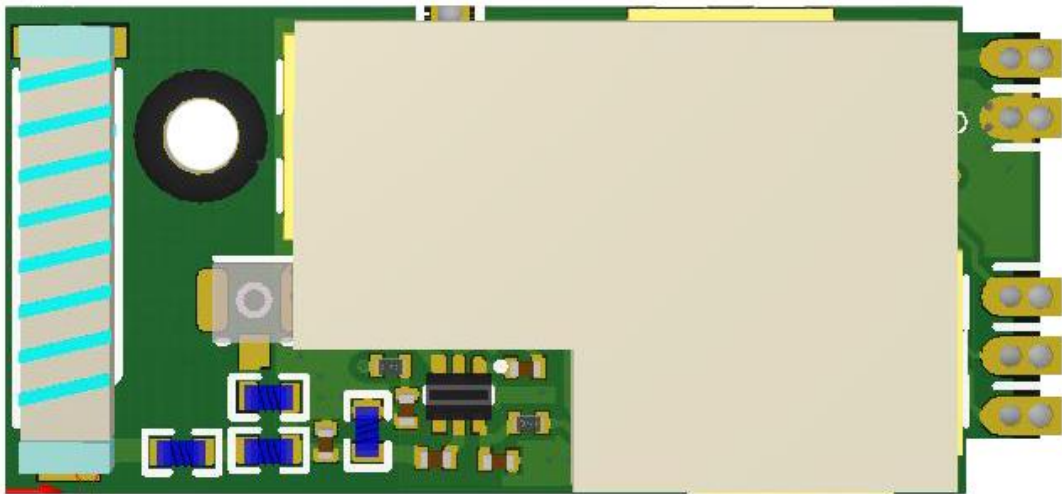


WENSHING® **TRW-LoRa20UARTC** RF MODULE

410~525MHz LoRa Transceiver Module



Version History

| Version | Date | Changes |
|---------|-------------------|-------------------------|
| V1.0 | November 07, 2017 | 1 st Edition |
| V1.1 | January 24, 2018 | 2 st Edition |

Features

This module includes MCU, high sensitivity, low consumption and anti-interference. Able to reach 1.5km (Straight line, open area). Use high efficiency cyclic error-detection and correction code to greatly enhance sensitivity and anti-interference. Transmit power 100mW, low consumption and wake on wireless.

LoRa provides ultra-long range while maintaining low current consumption wireless platform is the prevailing technology choice for building IoT networks worldwide. LoRa spread spectrum technology brings farther communication distance and covers community with thousands of people. Suit for meter-reading, smart home and integrated burglarproof alarm equipment. LoRa technology is being integrated in to cars, street lights, manufacturing equipment, home appliances, wearable devices – anything, really. Lora technology is making our world a Smart Planet. The patented modulation technique the device can achieve sensitivity of over -140 dBm. The device also support high performance (G) FSK for WMBus, FCC Part 90, 15.4g, and other legacy modes.

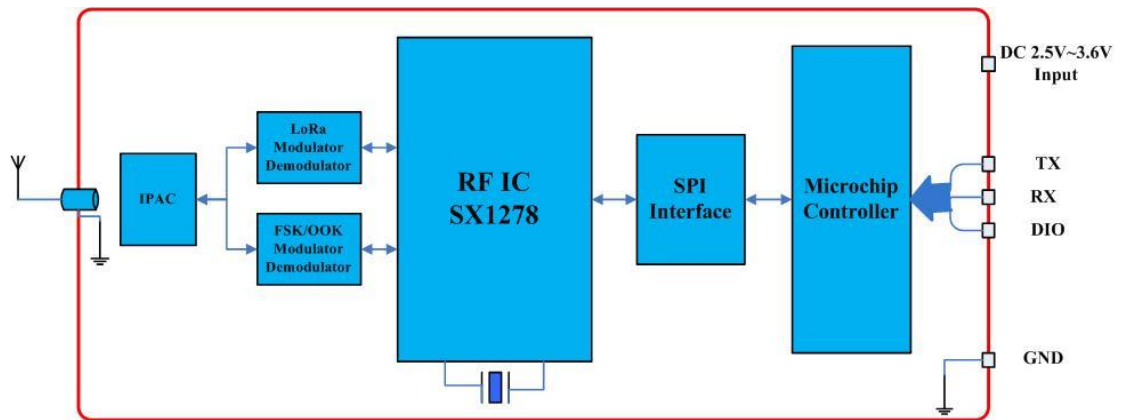
Applications

- Security surveillance system
- Data collecting from volt meter, water meter, and slot machine.
- Wireless modem
- Mudslide detecting site
- Weather detecting site

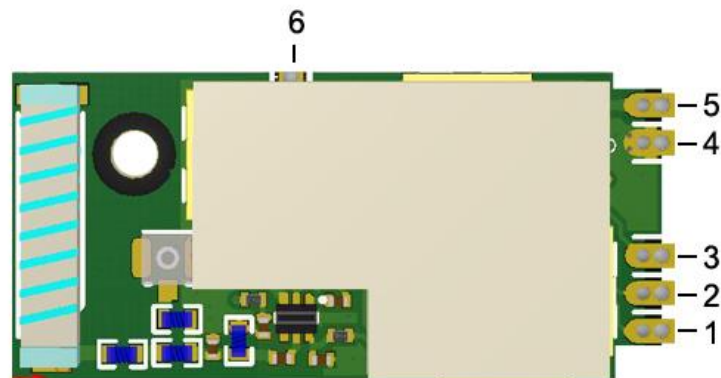
Electrical Specification

| Parameter | Description | Min | Typ | Max | Unit |
|-----------|--------------------------------------|-----|-----|-------|------|
| VDD | Supply Voltage Range | 2.5 | 3.3 | 3.6 V | V |
| FREQ | Operating frequency | 410 | | 525 | MHz |
| IDC_ST | Standby mode power consumption | | 5 | | mA |
| IDC_TX | Transmit power consumption (7~20dBm) | 24 | | 124 | mA |
| IDC_RX | Receive power consumption | | 15 | | mA |

Internal Block Diagram



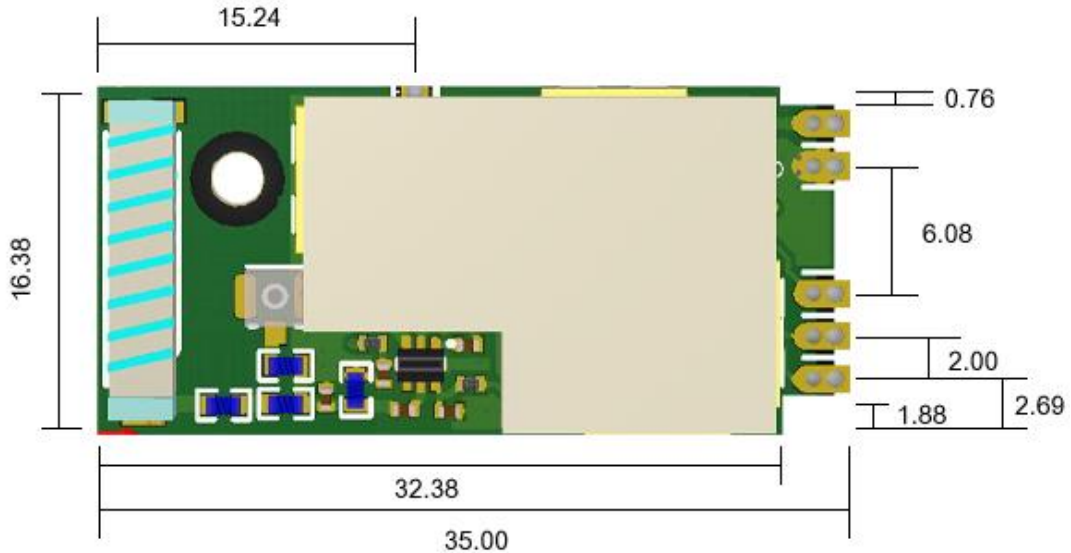
Pin Assignment



| Pin | Name | I/O | Description |
|-----|------|-------|----------------------------|
| 1 | TX | O | UART Interface TX Data Out |
| 2 | RX | I | UART Interface RX Data In |
| 3 | DIO | I/O | TBD |
| 4 | GND | GND | Ground |
| 5 | VDD | Power | Power Supply 2.5~3.6 |
| 6 | GND | GND | Ground |

Dimension

Unit : mm



Command communication mode

● Enter set-up mode

Transmit value = 0xFF FF FF 55 EE, total 5 bytes

Return value = 0xFF FF FF 55 EE, total 5 bytes

- ✧ It is only allowed to read or modify all parameters in set-up mode. Set-up mode can't be entered after being powered up for 30 seconds.

● Leave set-up mode

Transmit value = 0xFF FF FF 55 CC

Return value = none, if UART doesn't receive any data within 30 seconds in set-up mode, it will automatically leave set-up mode.

- ✧ It is only allowed to transmit or receive data after leaving set-up mode.

● Read product name and version

Transmit value = 0xFF FF FF 55 AA BB FD

Return value (Character) = LR_434_V003

- ✧ There are total 11 bytes of data, first 6 characters stand for product name, last 4 characters stand for the firmware name. (To show the difference within versions)

64+53=-11dBm). Display range: -192~63dBm. Bit7 set 0 as RSSI closing hint.

| Value | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|----------------|----|-----|-----|-----|------|------|------|------|
| LoRa Rate(bps) | 75 | 150 | 300 | 600 | 1200 | 2400 | 4800 | 9600 |

- ✧ 8th~10th byte: Working frequency, calculation method: MHz*1000=KHz, then convert to hexadecimal system.
Example: When it is in 434MHz working frequency, $434*1000=434000=0x06\ 9F\ 50$ fills 8th byte with 06, 9F in 9th, 50 in 10th; when it is in 432.5MHz working frequency, $432.5*1000=432500=0x06\ 99\ 74f$ fills 06 in 8th byte, 99 in 9th, 74 in 10th.
- ✧ 11th byte:
Bit0~Bit2: Output power, set-up range 0~7

| Output Power | | |
|--------------|-----------|-------------------|
| dBm | Set Value | Hex (Bit0 ~ Bit2) |
| 5 | 0 | 000 |
| 8 | 1 | 001 |
| 10 | 2 | 010 |
| 12 | 3 | 011 |
| 14 | 4 | 100 |
| 16 | 5 | 101 |
| 18 | 6 | 110 |
| 20 | 7 | 111 |

Bit3 ~ Bit5 : Invalid characters, fixed value 000

Bit6 ~ Bit7 : Device works in 4 modes as below:

- **Mode 1 (Long-figure data mode: Set-up value 00)**

In this mode, all devices with the same GID value can receive data. This mode is targeted for data over 127 bytes.

- **Mode 2 (ID data mode 1: Set-up value: 01)**

In this mode, all devices with the same GID value can transmit data to specified SID, to achieve one-to-multiple-transmission, but single data size should not exceed 127 bytes.

Transmit mode: First byte of the data is the SID of receiving end, from second byte on are the data.

Example: device A's SID value is 55, device B is 88 and both have the same GID. In mode 2, if device A transmit a 5 bytes data 0x1324567890 to B, then B will receive a 6 bytes data 0x551234567890. The first byte stands for the SID of the transmitting end.

● **Mode 3 (ID data mode 2: Set-up value 10)**

In this mode, it can transmit data to device with specified GID and SID, to achieve one-to-multiple-transmission, single data size including specified GID and SID should not exceed 127 bytes.

Transmission mode: The data should be transmitted through the order of 13th byte to 32th byte.

Example: device A shows GID= AAAA, SID=55, device B shows GID=BBBB, SID=88, device C shows GID=CCCC, SID=99.

If device A transmits a 5 bytes data 0x1234567890 to B, the data sent is a0x04FFBBBB881234567890, total 10 bytes, then B receives a 5 bytes data 0x1234567890.

If device A transmits data 0x1234567890, total 5 bytes to device C Through device B, then the transmitted data will be 0x08FFBBBB88FFCCCC991234567890, total 14 bytes. And B won't receive any data, while C receive 0x1324567890, total 5 bytes of data.

● **Mode4 (Saved ID data mode: set-up value 11)**

In this mode, route of GID and SID can be saved beforehand. When transmitting data, it will automatically use the route saved to transfer data for maximum 14 times. Single data size including pre-saved GID and SID route should not exceed 127 bytes.

- ◇ 12th byte: Invalid character, fixed value 0x00
- ◇ 13th to 32th byte : Pre-saved route, only activated in Mode 4 (Saved ID data mode).
- ◇ 13th byte: Indicates the number of valid data within 14th to 32th byte.
- ◇ 14th to 32th byte:

◆ Example1: 04 FF 12 34 55 11 22 33 44 ~ 00

The 13th byte shows 4 bytes of valid data: FF 12 34 55.

FF 12 34 55 means GID=1234, SID=55. When the device receives data through UR, it will automatically transmit it to the device with GID= 1234, SID=55.

◆ Example 2: 05 FF 12 34 55 11 22 33 44 ~ 00

The 13th byte shows 5 bytes of valid data: FF 12 34 55 11.
 FF 12 34 55 11 means GID=1234, SID=55 and 11. When the device receives data through UR, it will automatically transmit it to the device with GID= 1234, SID=55, then transfer it to the device with GID= 1234, SID=11.

◆ Example 3: 06 FF 12 34 55 11 22 33 44 ~ 00

The 13th byte shows 6 bytes of valid data: FF 12 34 55 11 22.

FF 12 34 55 11 22 means GID=1234, SID=55, 11 and 22. When this device receives data through UR, it will automatically transmit it to the equipment with GID= 1234, SID=55, then transfer to the device with GID= 1234, SID=11, then the one with GID= 1234, SID=22.

◆ Example 4: 08 FF 12 34 55 FF 45 67 88 44 ~ 00

The 13th byte shows 8 bytes of valid data: FF 12 34 55 FF 45 67 88.

FF 12 34 55 FF 45 67 88 means GID=1234, SID=55 and GID=4567, SID=88. When the device receives data through UR, it will automatically transmit it to the device with GID= 1234, SID=55, then transfer to the device with GID=4567、SID=88.

AT Command mode (Must add newline character at the end of every command)

| Send | Return | Function |
|-----------------|--|----------------------------------|
| AT+RSSI: ? | +RSSI: -101dBm | Read RSSI |
| AT+PacketRSSI ? | +PacketRSSI: -101dBm +PacketRSSI: No data packet received | Read the RSSI of the last packet |