIT = 0x24, OPERATINGTIME_MM_UNIT = 0x25, OPERATINGTIME_HH_UNIT = 0x26, OPERATINGTIME_DD_UNIT = 0x27, POWER_W_MIN_UNIT = 0x28, POWER_W_MAX_UNIT = 0x2F, POWER_J_H_MIN_UNIT = 0x30, POWER_J_H_MAX_UNIT = 0x37, VOLUME_FLOW_M3_H_MIN_UNIT = 0x38, VOLUME_FLOW_M3_H_MAX_UNIT = 0x3F, VOLUME_FLOW_M3_M_MIN_UNIT = 0x40, VOLUME_FLOW_M3_M_MAX_UNIT = 0x47, VOLUME_FLOW_M3_S_MIN_UNIT = 0x48, VOLUME_FLOW_M3_S_MAX_UNIT = 0x4F, MASS_FLOW_KG_H_MIN_UNIT = 0x50, MASS_FLOW_KG_H_MAX_UNIT = 0x57, FLOW_TEMPERATURE_C_MIN_UNIT = 0x58, FLOW_TEMPERATURE_C_MAX_UNIT = 0x5B, RETURN_TEMPERATURE_C_MIN_UNIT = 0x5C, RETURN_TEMPERATURE_C_MAX_UNIT = 0x5F, TEMPERATURE_K_MIN_UNIT = 0x60, TEMPERATURE_K_MAX_UNIT = 0x63, EXTERNAL_TEMPERATURE_C_MIN_UNIT = 0x64,

	EXTERNAL_TEMPERATURE_C_MAX_UNIT = 0x67, PRESSSURE_BAR_MIN_UNIT = 0x68, PRESSSURE_BAR_MAX_UNIT = 0x6B, EBYTE_TRANSPARENT_METER = 0xFF

- Table 5-2 -

5.3. 45 AT+ERRORLOGEN? - Enable radio frequency data error log query command

AT+ERRORLOGEN? - Enable RF data error log query command	
Execution instructions: AT+ERRORLOGEN?	response: +OK AT+ERRORLOGEN=[PARAM]
	Parameter Description: PARAM: value range 0/1, factory default value 0 0: Close log 1: Turn on log
example: Send: AT+ERRORLOGEN? \r\n Response: +OK AT+ERRORLOGEN=0 \r\n	

5.3. 46 AT+ERRORLOGEN - Enable radio frequency data error log setting command

AT+ERRORLOGEN - Enable radio frequency data error log setting command [effective after restart]	
Execution instructions: AT+ERRORLOGEN=[PARAM]	response: +OK
	Parameter Description: PARAM: value range 0/1, factory default value 0 0: Close log 1: Turn on log
example: Send: AT+ERRORLOGEN=0 \r\n Response: +OK \r\n	

5.3. 47 AT+ R2CHNUM? – R2 mode channel number query command

AT+ R2CHNUM - R2 mode channel number query command	
Execution instructions: AT +R2CHNUM?	response: +OK R2 CHANNEL NUM=[PARAM]
	Parameter Description: **of** this parameter is related to the R2 channel mode (see **5.3.77**) Channel mode 0: value range 0 ~3 , factory default value is 0 Channel mode 1: value range 0 ~9
example: Send: AT+ R2CHNUM? \r\n Response: +OK R2 CHANNEL NUM=0 \r\n	

5.3. 48 AT+ R2CHNUM - R2 mode channel number setting command

AT+ R2CHNUM - R2 mode channel number setting command (valid after restart)	
Execution instructions: AT +R2CHNUM=[PARAM]	response: +OK
	Parameter Description: **of** this parameter is related to the R2 channel mode (see **5.3.49**) Channel mode 0: value range 0 ~3 , factory default value is 0 Channel mode 1: value range 0 ~9
example: Send: AT+ R2CHNUM=1 \r\n Response: +OK \r\n	

5.3. 49 AT+ R2CHMODE ? – R 2 channel mode query

AT+R 2CHMODE ? – R 2 channel mode query command	
Execution instructions: AT+ R2CHMODE ?	response: +OK R2CHMODE=0
	Parameter Description: PARAM : 0/1, , the default parameter at power-on is 0 0 : R 2 is enabled by default , 4-channel mode, that is, 4 channels can be configured in R2 mode. 1 : Enable R 2 , 10 channel mode, that is, 10 channels can be configured in R 2 mode.
example: Send: AT+R 2CHMODE ? \r\n Response: +OK R2CHMODE=0 \r\n	

5.3. 50 AT+ R2CHMODE= – R2 channel mode setting

AT+R 2CHMODE= - R 2 channel mode setting command	
Execution instructions: AT+ R2CHMODE=1	response: +OK
	Parameter Description: PARAM : 0/1, , the default parameter at power-on is 0 0 : R 2 is enabled by default , 4-channel mode, that is, 4 channels can be configured in R2 mode. 1 : Enable R 2 , 10 channel mode, that is, 10 channels can be configured in R 2 mode. Notice: If you set A T+ R 2CHMODE=0 normally and return E RROR PARAMS , please first check whether the value of R 2CHNUM meets the channel mode. For example: R2CHNUM=9 , setting A T+ R 2CHMODE=0 will fail.
example: Send: AT+R 2CHMODE=1 \r\n Response: +OK \r\n	

5.3. 51 AT+ R2CHNUM_CUR - R2 mode channel number setting command

AT+ R2CHNUM_CUR - R2 mode channel number setting command (effective immediately, not saved when power off)	
Execution instructions: AT +R2CHNUM_CUR=[PARAM]	response: +OK
	Parameter Description: PARAM: value range 0 ~3 (corresponding to 4 channels, R2 mode can have 4 channels for use) , the factory default value is 0
example: Send: AT+ R2CHNUM_CUR=1 \r\n Response: +OK \r\n	

5.3. 52 AT+ PRINTMODE? –Module output mode query command

AT +PRINTMODE? –Module output mode query command	
Execution instructions: AT +PRINTMODE?	response: +OK PRINT MODE=[PARAM]
	Parameter Description: PARAM: value range 0 ~2 , factory default value is 0
example: Send: AT +PRINTMODE? \r\n Response: +OK PRINT MODE=0 \r\n	

5.3. 53 AT+ PRINTMODE –Module output mode setting instructions

AT +PRINTMODE –Module output mode setting command (valid after restart)	
Execution instructions: AT +PRINTMODE=[PARAM]	response: +OK
	Parameter Description: PARAM: value range 0 ~2 , factory default value is 0 O : A SCII code, formatted output 1 : ASCII code, hex String output 2 : H ex code is output directly
example: Send: AT +PRINTMODE=0 \r\n Response: +OK \r\n	

5.3. 54 AT+ PRINTMODE_CUR –Module output mode setting instructions

AT +PRINTMOD_CUR –Module output mode setting command (effective immediately, not saved when power off)	
Execution instructions: AT +PRINTMODE_CUR=[PARAM]	response: +OK
	Parameter Description: PARAM: value range 0 ~2 , factory default value is 0 O : A SCII code, formatted output 1 : ASCII code, hex String output 2 : H ex code is output directly
example: Send: AT +PRINTMODE_CUR=0 \r\n Response: +OK \r\n	

5.3. 55 AT+ APPHEADER? – The WM-B us application layer uses protocol header query instructions

AT +APPHEADER? –WM-B us application layer uses protocol header query instructions	
Execution instructions: AT +APPHEADER?	response: +OK APP HEADER=[PARAM]
	Parameter Description: PARAM: value range 0 ~2 , factory default value is 0
example: Send: AT +APPHEADER? \r\n Response: +OK APP HEADER=0 \r\n	

5.3. 56 AT+ APPHEADER - WM-B us application layer uses protocol header setting instructions

AT +APPHEADER –WM-B us application layer uses protocol header setting instructions (restart to take effect)	
Execution instructions: AT +APPHEADER=[PARAM]	response: +OK
	Parameter Description: PARAM: value range 0 ~2 , factory default value is 0 0 : No application layer protocol header 1 : Short application layer protocol header 2 : Long application layer protocol header
example: Send: AT +APPHEADER=1 \r\n Response: +OK \r\n	

5.3. 57 AT+ APPHEADER_CUR - WM-B us application layer uses protocol header setting instructions

AT + APPHEADER_CUR –WM-B us application layer uses protocol header setting instructions (effective immediately, not saved when power off)	
Execution instructions: AT	response: +OK

+APPHEADER_CUR=[PARAM]	Parameter Description: PARAM: value range 0 ~2 , factory default value is 0 0 : No application layer protocol header 1 : Short application layer protocol header 2 : Long application layer protocol header
example: Send: AT +APPHEADER_CUR=1 \r\n Response: +OK \r\n	

5.3. 58 AT+ INSTALLED_DEVICE? – Query command for the list of installed devices on the other end

AT +INSTALLED_ DEVICE? – Query command for the list of installed devices on the other end	
Execution instructions: AT +INSTALLED_ DEVICE?	response: [[PARAM1],[PARAM2],[PARAM3],[PARAM4],[PARAM5],[PARAM6]] * 40
	Parameter Description: PARAM 1 : The id of the device in the installation list PARAM2 : Installation device information-manufacturer id P ARAM3 : Installation device information-device type PARAM4 : Installation device information-device version number P ARAM5 : Installation device information-device id P ARAM6 : Installation device information-encryption key
example: Send: AT +INSTALLED_DEVICE? \r\n response: id:0 manf:EBT deviceType:7 veris on:1 devId:1 key:000102030405060708090A0B0C0D0E0F id:1 manf:@@@ deviceType:0 verison:0 devId:0 key:00000000000000000000000000000000 id:2 manf:@@@ deviceType:0 verison:0 devId:0 key:00000000000000000000000000000000 id:3 manf:@@@ deviceType:0 verison:0 devId:0 key:00000000000000000000000000000000 id:4 manf:@@@ deviceType:0 verison:0 devId:0 key:00000000000000000000000000000000 id:5 manf:@@@ deviceType:0 verison:0 devId:0 key:00000000000000000000000000000000 . . . id:39 manf:@@@ deviceType:0 verison:0 devId:0 key:00000000000000000000000000000000	

5.3. 59 AT+ IN_INSTALL –Instructions to enter installation mode

AT +IN_INSTALL –Enter installation mode	
Execution instructions: AT +IN_INSTALL	response: INSTALL MODE
	Function Description: Meter side: Entering installation mode, the module will send an installation request every 10s until the device receives installation confirmation from the main device. Other side: Enter the installation mode, the module will only output the received installation information, and the user can cooperate with the AT+INSTALL_DEVICE command to install the device. Repeater side: This command is not supported.
example: Send: AT +IN_INSTALL \r\n Response: I NSTALL MODE \r\n	

5.3. 60 AT+ OUT_INSTALL – Exit installation mode command

AT +OUT_INSTALL –Exit installation mode

Execution instructions: AT +OUT_INSTALL	response: OUT INSTALL MODE
	Function Description: Meter side: Exit installation mode and enter normal sending logic. Other side: Exit installation mode and enter normal output mode. Repeater side: This command is not supported.
example: Send: AT +OUT_INSTALL \r\n Response: O UT I NSTALL MODE \r\n	

5.3. 61 AT+ INSTALLDEVICE – other terminal, install device instructions

AT +INSTALLDEVICE -install equipment	
Execution instructions: AT +INSTALLDEVICE= [PARAM1],[PARAM2],[PARAM3],[PARAM4],[PARAM5]	response: +OK
	Parameter Description: PARAM1: manufacturer ID PARAM2: device type PARAM3: device version number PARAM4: device id PARAM5: device key
example: Send: AT+INSTALLDEVICE=EBT,8,1,1,000102030405060708090A0B0C0D0E0F \r\n Response: + OK \r\n Notice: This command can only be used by the host in installation mode.	

5.3. 62 AT+ DEVICESFLUASH – Instructions to write installation device information into flash

AT +DEVICEFLUSH – Instructions to write installation device information into flash	
Execution instructions: AT +DEVICESFLUSH	response: +OK
	Note: The AT+INSTALL_DEVICE command only writes device information into RAM. After finally completing the writing of all installed device information, you need to use this command to write the information into flash.
example: Send: AT+DEVICESFLUSH \r\n Response: + OK \r\n	

5.3 .63 AT+ DEVICEDELET –Delete slave device information command

AT +DEVICEDELET –Delete slave device information command	
Execution instructions: AT +DEVICEDELET=[PARAM]	response: +OK
	Parameter Description: PARAM: The id of the device in the list
example: Send: AT +DEVICEDELET=0 \r\n Response: + OK \r\n	

5.3.64 AT+READMSGMAIL – Read the default message command in msg mail

AT+READMSGGMAIL – Read the preset message in msg mail	
Execution instructions: AT+READMSGMAIL=[PARAM1]	response: +OK MSGMAIL=[PARAM2],[PARAM3],[PARAM4],[PARAM5]

	Parameter Description: PARAM1: Default message number 0~14 PARAM2: Number in the installed device PARAM3: command code PARAM4: Whether the payload of this message is encrypted PARAM5: User data that needs to be sent
example: Send: AT+READMSGMAIL=0 \r\n Response: +OK MSGMAIL=0,8,0,010203040506070809 \r\n	

5.3.65 AT+SETMSGMAIL – Set the default message command in msg mail

AT+SETMSGMAIL – Set the default message command in msg mail (effective immediately, not saved when power off)	
Execution instructions: AT+SETMSGMAIL=[PARAM1],[PARAM2],[PARAM3],[PARAM4],[PARAM5]	response: +OK MSGMAIL=0,8,0,010203040506070809
	Parameter Description: PARAM1: Default message number 0~14 PARAM2: Number in the installed device PARAM3: command code PARAM4: Whether the payload of this message is encrypted PARAM5: User data that needs to be sent
example: Send: AT+SETMSGMAIL=2,0,8,0,010203040506070809 \r\n Response: +OK \r\n	

5.3.66 AT+PARENTINFO – slave device obtains master device information command

AT+PARENTINFO – slave device obtains master device information command	
Execution instructions: AT+PARENTINFO?	response: +OK MSGMAIL=[PARAM1],[PARAM2],[PARAM3],[PARAM4],[PARAM5]
	Parameter Description: PARAM1: Is it installed on the master station device? PARAM2: Manufacturer id of the main station device, PARAM3: Device type of master device PARAM4: Device id of the master device PARAM5: Device version of the master device
example: Send: AT+PARENTINFO? \r\n Response: +OK PARENT=0,@@@,0,0,0 \r\n	

5.3.67 AT+REMOVEPARENT – Delete master station equipment and related information commands

AT+REMOVEPARENT - Delete master station equipment and related information commands	
Execution instructions: AT+REMOVEPARENT	response: +OK
	Instruction description: When the device wants to be reinstalled on a new collector device, the original master device needs to be deleted first.
example: Send: AT+REMOVEPARENT \r\n Response: +OK \r\n	

5.3. 68 AT+ KEY? – Encryption key query command of meter device

AT +KEY? - Encryption key query command of meter device	
Execution instructions: AT +KEY?	response: +OK KEY=[PARAM]
	Parameter Description: PARAM: 16-byte key (note that due to the encryption algorithm used, the key must be 16 bytes), the default is 00112233445566778899AABBCCDDEEFF (hex string) .
example: Send: AT +KEY? \r\n Response: +OK KEY=00112233445566778899AABBCCDDEEFF \r\n	

5.3. 69 AT+ KEY – Meter device encryption key setting instructions

AT +KEY - Encryption key setting command for meter device (restart to take effect)	
Execution instructions: AT +KEY= [PARAM]	response: +OK
	Parameter Description: PARAM: 16-byte key (note that due to the encryption algorithm used, the key must be 16 bytes), the default is 00112233445566778899AABBCCDDEEFF.
example: Send: AT +KEY=000102030405060708090A0B0C0D0E0F \r\n Response: +OK \r\n	

5.3. 70 AT+ AUTOMSG? –Whether the master-slave function enables query instructions

AT +AUTOMSG? - Is the master-slave function enabled to query the command?	
Execution instructions: AT +AUTOMSG?	response: +OK AUTOMSG=[PARAM]
	Parameter Description: PARAM: 0/1,, the default parameter on power-on is 0 0: Not enabled 1: enable
example: Send: AT +AUTOMSG? \r\n Response: +OK AUTOMSG=0 \r\n	

5.3. 71 AT+ AUTOMSG_CUR –Master- slave function setting instructions

AT +AUTOMSG - master-slave station function setting command (effective immediately and not saved when power off)	
Execution instructions: AT +AUTOMSG_CUR= [PARAM]	response: +OK
	Parameter Description: PARAM: 0/1,, the default parameter on power-on is 0 0: Not enabled 1: enable
example: Send: AT +AUTOMSG_CUR=1 \r\n Response: +OK \r\n	

5.3. 72 AT+ AUTOMSG_DEF –Master- slave function setting instructions

AT +AUTOMSG - master-slave station function setting command (valid after restart, saved after power off)	
Execution instructions:	response: +OK

AT +AUTOMSG_DEF=[PARAM]	Parameter Description: PARAM: 0/1,, the default parameter on power-on is 0 0: Not enabled 1: enable
example: Send: AT +AUTOMSG_DEF=1 \r\n Response: +OK \r\n	

-RF data error log description table-

Error code	illustrate
1	Reception aborted
2	Frame errors, including CRC check errors, block decoding errors, illegal frame length
3	Other errors
4	RF self-check error

- Table 5-3 -

5.3.73 AT+SLEEP - Enter deep sleep command

AT+SLEEP - Enter deep sleep command	
Execution instructions: AT+SLEEP	response: DEEP SLEEP
	Parameter description: None
example: Send: AT+SLEEP \r\n Response: DEEP SLEEP \r\n illustrate: After the module enters deep sleep, the module's radio frequency circuit and MCU stop working. The AT command cannot wake up the module and can only be woken up through pin (13) PA00.	

5.3.74 AT+EXIT - Exit configuration mode command

AT+EXIT - Exit configuration mode command	
Execution instructions: AT+EXIT	response: +OK EXIT AT MODE
	Parameter description: None
example: Send: AT+EXIT \r\n Response: +OK EXIT AT MODE \r\n	

5.3.75 AT+RESET - Module restart command

AT+RESET - Module restart command	
Execution instructions: AT+RESET	Response: RESET
	Parameter description: None
example: Send: AT+EXIT \r\n Response: RESET \r\n (after module restart) Serial port response: DEVICE ROLE:[ROLE] \r\n WAKE 0	

5.3.76s AT+RESTORE - Restore factory settings command

AT+RESTORE - Restore factory settings command	
Execution instructions: AT+RESTORE	Response: RESTORE RESET
	Parameter description: None
example: Send: AT+RESTORE \r\n Response: RESTORE \r\n RESET \r\n (after module restart) Serial port response: DEVICE ROLE:[ROLE] \r\n WAKE 0	

5.3. 77 AT+HARDWAREVER? - Hardware version query command

AT+HARDWAREVER? - Hardware version query command	
Execution instructions: AT+HARDWAREVER?	response: +OK HARDWARE VERSION:xxx
	Parameter description: None
example: Send: AT+HARDWAREVER? \r\n Response: +OK HARDWARE VERSION:1.0.0 \r\n	

5.3. 78 AT+SOFTWAREVER? - Software version query command

AT+SOFTWAREVER? - software version query command	
Execution instructions: AT+SOFTWAREVER?	response: +OK SOFTWARE VERSION:xxx
	Parameter description: None
example: Send: AT+SOFTWAREVER? \r\n Response: +OK SOFTWARE VERSION:1.0.0 \r\n	

Chapter 6 Quick Start

6. 1 Protocol mode data transmission

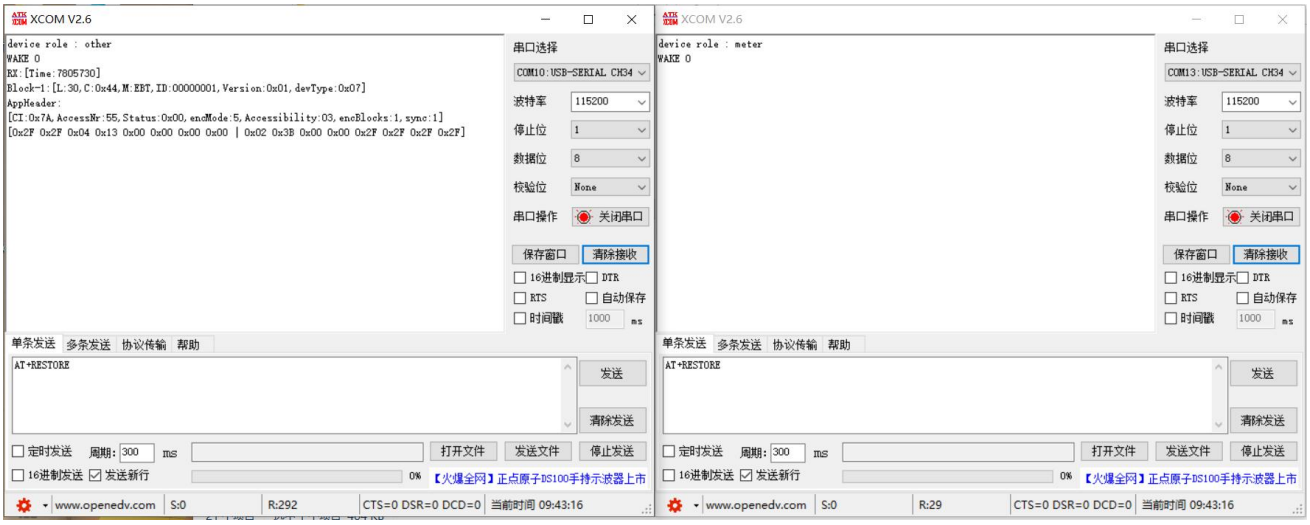
6.1.1 Preparation

Prepare two modules A and B. The factory default module is Meter role.

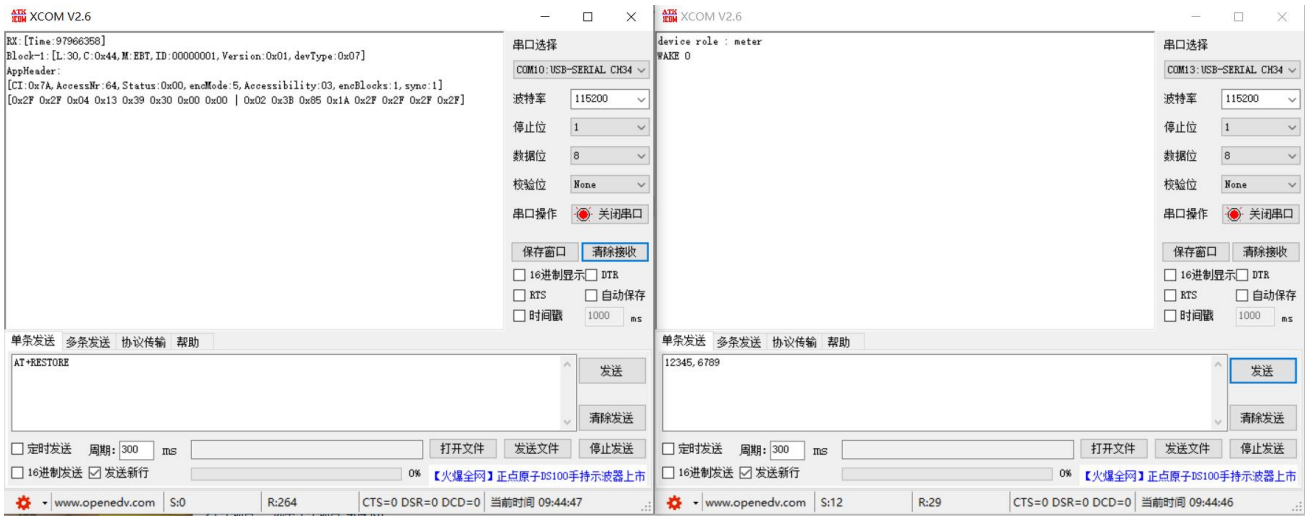
6.1.2 Parameter settings

1. Modify module A to the role of Other:
   - The module is connected to the serial port and powered on;
   - Enter ++++ to enter configuration mode;
   - Enter AT+DEVICEROLE=0, press Enter and line feed (check Send New Line), and set module A to the Other role;
   - Enter AT+RESET to restart to take effect.

2. Power on module B and use the factory default parameters (role is Meter). The module will start to automatically cycle and regularly send standard frames when used as a water meter. As shown in the figure below, standard frames are automatically sent after power-on.



3. When module B reports its own data as a Meter, it only needs to send the corresponding data according to its own configuration, as shown in the figure below. Detailed unpacking is in accordance with EN 13757-3 specifications. The user data sent in the figure is two blocks. The first data block is 0x04 0x13 0x39 0x30 0x00 0x00, which means the instantaneous value is $12345 \times 1e^{-3}\ m^3$. The second data block is 0x02 0x3B 0x85 0x1A, which means the instantaneous value is $6789 \times 1e^{-3}\ m^3\ /\ h$.



6.2 EBT transparent transmission mode data transmission

6.2.1 Preparation

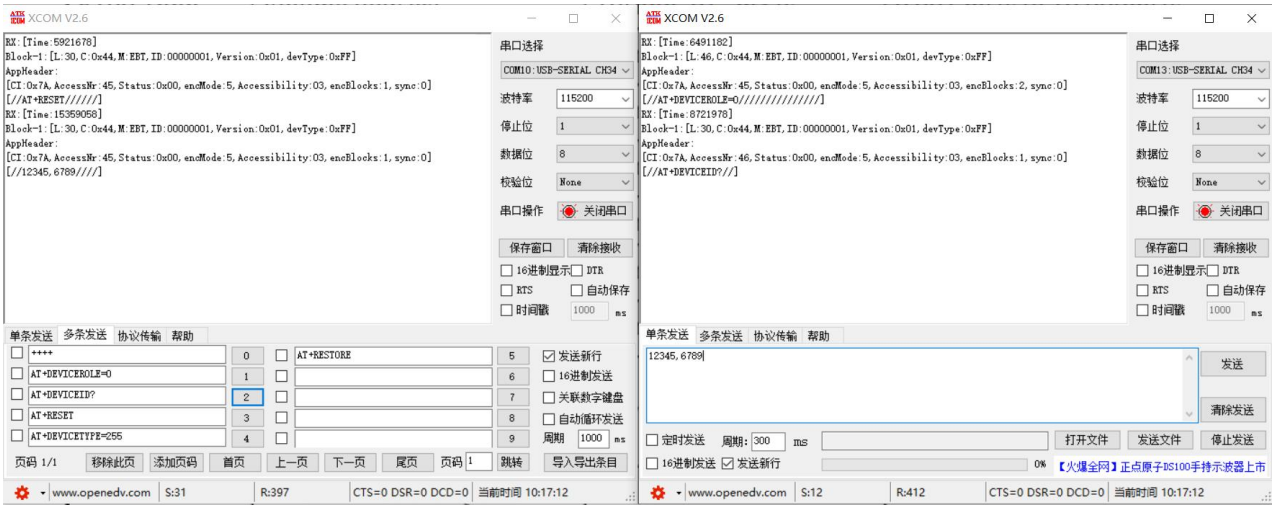
Prepare two modules A and B , and restore the parameters of these two modules to factory default values.

6.2.2 Parameter settings

1. Change module A to the Other role and the device type to EBT type:
   - The module is connected to the serial port and powered on;
   - Enter ++++ to enter configuration mode;
   - Enter AT+DEVICEROLE=0, press Enter and line feed (check Send New Line), and set the module A role to the Other role;
   - Enter AT+DEVICETYPE=255, press Enter and line feed (check Send New Line), and set the module A device type to EBT type;
   - Enter AT+RESET to restart to take effect.

2. Modify module B to the Meter role and the device type to the EBT type:
   - The module is connected to the serial port and powered on;
   - Enter ++++ to enter configuration mode;
   - Enter AT+DEVICEROLE=1, press Enter and line feed (check Send New Line), and set the module A role to the Other role;
   - Enter AT+DEVICETYPE=255, press Enter and line feed (check Send New Line), and set the module A device type to EBT type;
   - Enter AT+RESET to restart to take effect.

3. The device can then transparently transmit data directly. (Note that in this mode, data can only be sent when the module is in wake-up mode). As shown in the figure below, note that the slashes in the user's transmitted data are the reasons for using encryption mode. After sending without encryption, the slashes can be eliminated.



6.3 Installation and binding of equipment

If you need to use the automatic recovery function of the master and slave stations, you must first complete the device installation.

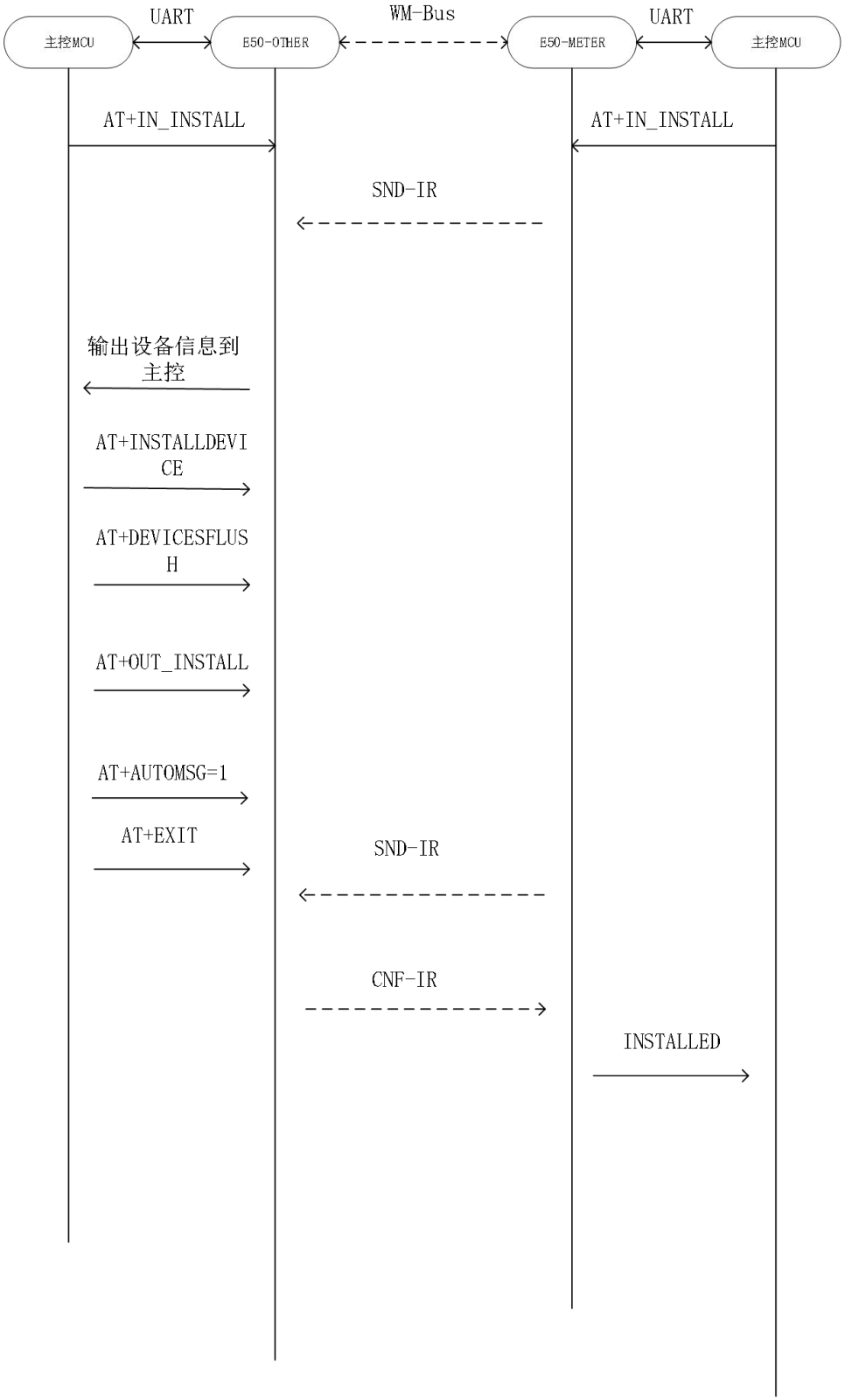
6.3.1 Brief description

When installing Meter equipment, please first determine the device-related address parameters (manufacturer ID, device type, device version, device ID, etc.) to maintain the uniqueness of the device.

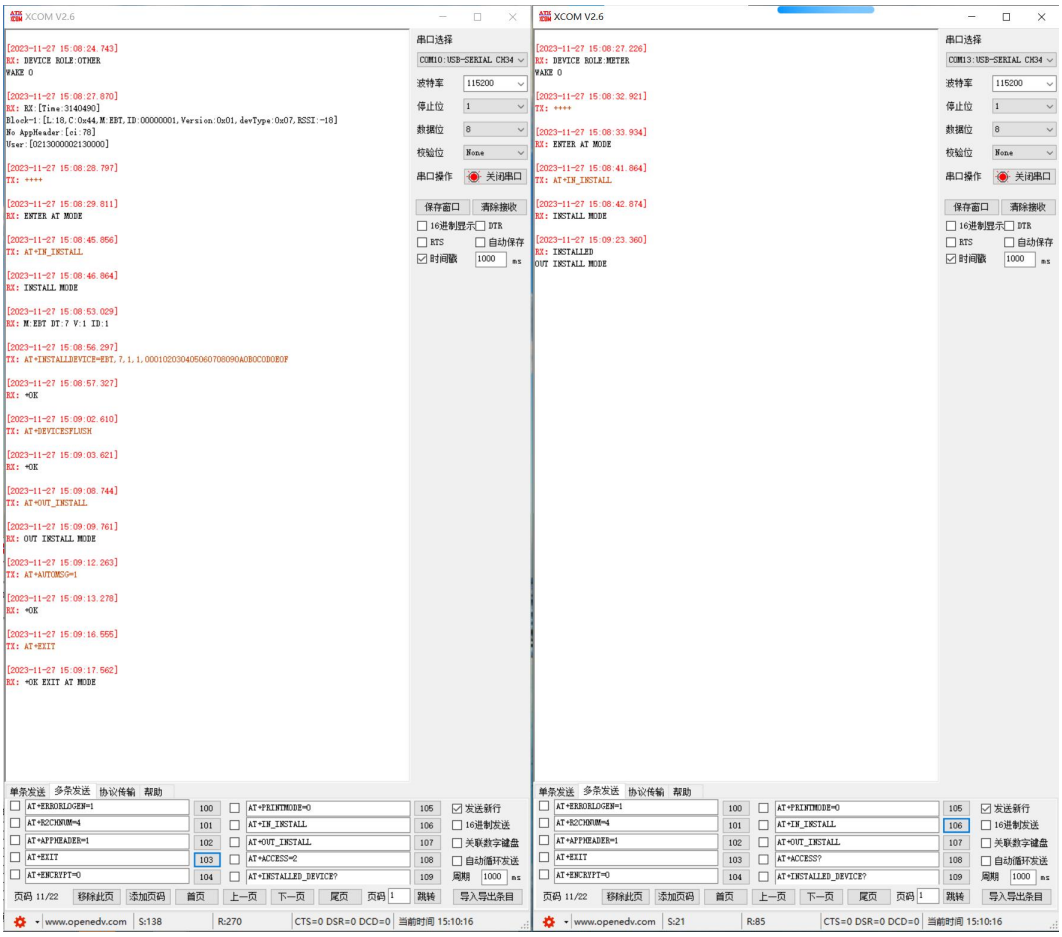
In the other-side role, it supports storing information of up to 40 installed devices, as well as encryption keys.

There can only be one master device on the Meter side.

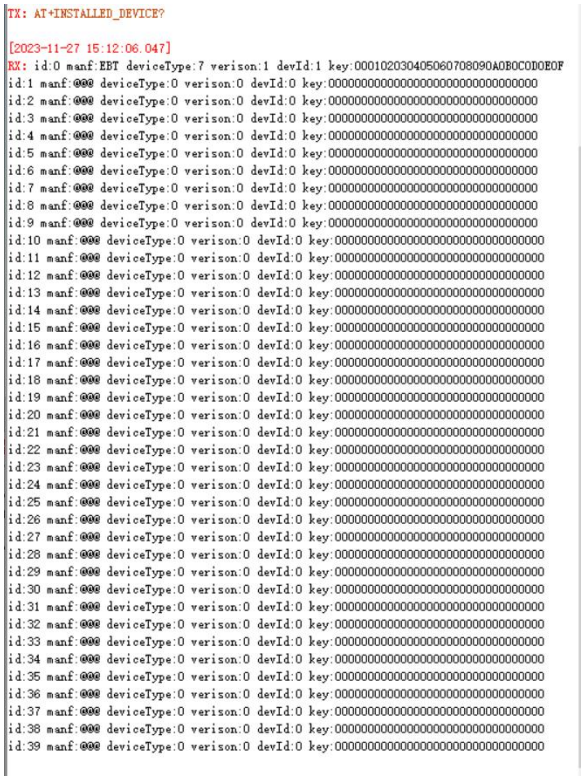
6. 3. 2 Equipment installation flow



6. 3. 3 Installation example



After entering the above process, the host can use AT+INSTALLED_DEVICE? to query the slave devices that have been installed on the device, as shown in the figure.



The slave device can also query its master device through AT+PARENTINFO?. If reinstallation is required, The host and slave machines need to use AT+DEVICEDELET and AT+REMOVEPARENT to delete relevant device information separately.

6.4 Master and slave stations automatically reply

6. 4 . 1 C CODE automatic reply type table

C Code	name	direction	describe	Reply to C code	Reply name
0x46	SND-IR	Meter to other	Meter side initiates installation request	0x06	CNF-IR
0x47	ACC-NR	Meter to other	Meter side provides access window	0x73/0x53	SND-UD
				0x5a/0x7a	REQ-UD1
				0x5b/0x7b	REQ-UD2
				no reply	no reply
0x48	ACC-DMD	Meter to othe	Mete side requests access	0x73/0x53	SND-UD
				0x5a/0x7a	REQ-UD1
				0x5b/0x7b	REQ-UD2
				no reply	no reply
0x44	SND-NR	Meter to other	No reply reported on the Meter side	0x73/0x53	SND-UD
				0x5a/0x7a	REQ-UD1
				0x5b/0x7b	REQ-UD2
				no reply	no reply
0x53/0x73	SND-UD	Other to meter	Other terminal sends user-defined commands	0x00/0x20	ACK
				0x08/0x28	RSP-UD
0x5a/0x7a	REQ-UD1	Other to meter	Other side requests first-level user data	0x00/0x20	ACK
				0x08/0x28	RSP-UD
0x5b/0x7b	REQ-UD2	Other to meter	Other side requests secondary user data	0x00/0x20	ACK
				0x08/0x28	RSP-UD

Note: In the automatic reply of the master and slave stations, only long header can be used for data packet framing by default. The destination address of the frame message will be reflected in the address field in the long header.

Level 1 user data is defined as emergency user data according to EN13737-4.

Secondary user data is defined as ordinary user data according to EN13737-4.

In the above C Code field framing, according to the standard, only SND-UD, and the first-level RSP-UD can carry user data.

6. 4. 2 frame message long Description of address information in header field

The data in the ordinary automatic reply message will be determined based on the link layer data in the received message. The data in SND-UD , REQ-UD1 , and REQ-UD2 will be determined based on the preset installed device index.

6. 4. 3 Example of using preset messages to make automatic replies

Prepare a pair of master-slave modules that have been installed and bound.

- The module is connected to the serial port and powered on ;
- Enter ++++ to enter configuration mode;
- Both the master and slave modules use the AT+AUTOMSG=1 module to enter the automatic reply mode.

In this mode, messages from non-installed devices will be automatically filtered ;

- The host sets a command message AT+SETMSGMAIL=2,0,3,0,010203040506070809 ;
- If the slave does not set relevant data, the slave will automatically reply with ACK data. As shown in the picture

- If the slave machine sets the R SP-UD message, the slave module will automatically reply to the data in the preset message. As shown in the picture

6.5 One-way relay

By setting AT+DEVICEROLE=2, the module can be set as a one-way repeater. Repeater mode supports S, T, and C modes. The repeater cannot enter installation mode, cannot install equipment, and cannot actively initiate installation requests. The repeater will repeat all the table-side messages it receives. And send the message after a random 5-20s delay after receiving the first message. The repeater supports processing of up to 15 messages simultaneously.

6. 5 . 1 Example of using T mode one-way relay

- SetAT +DEVICEROLE=2 ;
- Set AT+WMBUSMODE=2 ;
- A T+RESET ;

The other two modules are the receiving end and the meter end. The usage scenario of the repeater mode is that when the receiving end and the meter end are far apart, the repeater can extend the reporting distance of the meter end to a certain extent. The figure below shows the message output of a one-way repeater operation. The left 1 is the receiving end, the middle is the meter end, and the right 1 is the relay end.

As shown in the figure above, the table end first initiated the message. Due to the short distance, both the relay end and the receiving end received the message. Then about 10 seconds later, the relay end repeated the message to the receiving end.

Chapter 7 Precautions

7.1 Instructions for use

- The Other role only supports data from 2); S2 (Mode 4) can receive data from S1 (Mode 3) and S2 (Mode 4); C2 (Mode 6) can receive data from C1 (Mode 5) and C2 (Mode 6); R2 (Mode 7) can only receive data from R2 (Mode 7);
- EBT transparent transmission mode Devicetype=255, data can only be sent in wake-up mode;
- Note that due to the power supply design and the chip characteristics of Xinke EFR32FG23 SOC, using DC-DC to power some internal modules of the chip can effectively reduce power consumption, but it will increase the minimum operating voltage of the chip V > 2.5V;

7.2 Hardware design

- It is recommended to use a DC regulated power supply to power the module. The power supply ripple coefficient should be as small as possible, and the module must be reliably grounded;
- Please pay attention to the correct connection of the positive and negative poles of the power supply. Reverse connection may cause permanent damage to the module;
- Please check the power supply to ensure that it is within the recommended power supply voltage. If it exceeds the maximum value, it will cause permanent damage to the module;
- Please check the stability of the power supply. The voltage cannot fluctuate greatly and frequently;
- When designing the power supply circuit for the module, it is often recommended to reserve more than 30% margin, so that the whole machine can work stably for a long time;
- The module should be kept as far away as possible from power supplies, transformers, high-frequency wiring and other parts with high electromagnetic interference;
- High-frequency digital traces, high-frequency analog traces, and power traces must be avoided under the module. If it is absolutely necessary to pass under the module, assuming that the module is welded on the Top Layer, lay copper ground on the Top Layer of the module contact part (all copper and well grounded), it must be close to the digital part of the module and routed on the Bottom Layer;
- Assuming that the module is welded or placed on the Top Layer, it is also wrong to route traces randomly on the Bottom Layer or other layers, which will affect the module's spurious and receiving sensitivity to varying degrees;
- Assuming that there are devices with large electromagnetic interference around the module, which will also greatly affect the performance of the module, it is recommended to stay away from the module according to the intensity of the interference. If the situation allows, appropriate isolation and shielding can be done;
- Assuming that there are traces with large electromagnetic interference around the module (high-frequency digital, high-frequency analog, power traces), it will also greatly affect the performance of the module. It is recommended to stay away from the module according to the intensity of the interference. If the situation allows, you can make appropriate adjustments. isolation and shielding;
- If the communication line uses 5V level, a 1k-5.1k resistor must be connected in series (not recommended, there is still a risk of damage);

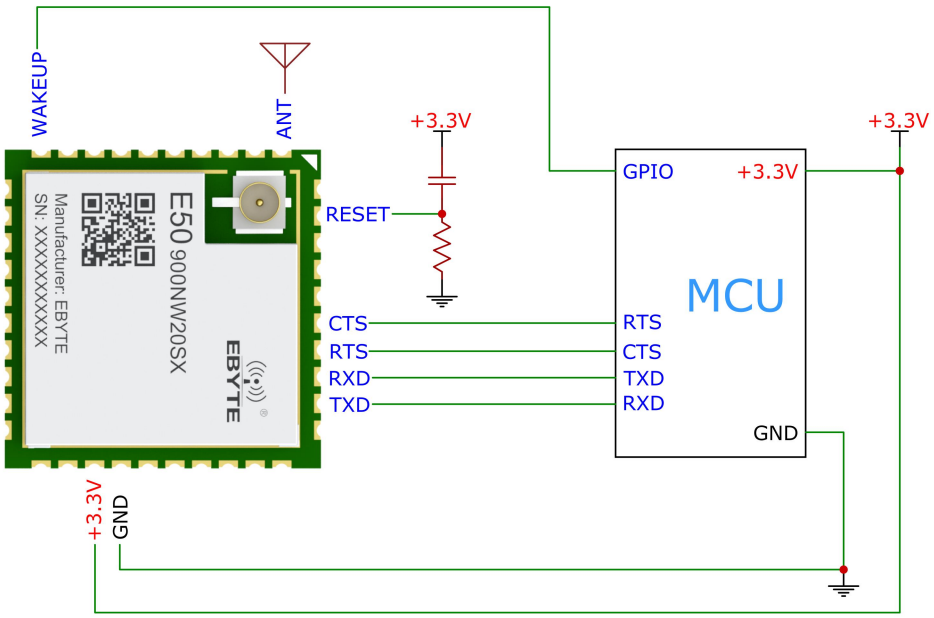
- Try to stay away from some TTL protocols whose physical layer is also 2.4GHz, such as USB3.0;
- The antenna installation structure has a great impact on module performance. Make sure the antenna is exposed, preferably vertically upward. When the module is installed inside the case, you can use a high-quality antenna extension cable to extend the antenna to the outside of the case;
- The antenna must not be installed inside a metal shell, as this will greatly reduce the transmission distance.
- If the module is connected to an external MCU, it is recommended to add a 200R protection resistor to the RXD/TXD of the external MCU.

7.3 Software writing

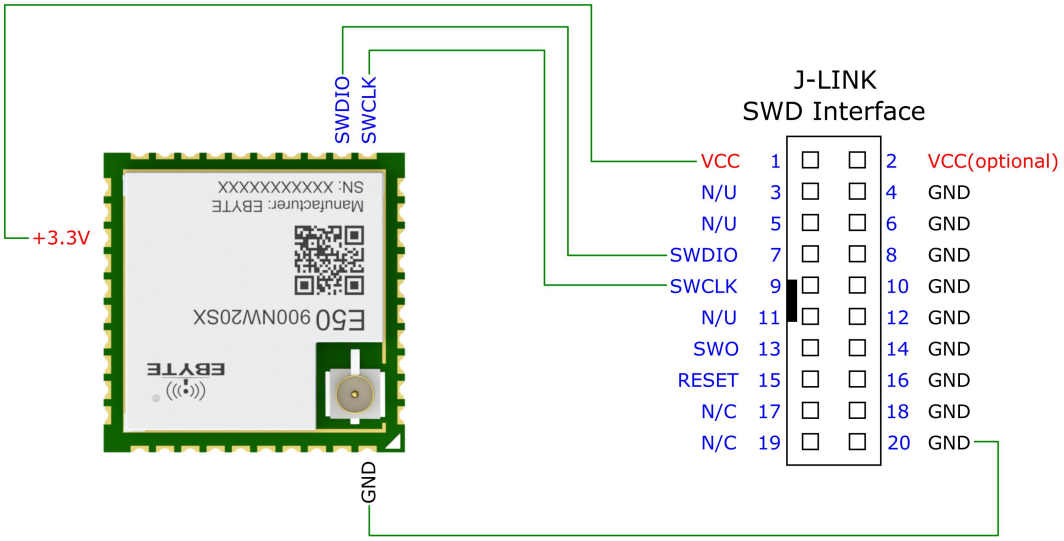
- This module is equipped with the EFR32FG2 3 series chip, and its driving method is completely the same as the EFR32FG2 3 series chip. Users can operate according to the EFR32FG2 3 series chip manual;
- A 39MHz high-frequency crystal oscillator has been connected inside the module;
- A 32.768KHz low-frequency crystal oscillator has been connected inside the module;
- EFR32FG23A020F256GM40-B chip: Download information from Silicon Labs official website ;
- Silicon Labs Wi-SUN SDK download address: gecko_sdk (Github) ;

Chapter 8 Reference Circuit

8.1 Basic circuit wiring diagram



8.2 J-Link program download/debugging wiring diagram



Chapter 9 Frequently Asked Questions

9.1 The transmission distance is not ideal.

- When there are straight-line communication obstacles, the communication distance will be correspondingly attenuated;
- Temperature, humidity, and co-channel interference will cause the communication packet loss rate to increase;
- The ground absorbs and reflects radio waves, and the test effect is poor when close to the ground;
- Seawater has a strong ability to absorb radio waves, so the seaside test results are poor;
- If there are metal objects near the antenna, or if it is placed in a metal case, the signal attenuation will be very serious;
- The power register setting is wrong and 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. The lower the voltage, the smaller the power generated;
- There is a poor match between the antenna and the module or there is a problem with the quality of the antenna itself.

9.2 Modules are easily damaged

- Please check the power supply to ensure that it is within the recommended power supply voltage. If it exceeds the maximum value, it will cause permanent damage to the module;
- Please check the stability of the power supply. The voltage cannot fluctuate greatly and frequently;

- Please ensure anti-static operation during installation and use, as high-frequency devices are sensitive to static electricity;
- Please ensure that the humidity during installation and use should not be too high, as some components are humidity-sensitive devices;
- If there are no special needs, it is not recommended to use it at too high or too low temperature.

9.3 Bit error rate is too high

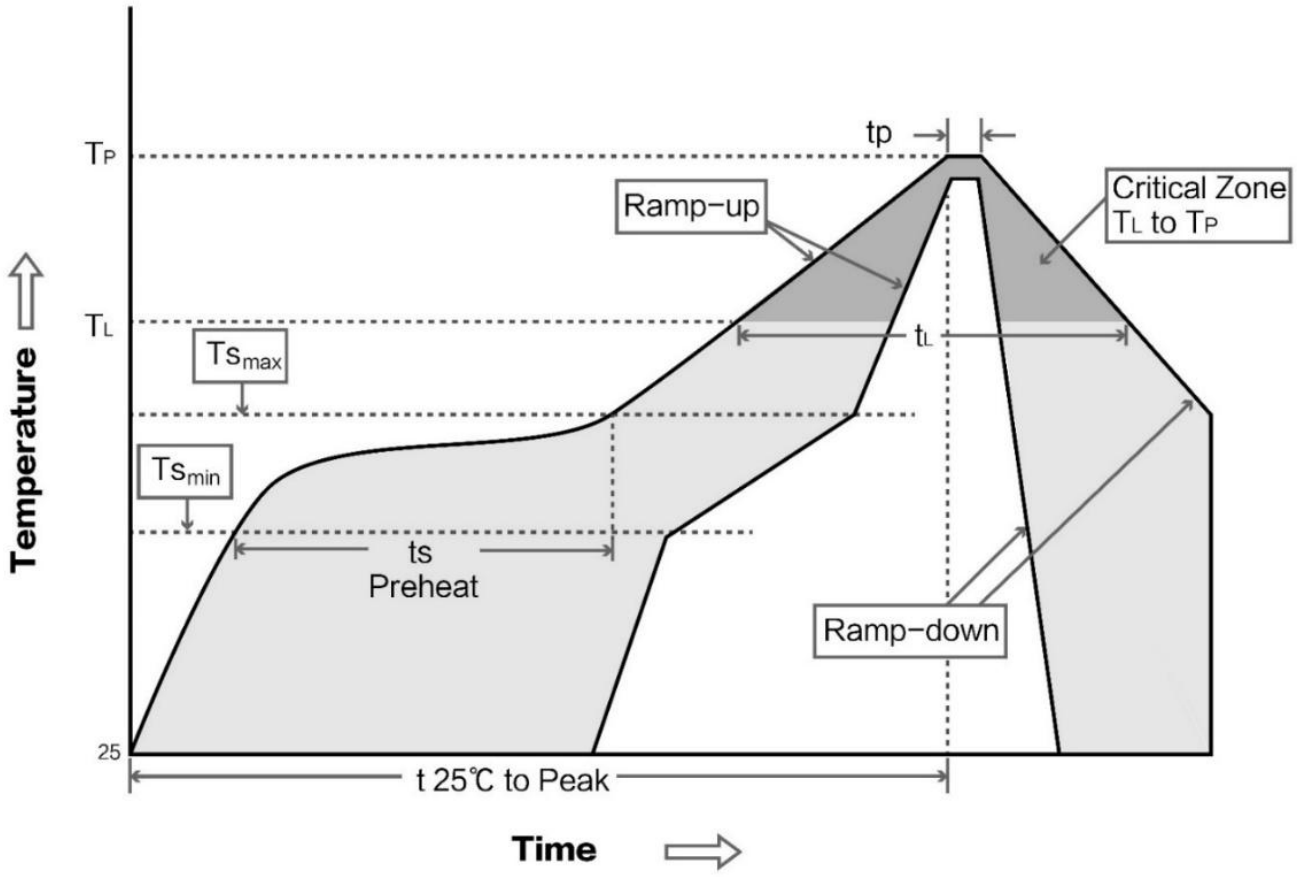
- If there is co-channel signal interference nearby, stay away from the interference source or modify the frequency or channel to avoid interference;
- The clock waveform on SPI is not standard. Check whether there is interference on the SPI line. The SPI bus line should not be too long;
- Unsatisfactory power supply may also cause garbled code, so be sure to ensure the reliability of the power supply;
- Poor quality or too long extension cords and feeders can also cause a high bit error rate.

Chapter 10 Welding Operation Guidance

10.1 Reflow soldering temperature

Profile Feature	Curve characteristics	Sn-Pb Assembly	Pb-Free Assembly
Solder Paste	solder paste	Sn63/Pb37	Sn96.5/Ag3/Cu0.5
Preheat Temperature min (Tsmin)	Minimum preheat temperature	100℃	150℃
Preheat temperature max (Tsmax)	Maximum preheating temperature	150℃	200℃
Preheat Time (Tsmin to Tsmax)(ts)	Preheat time	60-120 seconds	60-120 seconds
Average ramp-up rate(Tsmax to Tp)	average rate of rise	3℃/second max	3℃/second max
Liquidous Temperature (TL)	liquidus temperature	183℃	217℃
Time(tL)Maintained Above(TL)	time above liquidus	60-90 seconds	30-90 seconds
Peak temperature（Tp）	peak temperature	220-235℃	230-250℃
Aveage ramp-down rate（Tp to Tsmax）	average rate of decline	6℃/second max	6℃/second max
Time 25℃ to peak temperature	Time from 25℃ to peak temperature	6 minutes max	8 minutes max

10.2 Reflow soldering curve



Chapter 11 Related Models

Product number	Chip solution	carrier frequency Hz	Transmit power dBm	Test distance km	Package form	Product Size mm	Communication Interface
E22-400M22S	SX1268	433/470M	22	7	patch	14*20	SPI
E22-900M22S	SX1262	868/915M	22	7	patch	14*20	SPI
E22-400M30S	SX1268	433/470M	30	12	patch	24*38.5	SPI
E22-900M30S	SX1262	868/915M	30	12	patch	24*38.5	SPI
E22-230T22S	SX1262	230M	22	5	patch	16*26	TTL
E22-400T22S	SX1268	433/470M	22	5	patch	16*26	TTL
E22-900T22S	SX1262	868/915M	22	5	patch	16*26	TTL
E22-230T30S	SX1262	230M	30	10	patch	25*40.5	TTL
E22-400T30S	SX1268	433/470M	30	10	patch	25*40.5	TTL
E22-900T30S	SX1262	868/915M	30	10	patch	25*40.5	TTL

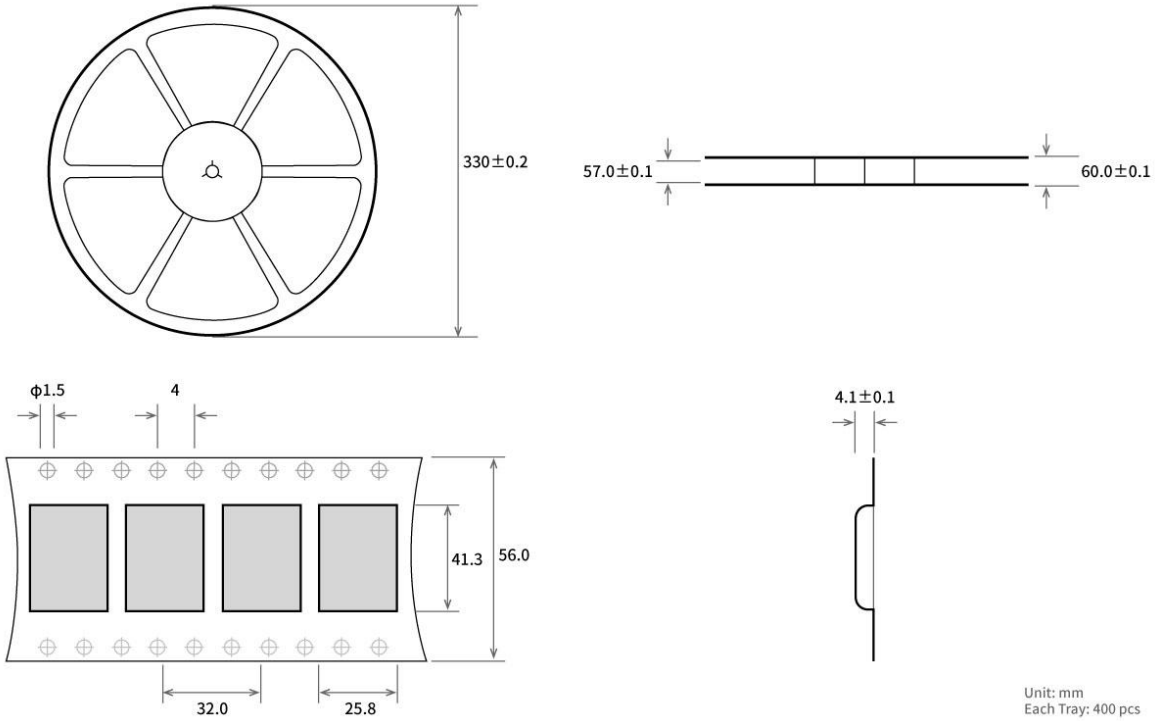
Chapter 12 Antenna Guide

12.1 Antenna recommendations

Antennas play an important role in the communication process. Often poor-quality antennas will have a great impact on the communication system. Therefore, our company recommends some antennas as antennas that support our wireless modules and have excellent performance and reasonable prices.

Product number	type	frequency band Hz	interface	Gain dBi	high mm	feeder cm	Features
TX433-NP-4310	flexible antenna	433M	welding	2.0	43.8*9.5	-	Built-in flexible, FPC soft antenna
TX433-JZ-5	glue stick antenna	433M	SMA-J	2.0	52	-	Ultra-short straight, omnidirectional antenna
TX433-JZG-6	glue stick antenna	433M	SMA-J	2.5	62	-	Ultra-short straight, omnidirectional antenna
TX433-JW-5	glue stick antenna	433M	SMA-J	2.0	50	-	Bend glue stick, omnidirectional antenna
TX433-JWG-7	glue stick antenna	433M	SMA-J	2.5	75	-	Bend glue stick, omnidirectional antenna
TX433-JK-11	glue stick antenna	433M	SMA-J	2.5	110	-	Bendable glue stick, omnidirectional antenna
TX433-JK-20	glue stick antenna	433M	SMA-J	3.0	210	-	Bendable glue stick, omnidirectional antenna
TX433-XPL-100	suction cup antenna	433M	SMA-J	3.5	185	100	Small suction cup antenna, cost-effective
TX433-XP-200	suction cup antenna	433M	SMA-J	4.0	190	200	Neutral suction cup antenna, low loss
TX433-XPH-300	suction cup antenna	433M	SMA-J	6.0	965	300	Large suction cup antenna, high gain
TX490-JZ-5	glue stick antenna	470/490M	SMA-J	2.0	50	-	Ultra-short straight, omnidirectional antenna
TX490-XPL-100	suction cup antenna	470/490M	SMA-J	3.5	120	100	Small suction cup antenna, cost-effective

Chapter 13 Batch Packaging Methods



Revise history

Version	Revision date	Revision Notes	Maintenance man
1.0	2023-09-15	Manual release	Ning
1.1	2024-03-05	Add and modify content	Ning
1.2	2024-04-01	Modify the module display orientation	Ning

About us

Technical support: support@cdebyte.com

Documents and RF Setting download link: https://www.cdebyte.com

Thank you for using Ebyte products! Please contact us with any questions or suggestions: info@cdebyte.com

---

Web: www.ebyte.com

Address: B5 Mould Park, 199# Xiqu Ave,Chengdu, Sichuan, China