SRF 2400M

2.4 GHz Wireless Networked UART Module

User's Guide (Rev. 1.01)

Tel: +82 (0)2 470-0491   Fax: +82 (0)2 474-0659

address : *soft*DSP Co., Ltd. Jungil Bldg 203, 552-1 Sungnae-dong, Kangdong-ku, Seoul, Korea

Home page: http://www.softdsp.com

CONTENTS

1 INTRODUCTION 3
1.1 Overview 3
1.2 Features 3
1.3 Applications 3
1.4 Block diagram 4
1.5 System Interface 6
1.6 Pin Description 7
1.7 Dimensions 9
1.8 Specifications 9

2 UART COMMANDS 12
2.1 UART specifications 12
2.2 Control Code 12
2.3 Commands 13
2.4 Packet Structure 13
2.5 Command description 14
2.5.1 Communication Configure (Command type: 0x10) 14
2.5.2 Network Configure (Command type: 0x20) 15
2.5.3 Send data (Command type: 0x25) 19
2.5.4 Receive data (Command type: 0x26) 20
2.5.5 Sleep (Command type: 0x21) 21
2.5.6 Wakeup (Command type: 0x22) 23
2.6 C code example 24

3 MAC protocol features 26
3.1 Automatic Network Configuration 26
3.2 Multi Channel/Multi-Hop for expanding communication area 28
3.3 Reliable Data Transmission 29
3.4 TX/RX buffering 30

1 INTRODUCTION

1.1 Overview

SRF 2400M is a 20 dBm Direct Sequence Spread Spectrum (DSSS) wireless module that allows bi-directional wireless communication between systems using a standard UART protocol.

SRF 2400M has a ready-to-use, highly advanced Medium Access Control (MAC) protocol which includes novel features such as automatic network configuration, multi-hop message packet delivery and error detection/correction with data retransmission to ensure reliable RF link in the 2.4 GHz ISM band. Unlike most other wireless modules in the market that does not have an integrated MAC protocol, the user can send and receive data from any of the modules in arbitrary network topologies without any experience or knowledge in the wireless communication field with just simple UART commands.

Boasting excellent sensitivity, extended range using multi-hop networking and highly robust communication protocol against interference, SRF 2400M is perfect for applications such as Supervisory Control and Data Acquisition (SCADA), remote meter reading, home automation, security, instrument monitoring, point of sale systems (POS), and countless other applications.

1.2 Features

- 2.4 GHz ISM band, DSSS communication. No licensing required
- Automatic network configuration
- Multiple channel
- Ready-to-use advanced MAC protocol
- Up to 1300 ft transmission range depending on line-of-sight
- Multi-hop self-organized network
- Up to 50 Kbytes of data stream TX/RX using a single UART command.

1.3 Applications

- SCADA
- Remote monitoring
- Industrial automation and Controls

- Instrument monitoring
- Point of Sale Systems (POS)
- Distributed perimeter Security
- Physical security
- Networking of personnel and physical assets and Machinery health and status monitoring.
- Home automation
- Distributed surveillance
- Telemetry

1.4 Block diagram

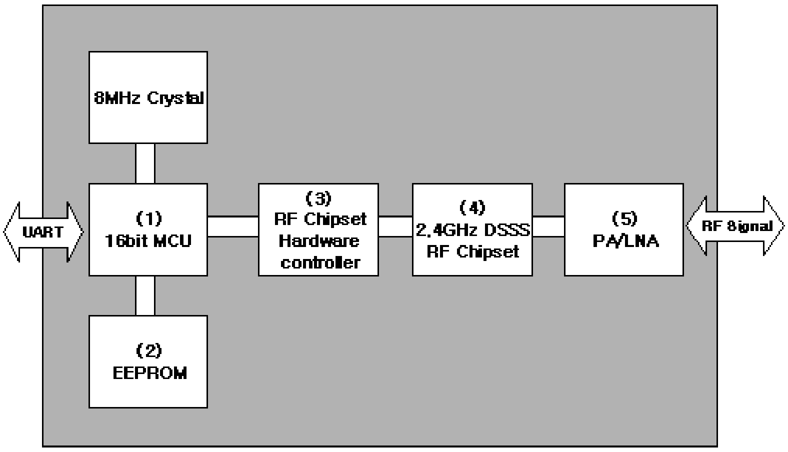


Figure 1 SRF 2400M block diagram

Figure 1 shows the block diagram of SRF 2400M. SRF 2400M is composed of 5 main blocks; 16 bit MCU, EEPROM, RF chipset hardware controller, 2.4 GHz DSSS RF chipset and PA/LNA.

(1) 16 bit MCU block

16-bit MCU block controls the whole system. This block is in charge of UART and wireless communication. It handles all the control signals and data processing for wireless communication based on its Medium Access Control (MAC) protocol

EEPROM block is connected to the MCU and stores data from the MCU. Its main purpose is to buffer long data streams between UART and RF, for more robust communication.

(2) Memory block

Memory block is composed of EEPROM memory. EEPROM is connected to MCU so that MCU can read and write stream data. By buffering data stream, SRF 2400M can communicate with less error. The EEPROM can buffer up to 50 Kbytes and thus, up to 50 Kbytes of long data stream can be sent and received.

(3) RF controller block

RF chipset HW controller operates with the 2.4 GHz DSSS RF chipset and takes charge of low-level control operations such as start of packet search, conversion of fast serial input to 8 bit words, buffering of incoming and outgoing data, so that the MCU is left with more room to work with for high-level operations. It gives and receives commands from the MCU via parallel interface bus.

(4) 2.4 GHz DSSS RF IC block

2.4 GHz DSSS RF chipset is the physical layer element (PHY), allowing wireless transmission and reception of bits from one end to the other. The communication link between the RF chipset is a Direct Sequence Spread Spectrum (DSSS) pulse pipe. This block sends/receives serial data to the RF chipset HW controller block and also sends Received Signal Strength Indicator (RSSI) signal to the MCU so that the RF link strength of the received RF signal can be measured.

(5) PA/LNA block

The PA/LNA block amplifies both TX/RX RF signals for the purpose of extending the communication range. A Power Amplifier (PA) is in the TX path and a Low Noise Amplifier (LNA) in the RX path to amplify both the sending and receiving RF signals. The PA boosts the output power to 20 dBm and the LNA improves the RX sensitivity to -90 dBm. The TX/RX path in the PA/LNA block is switched by the RF chipset HW controller for bi-directional operation.

1.5 System Interface

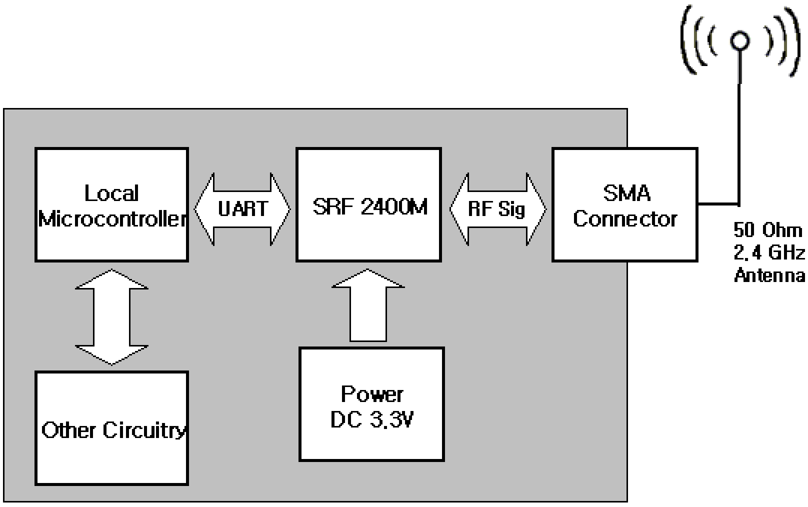


Figure 2 Typical interface of SRF 2400m with local system

Figure 2 shows a typical interface of SRF 2400M to the local system. The pins that are needed for basic operation are the power pins (+3.3 V, GND), UART TX/RX pin, and the RF signal pin. SRF 2400M interfaces with a local Microcontroller via UART and an external 50 ohm 2.4 GHz antenna is connected to the RF_out pin via SMA connector. With this simple interface, your local system can have an easy access to wireless communication.

1.6 Pin Description

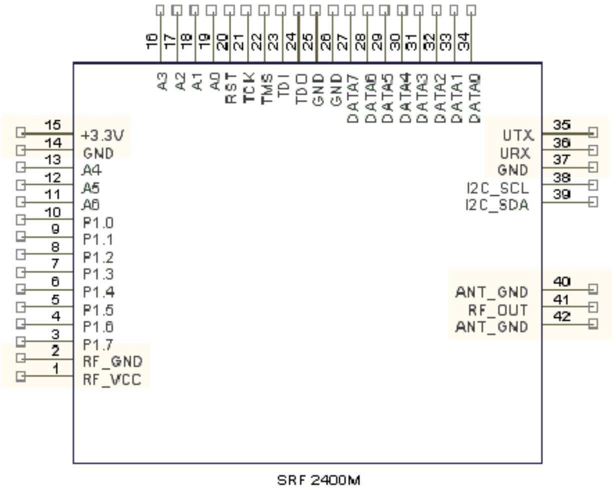


Figure 3 Pin Configuration of the SRF 2400M

Figure 3 shows the pin configuration of the SRF 2400M. SRF 2400M has a total of 42 pins and only 10 of these pins (highlighted in figure 3) can be connected for a complete wireless networked UART operation.

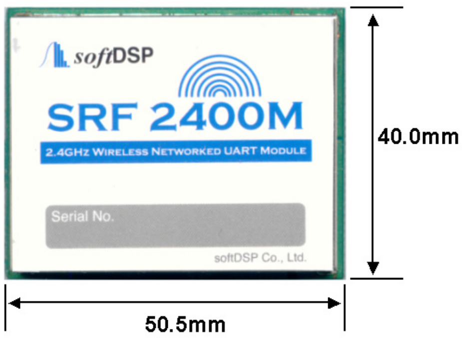
A full pin description of SRF 2400M is shown in figure 4.

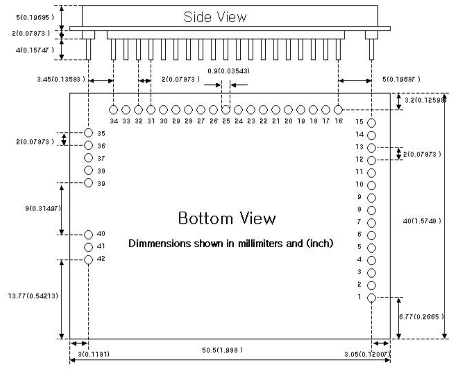
Pin No.	Pin Name	I/O Type	Description
1	**RF VCC**	**I**	**VCC (+3.3V) for RF circuitry**
2	**RF GND**	**-**	**GND for RF circuitry**
3	P1.7	I/O	NC
4	P1.6	I/O	NC
5	P1.5	I/O	NC
6	P1.4	I/O	NC
7	P1.3	I/O	NC
8	P1.2	I/O	NC
9	P1.1	I/O	NC
10	P1.0	I/O	NC
11	A6	I	NC
12	A5	I	NC
13	A4	I	NC

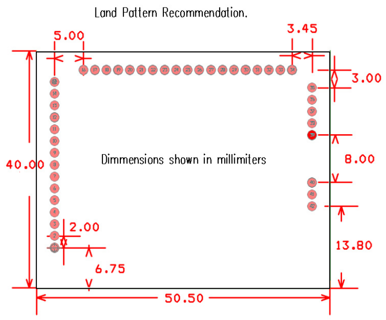
14	**GND**	**-**	**GND**
15	**+3.3V**	**I**	**+3.3V**
16	A3	I	NC
17	A2	I	NC
18	A1	I	NC
19	A0	I	NC
20	RST	I	Firmware download pin (Not used)
21	TCK	I	Firmware download pin (Not used)
22	TMS	I	Firmware download pin (Not used)
23	TDI	I	Firmware download pin (Not used)
24	TDO	O	Firmware download pin (Not used)
25	GND	-	GND
26	GND	-	GND
27	DATA7	I/O	NC
28	DATA6	I/O	NC
29	DATA5	I/O	NC
30	DATA4	I/O	NC
31	DATA3	I/O	NC
32	DATA2	I/O	NC
33	DATA1	I/O	NC
34	DATA0	I/O	NC
35	**UTX**	**O**	**UART transmit data out**
36	**URX**	**I**	**UART receive data in**
37	**GND**	**-**	**GND**
38	I2C SCL	I/O	I2C serial clock. Used to connect external EEPROM for extended memory.
39	I2C SDA	I/O	I2C serial data. Used to connect external EEPROM for extended memory.
40	**ANT GND**	**-**	**Antenna GND**
41	**RF OUT**	**O**	**50ohm RF signal output**
42	**ANT GND**	**-**	**Antenna GND**

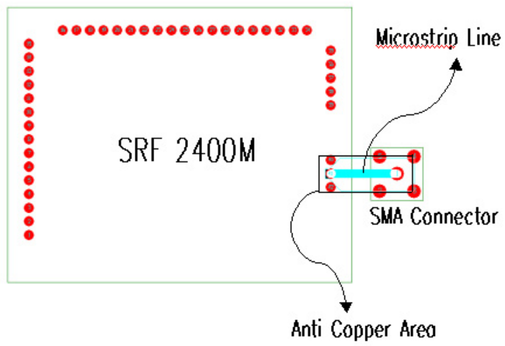
Figure 4 Pin Description of the SRF 2400M

1.7 Dimensions









1.8 Specifications

Hardware specification

CPU :	16-bit RISC architecture, 8 MIPS processing speed	
Data storage memory :	50 Kbytes	
Size :	1.99" x 1.58"" (L x W)	
	5.05 cm x 4.00 cm	
Operating voltage:	3.3VDC	
Operating current:	Transmitting / Receiving	100mA / 50mA
	Standby	50mA
	Power-down	100uA

UART specification

Max. Baud rate :	56,000 bps (fixed)
Interface :	Standard 3-line interface
Format :	1 start bit, 8 bit data,1 stop bit, no parity. (fixed)

UART interface: Unit: V

		MIN	TYP	MAX
Input (URX)				
VIL	Low-level input voltage	0		0.6
VIH	High-level input voltage	2.64		3.3
Output (UTX)				
VOH	High-level output voltage	3.05		3.3
VOL	Low-level output voltage	0		0.25

Functions

Network configuration :	Automatically detects nodes and the configures the network
Long data stream TX/RX :	Sends/Receives stream of data 1 ~ 50 Kbytes long with a single UART command.
Sleep/Wakeup :	Lowest power consumption is achieved using sleep function when not used

Configurable communication protocol

Group Address :	1 ~ 255 (Modules can only communicate with other modules with the same group address)
Local Address :	1 ~ 250 (1: Basestation)
Max. retransmission number :	10 ~ 255
Max. Multi-Hop number :	1 ~ 10
Inter-module RF link strength :	High, Normal, Low. (Reliable-link search criteria for automatic network configuration)

RF communication features

Communication Link :	2.4 GHz DSSS (Direct Sequence Spread Spectrum)
Bit rate :	1 Mbps
Output power :	+ 20 dBm
RX sensitivity :	- 90 dBm
RF_out output impedance :	50 ohm
Range :	1300 feet (400 meters) line of sight outdoor
Communication mode :	Bi-directional, half-duplex.

MAC protocol

Advanced CSMA :	Detects and avoids RF signals from both other SRF 2400M modules and 2.4 GHz ISM band devices such as Bluetooth, WLAN. Multiple channel communication, system co-existence supported.
Automatic retransmissions :	Handshaking with automatic retransmissions in case of error.
Error Detection/Correction :	Detection: 16 bit CRC Correction: Hamming Code
RTS/CTS :	To avoid the "Hidden Node " problem, RTS/CTS protocol is supported like 802.11 WLAN protocol.
Long data stream store & forward :	Automatically stores long data stream (more than 25 bytes) received from UART to local EEPROM before forwarding it to RF. When received by the destination module, the data is stored in EEPROM before forwarding it to UART.
TX/RX packet buffering :	Up to 5 TX/RX packets can be buffered for more robust communication, especially in congested network traffic.

2 UART COMMANDS

The serial interface serves as a direct connection from the SRF 2400M to the microcontroller. And the RS232 driver serves as a direct connection from the SRF 2400M to the computer.

The SRF2400M responds to Commands and send a data record. The command and data systax is defined below.

2.1 UART specifications

Communication type	UART
Baud rate	56,000 bps
Frame Format	Start Bit : 1
	Data Bit : 8
	Parity Bit : None
	Stop Bit :1

2.2 Control Code

Symbol	Length & Value	Description
STX	1 Byte : 0X02	Start Of Text
ETX	1 Byte : 0X03	End Of Text
LEN	2 Bytes (Little Endian)	Length of Data (Lower Byte is stored in lower address)
CMD	1 Byte	Command

CRC	2 Bytes (Little Endian)	16bit Cyclic Redundancy Check (Lower Byte is stored in lower address)

2.3 Commands

COMMAND	Description
0x10	Communication Configure. Sets variables for basic communication. Ex) Group address, Local address.
0x20	Network configure. Options can be changed.
0x25	Send data. Sends data to other modules. Basestation can send data to any of the modules in the network. Other modules can only send data to basestation. Up to 50kbytes of data stream can be sent.

2.4 Packet Structure

STX	LEN	CMD	Data	CRC	ETX
1byte	2 Bytes	1 byte	0~50 Kbytes	2 bytes	1 byte

CRC range

2.5 Command description

2.5.1 Communication Configure (Command type: 0x10)

2.5.1.1 Command Format

LEN:	0x0A (10)
CMD:	0x10
Datat0:	Group address
	1~255 (Default: 0x70)
Data1:	Local address
	1~250 (Default: 0x01)
	1: Basestation
	2~250: node
Data2:	Max. Retransmission no.
	10 ~ 255 (Default: 20)
Data3 ~ data9:	Reserved
	Don't Care

2.5.1.2 Explanation

This command is for basic configuration for operation. It includes configuring group address, local address and maximum retransmission number.

The group address distinguishes one network from the other. The same group address has to be assigned for all the modules in the same network. Different networks can coexist when different group addresses are assigned for each module in the same network. A value of 1~255, with the default value of 112(0x70) is assigned for group address.

The local address distinguishes one module from the other. In the same network, every module has to be assigned a different local address, and there can only be one basestation for every network. A value of 1~250 can be assigned as the local address, and 1 is for basestation and the others for nodes. Default value is 1.

Max retransmission number assigns the number of additional retransmissions when a packet delivery is unsuccessful. A value of 10~255 can be assigned with a default value of 20.

SRF 2400M saves all the communication configure information in its local EEPROM, so that after power-up it restores all the information without losing it.

Note that CRC value is NOT used, so that any value can be written in the CRC field.

2.5.1.3 Example

Group Address = 112

Local address = 1

Max. Retransmission no. = 20

STX	LEN(L)	LEN(H)	CMD	Data0	Data1	Data2	Data3	…	Data9	CRC(L)	CRC(H)	ETX
0x02	0x0A	0x00	0x10	0x70	0x01	0x14	D.C.	…	D.C.	D.C.	D.C.	0x03

2.5.2 Network Configure (Command type: 0x20)

2.5.2.1 Command Format

LEN:	0x0A (10)
CMD:	0x20
Datat0:	Maximum hop count
	1 ~ 10 (Default: 10)
Data1:	Link power
	0: Default, 1: High, 2: Highest (Default 0)
Data2:	Flood number
	1 ~ 5 (Default: 5)
Data3 ~ data9:	Reserved
	Don't Care

2.5.2.2 Explanation

Configures the network from the basestation, assigning a parent module for each module in the network. This command can only be sent to the basestation. When this command is executed in the basestation, then the basestation "floods" the network by sending out a broadcast message that is relayed to all the modules in the network via each module. Each module in the network listens to the flooding messages sent to them and decides which sending module it would decide as its parent module. The decision-making is done depending on the RF power level measured when the message is received and also statistical data processing, so that the parent module most often received has the top priority for the final decision. Maximum hop count, link power and flood number variables can be assigned for more flexible network configuration operation.

(1) Maximum Hopping Number

Maximum hop count decides the maximum hop count number allowed in the configured network. For instance, when this value is set to 1, then only 1-to-N network configuration is allowed with no multi hop. A value of 1 ~ 10 can be assigned, with 10 being the default value.

(2) Link power

Link power assigns the threshold RF power level for deciding the parent module. Each module ignores the flooding packets that have RF power levels lower than the configured threshold value. So the RF power level has to be at least higher than the threshold level for the module to consider deciding it as its parent address. Assigning a high value to the link power variable will result in stronger RF link between the parent and the child module, but greater multi-hops may result since the modules would choose a parent module that is closer to them, with a higher RF link power. Also, it is more likely to have "undetected" modules with a stronger RF threshold level, since shorter distances between the modules is allowed to satisfy the higher threshold level. A value of 0 ~ 2 can be assigned, with 0 being the default value.

(3) Flood number

Flood number decides how many times the basestation floods the network before each module decides its parent module. Each time the network is flooded, each module stores the received flooding message information. After the maximum flooding number, each module decides its parent address based on statistics. Higher flooding number will result in more reliable parent module selection, but longer time would be required to configure the network. It takes approximately 500 ms for every flooding number. So it would take around 2.5 seconds for 5 flooding number. A value of 1~5 can be assigned for the flood number, with 5 being the default value.

Note that CRC value is NOT used, so that any value can be written in the CRC field.

2.5.2.3 Example

Maximum hop count = 10

Link power = 0

Flood number = 5

STX	LEN(L)	LEN(H)	CMD	Data0	Data1	Data2	Data3	...	Data9	CRC(L)	CRC(H)	ETX
0x02	0x0A	0x00	0x20	0x0A	0x00	0x05	D.C.	...	D.C.	D.C.	D.C.	0x03

2.5.2.4 Network Configuration Example

After around 0.5x(Flood number) + 1 seconds, the configured network information data stream is sent from the basestation to the local microcontroller connected to the basestation via UART. The local microcontroller connected to UART either receives this data stream for reference or ignore it. If the system receives the data stream, it gains information such as how many modules are detected, what their local addresses are, and the topology of the network.

Note that CRC value is NOT used, so that any value can be written in the CRC field.

The data stream format for configured network information is as follows;

Dummy:	0xff (2 bytes, this is added for safer synchronization)
LEN:	0xFB (251)
CMD:	0x26
Data0:	Source address
	0x01 (Basestation)
Data1 ~ D250:	Configured network information

Each array index is mapped directly to the local address number and the value stored in the array is the parent address of that address. '0' means that there are no parent address assigned, and that particular module is not detected in the network.

For example, there is a network as shown in figure 5.

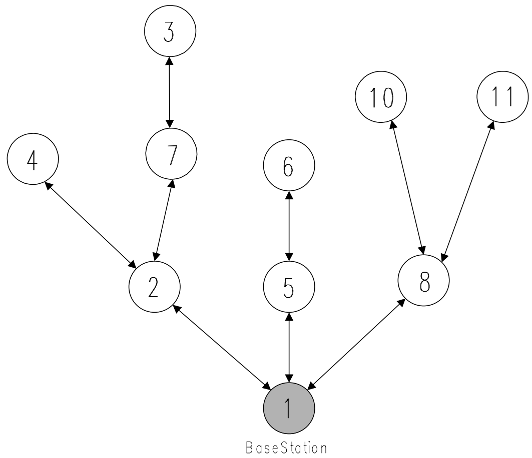


Figure 5 Configured network example

For the configured network in figure 5, the parent addresses of the configured modules are as follows;

Node Address	1	2	3	4	5	6	7	8	9	10	11
Parent Address	xx	1	7	2	1	5	2	1	0	8	8

In this case, the data stream sent by the basestation is as follows;

Dummy	Dummy	STX	LEN(L)	LEN(H)	CMD	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7
0xff	0xff	0x02	0xFB	0x00	0x26	0x01	0x00	0x01	0x07	0x02	0x01	0x05	0x02

Data8	Data9	Data10	Data11	Data12	...	Data249	Data250	CRC(L)	CRC(H)	ETX
0x01	0x00	0x08	0x08	0x00	...	0x00	0x00	D.C.	D.C.	0x03

2.5.3 Send data (Command type: 0x25)

2.5.3.1 Command Format

LEN:	Depends on the number of data field (1 byte ~ 50 Kbytes)
CMD:	0x25
Data0:	Destination address 1 ~ 250 If the module is not the basestation, then SRF 2400M forcibly sets the destination address to 1
Data1 ~ Data(N-1):	Arbitrary data set by the user

2.5.3.2 Explanation

This command sends data stream with length from 1byte to 50 Kbytes to one of the modules in the network. All the modules in the network can only send data to the basestation and the basestation can send to any of the modules in the network.

Data0 is the destination address, and when the module is not the basestation, then the destination address can only be 1. If by any chance, the module is not the basestation and the destination address was set to a value different from 1, then the hardware forcefully changes this value to 1. Note that a data stream with destination address that is not detected in the network will not be sent.

Data1 ~ Data(n-1), the user is free to put in any values with length from 1byte to 50kbytes.

Note that for data streams less than 25bytes, no EEPROM store & forward mode is used and only one RF packet is used for transmission. For data streams more than 25bytes, EEPROM store & forward mode is used and more than one RF packet is used for transmission.

When sending data streams of more than 25 bytes and when the EEPROM store and forward mode is used, it is advisable to make some kind of handshaking protocol between the sending and the destination module using the data send and receive commands. This would enable the sending module to know when the data stream sent is received by the destination module so that it would know the right timing to send the next data stream if required. Also, it is advisable that the basestaion act as master and the other modules act as slave, so that the modules send data only in response to a data send command from the basestation. This would avoid the situation where data streams from two sources are stored in the same EEPROM and corrupt the data.

For short data stream less or equal to 25 bytes, the timing for the next send command is not so critical since the sending operation is faster than the UART operation, and thus it could be sent at any time. Also, it is not obligatory to work the basestation as master and the other modules as slave.

Note that CRC value is only for the receiving system at the destination module, so the hardware does not reject the packet even if the CRC value is incorrect.

2.5.3.3 Example

The data stream send format is as follows;

Destination address = 5

Data1 ~ Data5 = 0x02, 0x05, 0x74, 0x23, 0x72

STX	LEN(L)	LEN(H)	CMD	Data0	Data1	Data2	Data3	Data4	Data5	CRC(L)	CRC(H)	ETX
0x02	0x06	0x00	0x25	0x05	0x02	0x05	0x74	0x23	0x72	CRC(L)	CRC(H)	0x03

2.5.4 Receive data (Command type: 0x26)

2.5.4.1 Command Format

Dummy:	0xff (2 bytes, this is added for safer synchronization)
LEN:	Depends on the number of data field (1 byte ~ 50 Kbytes)
CMD:	0x26
Data0:	Source address 1~250 If the module is not the basestation, then the source address is always 1.
Data1 ~ Data(N-1):	Arbitrary data set by the user

2.5.4.2 Explanation

This is not a command, but a corresponding data stream received by the destination address when the same data stream is sent by the sender address using command type 0x25. When a module sends a data stream with command type 0x25, the only difference with the data stream received by the destination address is that the command type is changed from 0x25 to 0x26 and the destination address field is changed to source address value.

Like the data send command, data stream with length from 1 byte to 50 Kbytes can be received. Also, the basestation can receive data streams from any of the modules in the network and the any module in the network can receive data from the basestation.

Data0 is the source address, informing the destination module of the source module that sent the data stream.

2.5.4.3 Example

The data stream receive format is as follows;

Source address = 3

Data1 ~ Data5 = 0x02, 0x05, 0x74, 0x23, 0x72

Dummy	Dummy	STX	LEN(L)	LEN(H)	CMD	Data0	Data1	Data2	Data3	Data4	Data5	CRC(L)	CRC(H)	ETX
0xff	0xff	0x02	0x06	0x00	0x26	0x03	0x02	0x05	0x74	0x23	0x72	CRC(L)	CRC(H)	0x03

2.5.5 Sleep (Command type: 0x21)

2.5.5.1 Command Format

LEN:	0x0A(10)
CMD:	0x21
Data0:	Sleep period : 0 ~ 10
	0: 100ms

	1: 200ms
	2: 500ms
	3: 1s
	4: 2s
	5: 5s
	6: 10s
	Others: 500ms
Data1 ~ Data(N-1):	Reserved
	Don't Care

2.5.5.2 Explanation

Make the modules in the network to go to sleep with a sleep period specified by the user. This command can only be sent to the basestation. When this command is executed in the basestation, then the basestation "floods" the network by sending out a broadcast message that is relayed to all the modules in the network via each module. When the modules receive the sleep message from the basestation, then it will go to sleep after around 500ms. Once the modules go to sleep, then they will go to sleep for the specified sleep period, shutting down every hardware component to keep the power consumption to its lowest possible level. After the sleep period elapses, the module wakes up for a very short time (~3ms) and listen for any Wakeup command. If it receives a Wakeup command, then it will come out of the sleep state and operate normally.

If it doesn't receive a Wakeup command, then it ill go back to sleep for the specified sleep period and repeat the process again. Sleep period decides the period of the sleep state, before it wakes up to listen for any wakeup command. A value from 0 to 6 can be assigned. 0 is for 100ms sleep period, 1 is for 200ms etc.

2.5.5.3 Example

Sleep period = 2

STX	LEN(L)	LEN(H)	CMD	Data0	Data1	…	Data9	CRC(L)	CRC(H)	ETX
0x02	0x0A	0x00	0x21	0x02	D.C.	…	D.C.	D.C.	D.C.	0x03

2.5.6 Wakeup (Command type: 0x22)

2.5.6.1 Command Format

LEN:	0x0A(10)
CMD:	0x22
Data1 ~ Data(N-1):	Reserved
	Don't Care

2.5.6.2 Explanation

Make the modules in the network to wakeup from the sleep state. This command can only be sent to the basestation. When this command is executed in the basestation, then the basestation "floods" the network by sending out a broadcast message that is relayed to all the modules in the network via each module. The broadcast message from the basestation is repeatedly transmitted for n times depending on the sleep period of the modules. The value of n is determined automatically by the basestation so that it is long enough for the modules to listen to atleast one wakeup command message during it's sleep period. When the module listens to the wakeup command messgae, then it comes out of the sleep state, and operates in normal state.

2.5.6.3 Example

STX	LEN(L)	LEN(H)	CMD	Data0	Data1	...	Data9	CRC(L)	CRC(H)	ETX
0x02	0x0A	0x00	0x21	0x02	D.C.	...	D.C.	D.C.	D.C.	0x03

2.6 C code example

Below is the C source code that can be used at the receiving side to synchronize the receiving packet with the start byte (STX) and to extract the data field from the packet;

```
//START byte search
if (START_BYTE_DETECT == false)
{
        if(CURRENT_RX_DATA == 0x02)
        {
                START_BYTE_DETECT = true;
                data_count = 0;
        }
//If START byte is detected
}
else
{

        //Reconstruct the 16bit LENGTH field
        if (data_count == 0)
        {
                rec_length = CURRENT_RX_DATA;
        }
        else if (data_count == 1)
        {
                rec_length += (CURRENT_RX_DATA<<8 & 0xff00);
                //Check the command type
        }
        else if (data_count == 2)
        {
                if (CURRENT_RX_DATA != 0x26)
                {
                        //If 0x26 command is not received
                }
```

```
            //Store the sending data local address
        }
        else if(data_count == 3)
        {
            SENDING_ADDRESS = CURRENT_RX_DATA;
            //Store data
        }
        else if (data_count <= (rec_length + 2))
        {
            DATA[data_count - 3] = CURRENT_RX_DATA;
            //CRC calculation and ETX detection routine added here
        }
        else if(data_count <= (rec_length + 4))
        {
            //End of RX packet, initialize
        }
        else
        {
            START_BYTE_DETECT = false;
            data_count = 0;
        }

        //Increment counter
        data_count ++;
}
```

3 MAC protocol features

MAC stands for Media Access Control MAC layer can prevent collision between the devices that uses the common media. Besides, SRF 2400M's MAC protocol has the flollowing advanced functions

(1) Automatic Network Configuration
(2) Multi-channel / Multi-hop for expanding communication area.
(3) Reliable Data Transmission
    1) Error Detection / Error Correction Code
    2) "Hidden Node" Prevention by CTS / RTS
    3) Automatic Data Fragmentation
(4) TX/RX buffering

SRF 2400M MAC protocol can be upgraded later.

3.1 Automatic Network Configuration

SRF 2400M MAC protocol supports automatic network configuration feature, which greatly eases the installation procedure. There is no need for strenuous network configuration procedures as one would encounter setting up the TCP/IP to access the internet. The protocol and messaging strategy supports random topologies.

Selecting one of the modules as basestation (local address = 1), the other modules with a different local address can be automatically detected by the basestation using automatic network configure command. When this command is sent to the basestation floods the network and each module will choose its parent module depending on the RF strength measured by the receiving module and some processing based on statistics. When all the modules select their parent module, then each of them sends its parent local address to the basestation so that the basestation knows the parent module information of each module, and thus the whole network topology information.

Any of the network topologies in figure 6 a), b), c) can be configured with a single UART command given to the basestation.

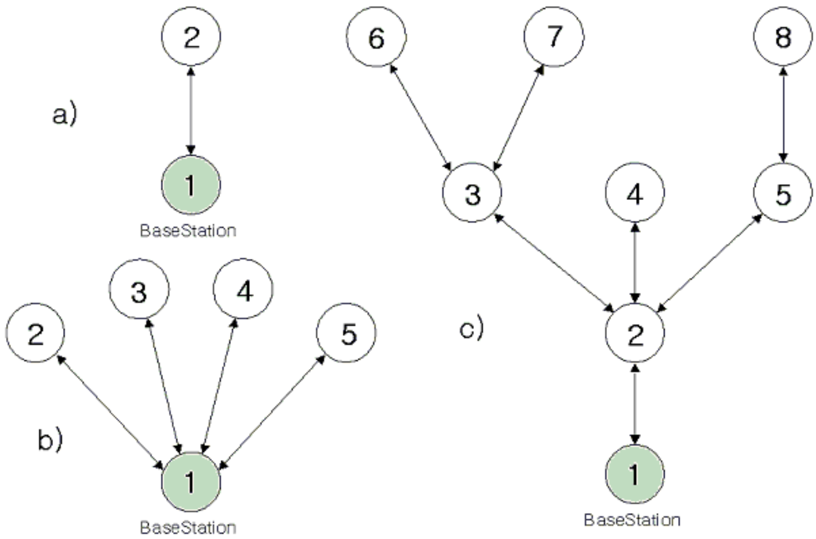


Figure 6 Configurable network topologies with SRF 2400M

A simple procedure for configuring the network is as follows;

1) Configure all the modules with the same group address with communication configure UART command from the local Microcontroller.
2) Configure all the modules with a different local address, with the basestation set as 1 with communication configure UART command.
3) Send a network configure UART command to the basestation module.

The network will be configured automatically and the network topology information is also sent to the UART from the basestation module.

More than one network can coexist when a different group address is assigned for a different network. Figure 7 shows such a case. group address 1,2,3 are assigned to three different networks, and there are three basestations assigned for each network. All three networks can coexist.

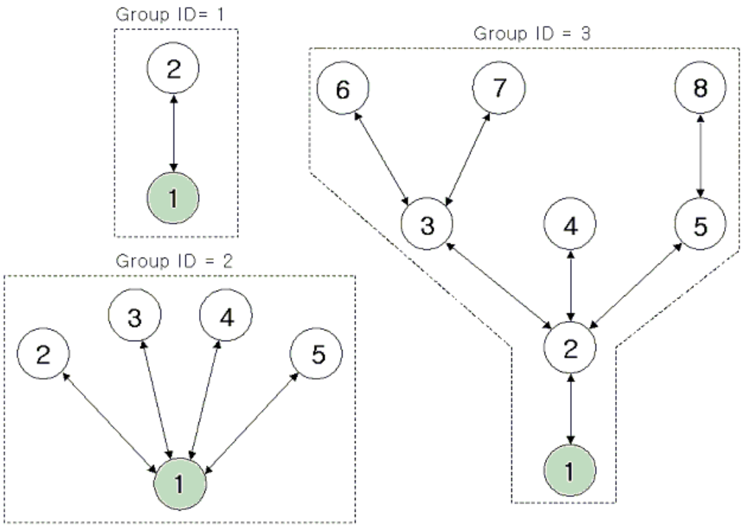


Figure 7 Assigning different Group address allows more than one network to coexist.

3.2 Multi Channel/Multi-Hop for expanding communication area

SRF 2400M MAC protocol is based on Carrier Sense Multiple Access (CSMA) for multi channel communication and also multi-hop communication is supported. CSMA protocol detects any nearby 2.4 GHz RF signals and allows transmissions only when the ' air is free ' so that collision can be avoided. The protocol supports two kinds of carrier sensing methods. The protocol allows any nearby RF signals from either SRF 2400M modules or other 2.4 GHz ISM band systems such as other WLAN / Bluetooth systems and microwave oven to be detected. This allows the SRF 2400M modules to coexist with other 2.4 GHz ISM band devices.

SRF 2400M MAC protocol offers multi-hop communication. This enables the network configuration to have arbitrary topology and enables the communication range to be extended further. For instance, the range can be increased three-fold using three nodes configured as a three-hop network. SRF 2400M is especially useful in fields where cable installation is costly or tedious.

3.3 Reliable Data Transmission

(1) Error Detection / Error Correction Code

SRF 2400M MAC protocol offers handshaking data communication with both error detection and error correction algorithms for reliable data transmission. When a packet is received, the protocol first tries to detect and correct the error, and if the error cannot be corrected the source module retransmits the packet.

(2) "Hidden Node" Prevention by CTS / RTS

SRF 2400M MAC protocol supports Request To Send (RTS)/Clear To Send(CTS) protocol like 802.11 protocol to avoid "hidden-node problem" as shown in figure 8 which is an intrinsic property in multi-hop communication. The " hidden node " problem occurs when both node A and C need to send data to B. Note that the communication range of node A and C (dotted circles) can reach node B but not the other nodes. So, when C is talking, then A can't listen and vice versa. When node C is talking to B, node A cannot sense the carrier sent from node C using CSMA protocol so it will think that the ' air is free' when it wants to send data to B. Since both node A and C are talking, node B will receive corrupt messages.

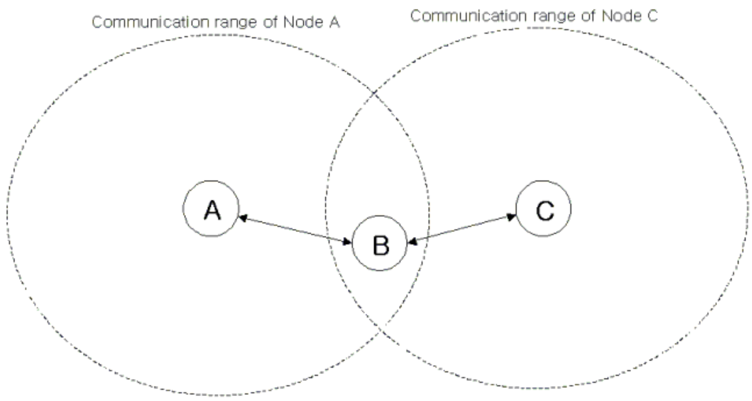


Figure 8 "Hidden node" problem example

Using RTS/CTS, any node that is willing to send data to other node, will ask if the destination node is free before sending. So, for the above case, when node B is communicating with node C, then node B would not allow node A to send data to itself before node C finishes the transmission. This way, there won't be a case when both node A and C talks to B and thus the " hidden node" problem is avoided.

(3) Automatic Data Fragmentation

SRF 2400M MAC protocol also supports automatic data fragmentation when sending/receiving a long data stream.

When the data stream is longer than a single RF data packet length (25 bytes) then store & forward mode of data transmission is supported for more reliable communication. The data stream is first stored in EEPROM when received from UART and when finished, the MCU reads the stored data one RF data packet unit (25 bytes) at a time and forwards to RF, until all the stored data is sent. The receiving side operates vice versa. It stores the received data in its local EEPROM and after all the data is received, it then reads the data from EEPROM and forwards to UART in the same format it was initially received from UART in the sending side.

3.4 TX/RX buffering

SRF 2400M MAC protocol can buffer up to 5 RX/TX packets of any order. This enables the communication to be more robust, especially in multi-hop communication topology where transmission traffic can be very congested. Buffering protocol allows the module to stack received packets without losing it until the channel to the next hop module is clear and then send all the stacked packets one at a time.