D A T A   S H E E T

YLX-TRM8053-500-05

Long Range Wireless Module

868-870MHz, 500mW Transmit Power, -114dBm Receiver Sensitivity

2011 Jan

Y-Lynx
Rue de Galilée15
CH - 1400 YVERDON
SWITZERLAND

Phone: +41 24 423 92 05
Email: info@y-lynx.com
Web site: http://www.y-lynx.com

Contents

IMPORTANT NOTICE

Documentation updates
Since Y-Lynx products are constantly evolving to meet customer needs, some technical information may differ from those described in this document. Please refer to our web site at www.y-lynx.com to obtain the latest documentation available.

Recommended reading
Other useful documents can be found on our web site www.y-lynx.com

Communication Controller version
Communication Controller software revision: rev1.05

1. GENERAL DESCRIPTION

The YLX-TRM8053-500-05 Radio Modem is Y-Lynx's longest range wireless product. The Modem transfers a serial data stream between two or more nodes.

Compact, the YLX-TRM8053-500-05 can replace easily miles of cables in industrial applications. Based on the Y-Lynx's Communication Controller, LibIC-5305, the radio modem resists to interference and enables collocated client and network operation without degrading data integrity and range performances thanks to TDMA / FDMA or FHSS (Frequency Hopping Spread Spectrum) embedded protocols.

The long range radio modem is pin to pin compatible with the 25mW radio modem: the YLX-TRM8053-025-05.

1.1 Features

RF Performances:

- 868-870MHz Free license band
- 500mW output power
- High reception sensitivity down to -114dBm @ 4.8kbps
- Low current consumption
  - 20mA in Receiver Mode
  - 650mA @ 500mW output power
  - <1uA in shut-down Mode
- Range
  - Outdoor line-of-sight: up to 40km w/ dipole antenna

Protocol and Networking

- Point to point, point to multi-point and peer to peer networks.
- API commands to control packet routing
- Embedded protocol
  - Selectable number of hops
  - Dynamic frequencies allocation
- Retries and acknowledgement
- Digital RSSI output
- SPI or UART host interface
- Hardware protocol Status Tracking
- Safety Mode (Default settings available from ROM memory)
- Two generic input / output

Easy to use

- 2.4V to 3.6V power supply
- Advance configuration available
- Transparent operation
- Small form factor and fit easily into a wide range of application
- Fully assembled and tested

1.2 General Specifications

YLX-TRM8053-500-05 Radio Modem

RF Performance		
Transmit output power (software digitally selectable)	500mW (27dBm) Maximum	
Outdoor line-of-sight range	40km w/ dipole antenna	
RF Data rate	4.8kbps	152.3kbps
Receiver Sensitivity	-114dBm	-106dBm
Power requirements		
Transmitter current at 3.3V		
Power output 500mW	650mA	
Receiver current	20mA	
Shut down Mode	<1uA	
General		
Frequency	868 - 870 /MHz	
Modulation	FSK (Frequency Shift Keying)	
Protocol	TDMA / FDMA / FHSS	
Network topologies	Point to Point, Point to Multipoint, Peer to peer	
Channel Capacity	50 hops sequence selectable with 500Hz precision	
Host Interface	SPI / UART with optional flow control	
	1’200 bps	115’200bps
Physical Properties		
Board size	74.7 mm x 29.1 mm x 6.5mm	
Output impedance	50 Ohm unbalanced	
Antenna connector	MMCX	Pad
Digital connector	40 pins	Drop-in

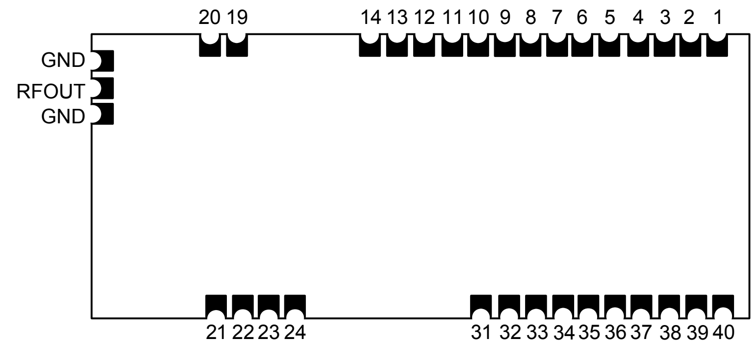
1.3 Electrical Specifications

PARAMETERS	MIN	NOM	MAX	UNITS
Supply Voltage during normal operation, Vcc	2.4	3.3	3.6	V
Supply Voltage during flash programming	2.7	3.3	3.6	V
Supply Voltage, Vss	0.0		0.0	V
V_{OH} High Level output voltage – Vcc=3.3V	Vcc-0.6		Vcc	V
V_{OL} Low Level output voltage – Vcc=3.3V	Vss		Vss+0.6	V

1.4 Pin Definition

The YLX-TRM8053-500-05 has simple interface with the OEM external application. The table below shows the connector/pad pin number and associated functions.

YLX-TRM8053-500-05 Pin Number



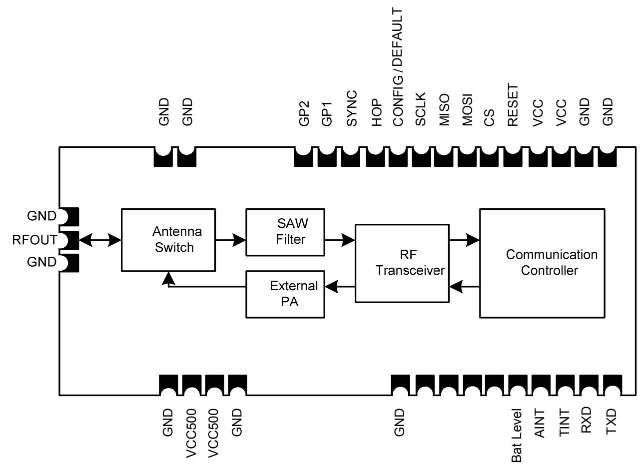
Pin Signal Description

Pin Number	Type	Signal Name	Function
1	-	GND	Ground
2	-	GND	Ground
3	-	VCC	Digital Power supply 2.4V – 3.6V
4	-	VCC	Digital Power supply 2.4V – 3.6V
5	I	RESET	Reset. Active Low
6	I	CS	SPI – Chip Select
7	I	MOSI	SPI – Master Out (Slave In)
8	O	MISO*	SPI – Master In (Slave Out)
9	I	SCLK*	SPI – Serial Clock
10	I	CONFIG / DEFAULT	Configuration Enable. Active Low Reset to default configuration at the power-up
11	O	HOP	Pulses high when the module is hopping
12	O	SYNC	Active high when a client is synchronized with a Server
13	I/O	GP1	General Purpose I/O Can be connected to any internal signals
14	I/O	GP2	General Purpose I/O Can be connected to any internal signals
15 -18	✕ Reserved, Do not connect		
19	-	GND	Ground
20	-	GND	Ground
21	-	GND	Ground
22	-	VCC	Digital Power supply 2.4V – 3.6V
23	-	VCC	Digital Power supply 2.4V – 3.6V
24	-	GND	Ground
25-30	✕ Reserved, Do not connect		
31	-	GND	Ground
32-35	✕ Reserved, Do not connect		
36	I	BAT_LEVEL	Input to the Battery Level Detector
37	I	AINT	Application Interrupt
38	O	TINT	Terminal Interrupt
39	O	RXD	UART – Receive Data
40	I	TXD	UART – Transmit Data

** Internal pull-up of 15k Ohm*

2. MODULE OPERATION

2.1 Block Diagram



2.2 Module Description

The YLX-TRM8053-500-05 contains a communication controller with embedded radio protocol, a RF transceiver, an external power amplifier to deliver 500mW, a SAW filter placed on the receiver chain and an antenna switch.

The communication controller handles the radio packet protocol, the communication interfaces, and the generic input output signals and controls the RF transceiver. The host sends data either on the TXD pin or MOSI pin, and buffered in the communication controller. Data packet is then assembled with the communication overhead as address information, before it is transmitted by RF. The RF transceiver modulates the data to be transmitted; the MAC method (FHSS, TDMA, FDMA) used to enhance the communication depends on the user settings. The last stage of the transmitter is an external power amplifier to obtain four output power levels up to 500mW (+27dBm). The antenna switch is driven directly by the communication controller and no specific command is required.

Received RF data go through a SAW filter for a better adjacent channel rejection and are demodulated by the RF transceiver. Data are then checked for correct address and redundancy check (if the option has been enabled) by the communication controller. If the address matches the recipient module own address (for more information on recipient address, please refer to Chapter 3.3-Address Header) and no error was detected, then after removing the overhead, data are assembled by packet of 8-bits and sent to the host on either the RXD line or the MISO line.

When the CONFIG/DEFAULT pin is driven to low, the communication controller enters in Configuration Mode and interprets data received on the UART or on the SPI as configuration commands. There are four levels of commands, the Application commands (for example: the group notification), the Network commands (for example: address assignment), Data Link commands (for example: Number of hops in FHSS Mode) and Physical commands (for example: RF data rate). Changes are then stored in the internal memory of the communication controller (RAM memory) and can be permanent by saving in the internal Flash memory. The CONFIG/DEFAULT pin can also be used, at the power up, to set the YLX radio modem with the default configuration and then to reset the changes from the RAM or Flash memory.

Additional digital signals are available from the YLX-TRM8053-500-05 module:

- The HOP pin informs the host that a new time slot (channel) will begin very soon and the complete data packet which has been stored in the Transmit buffer will be sent during this new RF channel.

- The SYNC pin is used by the host to monitor if the radio modem is synchronized with the Server (beacon server) and then is able to receive or transmit data.

- The TINT signal (Terminal Interrupt) becomes active when a modem contains data to be transferred to the host, which may be either received RF data or a command acknowledge.

- The AINT pin (Application Interrupt) has to be used by the host to select SPI data flow direction.

- GP1/GP2 pins (General Purpose) can be used to control an external device as well as to inform an external device on the current status of the module. If can be also used in combination with a GPS receiver to synchronized the wireless network thanks to the PPS signal (Pulse Per Second).

2.3 Network Architecture

Each network should consist of only one Server. There should never be two Servers on the same RF network as the interference between the two Servers will severely hinder RF communications.

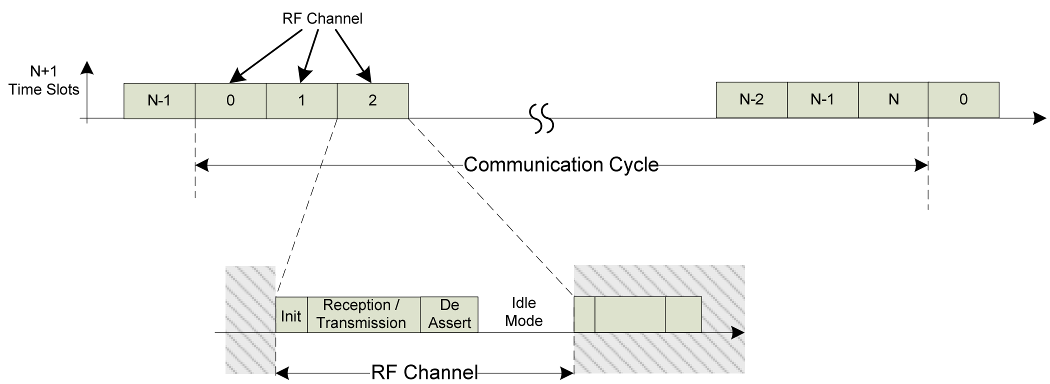
The Server controls the system by sending out regular beacons (transparent to the host) which contains system timing information. This timing information is used to synchronize Client to the Server.

Y-Lynx radio modem runs a Peer-to-Peer type architecture where all the transceivers, whether Servers or Clients, can communicate with all other transceivers.

3 THEORY OF OPERATION

3.1 Communication Cycle

The YLX-TRM8053-500-05 protocol is based on a very innovative and powerful architecture. The system uses communication cycles that mean that the time is divided into cycles, each cycles is constituted of several slots (called also RF channel). An RF channel is a short period of time where the wireless module can transmit or receive data.



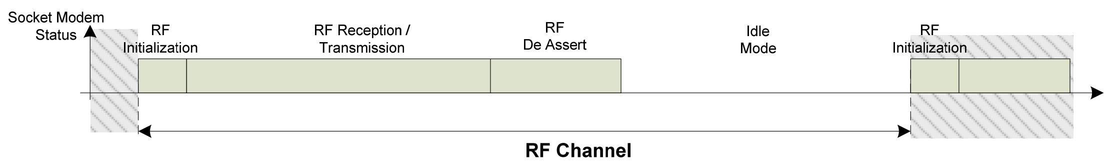
Thanks to the extended protocol option, it is also possible to pre-defined each channel behavior for each wireless nodes (wireless module). For example to prevent a wireless module to transmit data during the RF channel #1. For more information about the Channel behavior settings, please refer to the chapter: Advanced Operation – Channel Configuration Settings.

3.2 RF Channel

An RF message is sent or received during a RF channel. Three sub-Modes are used to define a channel:

- RF initialization: This sub-Mode allows the setting of the incoming RF channel. When the module shifts in this sub-Mode, complete data frame stored in transmit buffer will be send on this channel.
- Transmit or receive RF data: When the module shifts in this Mode, the module is set in receiver if no data need to be transmitted through RF. After the end of transmission or after the end of message reception the radio modem switches in the next sub-Mode, RF De-Assert Mode.
- RF De-Assert Mode: The De Assert sub-Mode is used to initiate the transfer of received data from RF to the host through the UART/SPI interface. If no RF message has been received, the radio modem switches automatically in Idle Mode.

An RF channel is defined by the succession of these sub-Modes as shown in the figure below:



Few parameters are necessary to qualify an RF Channel:

- RF Channel ID: This parameter is the identification code of the channel; it is generated automatically by the radio modem and is linked to the RF frequency. This ID depends also on number of channels used in the network. For more information please refer to Command Description.
    - CHANNEL_FREQ
    - CHANNELS_COUNT

- RF Channel frequency: this parameter defines on which frequency the RF message will be sent or received. It is important that the same RF frequency is used by the transmitter and the receiver to obtain reliable RF communication. For more information please refer to Command Description.
    - CHANNEL_FREQ
    - CHANNELS_COUNT

- RF Channel duration: In order to provide a maximum of flexibility, the radio Modem protocol is not based on fixed timing reference which allows to address different sort of protocol as TDMA, FDMA and FHSS. The RF channel duration is used to specify the time spent on channels. For more information please refer to Command Description.
    - CHANNEL_DURATION

3.3 Address Header

The Address Header is sent each time a new communication sequence begins. The Address Header contains the application and network addressing information (ID) that filters incoming RF data. The receiving module checks for matching the address header including the destination (Device) address. Data that does not pass through all the three filters is discarded.

- Application address: 8 bits
- Network address: 6 bits
- RF address: 8 bits

3.4 Addressing Options

A wireless transmission can be addressed to a specific radio modem or group of radio modems by using the destination address and the multicast address. A receiving module will only accept a packet if it determines the packet is addressed to it, either as a multicast or unicast packet. The receiving module makes this determination by inspecting the destination address of the packet and comparing it to its own address and multicast address.

3.5 Acknowledge Mode

In Acknowledge mode, the RF packet is sent out by a radio modem to another radio modem designated by the destination address. The sending module will send the data until receiving an acknowledge from the addressed module or until having finished all the retries allowed by the user. As only one data communication can take place in a given channel, several channels are used for the retries.

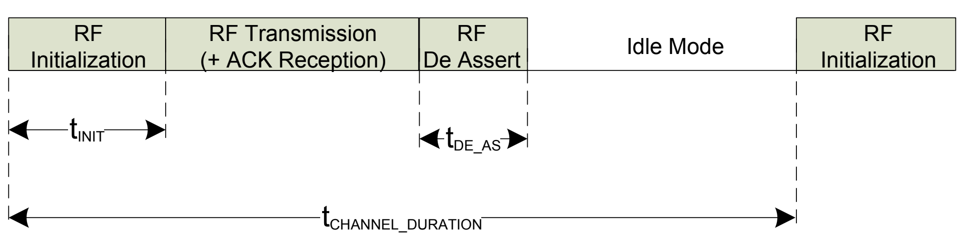
3.6 Timing Information

The figures and tables below show timing information for the module when switching from operating modes and timing information to respect for a proper behavior.

Only two parameters do not depend on the operating mode: RF initialization time and Channel duration time. These two parameters are fixed, one by the system itself, the other by a user command.

3.6.1 In Transmitter Mode

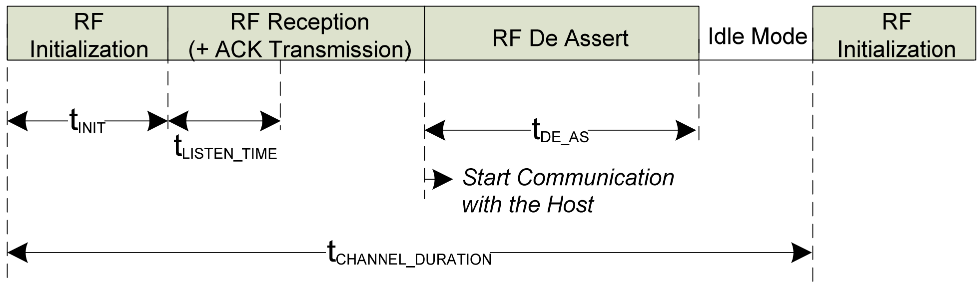
In Transmitter Mode, the transmission starts as soon as the RF initialization is finished. When the RF transmission is over, the module switches automatically in De-Assert Mode. As De-Assert Mode is used in Receiver Mode, t_{DE_AS} is very short. The module stays in Idle Mode until the end of the channel, specified by $t_{CHANNEL\_DURATION}$, no other operation can be done during this time. To increase the RF traffic on the network the Channel Duration must be chosen carefully according to the number of bytes to transmit and the RF bit rate.



Symbol	Description Note
$t_{CHANNEL\_DURATION}$	Time of a complete channel sequence
t_{INIT}	Fixed time, set parameters used by the next channel
t_{DE_AS}	Time used to initiate the transfer of RF data to the host

3.6.2 In Receiver Mode with message received

A received message is transferred to the host only if the CRC (depends on the settings) has been verified and if the same frame as not already been forwarded (no reception of the ACK); this verification can be observed on TINT output pin.

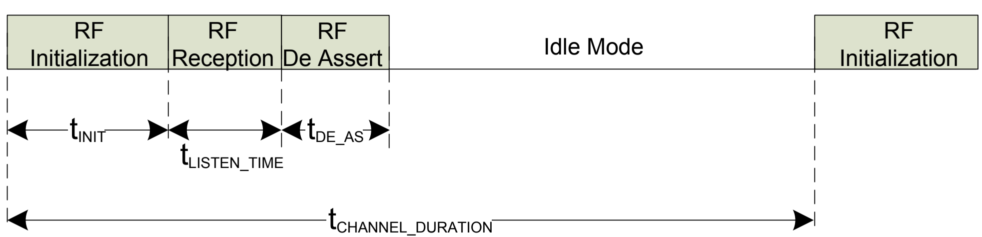


Symbol	Description Note
$t_{CHANNEL\_DURATION}$	Time of a complete channel sequence
t_{INIT}	Fixed time, set parameters used by the next channel
t_{LISTEN_TIME}	Time where the modem is set in receiver and track address header
t_{DE_AS}	Time used to initiate the transfer of RF data to the host

3.6.3 In Receiver Mode with no message received

If no Address Header is received during this period, the radio modem switches automatically in RF De-Assert Mode followed by the Idle Mode.

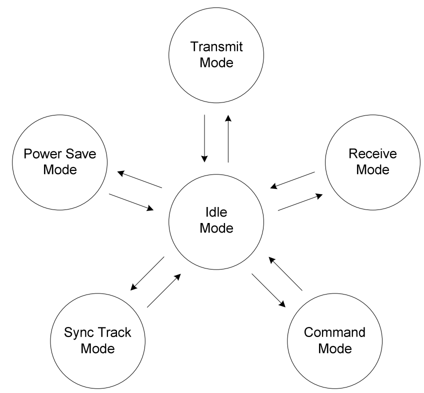
The time spent in De-Assert Mode is very short as no RF message needs to be transmitted to the host.



Symbol	Description Note
$t_{CHANNEL\_DURATION}$	Time of a complete channel sequence
t_{INIT}	Fixed time, set parameters used by the next channel
t_{LISTEN_TIME}	Time where the modem is set in receiver and track address header
t_{DE_AS}	Time used to initiate the transfer of RF data to the host

4 OPERATING MODES

The YLX-TRM8053-500-05 operates in six different Modes. The radio modem can be in one Mode at a time.



4.1 Idle Mode

When not receiving or not transmitting data, the radio modem is set in Idle Mode. The power consumption is equivalent to the Stand-By Mode.

The radio modem shifts into other operating Modes under the following conditions:

- Serial data is received in the transmit buffer (Transmitter Mode)
- A new time slot is starting (Receive Mode)
- A new synchronization slot is starting (Sync Track Mode)
- A new command is requested (Command Mode)
- Power Save Mode condition is met (Power Save Mode)

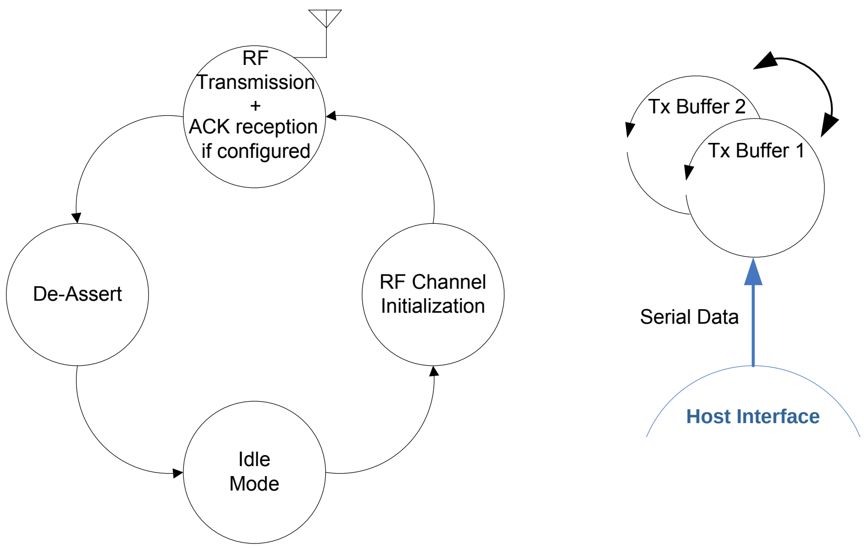
The module automatically transitions back to the Idle Mode after responding to these conditions.

4.2 Transmit Mode

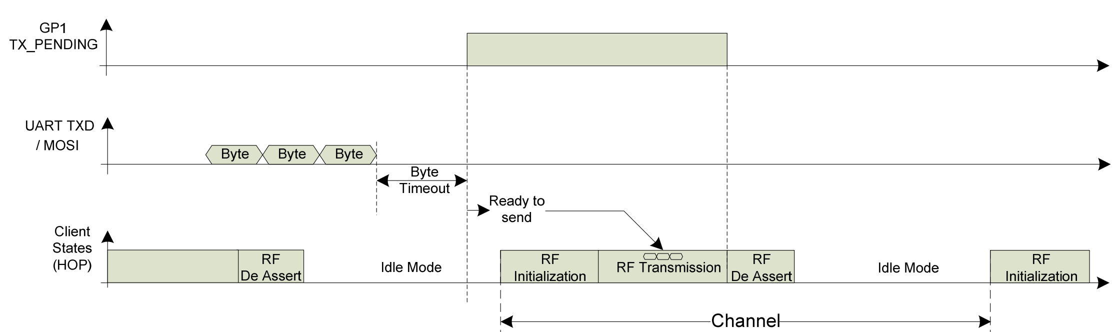
Based on the state machine described in the figure below, the modem enters in Transmit Mode only if a complete serial data frame has been received in one of Tx buffer otherwise the module enters in the RF reception mode.

The interface between the module and the host is asynchronous, that means that the host can send serial data to Tx buffers at any time. The mechanism to transmit data over RF and the mechanism to fill buffers are independents. Only complete frame will be sent on the following time slot.

The Transmit Mode can be described with four states. The priority is given to the RF communication as it is the most critical state, so transitions timing are based on RF rendezvous.

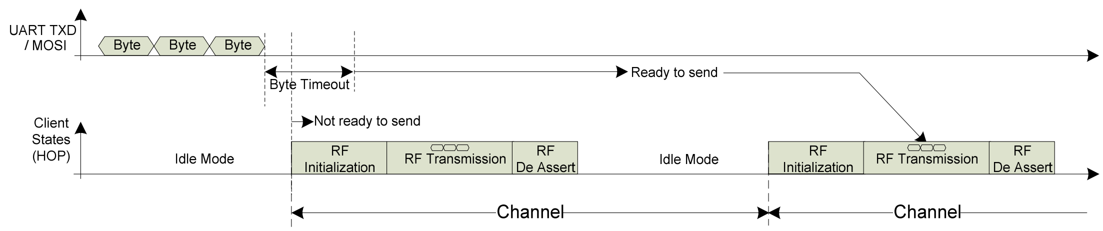


When no RF communication is planned, the radio modem is available to treat other tasks. The radio modem prepares the next channel configuration for the next time slot. When a new time slot starts, the radio modem sends data present on the Tx buffer over the air. When the transmission is over, the module returns in Idle mode and waits the next slots. Serial data are sent on the following time slot as the Interface Time out occurs before the transition to the Channel initialization.



Serial data sent on the following time slot

If the frame is not complete, Interface Timeout not detected before the beginning of the RF channel initialization, the serial data will be sent on the next time slot.



Serial data are sent on the next time slot

4.2.1 End of Data Frame

A RF frame is ready to be sent over the air if at least one of the enabled conditions is met.

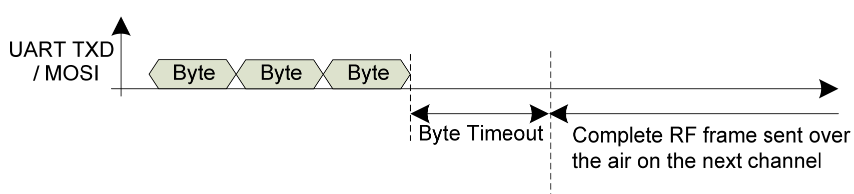
- A new RF communication channel starts
- A byte timeout occurs
- The maximum size of the frame is reached
- The CONFIG/DEFAULT pin is switch from DEFAULT (data mode) to CONFIG

As the last two conditions are always activated, the two others can be configured thanks to a dedicated command. CMD_SET_HOST_DATA_END_CONDITION (0xDC), for more information about this command, please refer to the command description guide.

Design Tips:

Only complete RF frame is sent on the following time slot through RF. One condition for a complete RF Frame is based on the interface Byte Timeout* between two bytes.

This parameter is set by the following command: CMD_SET_BYTE_TIMEOUT



** Byte Timeout: Maximum time between consecutive bytes.*

4.2.2 Maximum Size of RF message

The size of a RF message depends on the parameters set in the wireless module as channel duration, RF bit rate, size of the overhead and if the message has to be acknowledged. The Communication Controller embedded in the wireless module is able to calculate the maximum RF message length that can be sent in one channel. The command CMD_GET_RF_FRAME_MAX_SIZE (0xCB) gives the actual maximum value of bytes that can be sent by the modem.

The Communication Controller gives also the possibility to define the maximum RF message length, by using the following command:
CMD_SET_USER_RF_FRAME_MAX_SIZE (0xCE).

So, the maximum RF frame used by the wireless module will be the lowest value between the RF_FRAME_MAX_SIZE and the USER_RF_FRAME_MAX_SIZE.

Design Tips:

In Config Mode, the frame sent by the host to the modem does not depend on FRAME_MAX_SIZE so if a command is too long, a context error can be generated (ERR_CONTEXT) please for more information, refer to the command description guide.

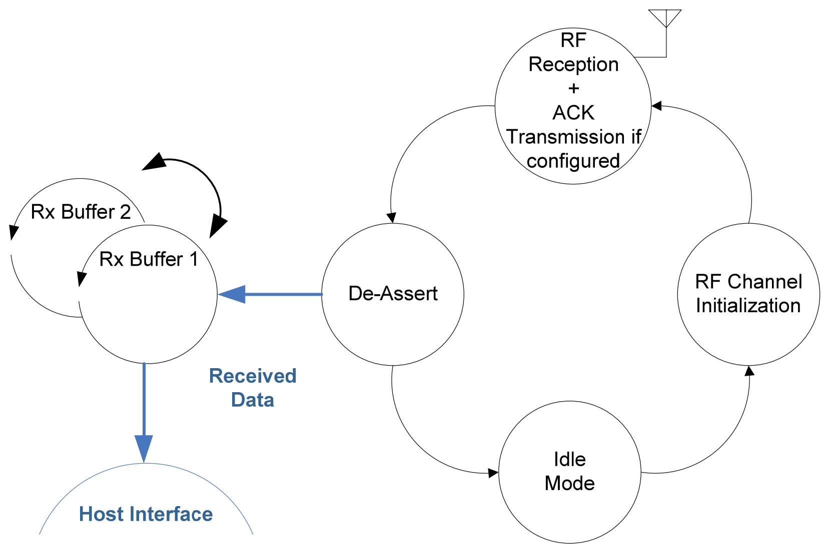
Channel duration [ms]	Maximum size (bytes) of pload accordingly with the RF bit rate and the ACK mode											
	152340		76170		38085		19042		9521		4760	
	Ack		Ack		Ack		Ack		Ack		Ack	
	off	on	off	on	off	on	off	on	off	on	off	on
10	83	26	32	0	0	0	0					
20	128	128	127	93	52	23	16	0	0			
30			128	128	101	71	40	15	9			
40					128	119	64	39	21	0	0	
50						128	87	62	33	10	6	
60							111	86	45	22	12	
70							128	110	57	34	18	0
80								128	69	46	24	2
90									81	58	30	8
100									93	70	36	14
110									105	82	42	20
120									117	94	48	26
130									128	106	54	32
140										118	60	38
150										128	66	44
160											72	50
170											77	56
180											83	62
190											89	68
200											95	74
210											101	80
220											107	86
230											113	92
240											119	98
250											125	103
260											128	109
270												115
280												121
290												127
300												128

4.3 Receiver Mode

The radio modem enters in Receiver Mode if no data are ready to be transmitted. As for the Transmitter Mode, the transitions timing are based on the RF rendezvous.

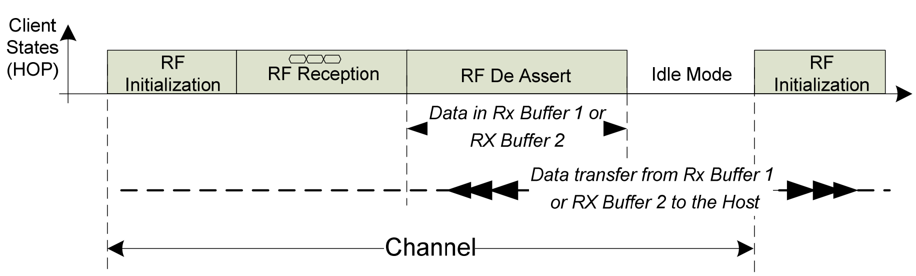
The Receiver Mode can be described with four states; RF channel initialization and De asserted state correspond to the same state used in the Transmit Mode mechanism.

In addition, data are forwarded to the host after the complete reception of the RF frame and is initiated during the De-Assert Mode.



When no RF communication is planned, the module is available to treat other tasks. The module prepares the next channel configuration for the next time slot. When a new time slot starts and if no data are ready to be transmitted, the radio modem transits in Receiver Mode. The received data are stored in one of Rx buffers until the radio modem enters in the De-Assert Mode where the received RF data will be transmitted to the host interface by UART or SPI, the host is informed by TINT pin level.

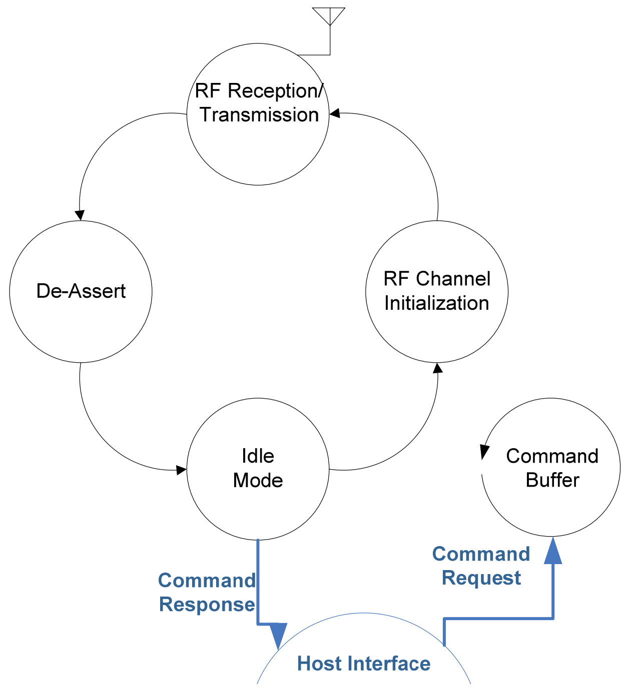
The figure below presents the case where RF data are received by the radio modem during the RF reception mode and transferred to the host during the De-Assert Mode.



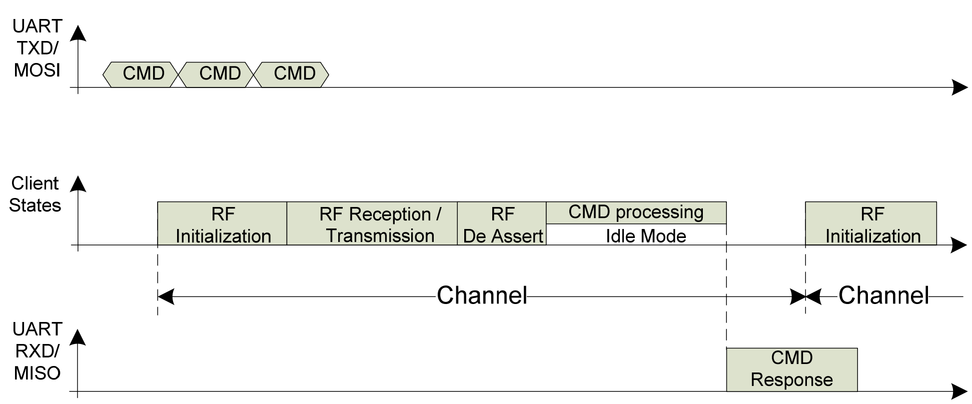
RF data are received and transferred to the host

4.4 Command Mode

As described in the Transmitter Mode with Tx buffers, the mechanism to send command to the system is independent from the mechanism of RF data reception/transmission, which means that the host can send a command request. The CTS signal mapped on GP1 is used to indicate when the Command Buffer is not available (please refer to GP1_SIGNAL command). The response to this command is given to the host interface only during the Idle Mode so after RF reception / transmission.



The figure below presents the case where a Command is transmitted to the module during the RF reception or transmission mode. The command processing and response are given during the next Idle Mode.

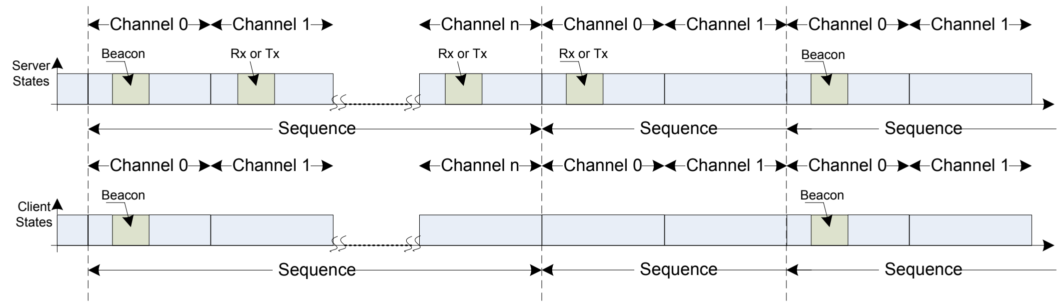


Command Mode and RF activity

Note: During the RF De-Assert phase, the RF data will be encapsulated if Command Mode is active. Please refer to the chapter Enter/Exit Command Mode.

4.5 Synchronization Track Mode

The synchronization Track Mode is used to reduce the power consumption without losing the synchronization message (Beacon) from the Network Server. The Beacon is sent by the Network Server during the first time slot of the frequency hopping sequence but not necessary every sequence.



The Beacon can be sent by the network server every “n” sequence, for example if a sequence is based on 50 hops per second and the Beacon Period is set to “3”, a beacon will be sent every 150 hops so every 3 seconds.

4.6 Power Save Mode

The YLX-TRM8053-500-05 offers two Power Save Modes. Each mode can be configured directly by the host controller:

- Power Sleep Mode
- Power Shut-down Mode

4.6.1 Power Sleep Mode

In Sleep Mode the radio modem keep the synchronization and is able to receive the beacon signal from the server. In addition, in this mode, the radio modem is also able to transmit a RF frame with or without acknowledgment.

The main difference with the normal mode is that in Sleep mode, the radio modem does not switch in receiver mode in each channel, but only in the beacon channel to keep the synchronization as described the parameter BEACON_PERIOD and the parameter BEACON_LISTEN_PERIOD.

To set the radio modem in Power Sleep Mode, the command Power_Mode is used with an argument equal to 0x01.

4.6.2 Power Shut-Down Mode

In Power Shut-down Mode, the radio transceiver is off and the communication controller is in low power mode. The radio modem does not receive or detect incoming data packet from the RF, only the UART or SPI interface is running.

To set the radio modem in power shut-down mode, the command CMD_SET_POWER_MODE is used with an argument equal to 0x02.

To transmit or receive data through RF, the power save mode must be set in normal mode (argument equal to 0x00).

Please refer to the command description (CMD_SET_POWER_MODE) for more information on the power save mode.

5 HARDWARE DESCRIPTION

5.1 Hardware Interfaces

The YLX-TRM8053-500-05 module interfaces to a host through different modes:

- Through its asynchronous serial port, the module can communicate with any UART (Figure 5.1)
- Through its synchronous serial port, the module can communicate with any SPI Master. The YLX-TRM8053-500-05'SPI is configured as slave. (Figure 5.2)

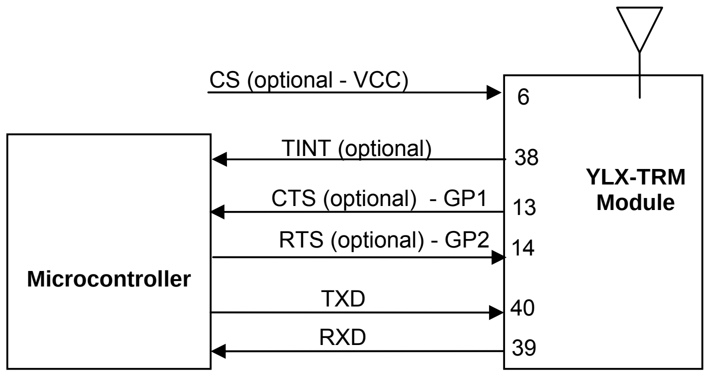


Figure 5.1: Interface through UART

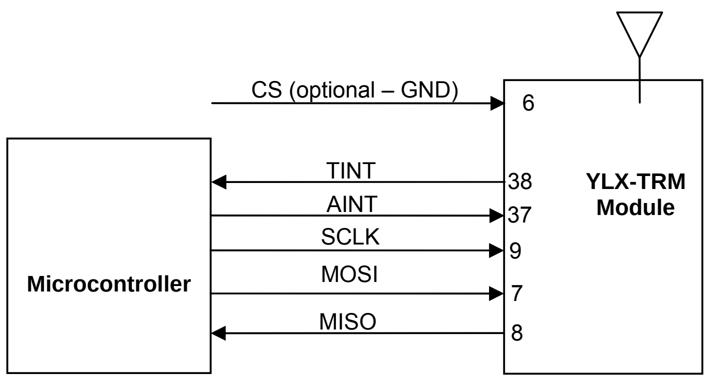


Figure 5.2: Interface through SPI

5.2 UART

The serial UART communication consists of two UARTs connected each other, as described in figure 5.1, with compatible parameters (baud rate, parity, start and stop bits) flow control to have successful communication.

Each data packet consists of a start bit (low level), 8 data bits (LSB first) and one or two stop bits (high level). The following diagram (Figure 5.3) illustrates the serial UART pattern to transfer data or commands to the YLX radio modem.

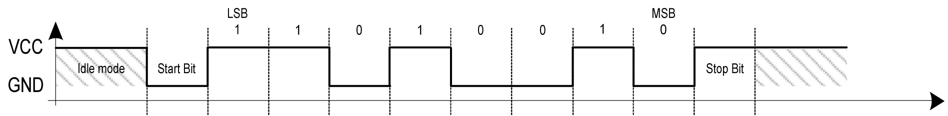


Figure 5.3: UART Data Packet (example 0x4B)

The UART communication implemented in the modem can be parameterized to insure compatibility with the largest set of hosts. The baud rate is set through the command CMD_SET_HOST_BAUDRATE and the other parameters (RTS/CTS polarity, Parity, Stop Bit) through the command CMD_SET_HOST_UART SETTINGS.

The host interface selection between UART and SPI is done using the Chip Select pin (CS) with low level to select the SPI interface and high level to select UART interface.

WARNING:

When the SPI has been selected once, the UART is disabled until next reset.

5.3 UART Flow Control: RTS - CTS

The RTS and the CTS functionalities are dissociated, thus each signals can be independently activated/deactivated.

CTS is a modem output signal which can be mapped on GP1 pin and used to inform the host when the modem is no more ready to receive data.
Please refer to CMD_SET_GP1_SIGNAL.

RTS is a modem input signal which can be mapped on GP2 and used by the host to prevent the modem from sending more data. When RTS is not mapped on GP2, it is considered as active and so the modem will send data to the host when it is required.
Please refer to CMD_SET_GP2_SIGNAL.

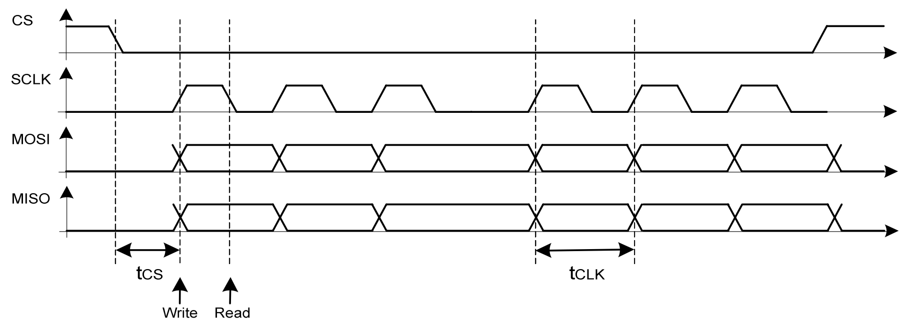
RTS and CTS signals are by default active low "0" and the communication is allowed. The polarity of either signal can be changed through the following command CMD_SET_HOST_UART_SETTINGS.

5.4 SPI

The serial SPI communication consists of two SPI connected each other (Master and Slave) as described in figure 5.2. The Y-Lynx radio modems have slave SPI interface. The host needs to be configured with a Master SPI to have successful communication.

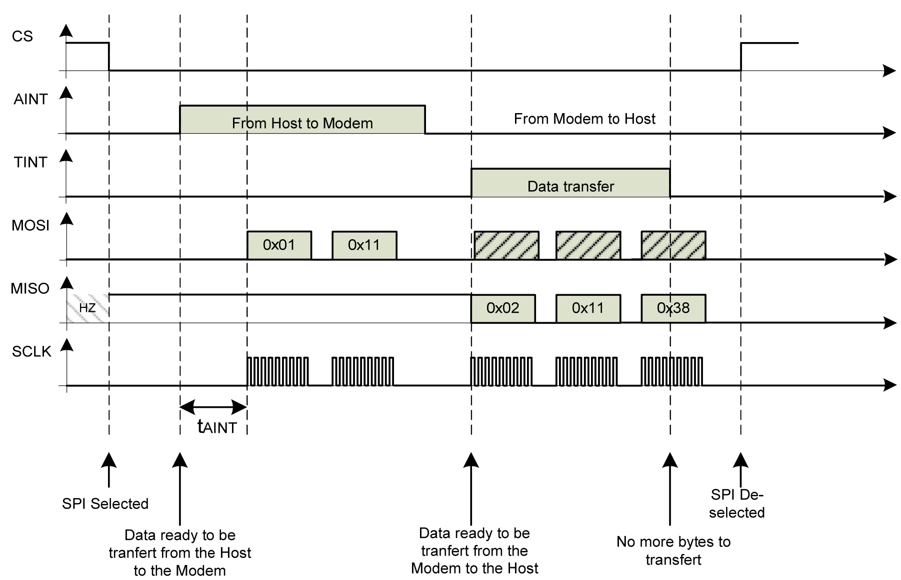
The SPI bus is based on four signals, the chip select (CS), the Serial Clock (SCLK) and two signals for data. One is from the master to the slave (MOSI) the second is from the slave to the master (MISO).

To start a communication with the radio modem by using the SPI bus, the host drives low the chip select. On each rising edge of the serial clock, generated by the host, the data is written by the Master on the MOSI pin and by the slave on MISO pin. The reading of data is done on the falling edge by master on MISO pin and by the slave on MOSI pin. The following diagram illustrates the serial SPI pattern to transfer data or commands.



When the SPI is selected, the signals AINT and TINT are used to indicate that data are ready to be transmitted to the Modem (AINT=“1”) or to the Host (AINT=”0” and TINT=”1”).

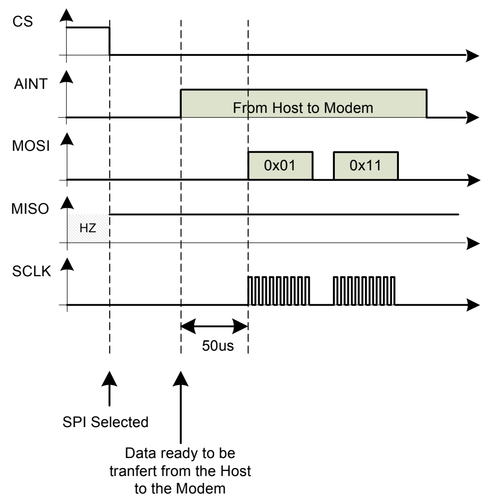
In the example below, the host sends the command GET_LIBIC_VERSION (0x11), the modem gives the following answer: 0x38



Name	Value
t_{CS}	50µs
t_{CLK}	1µs
t_{AINT}	50µs

5.5 AINT

When the SPI is selected to communicate between the host and the modem, the Application Interrupt (AINT) signal has to go high 50us before to start the SPI communication to transfer data (RF Data or Command) to the modem.

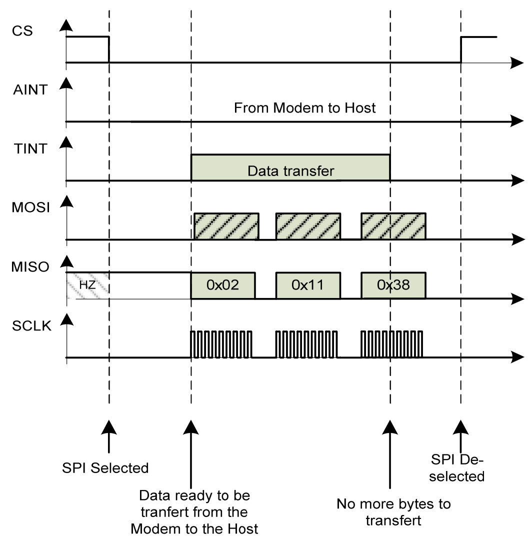


AINT	Communication
0	No communication or communication from Modem to Host
1	From Host to Modem

When the UART is selected, the Application Interrupt (AINT) has no influence.

5.6 TINT

When the SPI is selected to communicate between the host and the modem, the Terminal Interrupt (TINT) signal generated by the modem goes high to indicate to the host that data (RF Data or Command Response) are ready to be transferred.



AINT	TINT	Communication
0	0	No communication
0	1	From Modem to Host
1	0	From Host to Modem
1	1	From Host to Modem

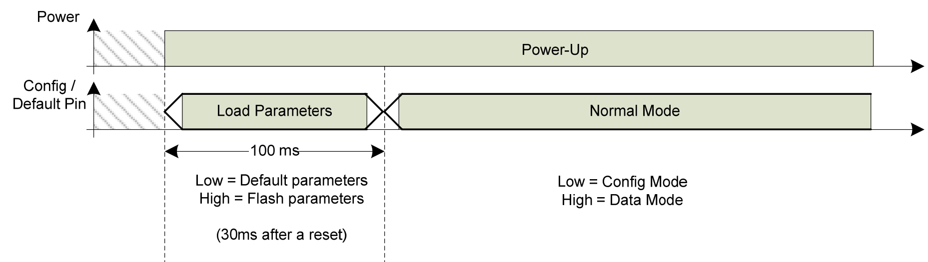
When the UART is selected, the Terminal Interrupt (TINT) has the same behavior as described with the SPI communication. But in UART mode, TINT level depends also on the flow control CTS signal.

5.7 CONFIG/DEFAULT Pin

This pin offers two different functionalities: the access to the Configuration Mode and the up-load of the default parameters in RAM memory to guarantee a reliable RF communication.

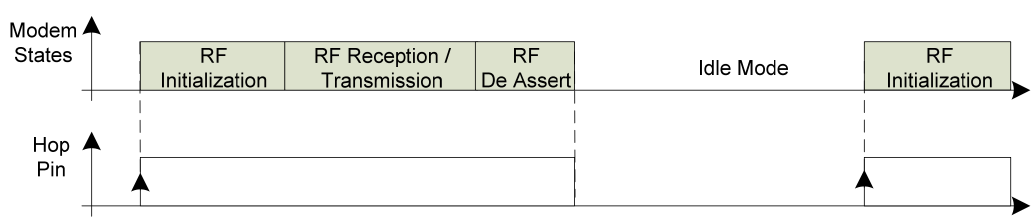
When the pin “Configuration / Default Mode” is driven low during 2.5ms after the power up or after a RESET, the radio modem enters in Default Mode and loads the default configuration stored in ROM memory (during production). Default Mode is available only at the power-up and after a reset; it is not possible to re-load the default configuration when the radio modem is in operation by using the Configuration/Default Mode pin. By the way, it’s feasible thanks to a dedicated command, LOAD_DEFAULT_PARAMETER or with the break sequence when using the UART interface (20 breaks sent consecutively).

When this pin is driven low after the power up (6ms later) or after the load of the default parameters (3.5ms later), the radio modem enters in Configuration Mode until the signal switches to high. This mode is used to set/read parameters thanks to predefined commands listed in Chapter 7. This mode is accessible at any time during the radio modem operation.



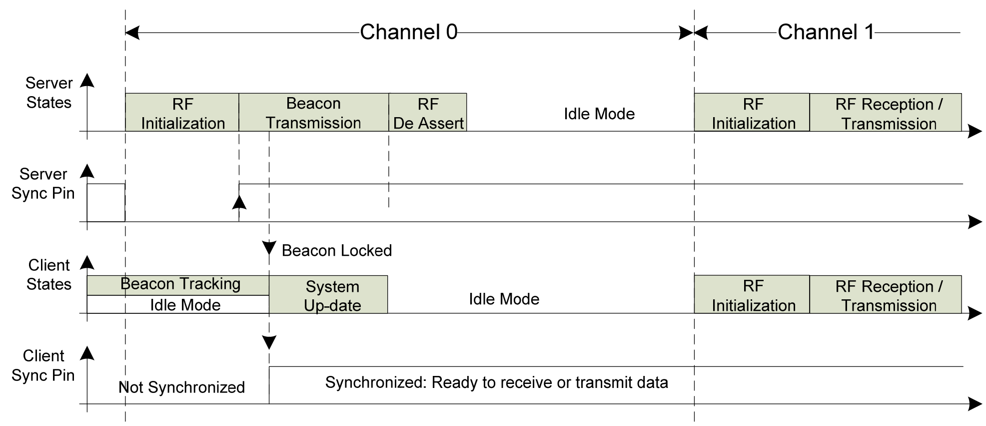
5.8 Hop Pin

The HOP pin goes high at the beginning of a time slot (a Hop) and goes low at the end of the slot (during the Idle). The Host is not required to monitor the Hop pin. This signal can be used by host to synchronize the communication with the radio modem.



5.9 SYNC Pin

The SYNC pin is used to inform the host on the synchronization status of the radio modem. The radio modem is considered synchronized when the beacon transmitted by the network Server is received or through an external synchronization signal likes the Pulse Per Second signal from a GPS module (for more information please refers to the chapters 6.1 and 6.2). If the radio modem is not synchronized, data received from the host, through the UART or SPI, are discarded and are not transmitted through RF. The Beacon Tracking mechanism is independent from the Command Mode, so the radio modem is able to receive and response to commands from the host during the Idle Mode. The SYNC pin signal pulses low ("0") at the beginning of each sequence, this signal can be used as a network synchronization signal for the application needs.



5.10 GP1 / GP2

GP1 and GP2 serve as general purpose pins. Reading and writing of these pins can be performed using Application Commands (details can be found in the Application Command Reference) or through commands received by RF.

Some useful internal signals can also be mapped on these pins, please refer to the list of commands (command 0xE0 to 0xE7).

5.11 POR: Power on Reset

The YLX-TRM8053-500-05 embeds a Power on Reset functionality which loads the module configuration in RAM memory. Two kinds of configuration exist, the default configuration defined by Y-Lynx and stored in ROM memory and the User configuration stored in Flash memory.

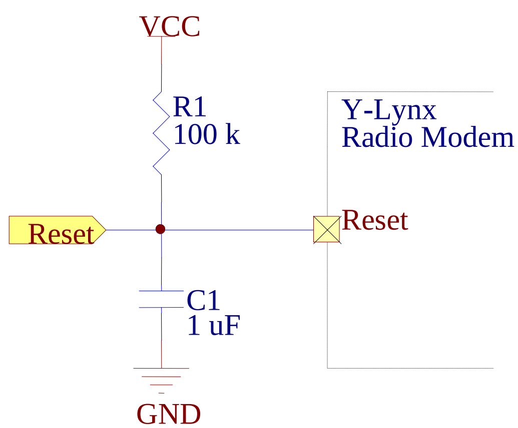
During the Reset, the Config/Default pin (pin #10) is also used to make the selection between Y-Lynx default configuration and User configuration; this is the Conditional configuration settings mode. But it is also possible to load a configuration whatever the level of Config/Default pin (Unconditional configuration settings).

The command RESET_MODE is used to define the source (Y-Lynx / User) and type (conditional / unconditional) of the configuration to load after the reset.

For more information, please refer to the list of commands (Commands 0xDE and 0xDF) by using the following argument:

Mode (hexa)	Config/Default pin level	Configuration source
0x00	Low	ROM – Y-Lynx settings
0x00	High	Flash – User settings
0x01	Low	Flash – User settings
0x01	High	ROM – Y-Lynx settings
0x02	X	Flash – User settings
0x03	X	ROM – Y-Lynx settings

To guarantee a proper behavior it's highly recommended to add an external 10µF capacitor on the RESET pin.



For more information on the default configuration settings (Y-Lynx settings), please contact Y-Lynx office.

6 ADVANCED OPERATIONS

6.1 External Synchronization Mode

To be able to communicate wirelessly, a module has to be synchronized by receiving an RF beacon signal from the Server. This beacon signal is used by the module to obtain timing information. The first channel of the communication cycle (channel 0) is often used to send the beacon.

But it is also possible to use an external signal as clock source, for example the pulse-per-second (PPS) signal delivered by most of GPS modules. This external clock has to be mapped on the general purpose I/O GP2. A specific command is used to change the synchronization source: GP2_SIGNAL (0xE4) with SYNC_IN (0x02) as parameter.

In addition the external synchronization source (SYNC_IN) has to respect a precise timing between two risings edges which has to be greater than:

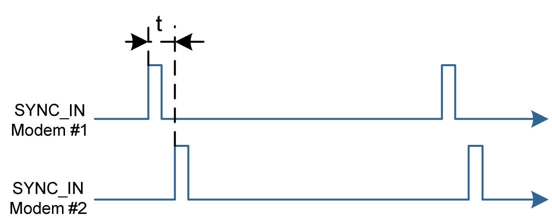
CHANNEL_COUNT x CHANNEL_DURATION - 250µs

For example, by default:
CHANNEL_COUNT =50
CHANNEL_DURATION = 2 (2 x 10ms)

So, the SYNC_IN rising edge period has to be greater than:
50 x 20E-3 – 250E-6 = 999.75ms

The precision of SYNC_IN must be also precise in order to guarantee a proper network behavior; the difference must be lower than

$t = \pm 50\mu s$

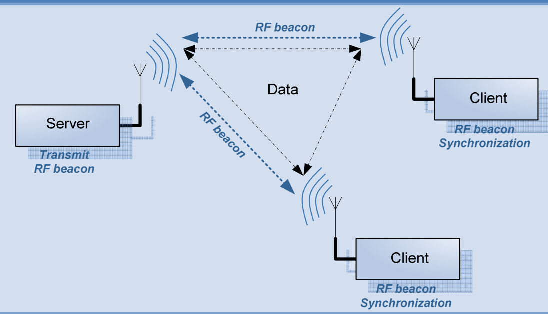
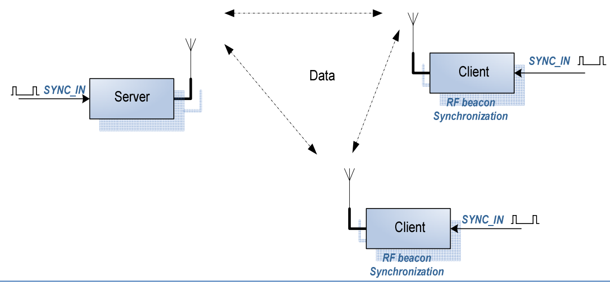
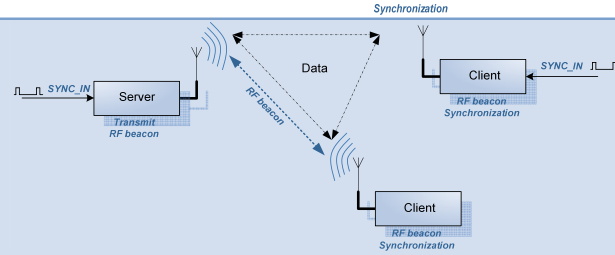


For more information about the external synchronization, please refer to Y-Lynx Application Note: AN005 Serverless Network

6.2 Network Synchronization

As described above, a node (wireless module) can not transmit data if it is not synchronized with the network. This condition allows reducing the risk of data collision.

Thanks to the possibility to synchronize a module with an external signal, there are three possibilities of architecture for a wireless network based on the Y-Lynx protocol (LibIC-5305)

Network Synchronization	Modules Configuration	Topology
RF Beacon	- One module set as Server (with SYNC_IN not mapped on GP2). Transmit RF beacon - Other modules set as Client (with SYNC_IN not mapped on GP2). Receive RF beacon	
SYNC_IN Signal	All modules are set as Beaconless Network (0x02 0x54 0x02)	
Mixed (RF beacon or Sync_in)	- One module set as Server with SYNC_IN mapped on GP2. Transmit RF beacon - Some modules are set as Client with SYNC_IN mapped on GP2. - Some modules are set as Client (with SYNC_IN not mapped on GP2). Receive RF beacon	

Depending on the chosen Network Synchronization topology, wireless modules have to be configured according to one on the following configuration.

Beacon Mode	RF Beacon synchronization (SYNC_IN not mapped)	External Synchronization (SYNC_IN mapped)
Client (0x00)	- Synchronized by RF Beacon - The beacon channel is not available for data transmission - The beacon channel is used to receive RF beacon - A server is required in the network	- Synchronized by SYNC_IN - The beacon channel is not available for data transmission - No server is required but one is tolerated in the network
Server (0x01)	- Self Synchronized (transmit the RF Beacon) - The beacon channel is not available for data transmission - The beacon channel is used to send RF beacon - No other Server is tolerated in the network	- Synchronized by SYNC_IN - The beacon channel is not available for data transmission - The beacon channel is used to send RF beacon - No other server is tolerated in the network
Beaconless Network (0x02)	Not allowed	- Synchronized by SYNC_IN - The beacon channel is available for data transmission - Other modules have to be set in Beaconless Network - No server nor clients are tolerated in the network

6.3 ENCRYPTION

When the encryption is enabled, the RF frame is encrypted (ciphered) before the transmission and un-encrypted (deciphered) at the reception stage. The encryption algorithm is based on a 32-bits key code; this code must be the same at the transmitter side and at the receiver side to allow a correct wireless communication. If the key code at the receiver side is not the same or if the encryption is disabled, the data stream is received but is different from the original (data sent by the transmitter).

To increase the data security it is highly recommended to use the extended protocol functionalities (c.f. SET/GET_EXTENDED_PROTOCOL).

It is also recommended to change the key code before each communication to guarantee data confidentiality.

Two commands are used to enable or disable the encryption function. For more information please refer to the list of commands, (command 0xA2 and 0xA3 as well as 0xA4 and 0xA5).

6.4 Remote Commands

Commands can be sent to the connected module as well as to a remote module (available only from the firmware version 1.03). To send commands to a remote module, the REMOTE_CMD_MODE (0x76) has to be enabled (cf: Communication Controller User's Manual).

A remote command is sent thanks to the CMD_SEND_COMMAND (0xD8) and the response is given through the indicator IND_RECEIVED_RESPONSE (0xD9).

The available remote commands are:

Command Name	Command Code
GET_VERSION	0x10
GET_LIBIC_VERSION	0x11
SET/GET_GP1_VALUE	0xE2 / 0xE3
SET/GET_GP2_VALUE	0xE6 / 0xE7
SET/GET_REMOTE_CMD_MODE	0x76 / 0x77
SET/GET_ALL_CHANNEL_TYPE	0x78 / 0x79
SET/GET_CHANNEL_TYPE	0x3C / 0x3D

Example:
Read the value of the remote GP1, (the remote GP1 has been configured as input).

Message sent by the connected module.
TXD: 0x03 0xD8 0x01 0xE3

Where:
0x03: Length of the command
0xD8: SEND_COMMAND code
0x01: length of the remote command
0xE3: code of the remote command (GET_GP1_VALUE)

The response received by the connected the module (the level of GP1 is high, "1")
RXD: 0x04 0xD9 0x02 0xE3 0x01

Where:
0x04: Length of the response
0xD9: IND_RECEIVED_RESPONSE code
0x02: length of the remote response
0xE3: code of the remote command (GET_GP1_VALUE)
0x01: response (GP1 is high)

6.5 Channel Configuration Settings

When the extended protocol is enabled then it is possible to define the configuration of each channel used during the communication cycles.

The command CHANNEL_TYPE (0x3C / 0x3D) allows to set or get the type of the channel according to argument described in the table below:

Channel behavior	TYPE
Idle mode	0x00
Receive mode only	0x01
Transmit mode only*	0x02
Receive and transmit (default mode)	0x03

**. an RF transmission occurs only if data are present in the TX_buffer.*

Example:
The RF channel #4 is set in receiver mode only, so the wireless module cannot transmit data during this period.

From the host (PC) to the wireless module
TXD: 0x03 0x3C 0x04 0x01

The response of the wireless module to the host
RXD: 0x01 0x3C

6.6 Channel Configurations Settings by Group

As described above, a RF channel behavior can be pre-defined. To facilitate the settings of all the channels defined in a communication cycle, the command ALL_CHANNELS_TYPE (0x78 / 0x79) can be used as following:

The configuration of each channel is set through two bits, the bit #1 for the transmission and the bit#0 for the reception

RF Channel	
Bit 1	Bit 0
Tx: Transmission	Rx: Reception

For example for an RF Channel the bits "10" means that transmission is allowed but not the reception.

In addition to reduce the communication between the host (PC) and the wireless module, the channel configurations settings are grouped by bytes according to the channel ID.

As the maximum communication cycle is constituted of 50 channels (0 - 49) and 2 bits per channel are required for the configuration, then 13 bytes (12.5) of 8-bits are used to configure the channels.

Byte number	Byte 12			Byte 1								Byte 0							
Channel ID	49			7		6		5		4		3		2		1		0	
Configuration	Tx	Rx		Tx	Rx	Tx	Rx	Tx	Rx	Tx	Rx	Tx	Rx	Tx	Rx	Tx	Rx	Tx	Rx

For example, all channel are set in idle mode (no transmission, no reception allowed), only channels 1 and 2 are set in reception only and channel 4 in transmission only. The settings bytes are then set as following

Byte number	Byte 1								Byte 0							
Channel ID	7		6		5		4		3		2		1		0	
Configuration	Tx	Rx	Tx	Rx	Tx	Rx	Tx	Rx	Tx	Rx	Tx	Rx	Tx	Rx	Tx	Rx
Binary Configuration	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0
Value in hex	**0x02**								**0x14**							

So the following command (TXD) has to be sent by the host to wireless module (byte 0 first) then the module response the following frame (RXD):

	Size	CMD	Configuration Bytes												
			0	1	2	3	4	5	6	7	8	9	10	11	12
TXD	0x0D	0x78	0x14	0x02	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
RXD	0x01	0x78													

6.7 CTS - Threshold

The CTS output signal, when mapped on GP1, is risen to inform the host that the modem is no more ready to receive data. This is done as soon as the modem buffer becomes full, thus after the complete reception of the last byte.

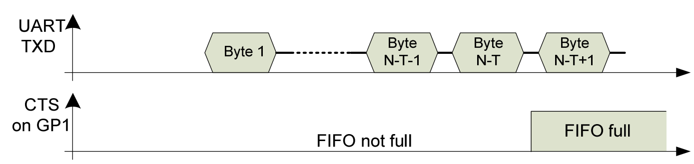
Depending on the host flow control implementation, the CTS signal can be sampled before it becomes valid and therefore the host may not be prevented from sending another byte which will be lost.

The CTS_THRESHOLD parameter (which corresponds to a number of bytes) can be used to fine tune the CTS signal behavior to insure that the host samples it in time. CTS is risen when CTS_THRESHOLD bytes remain in the modem FIFO. The followings commands are used:

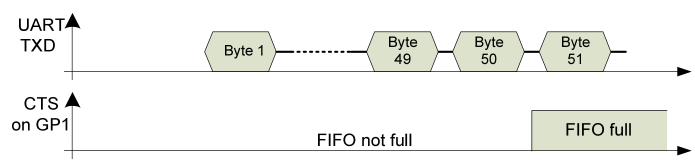
CMD_SET_CTS_THRESHOLD 0xDA
CMD_GET_CTS_THRESHOLD 0xDB

To set CTS_THRESHOLD value to “1” (the valid range is from 0 to 255), the host sends:
Tx: 0x02 0xDA 0x01
Rx: 0x01 0xDA
To get CTS_THRESHOLD value, the host sends:
Tx: 0x01 0xDB
Rx: 0x02 0xDB 0x01

The following figures show the CTS behavior related to the CTS_THRESHOLD

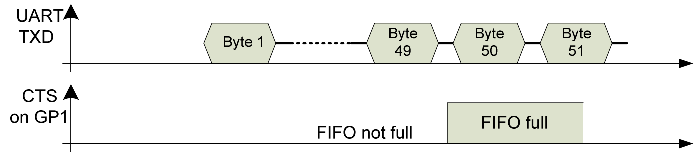


With CTS_THRESHOLD = T and HOST_FRAME_MAX_SIZE = N, the CTS signal is set after the N-Tth byte.*



With CTS_THRESHOLD = 0 and HOST_FRAME_MAX_SIZE = 50, the CTS signal is set after the 50th byte.*

*: For more information on the HOST_FRAME_MAX_SIZE command, please refer to the “Command Description” chapter.



With CTS_THRESHOLD = 1 and HOST_FRAME_MAX_SIZE = 50, the CTS signal is set after the 49th byte.*

*: For more information on the HOST_FRAME_MAX_SIZE command, please refer to the "Command Description" chapter.

6.8 END TO END FLOW CONTROL

The local flow control between a modem and its host can easily be extended to a global flow control between distant hosts. To obtain this "limited in time" global flow control, the receiver's host needs to map RTS on GP2 to control the flow and the sender's RF_RETRY_COUNT has to be set at a non zero value. The last requirement is to activate the modems RF_ACK_MODE and set other RF parameters to insure that the communication is possible.

The flow control efficiency is limited. A modem which receives an RF frame when its buffer is full, does not acknowledge the RF reception. The sender will then retry the transmission according to its RF_RETRY_COUNT parameter. As long as the RF frame is not acknowledged or RF_RETRY_COUNT reached, the sender sustains its CTS signal to inactive state to prevent that more data are received from the host.

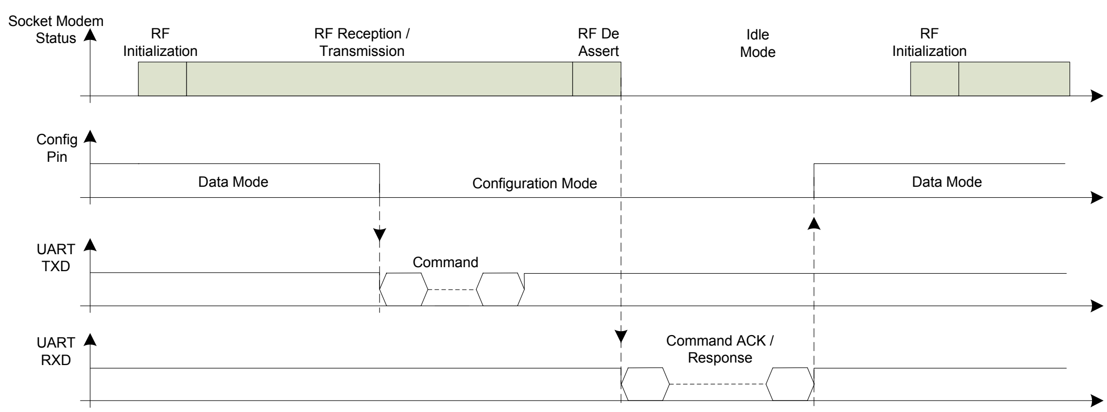
7 MODEM CONFIGURATION

The modem can be configured through four classes of commands and can be related to the OSI Stack.

- APP: Application Layer
- NWK: Network Layer
- DLK: Data Link Layer
- PHY: Physical Layer

7.1 Command Format

To set or read parameters, the modem must first enter "Command Mode" (state in which incoming characters are interpreted as commands). The command can be sent at any time to the radio modem but the response of this command is sent back to the host only during the Idle Mode.



For modified parameter values to persist in the radio modem registry, changes must be saved in Flash memory using SAVE_PARAMETERS Command. Otherwise, parameters are restored to previously saved values when the radio is powered off and then on again.

7.1.1 To Enter in Command Mode

To enter in Command mode two different ways are possible; through the CONFIG/DEFAULT pin or through a UART "Break" condition.

7.1.1.1 Through CONFIG/DEFAULT Pin

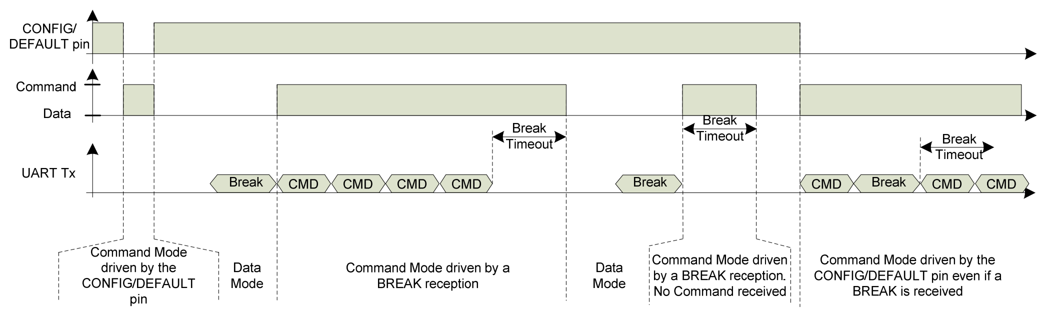
The CONFIG/DEFAULT pin (pin 10) must be driven low to enter in Command Mode and during all the configuration phase.

7.1.1.2 Through a "Break" condition

It is possible to enter in Command mode by sending a "Break" condition on the UART. A break condition is a period of 10 or more low bits received on the UART after a missing stop bit. For example, send a "0x00" at 1200bps to achieve a "Break" condition whatever the UART settings of the radio modem.

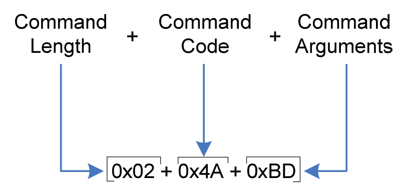
Related commands to the “Break” condition:
- CMD_SET_HOST_BREAK_TIMEOUT
- CMD_GET_HOST_BREAK_TIMEOUT
- CMD_EXIT_HOST_CONFIG_MODE

For more information, please refer to related commands descriptions.



7.1.2 To Send Commands

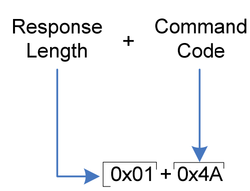
Send commands and parameters using the syntax shown below:



The preceding example would change the RF destination address to “BD” (#189).

7.1.3 Command Acknowledgement

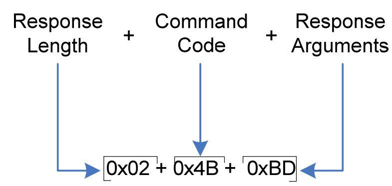
Each SET command sent to the Y-Lynx radio modem is acknowledged by receiving the command code as shown below:



The preceding example confirms the reception by the radio modem of 0x4A Command set (RF destination address).

7.1.4 Command Response

GET Commands requests an answer from the radio modem, the received response uses the syntax shown below:



The preceding example sends back the response to a GET RF_DEST_ADDRESS command (0x4B), the argument is the response to the command, 0xBD (#189) is the RF destination address programmed in the radio modem (the last destination address used by the module).

7.1.5 To Exit from Command Mode

7.1.5.1 Through CONFIG/DEFAULT Pin

To exit from the Command Mode, the CONFIG/DEFAULT pin must be driven high. However, a UART “Break” condition can still force the Command Mode.

7.1.5.2 Through a “Break” condition

The modem is set in Command Mode during a specific time and return after in Data Mode if no command is received by the time (each command reloads the timeout counter). This time is programmable through the following commands:

- CMD_SET_HOST_BREAK_TIMEOUT
- CMD_GET_HOST_BREAK_TIMEOUT

So, the modem returns in Data Mode after the Timeout depending on the status of the CONFIG/DEFAULT pin which can force the Command Mode.

The second way to exit from the Command Mode is to use the specific command:

- CMD_EXIT_HOST_CONFIG_MODE

For more information, please refer to related commands descriptions.

7.2 Command Reference

Table 3.1 Physical layer commands

Command Name	Command	Length*	Range	Description
CMD_SET_RF_POWER	0x60	2 bytes	0x00	Set the output RF power
CMD_GET_RF_POWER	0x61	1 byte	0x03	Get the output RF power
CMD_SET_RF_BITRATE	0x62	2 bytes	0x00	Set the RF data rate
CMD_GET_RF_BITRATE	0x63	1 byte	0x05	Get the RF data rate
CMD_SET_RF_ RECEIVER_MODE	0x6A	2 bytes	0x00 0x01	Set the Receiver in sensitivity or linearity mode
CMD_GET_RF_ RECEIVER_ MODE	0x6B	1 byte		Get the Receiver in sensitivity or linearity mode
CMD_SET_ALL_CHANNEL _TYPE	0x78	14 bytes		Set the type of all channels (receiver, transmitter or idle mode)
CMD_GET_ALL_CHANNEL _TYPE	0x79	1 bytes		Get the type of all channels
CMD_SET_CHANNEL_ TYPE	0x3C	3 bytes	0x00 0x03	Set the channel type (receiver, transmitter or idle mode)
CMD_GET_CHANNEL_ TYPE	0x3D	1 byte		Get the channel type

**Length: includes only command size (1 byte) and parameter range (the request parameter for a SET command and the response parameter for the GET command).*

Table 3.2 Data Link layer commands

Command Name	Command	Length	Range	Description
CMD_SET_ CHANNELS_COUNT	0x20	2 bytes	0x01 0x32	Set the number of used channels
CMD_GET_ CHANNELS_COUNT	0x21	1 byte		Get the number of used channels
CMD_SET_ CHANNEL_DURATION	0x22	2 bytes	0x01 0xFF	Set the duration of a channel
CMD_GET_ CHANNEL_DURATION	0x23	1 byte		Get the duration of a channel
CMD_SET_FREQ_ CORRECTION_MODE	0x28	2 bytes	0x00 0x01	Enable or disable the Frequency correction mode
CMD_GET_FREQ_ CORRECTION_MODE	0x29	1 byte		Get the setting of the Frequency correction mode
CMD_SET_RSSI_MODE	0x2A	2 bytes	0x00 0x03	Enable or disable the Receive Signal Strength Indicator on each channel
CMD_GET_RSSI_MODE	0x2B	1 byte		Get the setting of the Receive Signal Strength Indicator
CMD_SET_ CHANNEL_FREQ	0x2C	4 bytes	0x9A70 0x6590	Set the communication frequency per channel
CMD_GET_ CHANNEL_FREQ	0x2D	2 bytes		Get the communication frequency of one channel
CMD_GET_CHANNELS_ MAX_COUNT	0x2F	1 byte	0x32	Get the maximum of available channels
CMD_GET_LAST_ BEACON_INFO	0x31	2 bytes	-	Gives information on the last received Beacon
CMD_GET_LAST_TX_ FRAME_INFO	0x33	2 bytes	-	Gives information on the last transmitted frame
CMD_GET_LAST_RX_ FRAME_INFO	0x37	2 bytes	-	Gives information on the last received frame
CMD_SET_RF_ ADDRESS	0x44	2 bytes	0x00 0xFE	Set an address to the device
CMD_GET_RF_ ADDRESS	0x45	1 byte		Get the device address
CMD_SET_RF_ DEST_ADDRESS	0x4A	2 bytes	0x00 0xFF	Set a destination address

CMD_GET_RF_ DEST_ADDRESS	0x4B	1 byte		Get the destination address
CMD_SET_RF_ ACK_MODE	0x50	2 bytes	0x00 0x01	Enable/disable the ACK Mode
CMD_GET_RF_ ACK_MODE	0x51	1 byte		Get the status of ACK Mode
CMD_SET_RF_ CHECK_MODE	0x52	2 bytes	0x00 0x03	Set the redundancy check mode
CMD_GET_RF_ CHECK_MODE	0x53	1 byte		Get the status of the redundancy check mode
CMD_SET_BEACON_ MODE	0x54	2 bytes	0x00 0x02	Set the device in server, client or beaconless Mode
CMD_GET_BEACON_ MODE	0x55	1 byte		Get the mode of the device (server, client or beaconless)
CMD_SET_BEACON_ PERIOD	0x56	2 bytes	0x01 0x05	Set the number of sequence between two beacon sent by the server on channel “0”
CMD_GET_BEACON_ PERIOD	0x57	1 byte		Get the setting of the beacon period set in the server
CMD_SET_BEACON_ LISTEN_PERIOD	0x58	2 bytes	0x01 0x0A	Set the number of sequence between two beacon received by the client on channel “0”
CMD_GET_BEACON_ LISTEN_PERIOD	0x59	1 byte		Get the setting of the beacon listen period set in the client
CMD_SET_MAX_ BEACON_LOST	0x5A	2 bytes	0x00 0x0A	Set the number of beacon that a client can lost before to be considered as desynchronized (10 maximum)
CMD_GET_MAX_ BEACON_LOST	0x5B	1 byte		Get the number of beacon that the client can lost before to be considered as desynchronized
CMD_SET_RF_RETRY_ COUNT	0x5C	2 bytes	0x00 0x07	Set the number of retry that can be sent for a given RF packet
CMD_GET_RF_RETRY_ COUNT	0x5D	1 byte		Get the number of retry that can be sent for a given RF packet
CMD_SET_MAX_ BEACON_TRACK	0x5E	2 bytes	0x00 0xFF	Set the number of beacon period to try to resynchronized a client (255 maximum)
CMD_GET_MAX_ BEACON_TRACK	0x5F	1 byte		Get the number of beacon period used to resynchronized a client
CMD_SET_ EXTENDED_ PROTOCOL	0x74	2 bytes	0x00 0x01	Activate the extended protocol
CMD_GET_EXTENDED_ PROTOCOL	0x75	1 byte		Get the protocol option
CMD_SET_TX_ CHANNEL	0xC6	2 bytes	0x00 0x31	Used to set the channel restriction for the transmission
CMD_GET_TX_ CHANNEL	0xC7	1 byte		Get the channel restriction for a transmission
CMD_SET_TX_RETRY_ RESTRICTION	0xC8	2 bytes	0x00 0x01	Used to set the retries channel restriction for the transmission
CMD_GET_TX_RETRY_ RESTRICTION	0xC9	1 byte		Get the channel retries restriction for a transmission
RF_FRAME_MAX_SIZE	0xCB	1 byte	0x00 0x80	Get the absolute maximum size of a RF frame
CMD_SET_USER_RF_ FRAME_MAX_SIZE	0xCE	2 bytes	0x00 0x80	Used to set the maximum size of the RF frame defined by the user
CMD_GET_USER_RF_ FRAME_MAX_SIZE	0xCF	1 byte		Used to read the maximum size value of the RF frame defined by the user

Table 3.3 Network layer commands

Command Name	Command	Length	Range	Description
CMD_SET_NWK_ID	0x42	2 bytes	0x00 0x3F	Set an id to the network
CMD_GET_NWK_ID	0x43	1 byte		Get the id of the network
CMD_SET_RF_ MULTICAST	0x4C	2 bytes	0x00 0xFF	Register a node to one or more group
CMD_GET_RF_ MULTICAST	0x4D	1 byte		Get the registered group of the node
CMD_SET_RF_DEST MULTICAST	0x4E	2 bytes	0x00 0xFF	Define the destination multicast
CMD_GET_RF_DEST MULTICAST	0x4F	1 byte		Get the destination multicast

Table 3.4 Application commands

Command Name	Command	Length	Range	Description
CMD_GET_VERSION	0x10	1 byte	-	Get the version of the module
CMD_GET_LIBIC_ VERSION	0x11	1 byte	-	Get the LibIC version
CMD_SET_APPL_ID	0x40	2 bytes	0x00 0xFF	Set an id to the application
CMD_GET_APPL_ID	0x41	1 byte		Get the id of the application
CMD_SET_REMOTE_ CMD_MODE	0x76	2 bytes	0x00 0x01	Configure the module to accept remote command
CMD_GET_REMOTE_ CMD_MODE	0x77	1 byte		Get the command mode setting of the module
CMD_SET_CIPHER_MODE	0xA2	2 bytes	0x00 0x01	Set the module in cipher mode
CMD_GET_CIPHER_MODE	0xA3	1 byte		Get the cipher mode setting
CMD_SET_CIPHER_KEY	0xA4	5 bytes	0x00 0xFF	Set the cipher code (key)
CMD_GET_CIPHER_KEY	0xA5	1 byte		Get the cipher code
CMD_SET_HOST_ BYTE_TIMEOUT	0xC0	2 bytes	0x00 0x23	Configure the timeout between two bytes, used to determine the “end of frame”
CMD_GET_HOST_ BYTE_TIMEOUT	0xC1	1 byte		Get the programmed byte timeout value
CMD_SET_HOST_ BREAK_TIMEOUT	0xC2	2 bytes	0x00 0xFF	Configure the timeout after which a break condition is exited
CMD_GET_HOST_ BREAK_TIMEOUT	0xC3	1 byte		Get the timeout after which a break condition is exited
CMD_SET_HOST_ BAUDRATE	0xC4	2 bytes	0x00 0x07	Configure the baud rate of the UART interface with the host
CMD_GET_HOST_ BAUDRATE	0xC5	1 byte		Get the programmed baud rate
CMD_EXIT_HOST_ CONFIG_MODE	0xCA	1 byte	-	Exit configuration mode
CMD_SET_HOST_UART_ SETTINGS	0xCC	2 bytes	0x00 0x1F	Configure the UART parameters
CMD_GET_HOST_ UART_ SETTINGS	0xCD	1 byte		Get the UART parameters
CMD_SET_HOST_DATA_ MODE	0xD0	2 bytes	TX 0x00 0x3F RX 0x00 0xFF	Used to set the configuration of the data mode
CMD_GET_HOST_DATA_ MODE	0xD1	1 byte		Used to read the data mode configuration
CMD_SEND_DATA	0xD2	n bytes	-	Send data in configuration mode
IND_RECEIVED_DATA	0xD3	n bytes	-	Used to receive data in configuration mode
CMD_SEND_ECHO_ DATA	0xD4	n bytes	-	Allow to send an RF data frame to a hostless modem which will echo it back
CMD_SET_POWER_ MODE	0xD6	2 bytes	0x00 0x02	Used to set the power mode of the radio modem.

CMD_GET_POWER_ MODE	0xD7	1 byte		Used to read the power mode of the radio modem
CMD_SEND_COMMAND	0xD8	n bytes	-	Send a command to a remote wireless module
IND_RECEIVED_ RESPONSE	0xD9	n bytes	-	Used to receive the response to the sent command from the remote wireless module
CMD_SET_CTS_ THRESHOLD	0xDA	2 bytes	0x00 0xFF	Used to fine tune the CTS signal behavior.
CMD_GET_CTS_ THRESHOLD	0xDB	1 byte		Used to read the settings of CTS threshold
CMD_SET_HOST_DATA_ END_CONDITION	0xDC	2 bytes	0x00 0x03	Used to set the RF frame end condition
CMD_GET_HOST_DATA_ END_CONDITION	0xDD	1 byte		Reads the host data frame end condition
CMD_SET_RESET_MODE	0xDE	2 bytes	0x00 0x03	Determine which configuration has to be loaded after the reset and on which conditions
CMD_GET_RESET_MODE	0xDF	1 byte		Get the configuration of the reset mode
CMD_SET_GP1_SIGNAL	0xE0	2 bytes	0x00 0x04	Map a signal on GP1
CMD_GET_GP1_SIGNAL	0xE1	1 byte		Get the mapping on the GP1
CMD_SET_GP1_VALUE	0xE2	2 bytes	0x00 0x01	Write a signal level on GP1 when configured as output
CMD_GET_GP1_VALUE	0xE3	1 byte		Read the signal level on GP1 when configured as input
CMD_SET_GP2_SIGNAL	0xE4	2 bytes	0x00 0x05	Map a signal on GP2
CMD_GET_GP2_SIGNAL	0xE5	1 byte		Get the mapping on the GP2
CMD_SET_GP2_VALUE	0xE6	2 bytes	0x00 0x01	Write a signal level on GP2 when configured as output
CMD_GET_GP2_VALUE	0xE7	1 byte		Read the signal level on GP2 when configured as input
CMD_SET_USER_DATA	0xF0	5 bytes	-	Allow to save the user version number
CMD_GET_USER_DATA	0xF1	1 byte		Get the user version number
CMD_SAVE_ PARAMETERS	0xF2	2 bytes	0x00 0x08	Save the current configuration in Flash memory
CMD_LOAD_ PARAMETERS	0xF4	2 bytes	0x00 0x08	Load the Flash memory in RAM
CMD_LOAD_DEFAULT_ PARAMETERS	0xF6	2 bytes	0x00 0x08	Load the ROM memory in RAM

Table 3.5 and 3.6 Error command and Error codes

Command Name	Command	Length	Range	Description
IND_ERROR	0x03	2 bytes	-	Sent in response to a request when an error occurred

Error Code Name	Code	Description
ERR_FORMAT_CMD	0x02	The command does not respect the protocol
ERR_INVALID_ARGUMENT	0x03	An argument is invalid
ERR_OUTOFRANGE_ARGUMENT	0x04	An argument is out of range
ERR_TIMEOUT	0x05	Byte timeout occurred before complete command reception
ERR_CONTEXT	0x06	The command can not be executed with the current parameters
ERR_UNDEFINED_CMD	0x07	The module does not know how to interpret the given command

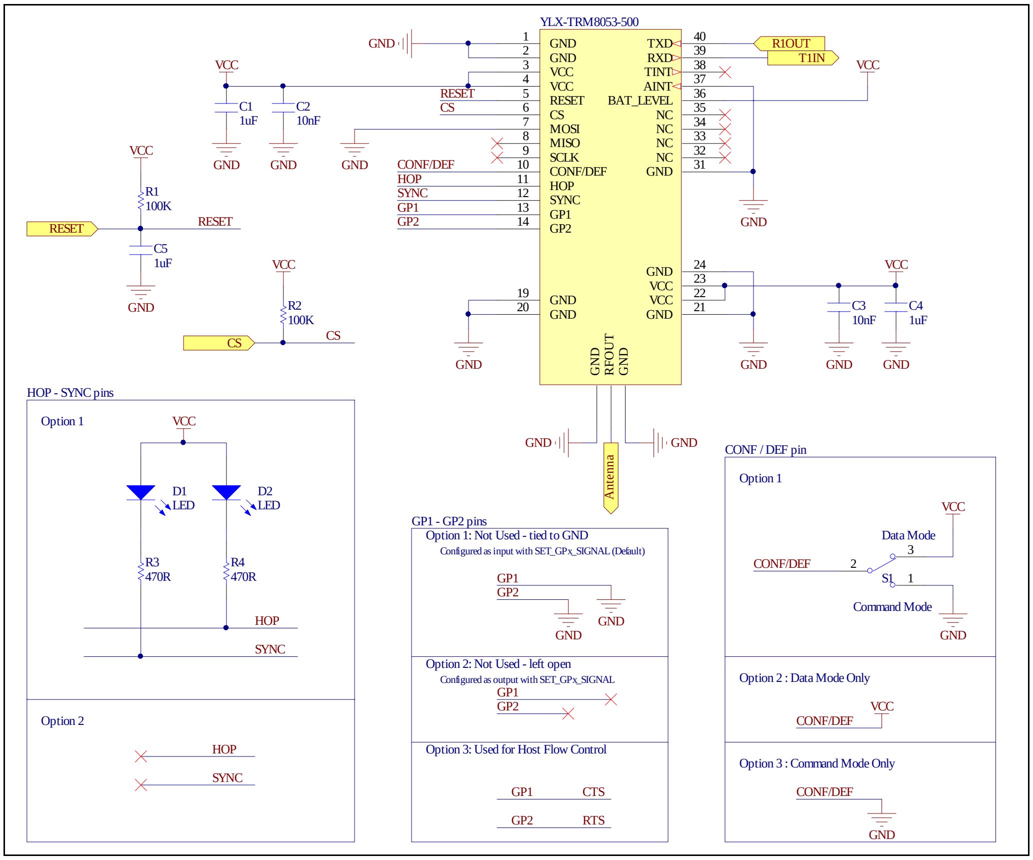
7.3 Command Description

A description of each command can be found in the LibIC5305 - Radio Communication Command Reference Guide.

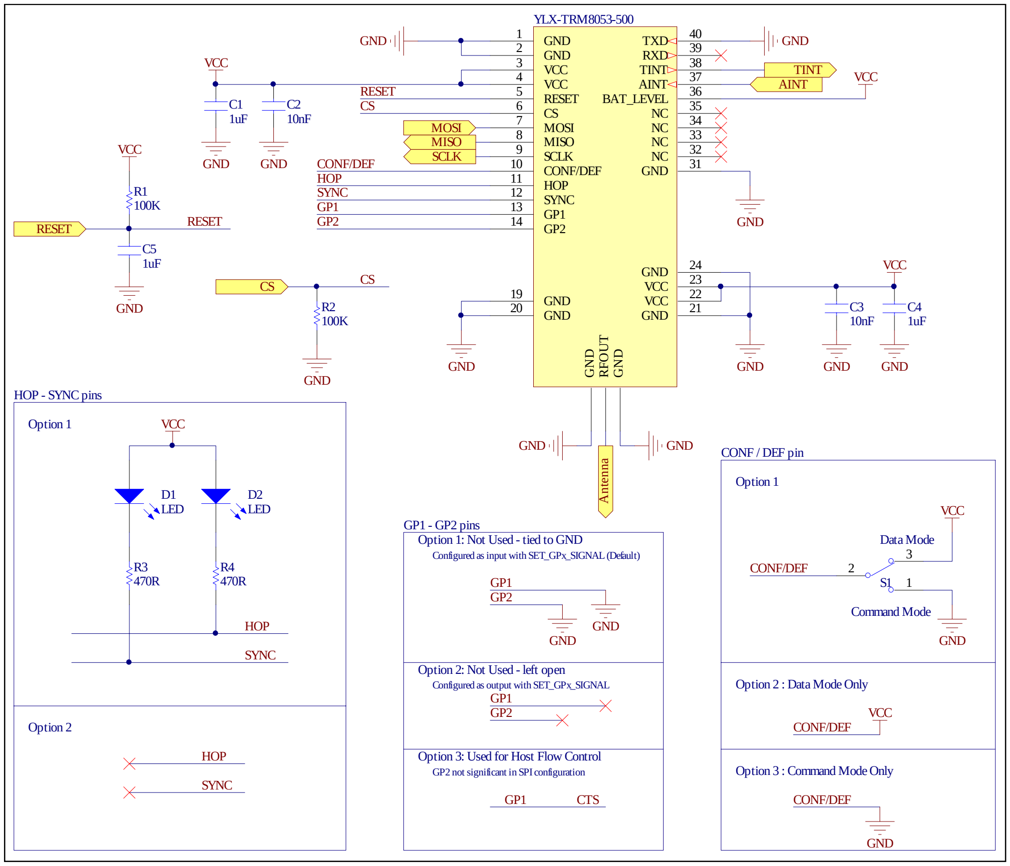
8 APPLICATION INFORMATION

8.1 Typical Applications

8.1.1 UART Interface



8.1.2 SPI Interface



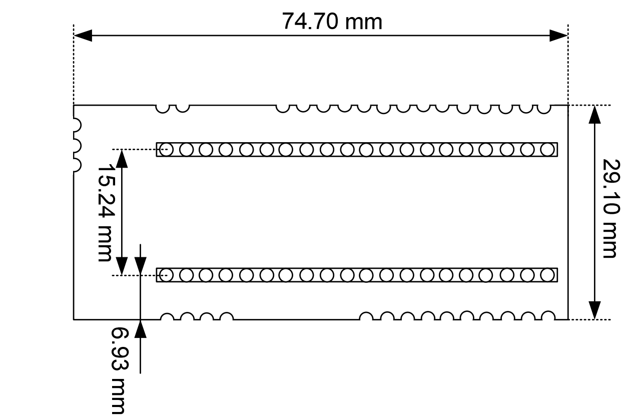
8.1.3 Typical Connection of pins

The correct termination of pins is listed in following table.

Pin name	UART Configuration	SPI Configuration	Comment
RESET	VCC	VCC	RC for power-on-reset
CS	VCC	GND	VCC for UART / GND for SPI
MOSI	GND	MOSI	
MISO	Open	MISO	Internal pull-up
SCLK	Open	SCLK	Internal pull-up
CONFIG/DEFAULT	VCC or GND	VCC or GND	VCC for Data / GND for Command c.f. Power On Reset Configuration
HOP	Open	Open	Modem Output (Option 2)
SYNC	Open	Open	Modem Output (Option 2)
GP1	GND	GND	Configured by default as input (Option 1)
GP2	GND	GND	Configured by default as input (Option 1)
BAT_LEVEL	VCC	VCC	
AINT	GND	AINT	
TINT	Open	Open	
RXD	RXD	Open	
TXD	TXD	GND	

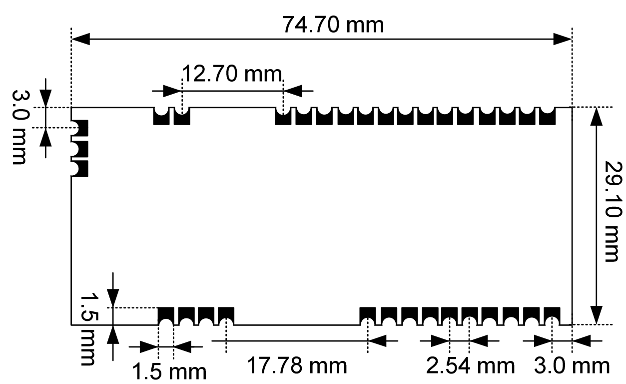
8.2 Packaging Information

8.2.1 40 pins Socket Version (Mechanical drawing)



View from the top

8.2.2 Drop'in Version (Footprint)



View from the top

Documentation History

Revision	Modifications	Date
Rev 1.6	Minor modifications	Jan 2011
Rev 1.5	New functionalities available - External Synchronization - Channel configuration - Remote commands - Encryption - Reset functionality	Oct 2010
Rev 1.4	- Reset schematic - Packaging Information	Jan 2009

Rev 1.3	Commands description split in a new documentation: LibIC5305 Reference Guide
	Rx, Tx buffers
	- End of RF frame
	- Size of RF frame (with Pload)
	- Power Save Mode (Add Standby mode)
	- Commands modification/up-date
	  - HOST_DATA_MODE_TX
	  - LAST_FRAME_RX_INFO
	  - LAST_FRAME_TX_INFO
		April 2008

Rev 1.2	New functionalities available - Host UART Settings - RTS – CTS pins assignment - CTS_THESHOLD command - HOST_FRAME_MAX_SIZE command Add Application Information chapter	November, 2007
Rev 1.1	Errata on UART pins assignment	July, 2007
Rev 1.0	Original version	December, 2006

Related Products and Documents

Related Documentations

Description	Type of document
LibIC5305 – Radio Communication Controller	Reference Guide
Errata Sheet	See www.y-lynx.com for details

YLX-TRM8053-500-05 Radio Modem

Description	Ordering Number
YLX-TRM8053-500-05 40 pins Socket Modem	YLX-TRM8053-500-05SIT
YLX-TRM8053-500-05 Drop'in Modem	YLX-TRM8053-500-05DIT

Development Kit

Description	Ordering Number
Radio Modem Starter Kit for YLX-TRM8053-500-05	RMSK-TRM8053-500-05

Contact Information

Address:

Y-Lynx Sarl
Rue de Galilée15
CH - 1400 YVERDON
SWITZERLAND

Phone: +41 24 423 92 05
Email: info@y-lynx.com
Web site: http://www.y-lynx.com

