



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



## Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China





**HumPRO™ Series 900MHz  
RF Transceiver Module  
Data Guide**

**Wireless made simple®**

**Warning:** Some customers may want Linx radio frequency (“RF”) products to control machinery or devices remotely, including machinery or devices that can cause death, bodily injuries, and/or property damage if improperly or inadvertently triggered, particularly in industrial settings or other applications implicating life-safety concerns (“Life and Property Safety Situations”).

**NO OEM LINX REMOTE CONTROL OR FUNCTION MODULE SHOULD EVER BE USED IN LIFE AND PROPERTY SAFETY SITUATIONS.** No OEM Linx Remote Control or Function Module should be modified for Life and Property Safety Situations. Such modification cannot provide sufficient safety and will void the product’s regulatory certification and warranty.

Customers may use our (non-Function) Modules, Antenna and Connectors as part of other systems in Life Safety Situations, but only with necessary and industry appropriate redundancies and in compliance with applicable safety standards, including without limitation, ANSI and NFPA standards. It is solely the responsibility of any Linx customer who uses one or more of these products to incorporate appropriate redundancies and safety standards for the Life and Property Safety Situation application.

**Do not use this or any Linx product to trigger an action directly from the data line or RSSI lines without a protocol or encoder/decoder to validate the data.** Without validation, any signal from another unrelated transmitter in the environment received by the module could inadvertently trigger the action.

**All RF products are susceptible to RF interference that can prevent communication.** RF products without frequency agility or hopping implemented are more subject to interference. This module does have a frequency hopping protocol built in, but the developer should still be aware of the risk of interference.

**Do not use any Linx product over the limits in this data guide.** Excessive voltage or extended operation at the maximum voltage could cause product failure. Exceeding the reflow temperature profile could cause product failure which is not immediately evident.

**Do not make any physical or electrical modifications to any Linx product.** This will void the warranty and regulatory and UL certifications and may cause product failure which is not immediately evident.

# Table of Contents

1	Description
1	Features
2	Ordering Information
2	Absolute Maximum Ratings
3	Electrical Specifications
5	Typical Performance Graphs
10	Pin Assignments
10	Pin Descriptions
12	Pre-Certified Module Pin Assignments
13	Module Dimensions
14	Theory of Operation
15	Module Description
16	Overview
18	Addressing Modes
20	Automatic Addressing
20	Address Register Use
21	Acknowledgements and Assured Delivery
22	Frequency Hopping Spread Spectrum
23	Compatibility with the 250 Series
23	Networking
24	Transmitting Packets
25	Receiving Packets
29	Using the Buffer Empty (BE) Line
30	Exception Engine
32	Carrier Sense Multiple Access (CSMA)
33	Using the Command Response ( $\overline{\text{CRESP}}$ ) Line
34	Using the $\overline{\text{CMD}}$ Line
34	External Amplifier Control
35	AES Encryption

# HumPRO™ Series 900MHz RF Transceiver Module Data Guide



## Description

The HumPRO™ Series is a frequency hopping spread spectrum (FHSS) transceiver designed for the reliable transfer of digital data. It has a very fast lock time so that it can quickly wake up, send data and go back to sleep, saving power in battery-powered applications. The module is available in the 915MHz frequency band.

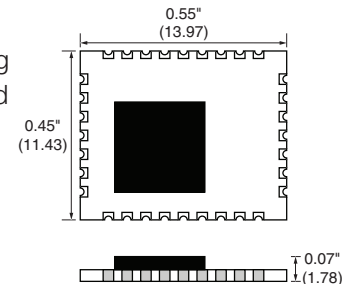


Figure 1: Package Dimensions

The module has several features that increase the data transfer reliability. It ensures that no other modules are transmitting before it begins transmitting data. Automatic acknowledgements ensure that the remote side received valid data. Multiple hopping patterns enable several systems to operate in proximity without interference. A standard UART interface is used for module configuration and data transfer. A few simple serial commands are all that are needed for configuration.

All modules have a unique 32-bit serial number that can be used as an address. Source and destination addressing support point-to-point and broadcast links. Address masking by the receiving module allows for creating subnets. Other network topologies can also be implemented.

Housed in a tiny compact reflow-compatible SMD package, the transceiver requires no external RF components except an antenna, which greatly simplifies integration and lowers assembly costs. Versions are available that have obtained FCC and Industry Canada modular certification.

## Features

- FHSS Algorithm
- Fast Lock (<30ms at 115kbps)
- Low power modes
- FCC and IC Pre-certified version
- Simple UART interface
- No external RF components required
- No production tuning required
- Tiny PLCC-32 footprint

- 38 [Using the MODE\\_IND Line](#)
- 39 [Using the PB Line](#)
- 40 [Restore Factory Defaults](#)
- 40 [Using the Low Power Features](#)
- 42 [The Command Data Interface](#)
- 43 [Reading from Registers](#)
- 44 [Writing to Registers](#)
- 45 [Command Length Optimization](#)
- 45 [Example Code for Encoding Read/Write Commands](#)
- 48 [The Command Data Interface Command Set](#)
- 95 [Typical Applications](#)
- 96 [Usage Guidelines for FCC Compliance](#)
- 96 [Additional Testing Requirements](#)
- 97 [Information to the user](#)
- 98 [Product Labeling](#)
- 98 [FCC RF Exposure Statement](#)
- 98 [Antenna Selection](#)
- 100 [Castellation Version Reference Design](#)
- 102 [Power Supply Requirements](#)
- 102 [Antenna Considerations](#)
- 103 [Interference Considerations](#)
- 104 [Pad Layout](#)
- 105 [Microstrip Details](#)
- 106 [Board Layout Guidelines](#)
- 107 [Helpful Application Notes from Linx](#)
- 108 [Production Guidelines](#)
- 108 [Hand Assembly](#)
- 108 [Automated Assembly](#)
- 110 [General Antenna Rules](#)
- 112 [Common Antenna Styles](#)
- 114 [Regulatory Considerations](#)

## Ordering Information

Ordering Information	
Part Number	Description
HUM-900-PRO	HumPRO™ Series Data Transceiver
HUM-900-PRO-CAS	HumPRO™ Series Data Transceiver with Castellation Connection
HUM-900-PRO-UFL	HumPRO™ Series Data Transceiver with u.FL Connector
EVM-900-PRO	HumPRO™ Series Carrier Board
EVM-900-PRO-CAS	HumPRO™ Series Carrier Board with Certified module, Castellation Connection
EVM-900-PRO-UFL	HumPRO™ Series Carrier Board with Certified module, UFL Connector
MDEV-900-PRO	HumPRO™ Series Master Development System

Figure 2: Ordering Information

## Absolute Maximum Ratings

Absolute Maximum Ratings				
Supply Voltage $V_{CC}$	-0.3	to	+3.9	VDC
Any Input or Output Pin	-0.3	to	$V_{CC} + 0.3$	VDC
RF Input		0		dBm
Operating Temperature	-40	to	+85	°C
Storage Temperature	-40	to	+85	°C

Exceeding any of the limits of this section may lead to permanent damage to the device. Furthermore, extended operation at these maximum ratings may reduce the life of this device.

Figure 3: Absolute Maximum Ratings



**Warning:** This product incorporates numerous static-sensitive components. Always wear an ESD wrist strap and observe proper ESD handling procedures when working with this device. Failure to observe this precaution may result in module damage or failure.

## Electrical Specifications

HumPRO™ Series Transceiver Specifications						
Parameter	Symbol	Min.	Typ.	Max.	Units	Notes
Power Supply						
Operating Voltage	$V_{CC}$	2.0		3.6	VDC	
TX Supply Current	$I_{CCTX}$					
900MHz at +10dBm			40.5	41.5	mA	1,2
900MHz at 0dBm			22	24	mA	1,2
RX Supply Current	$I_{CCRX}$		23.5	24.5	mA	1,2,3
Power-Down Current	$I_{PDN}$		0.7	1.4	μA	1,2
RF Section						
Operating Frequency Band	$F_C$	902		928	MHz	
Number of hop channels						
@ 19.2kbps RF Rate			50/64			
@ 152.34kbps RF Rate			26/32			
Channel spacing						
@ 19.2kbps RF Rate			375.9		kHz	
@ 152.34kbps RF Rate			751.81		kHz	
20 dB OBW						
@ 19.2kbps RF Rate			64		kHz	
@ 152.34kbps RF Rate			315		kHz	
Receiver BW						
@ 19.2kbps RF Rate			102		kHz	
@ 152.34kbps RF Rate			232		kHz	
FSK deviation						
@ 19.2kbps RF Rate			± 19.2		kHz	
@ 152.34kbps RF Rate			± 51		kHz	
Scan time / channel (avg)						
@ 19.2kbps RF Rate			1.2		ms	
@ 152.34kbps RF Rate			0.335		ms	
FHSS Lock time						
@ 19.2kbps RF Rate			63		ms	
@ 152.34kbps RF Rate			26		ms	
Modulation			2FSK			
Data Encoding			6/7 RLL			
Number of Hop Sequences			6			

HumPRO™ Series Transceiver Specifications						
Parameter	Symbol	Min.	Typ.	Max.	Units	Notes
Receiver Section						
Spurious Emissions				-47	dBm	
IF Frequency			304.7		kHz	
Receiver Sensitivity						5
@min rate		-98	-101		dBm	5
@max rate		-91	-94		dBm	5
RSSI Dynamic Range			85		dB	
CSMA RSSI Threshold			-70		dBm	
Transmitter Section						
Max Output Power	P <sub>O</sub>	+8.5	+9.5		dBm	6
Harmonic Emissions	P <sub>H</sub>		-41		dBc	6
Output Power Range	P <sub>H</sub>	-5		9	dB	6
Antenna Port						
RF Impedance	R <sub>IN</sub>		50		Ω	4
Environmental						
Operating Temp. Range		-40		+85	°C	4
Timing						
Module Turn-On Time						
Via V <sub>CC</sub>		63		173	ms	4
Via $\overline{\text{POWER\_DOWN}}$			4		ms	4
Via Standby			4		ms	4
Serial Command Response						
Volatile R/W			0.4	5	ms	8
NV Update			2.4	31.5	ms	8
Factory Reset		204		329	ms	14
Channel Dwell Time				400	ms	
CMD low to trigger TX with option TXnCMD	t <sub>TXnCMD</sub>	2			ms	13
Interface Section						
UART Data rate		9,600		115,200	bps	
Input						
Logic Low	V <sub>IL</sub>			0.3*V <sub>CC</sub>	VDC	
Logic High	V <sub>IH</sub>	0.7*V <sub>CC</sub>			VDC	

HumPRO™ Series Transceiver Specifications						
Parameter	Symbol	Min.	Typ.	Max.	Units	Notes
Output						
Logic Low, MODE_IND, BE	V <sub>OLM</sub>			0.3*V <sub>CC</sub>	VDC	1,9
Logic High, MODE_IND, BE	V <sub>OHM</sub>	0.7*V <sub>CC</sub>			VDC	1,9
Logic Low	V <sub>OL</sub>			0.3*V <sub>CC</sub>		1,10
Logic High	V <sub>OH</sub>	0.7*V <sub>CC</sub>				1,10
CRESP Hold Time		10			Bits	11
Flash (Non-Volatile) Memory Specifications						
Flash Write Cycles		22,000			cycles	12
1.	Measured at 3.3V V <sub>CC</sub>					9. 60mA source/sink
2.	Measured at 25°C					10. 6mA source/sink
3.	Input power < -60dBm					11. End of CMD_DATA_OUT stop bit to change in $\overline{\text{CRESP}}$
4.	Characterized but not tested					12. Number of register write operations
5.	PER = 1%					13. With CSMA disabled
6.	Into a 50-ohm load					14. Start of factory reset command to end of last ACK response
7.	No RF interference					
8.	From end of command to start of response					

Figure 4: Electrical Specifications

## Typical Performance Graphs

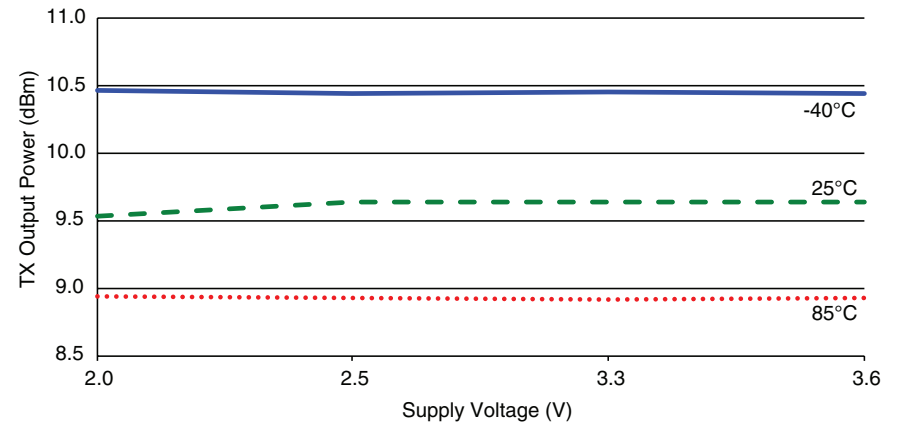


Figure 5: HumPRO™ Series Transceiver Max Output Power vs. Supply Voltage - HUM-900-PRO

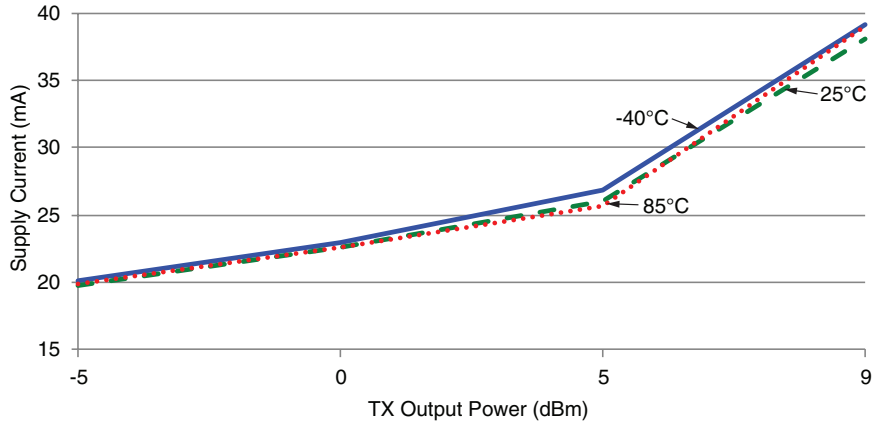


Figure 6: HumPRO™ Series Transceiver Average Current vs. Transmitter Output Power at 2.5V - HUM-900-PRO

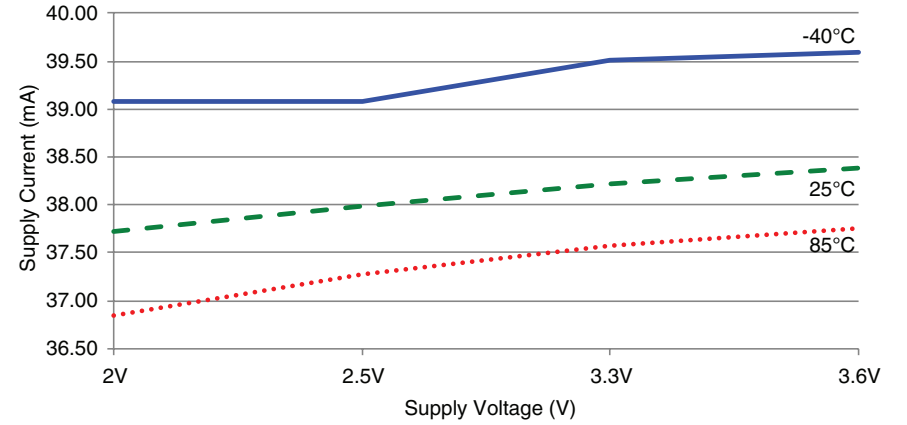


Figure 9: HumPRO™ Series Transceiver TX Current vs. Supply Voltage at Max Power - HUM-900-PRO

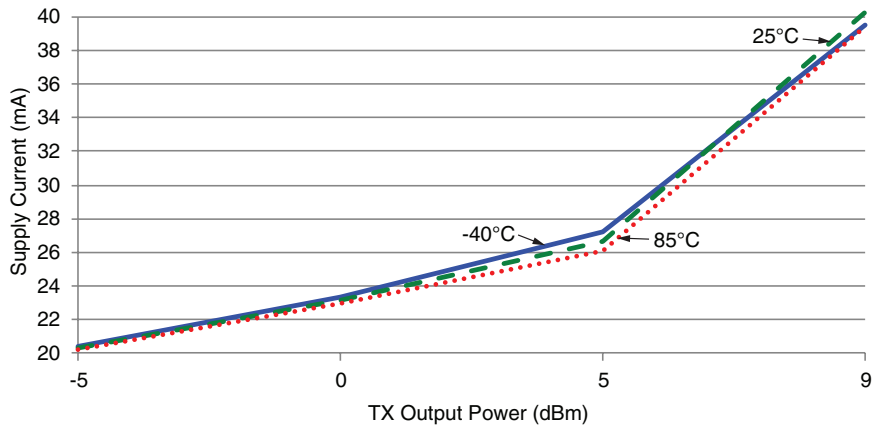


Figure 7: HumPRO™ Series Transceiver Average TX Current vs. Transmitter Output Power at 3.3V - HUM-900-PRO

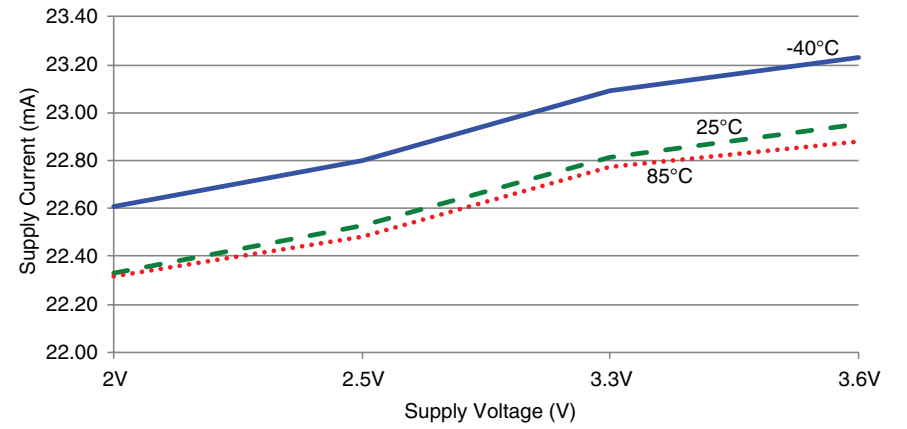


Figure 8: HumPRO™ Series Transceiver TX Current vs. Supply Voltage at OdBm - HUM-900-PRO

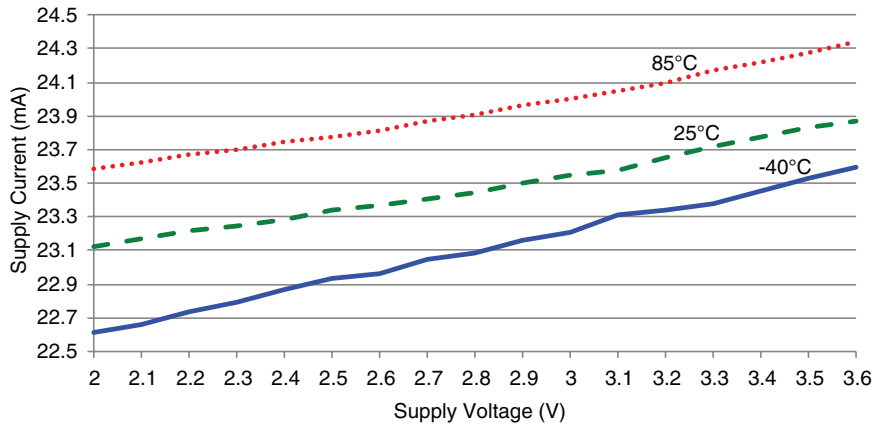


Figure 10: HumPRO™ Series Transceiver RX Scan Current vs. Supply Voltage, 9.6kbps - HUM-900-PRO

Current consumption while the module is scanning for a transmission. The current is approximately 0.5mA higher when receiving data at 9.6kbps.

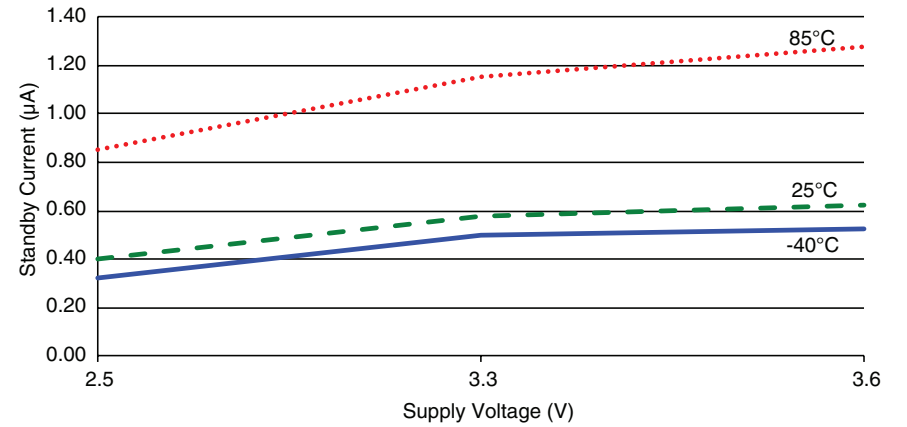


Figure 12: HumPRO™ Series Transceiver Standby Current Consumption vs. Supply Voltage - HUM-900-PRO

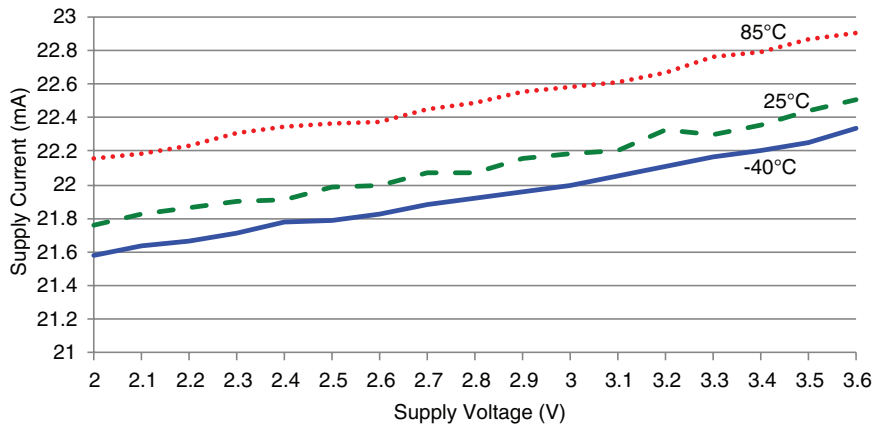


Figure 11: HumPRO™ Series Transceiver RX Scan Current vs. Supply Voltage, 115.2kbps - HUM-900-PRO

Current consumption while the module is scanning for a transmission. The current is approximately 2mA higher when receiving data at 115.2kbps.

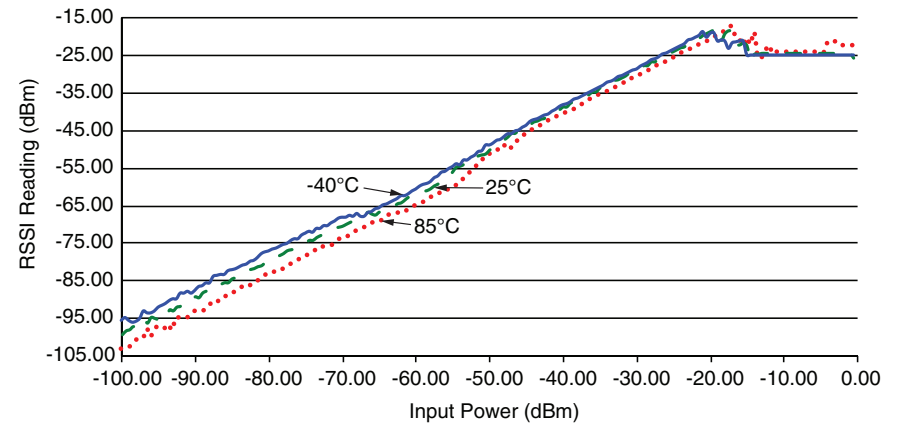


Figure 13: HumPRO™ Series Transceiver RSSI Voltage vs. Input Power - HUM-900-PRO



## Pin Assignments

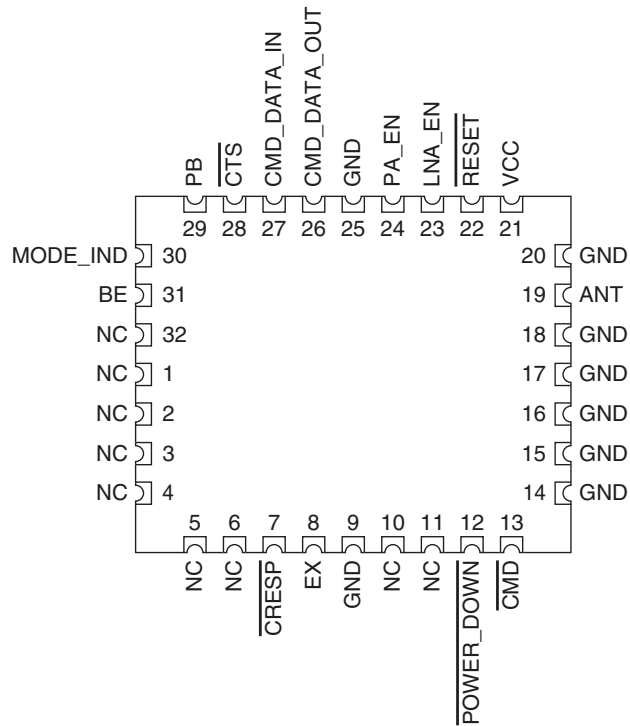


Figure 14: HumPRO™ Series Transceiver Pin Assignments (Top View)

## Pin Descriptions

Pin Descriptions			
Pin Number	Name	I/O	Description
1, 2, 3, 4, 5, 6, 10, 11, 32	NC	—	No Electrical Connection. Do not connect any traces to these lines.
7	$\overline{\text{CRESP}}$	O	Command Response. This line is low when the data on the CMD_DATA_OUT line is a response to a command and not data received over the air.
8	EX	O	Exception Output. A mask can be set to take this line high when an exception occurs.
9, 14, 15, 16, 17, 18, 20, 25	GND	—	Ground
12	$\overline{\text{POWER\_DOWN}}$	I	Power Down. Pulling this line low places the module into a low-power state. The module is not functional in this state. Pull high for normal operation. Do not leave floating.

Pin Descriptions			
Pin Number	Name	I/O	Description
13	$\overline{\text{CMD}}$	I	Command Input. When this line is low, incoming bytes are command data. When high, incoming bytes are data to be transmitted.
19	ANTENNA	—	50-ohm RF Antenna Port
21	VCC	—	Supply Voltage
22	$\overline{\text{RESET}}$	I	This line resets the module when pulled low. It should be high for normal operation. This line has an internal 10k resistor to supply, so leave it unconnected if not used.
23	LNA_EN	O	Low Noise Amplifier Enable. This line is driven high when receiving. It is intended to activate an optional external LNA.
24	PA_EN	O	Power Amplifier Enable. This line is driven high when transmitting. It is intended to activate an optional external power amplifier.
26	CMD_DATA_OUT	O	Command Data Out. Output line for data and serial commands
27	CMD_DATA_IN	I	Command Data In. Input line for data ( $\overline{\text{CMD}}$ is high) and serial commands ( $\overline{\text{CMD}}$ is low).
28	$\overline{\text{CTS}}$	O	UART Clear To Send, active low. This line indicates to the host microcontroller when the module is ready to accept data. When $\overline{\text{CTS}}$ is high, the module is busy. When $\overline{\text{CTS}}$ is low, the module is ready for data.
29	PB	I	Push Button input. This line can be connected to Vcc through a normally open push button. Button sequences can reset configurations to default and join modules into a network. <b>Pull low when not in use; do not leave floating.</b>
30	MODE_IND	O	Mode Indicator. This line indicates module activity. It can source enough current to drive a small LED, causing it to flash. The duration of the flashes indicates the module's current state.
31	BE	O	Buffer Empty. This line is high when the UART input buffer is empty, indicating that all data has been transmitted. If acknowledgment is active, it also indicates that the receiving module has acknowledged the data or a retry exception has occurred.

Figure 15: HumPRO™ Series Transceiver Pin Descriptions

## Pre-Certified Module Pin Assignments

The pre-certified version of the module has mostly the same pin assignments as the standard version. The antenna connection is routed to either a castellation (-CAS) or a u.FL connector (-UFL), depending on the part number ordered.

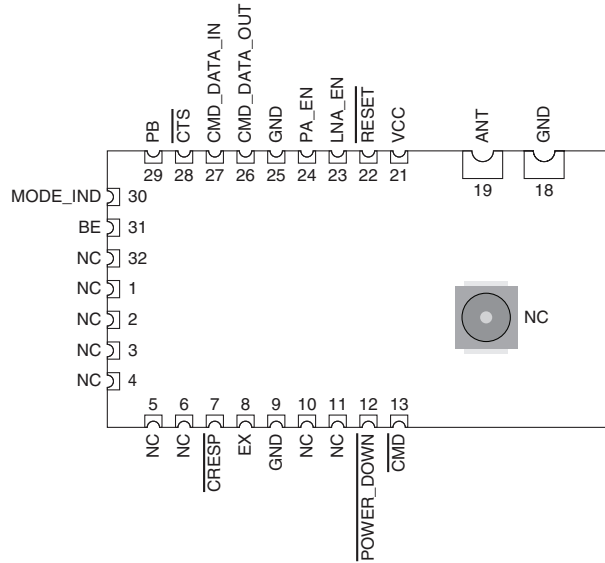


Figure 16: HumPRO™ Series Transceiver Pre-certified Version Pin Assignments - Castellation Connection (Top View)

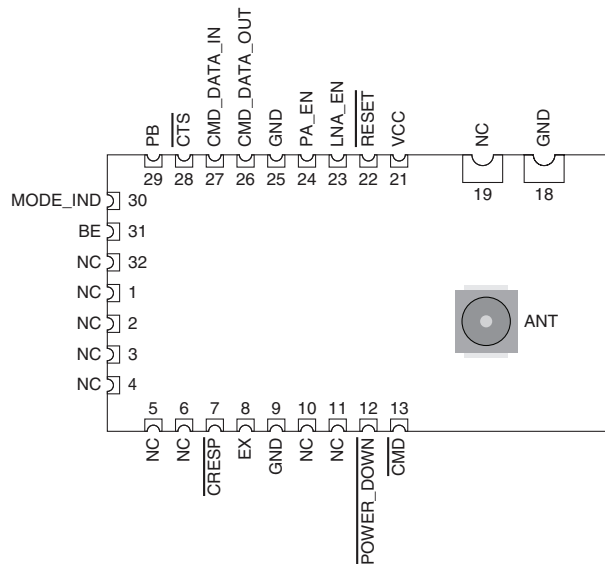


Figure 17: HumPRO™ Series Transceiver Pre-certified Version Pin Assignments - UFL Connection (Top View)

## Module Dimensions

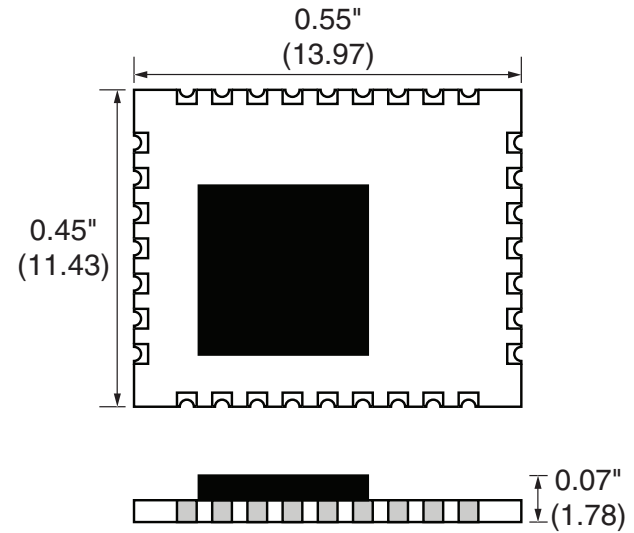


Figure 18: HumPRO™ Series Transceiver Dimensions

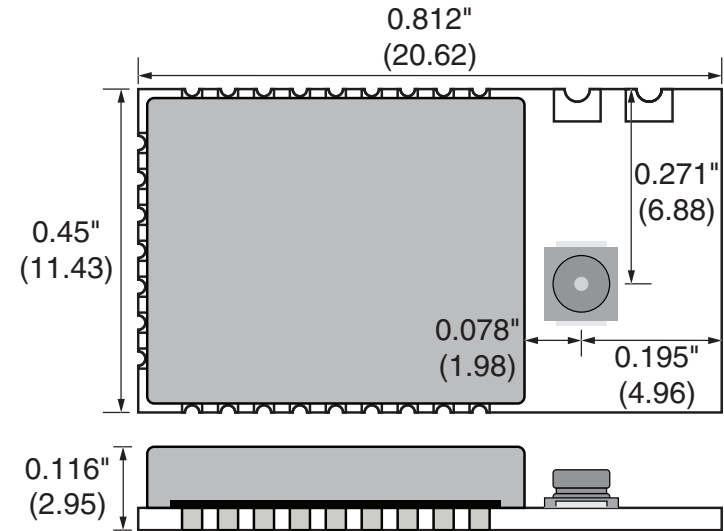


Figure 19: HumPRO™ Series Transceiver Pre-certified Version Dimensions



## Overview

The HumPRO™ Series RF transceiver module offers a number of features that make it suitable for many data transfer applications. This section provides a basic overview of the features while following sections dive into them in more detail.

## Addressing

The modules have a very powerful addressing method. Each module is given a unique 16 or 32 bit address. The receiving modules use an address mask that determines how it responds to a received transmission.

The addressing and masking allow for the creation of point-to-point, many-to-one and one-to-many wireless links. This allows the creation of many network topologies, such as star, tree and mesh. The routing for the network topology is managed outside the module.

The addressing is the primary configuration when getting started with the modules. RG-00105, the HumPRO™ Addressing Mode Reference Guide has details about configuring the addressing.

## Acknowledgements and Assured Delivery

The modules support assured delivery in the form of acknowledgements and retries. When the acknowledgements are enabled, the receiving device sends an acknowledge message to let the sender know that the transmission was received. If the sender does not get an acknowledgement it resends the message up to a configurable number of retries. If there is still no acknowledgement, the module triggers an exception to let the host processor know of the error.

## Command Mode and Data Mode

The module has two main interface modes controlled by the state of the  $\overline{\text{CMD}}$  line. Command mode routes the data coming in on the  $\text{CMD\_DATA\_IN}$  line to the processor for configuring the module. Data mode routes the data to the transmitter for transmission over-the-air. The  $\overline{\text{CMD}}$  line is normally controlled by an external microcontroller.

## Encryption

The module supports AES-128 encryption to provide a secure wireless link. All of the modules must have encryption enabled and be using the same key in order for communication to be successful. There are two ways of entering an encryption key: directly by writing the key to registers through the Command Data Interface or through a JOIN process.

## Streaming Data and Explicit Packets

The module's default configuration is for streaming data. At some UART rates the module sends the data at a higher rate over-the-air than it is input on the UART. This hides the time required for the protocol transactions and the frequency hopping. The result is that the data appears to stream through the module with no breaks in the data apparent to the host processor.

Alternatively, the module can be configured for explicit packet transmission. This allows the host processor to control when packets are sent and what data is in each packet

## Exceptions and Host Processor Interface

The module has several indicator lines that provide feedback to the host processor on the module's operation and current status. This includes an exception line (EX) that informs the processor when errors occur so that it can take steps to manage the issue gracefully. The state of the status lines can also be read through the module's Command Data Interface to reduce the number of hardware connections that are required.

## Command Data Interface

The module has a Command Data Interface that consists of a set of serial commands entered through a UART. These are shorter and simpler than AT commands that are popular with many modules. These commands control the configuration of the module as well as allow feedback on the operation and status of the module.

## Carrier Sense Multiple Access (CSMA)

The module implements a Carrier Sense Multiple Access method. It listens to the channel and makes sure that it is clear before it transmits. If the channel is in use, the module either waits for it to clear or hops to the next channel depending on its current state. This reduces the overall potential for interference and improves the robustness of the link.

## Addressing Modes

The module has very flexible addressing methods selected with the **ADDMODE** register. It can be changed during operation. The transmitting module addresses packets according to the addressing mode configuration. The receiving module processes all addressing types regardless of the **ADDMODE** configuration. If the received message matches the addressing criteria, it is output on the UART. Otherwise it is discarded. The **ADDMODE** configuration also enables assured delivery.

There are three addressing modes: DSN, User and Extended User. Each mode offers different communications methods, but all use source and destination addressing. The source address is for the transmitting unit, the destination address is the intended receiver. Each mode uses different registers for the source and destination addresses.

All three addressing modes can be configured to be compatible with the older 250 Series modules. The default operation has an additional level of masking on the receiving module that helps prevent interference from adjacent networks.

The following sections give brief descriptions of the three modes, but a detailed explanation and examples are given in RG-00105, the HumPRO™ Addressing Mode Reference Guide.

### DSN Addressing Mode

Device Serial Number Addressing mode is the simplest mode and supports point-to-point communications. Each module is programmed at the factory with a unique 4-byte serial number that cannot be changed. These bytes are found in the non-volatile read-only **MYDSN** registers (**MYDSN**[3-0]). DSN Addressing mode uses this serial number as an address. The transmitting unit's DSN is used as the source address and the intended receiver's DSN is written into the destination address registers (**DESTDSN**[3-0]). All modules within range hear the transmission, but only the module with the serial number that matches the destination address outputs the data on its UART. All others ignore the transmission.

### User Addressing Mode

User Addressing Mode is a more flexible method than DSN Addressing Mode. It uses the customer ID bytes (**CUSTID**[1-0]) for unencrypted messages and two of the user destination bytes (**UDESTID**[1-0]) as a destination address. The customer ID bytes are programmed at the factory and cannot be changed. These are determined by the factory for specific customers to prevent their systems from operating with any other systems. Contact Linx for more details.

The module's local address is contained in two of the user source ID registers (**USRCID**[1-0]). In this mode, **USRCID** [1-0] contain the node address and **USRCID** [3-2] must be 0 in the receiver.

In normal operation each module has a user ID mask (**UMASK**[3-0]) that splits the 32 address bits into up to three fields to provide a network address and address fields for sub-networks, supporting both individual addressing and broadcast addressing within the user's network. A detailed explanation and examples are given in Reference Guide RG-00105. The 16 bits in the **UDESTID**[1-0] registers are transmitted. The upper 16 bits of **USRCID**[3-2] in the receiver must be 0.

If acknowledgements are enabled, only the module with a user source ID that exactly matches the transmitted user destination ID responds. The mask is not used for this determination.

### Extended User Addressing Mode

Extended User Addressing mode is the same as User Addressing mode but uses 32-bit addresses. The two customer ID bytes are still used (**CUSTID**[1-0]) for unencrypted messages but four bytes are used for the user destination address (**UDESTID**[3-0]), user source ID (**USRCID**[3-0]) and user ID mask (**UMASK**[3-0]). This provides more addressing capabilities at the expense of more overhead in the packet.

### Network Addressing

Network Addressing is selected by setting **COMPAT** to 0x03. It allows the receiver to receive all messages sent in User Address or Extended User Address mode with a destination address matching the **USRCID** group 1 bits (continuous high-order zero bits in **UMASK**). For example, with **USRCID** = 0x12345678 and **UMASK** = 0x000FFFFFF, messages with destination address 0x123zzzzz, where z is any value, is received.

## Automatic Addressing

The module supports an automatic addressing mode that reads the Source Address from a valid received packet and uses it to fill the Destination Address register. This makes sure that a response is sent to the device that transmitted the original message. This also allows the host microcontroller to read out the address of the sending unit. The automatic addressing is enabled for the different addressing modes with register [AUTOADDR](#).

## Address Register Use

Figure 21 shows the address registers that are used with each addressing mode.

HumPRO™ Series Transceiver Address Registers						
COMPAT	0x00 (Relaxed Addressing)			0x02 (Normal Addressing)		
	0x04 (DSN)	0x06 (User)	0x07 (Ex User)	0x04 (DSN)	0x06 (User)	0x07 (Ex User)
ADDMODE	0x14 (DSN +ACK)	0x16 (User +ACK)	0x17 (ExUser +ACK)	0x14 (DSN +ACK)	0x16 (User +ACK)	0x17 (ExUser +ACK)
UDESTID[3-0]			X			X
UDESTID[1-0]		X			X	
USRC[3-0]			X		X	X
USRC[1-0]		X				
UMASK[3-0]			X		X	X
UMASK[1-0]		X				
DESTDSN[3-0]	X			X		

Figure 21: HumPRO™ Series Transceiver Address Register Use

## Acknowledgements and Assured Delivery

When a module transmits with assured delivery enabled, the receiving module returns an acknowledgement packet. The transmitting module waits for this acknowledgement for a preset amount of time based on the data rate. If an acknowledgement is not received, it retransmits the packet. If the receiver receives more than one of the same packet, it discards the duplicate packet contents but sends an acknowledgment. This way, duplicate data is not output by the module.

If the received destination address matches the local address, the receiving module immediately sends an acknowledgement. This packet lets the sending module know that the message has been received. An acknowledgement packet is sent immediately following reception; CSMA delay is not applied to these packets since permission belongs to the interacting modules. When the sending module receives the acknowledgement packet, it marks the current block of data as completed. If this is the last message in the queue, the sending module takes the BE line high to indicate that all outgoing data has been sent.

Assured delivery should only be used when addressing a specific module in a point-to-point link. It should not be used when multiple receivers are enabled. When address masking is used, only the receiver with an exact match to the address in the transmitted packet responds. If none of the enabled receivers has an exact match, then there is no response and the transmitting module continues to re-transmit the data until the max number of retries is attempted. This causes the transmitting module to appear slow or unresponsive. It also impedes valid communications.

## Frequency Hopping Spread Spectrum

The module uses Frequency Hopping Spread Spectrum to allow operation at higher power levels per regulations and to reduce interference with other transmitters. The module is configured for operation in one of 6 different hopping sequences. Each sequence uses 26 channels for the high RF data rate or 50 channels for the low RF data rate. Modules must use the same hopping sequence to communicate. Assigning different hopping sequences to multiple networks in the same area minimizes the interference.

When the module is awake and not transmitting, it rapidly scans all channels for a packet preamble. When a module starts transmitting at the beginning of a new channel, it transmits a packet with a long preamble of alternating 0 and 1 bits. This long preamble is sufficient to allow receiving modules to scan through all of the channels in the hopping sequence and find it. Modules that are scanning detect the preamble and pause on that channel, waiting for a valid packet.

If a packet is received with a valid CRC (unencrypted) or authentication (encrypted), the header is examined to determine whether the module should synchronize to the transmitter. Synchronization requires that the hop sequence matches and that the message is addressed to the receiver.

When synchronized, the receiver stays on the current channel to either transmit a packet or to receive an additional packet. Additional packets transmitted on the same channel within the time slot use short preambles since the receivers are already listening to the current channel.

At the end of the time slot for the current channel, all modules which locked to the original transmission switch to the next channel in the hop sequence. The first transmission on each new channel has a long preamble.

A receiver that has synchronized to a transmitter continues to stay in synchronism by staying on the received channel until the expiration of the time slot, then waiting on the next hop channel for the duration of the time slot. If no further packets are received, the receiver loses lock and reverts to scanning. This allows the receiver to stay synchronized for a short while if a packet is not received correctly.

The module supports the option to send the long preamble with every packet rather than just the first packet on each channel. This can be beneficial for systems that have modules asleep most of the time. It gives

modules that just woke up the chance to synchronize to any transmitted packet instead of having to wait for the transmitter to complete its time slot and jump to the next channel. This can reduce the synchronization time and power consumption of the sleeping nodes.

## Compatibility with the 250 Series

When DSN mode is used with a specific address, the module can communicate with 250 Series modules at UART data rates of 38,400 to 115,200 bps, non-encrypted. For other addressing modes, the HumPRO™ Series modules can be configured to operate with them. Setting the COMPAT register to 0x00 enables the compatible operation. This allows mixed-mode systems and upgrades of legacy products that still maintain backwards compatibility. Only the higher baud rates are compatible.

The main feature of compatibility operation is that it configures the same addressing methods used by the 250 Series. These methods are more susceptible to interference from adjacent networks of 250 Series modules which use DSN (GUI) broadcast messages. Please see Reference Guide RG-00105 for more details.

## Networking

The HumPRO™ Series modules can be used to create many types of wireless networks. The modules do not provide network routing since the internal memory size of the module would limit the overall network size. The HumPRO™ can work as the MAC/PHY layers of a network stack and the memory and processing speed of the external microcontroller can be sized according to the size of the network that is needed for the application.

This requires more software development, but avoids the cost of adding extra memory on the module for applications that don't need it. Linx can assist with network frameworks and concepts and can create custom designs on a contract basis. Contact Linx for more details.

## Transmitting Packets

In default operation when transmitting, the host microcontroller writes bytes to the `CMD_DATA_IN` line while the  $\overline{\text{CMD}}$  line is held high at the baud rate selected by the `UARTBAUD` register. The incoming bytes are buffered until one of the following conditions triggers the packet to be transmitted:

1. The number of bytes in the buffer exceeds the value in the Byte Count Trigger (`BCTRIG`) register.
2. The time since the last received byte exceeds the value in the Data Timeout (`DATATO`) register.
3. A `SENDP` command is written to the `CMD` register.
4. The  $\overline{\text{CMD}}$  line is taken low with option `PKOPT: TXnCMD = 1`.
5. The number of buffered bytes exceeds what can be sent before the radio must hop channels.

The first four conditions can be controlled by the host microcontroller. In the last case, the module transmits what it can in the remaining time then sends the rest on the next channel. This can cause the data to be divided up into multiple packets and is not within the control of the host micro.

In cases where all data needs to be sent in the same packet or where the microcontroller needs greater control over the radio, the HumPRO™ offers explicit control of packet transmission with options in the `PKTOPT` register.

When the `TXPKT` option is enabled (`PKTOPT` register, bit 0 = 1), the data is held until a `SENDP` command is written to the `CMD` register. Alternatively, if option `TXnCMD` is enabled (`PKTOPT` register, bit 1 = 1), then lowering the  $\overline{\text{CMD}}$  line triggers the packet transmission, reducing the number of UART transactions that are required. The `BCTRIG`, `DATATO` and hop-timing conditions are ignored when the `TXPKT` option is enabled.

Once triggered, the transmitted packet contains the bytes in the buffer as of the trigger event, even if more data bytes are received before the packet can be sent. Multiple outgoing packets can be buffered in this way.

If the full packet cannot be sent in the time remaining on the current channel, then it is held until the module hops to the next channel.

This option gives the host microcontroller very fine control over when packets are transmitted and what they contain.

## Receiving Packets

In default operation when receiving valid packets, the module outputs all received bytes as soon as the packet is validated (CRC checks pass if unencrypted or key-based verification if encrypted) and if the addressing permits it at the baud rate selected by the `UARTBAUD` register. No command or control bytes are output and no action is required of an external microcontroller. The first byte from a packet directly follows the last byte of the previously received packet.

In cases where the host microcontroller needs more control over the data or where dynamic configuration changes could set up race conditions between incoming data and outgoing commands, the module offers explicit control over received packets.

When the `RXPKT` option is enabled (`PKTOPT` register, bit 2 = 1), received data is output on the `CMD_DATA_OUT` line one packet at a time after a `GETPH`, `GETPD`, or `GETPHD` command is written to the `CMD` register. Writing one of these commands begins the received packet transfer cycle.

Two lines are used as flow control and indicators during the transfer cycle. The  $\overline{\text{CMD}}$  line is controlled by the host microcontroller. The module uses either the  $\overline{\text{CTS}}$  line or the  $\overline{\text{CRESP}}$  line as a status line, depending on the state of the `RXP_CTS` option in the `PKOPT` register.

When a valid packet is received, the `EX_RXWAIT` exception flag is set in the `EEXFLAG1` register. If the corresponding bit in the `EEXMASK1` register is set, then the `EX` line goes high. The host microcontroller can monitor the `EX` line or periodically check the `EEXFLAG` or `LSTATUS` registers to determine if data is ready to be read.

The transfer cycle is begun by writing a Get Packet Header (`GETPH`), Get Packet Data (`GETPD`), or Get Packet Header and Data (`GETPHD`) command to the `CMD` register. The module sends the command `ACK` byte and sets the selected status line high. Once the status line goes high, the host microcontroller sets the  $\overline{\text{CMD}}$  line high and the module outputs the received data. The command sent determines whether the bytes sent are the header, data, or header followed by data.

When all packet bytes have been sent the control line goes low. When the host microcontroller detects that the line is low, it sets  $\overline{\text{CMD}}$  low, completing the transfer cycle. The cycle is shown in Figure 22.



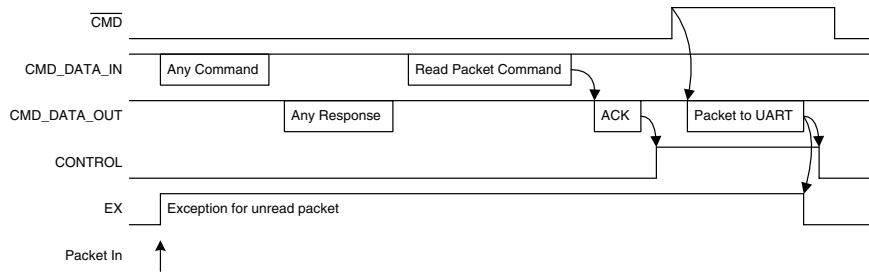


Figure 22: HumPRO™ Series Transceiver Received Packet Transfer Cycle

If a GETPH was sent and header data received, the following data can then be read by repeating the cycle with the GETPD command. If the next GETPx command is a GETPH or GETPHD, the data associated with the header read by GETPH is discarded and the header or header plus data of the following packet is returned.

If there is RF-received data waiting to be sent to the UART and the mask for EX\_RXWAIT is set in the [EEXMASK](#) register, EX is raised if it is low.

If there is no packet waiting when a GETPx command is sent, the control line is still taken high and not reset until after  $\overline{\text{CMD}}$  goes high, thereby performing a zero-byte transfer cycle.

The header and payload structures differ between encrypted packets and unencrypted packets. The header and data structures for explicit unencrypted packets are shown in Figure 23.

The Tag field identifies the start of the block and if it is the header information (0x01) or the packet data (0x02).

The Header Length field identifies the number of header bytes that follow.

The Frame Type field identifies what kind of packet was received. The values are shown in Figure 24.

The Hop ID field is the hop sequence number, 0 - 5.

The Sequence byte is incremented for each new packet, modulo 255. A received packet is discarded if the sequence byte matches the previously received packet to prevent delivering duplicate copies of an automatically retransmitted packet.

DSN Address Packet Header

Tag	Header Length	Frame Type	Hop ID	Sequence	Dest DSN	Source DSN	Data Length
0x01	1	1	1	1	4	4	1

User Address Packet Header

Tag	Header Length	Frame Type	Hop ID	Sequence	Cust ID	Dest Addr	Source Addr	Source DSN	Data Length
0x01	1	1	1	1	2	2 or 4	2 or 4	4	1

Packet Data

Tag	Data Length	Data
0x02	1	Data Length Bytes

Figure 23: HumPRO™ Series Transceiver Unencrypted Packet Header and Data Structure

HumPRO™ Series Transceiver Frame Types	
Frame Type	Packet Type
0x04	DSN Addressing Mode
0x06	User Addressing Mode
0x07	Extended User Addressing Mode
+0x10	Acknowledgements Enabled
+0x20	Encrypted Packet
+0x40	Long Preamble Packet

Figure 24: HumPRO™ Series Transceiver Frame Types

The Cust ID field is a number that can be assigned to a specific customer. Only modules with the same customer ID respond to unencrypted transmissions. By default, Cust ID is 0x7FFF for packets transmitted with COMPAT = 2 or 0xFFFF for packets transmitted with COMPAT = 0. This field is not used in DSN mode.

The Dest Addr field has the received destination address. This is 2 bytes long with User Addressing Mode and 4 bytes with DSN and Extended User Addressing Modes.

The Source Addr Field is the address of the transmitting module. This is 2 bytes long with User Addressing Mode and 4 bytes with DSN and Extended User Addressing Modes.

The Data Length byte indicates how many bytes of data are in the packet. This value is the same in the packet header and the associated data block.

The header and data structures for explicit encrypted packets are shown in Figure 25. The header and data blocks returned by the module are the decrypted message contents.

Encrypted DSN Address Packet Header

Tag	Header Length	Frame Type	Hop Key	Sequence	Dest DSN	Source DSN	EBlock Length	Payload Type
0x11	1	1	1	6	4	4	1	1

Encrypted User Address Packet Header

Tag	Header Length	Frame Type	Hop Key	Sequence	Dest Addr	Source Addr	Source DSN	EBlock Length	Payload Type
0x11	1	1	1	6	2 or 4	2 or 4	4	1	1

Encrypted Packet Data

Tag	Data Length	Data
0x12	1	Data Length Bytes

Figure 25: HumPRO™ Series Transceiver Encrypted Packet Header and Data Structure

The Tag, Header Length and Frame Type fields are the same as for unencrypted packets.

The Hop Key field uses the first three low-order bits to indicate the Hop Sequence number, which is the same as unencrypted packets. The upper two bits indicate which key is being used. Either the factory-set key that is used to securely transfer the network key or a network key that has been written or created by the JOIN process. This is shown in Figure 26.

HumPRO™ Series HopKey Byte Values	
HopKey Bit	Value
0 - 3	Hop Sequence Number, 1 to 5
4 - 5	= 0
6 - 7	Encryption key 0 = factory 1 = user network

Figure 26: HumPRO™ Series HopKey Byte Values

The Sequence bytes contain a counter that is incremented for each new transmitted message. The initial value is randomized when the module is reset. The extended sequence becomes part of an initialization vector which is used to vary the encrypted contents of identical packets. A received packet is discarded if the sequence byte matches the previously received packet to prevent delivering duplicate copies of an automatically retransmitted packet.

The Dest DSN, Source DSN, Dest Addr and Source Addr fields are the source and destination addresses, the same as in unencrypted packets.

The EBlock length field is the total number of bytes of data in the encrypted payload block. This length includes the Payload Type byte.

The Payload Type byte indicates what data is contained in the payload. 0x00 indicates that the payload is user data. 0x01 indicates that the payload is the 16-byte AES key followed by any user data. This is used for transferring the network encryption key during the JOIN process.

For the Encrypted Packet Data packet, the Data Length byte indicates the number of bytes of data payload that follow. This value is one less than the EBlock length in the header. The reason for this is that the Payload Type byte is included in the encrypted block, but is reported with the header since it is not user data.

## Using the Buffer Empty (BE) Line

The BE line indicates the state of the module's UART buffer. It is high to indicate that the UART input buffer is empty, indicating that all data has been transmitted. When the module receives data on the CMD\_DATA\_IN line and the  $\overline{\text{CMD}}$  line is high, the BE line is lowered until all data in the buffer has been processed by the protocol engine. If acknowledgement is not enabled, the BE line is raised as soon as the module transmits the outgoing packets. If acknowledgement is enabled, the buffer is not updated until either the data transmissions are acknowledged by the remote end or delivery fails after the maximum number of retries. When the BE line returns high, the EX line may be sampled, or the **EXCEPT** or **EEXFLAG** register polled to determine if an error occurred during transmission.

The state of the BE line can be read in the **LSTATUS** register, reducing the number of hardware connections that are needed.

## Exception Engine

The HumPRO™ is equipped with an internal exception engine to notify the host microcontroller of an unexpected event. If errors occur during module operation, an exception is raised. There are two methods of driving the EX pin when an exception condition exists:

1. From the **EXMASK** and **EXCEPT** registers (legacy operation)
2. From the **EEXMASKx** and **EEXFLAGx** registers (standard operation)

If EXMASK is non-zero, the first method is used, otherwise the second method is used.

For legacy operation with the 250 and 25 Series, the EX line is set and reset by the Exception (**EXCEPT**) register processing. It is set when an exception occurs and the exception code ANDed with the current Exception Mask (**EXMASK**) register is non-zero. It is reset when the **EXCEPT** register is read through a command. No other operations affect the state of EX. Setting **EXMASK** non-zero does not change the state of EX.

If an exception code is already present in the register when an error occurs, the new exception code overwrites the old value. Exception codes are organized by type for ease of masking. Figure 27 lists the exception codes and their meanings.

HumPRO™ Series Transceiver Exception Codes		
Exception Code	Exception Name	Description
0x08	EX_BUFOVFL	Incoming UART buffer overflowed.
0x09	EX_RFOVFL	Outgoing UART buffer overflowed.
0x13	EX_WRITEREGFAILED	Attempted write to register failed.
0x20	EX_NORFACK	Acknowledgement packet not received after maximum number of retries.
0x40	EX_BADCRC	Bad CRC detected on incoming packet.
0x42	EX_BADHEADER	Bad CRC detected in packet header.
0x43	EX_BADSEQID	Sequence ID was incorrect in ACK packet.
0x44	EX_BADFRAMETYPE	Attempted transmit with Invalid setting in reg:NETMODE or invalid packet type in received packet header

Figure 27: HumPRO™ Series Transceiver Exception Codes

The EX line can be asserted to indicate to the host that an error has occurred. The **EXCEPT** register must be read to reset the line. Figure 28 lists some example exception masks.

HumPRO™ Series Transceiver Example Exception Masks	
Exception Mask	Exception Name
0x08	Allows only EX_BUFOVFL and EX_RFOVFL to trigger the EX line
0x10	Allows only EX_WRITEREGFAILED to trigger the EX line
0x20	Allows only EX_NORFACK to trigger the EX line
0x40	Allows only EX_BADCRC, EX_BADHEADER, EX_BADSEQID and EX_BADFRAMETYPE exceptions to trigger the EX line
0x60	Allows EX_BADCRC, EX_BADHEADER, EX_BADSEQID, EX_BADFRAMETYPE and EX_NORFACK exceptions to trigger the EX line
0xFF	Allows all exceptions to trigger the EX line

Figure 28: HumPRO™ Series Transceiver Example Exception Masks

The exception mask has no effect on the exceptions stored in the exception register. It only controls which exceptions affect the EX line.

The extended exception registers offer more functionality with more exceptions and a separate bit for each exception. These registers are the default and should be used with new applications. When an exception sets an exception code in the **EXCEPT** register, the corresponding flag in the **EEXFLAG** register is also set.

The EX line is set and reset by the Extended Exception Flags (**EEXFLAG**) and Extended Exception Mask (**EEXMASK**) register processing. It is set whenever the **EEXFLAG** value ANDed with the **EEXMASK** value is non-zero. EX can change on any write to either of these registers that affects the result of ANDing the registers. Clearing an **EEXFLAG** register bit or value can leave EX set if there is another masked condition bit set.

The state of the EX line can also be read in the **LSTATUS** register, reducing the number of hardware lines that are required.

## Carrier Sense Multiple Access (CSMA)

CSMA is an optional feature. It is a best-effort delivery system that listens to the channel before transmitting a message. If CSMA is enabled and the module detects another transmitter on the same channel, it waits until the active transmitter finishes before sending its payload. This helps to eliminate RF message corruption and make channel use more efficient.

When a module has data ready to transmit and CSMA is enabled, it listens on the intended transmit channel for activity. If no signal is detected, transmission is started.

If a carrier is detected with an RSSI above the CSMA threshold in the [CRSSI](#) register, transmission is inhibited. If a signal below the threshold is detected that has a compatible preamble or packet structure, transmission is also inhibited.

If the module is synchronized from a recent packet transfer, it waits for a random interval, then checks again for activity. If the detected carrier lasts longer than the time allowed for the current channel, the module hops to the next channel in the hop sequence and again waits for a clear channel before transmitting.

If the module is not synchronized, it hops to the next channel and again checks for interference. When no activity is detected it starts transmitting.

## Using the Command Response ( $\overline{\text{CRESP}}$ ) Line

The  $\overline{\text{CRESP}}$  line is high when sending data bytes and low when sending command response bytes. This indicates to an external host microcontroller that the data on the `CMD_DATA_OUT` line is a response to a command and not data received over-the-air.  $\overline{\text{CRESP}}$  is held in the correct state at least one byte time after the last byte for the indicated source (command response or data, although it normally stays in the same state until a change is required).

If a data packet is received when the module is processing a command, it sends the command response, raises  $\overline{\text{CRESP}}$ , and then sends the received data bytes.

When reading or writing the module's register settings, it is possible for incoming RF data to intermix with the module's response to a configuration command. This can make it difficult to determine if commands were successfully processed as well as to capture the received RF data. Setting the `CMDHOLD` register to `0x01` causes the module to store incoming RF traffic (up to the RF buffer capacity) while the  $\overline{\text{CMD}}$  line is low. When the  $\overline{\text{CMD}}$  line is returned high, the module outputs the buffered data on the UART. This allows the external host microcontroller to have separate configuration times and data times instead of potentially having to handle both at once.

The  $\overline{\text{CRESP}}$  line stays low for at least ten bit times after the stop bit of the last command response. Figure 29 shows the timing.

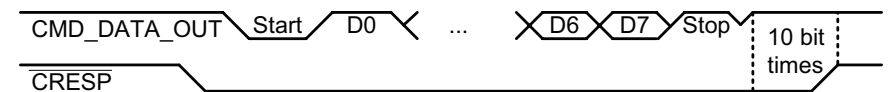


Figure 29: HumPRO™ Series Transceiver  $\overline{\text{CRESP}}$  Line Timing

## Using the $\overline{\text{CMD}}$ Line

The  $\overline{\text{CMD}}$  line informs the module where incoming UART data should be routed. When the line is high, all incoming UART data is treated as payload data and is routed to the transmitter to be sent over the air. If the  $\overline{\text{CMD}}$  line is low, the incoming UART data is treated as command bytes and is routed to the controller for processing.

Since the module's controller looks at UART data one byte at a time, the  $\overline{\text{CMD}}$  line must be held low for the entire duration of the command plus time for ten bits as margin for processing. Leaving the line low for additional time (for example, until the ACK byte is received by the application) does not adversely affect the module. If RF packets are received while the  $\overline{\text{CMD}}$  line is active, they are still processed and output on the module's UART (assuming  $\text{CMDHOLD}=0$  and  $\text{PKOPT:RXPKT}=0$ ). Figure 30 shows this timing.

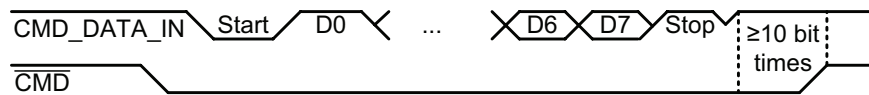


Figure 30: HumPRO™ Series Transceiver  $\overline{\text{CMD}}$  Line Timing

Commands can be entered sequentially without having to raise the  $\overline{\text{CMD}}$  line after each one. The  $\overline{\text{CMD}}$  line just needs to be raised to be able to enter data for transmission.

If the  $\text{CMDHOLD}$  register is  $0x01$  then any received data is held until the  $\overline{\text{CMD}}$  line is raised. This prevents received data from being intermingled with command responses.

## External Amplifier Control

The HumPRO™ Series transceiver has two output lines that are designed to control external amplifiers. The  $\text{PA\_EN}$  line goes high when the module activates the transmitter. This can be used to activate an external power amplifier to boost the signal strength of the transmitter. The  $\text{LNA\_EN}$  line goes high when the module activates the receiver. This can be used to activate an external low noise amplifier to boost the receiver sensitivity. These external amplifiers can significantly increase the range of the system at the expense of higher current consumption and system cost.

The states of the  $\text{PA\_EN}$  and  $\text{LNA\_EN}$  lines can be read in the  $\text{LSTATUS}$  register. This offers a quick way to determine the current state of the radio.

## AES Encryption

HumPRO™ Series modules with firmware version 2.0 and above offer AES encryption. Encryption algorithms are complex mathematical calculations that use a large number called a key to scramble data before transmission. This is done so that unauthorized persons who may intercept the signal cannot access the data. To decrypt the data, the receiver must use the same key that was used to encrypt it. It performs the same calculations as the transmitter and if the key is the same, the data is recovered.

The HumPRO™ Series module has the option to use AES encryption, arguably the most common encryption algorithm on the market. This is implemented in a secure mode of operation to ensure the secrecy of the transmitted data. It uses a 128-bit key to encrypt the transmitted data. The source and destination addresses are sent in the clear.

Encryption is disabled by default. There are two ways to enable encryption and set the key: sending serial commands and using the JOIN process.

### Writing an encryption key to the module with the CDI

The module has no network key when shipped from the factory. An encryption key can be written to the module using the CDI. The  $\text{CMD}$  register is used to write or clear a key. The key cannot be read.

The same key must be written to all modules that are to be used together. If they do not have the same key then they will not communicate in encrypted mode.

### The JOIN Process

The JOIN process is a method of generating an encryption key and distributing the key and addresses to associated modules through a series of button presses. This makes it very simple to establish an encrypted network in the field or add new nodes to an existing network without any additional equipment. It is also possible to trigger the JOIN process through commands on the Command Data Interface.

The JOIN process configures a star network with the central unit as system administrator. Other units are added to the network one at a time.

The hardware required is a pushbutton that is connected to the  $\text{PB}$  line. This takes the line to  $\text{VCC}$  when it is pressed and ground when it is released. An LED connected to the  $\text{MODE\_IND}$  line provides visual indication of the module's state.

A module is set as an administrator by pressing and holding the button for 30 seconds to start the Generate Key function. While the button is held, the MODE\_IND line is on. After 30s, the MODE\_IND line repeats a double blink, indicating that the function has begun. When the button is released the key and address generation are complete and the module is an administrator.

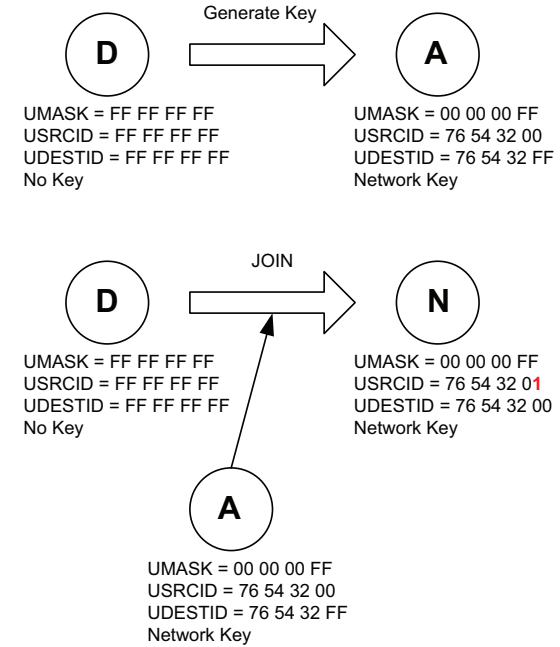
When Generate Key is performed, the unit is set as the network administrator. It generates a random 128-bit AES encryption key based on ambient RF noise and scrambled by an encryption operation. If UMASK is the default value (0xFFFFFFFF), it is set to 0x000000FF, supporting up to 255 nodes, and ADDMODE is set to Extended User Address with encryption (0x27) (or without encryption (0x07) if flag PGKEY in the SECOPT register is 0). UMASK and ADDMODE are not changed if UMASK is not 0xFFFFFFFF. A random 32-bit address is generated. By default, the lower 8 bits are 0, forming the network base address. Other nodes are assigned sequential addresses, starting with network base address +1. UDESTID is set to the bitwise OR of USRCID and UMASK, which is the network broadcast address.

A module becomes a node by joining with an administrator. This is done by pressing and releasing the PB button on both units. The modules automatically search for each other using a special protocol. When they find each other, the administrator sends the node the encryption key, UMASK and its network address. The UDESTID is set to the address of the administrator. The values are encrypted using a special factory-defined key. Once the JOIN process is complete, the MODE\_IND blinks on both units and they now operate together. This is shown in Figure 31 A.

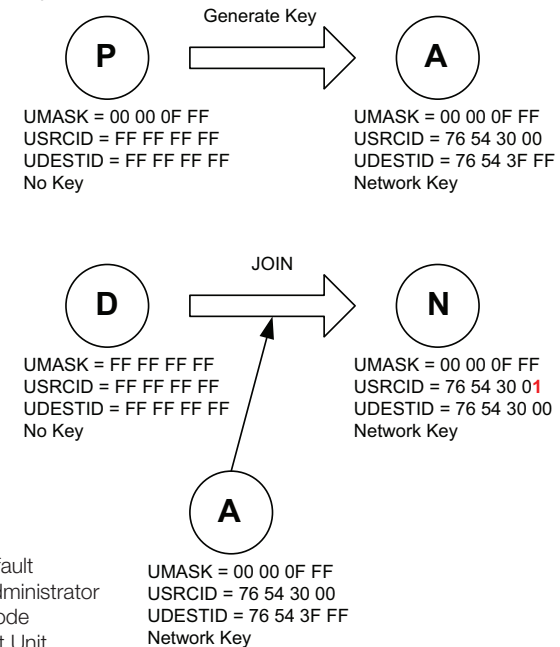
If UMASK is pre-set when Generate Key is initiated, then the JOIN process uses that mask and sets the address accordingly. This can allow more nodes in the network. This is shown in Figure 31 B. Likewise, the network key can be written to the module with the CDI interface and the JOIN process used to create an address and associate new modules. Or the administrator can be completely configured through the CDI and the JOIN process used to associate nodes in the field. This gives the system designer many options for configuration.

The SECOPT register is used to configure options related to the JOIN process. This allows the OEM to set desired values at the factory and allow final network configuration in the field. This includes disabling the ability to change the address, change the key and share the key. The built-in security prohibits changing a node to an administrator without changing the key.

### A) Key Generation and Network Join from Factory Default



### B) Key Generation and Network Join from Preset Mask



D = Factory Default  
 A = Network Administrator  
 N = Network Node  
 P = OEM Preset Unit

UMASK = 00 00 0F FF  
 USRCID = 76 54 30 00  
 UDESTID = 76 54 3F FF  
 Network Key

Figure 31: HumPRO™ Series JOIN Process



## Restore Factory Defaults

The transceiver is reset to factory default by taking the PB line high briefly 4 times, then holding PB high for more than 3 seconds. Each brief interval must be high 0.1 to 2 seconds and low 0.1 to 2 seconds. (1 second nominal high / low cycle). The sequence helps prevent accidental resets. Once the sequence is recognized, the MODE\_IND line blinks in groups of three until the PB line goes low. After PB goes low, the non-volatile configurations are set to the factory default values and the module is restarted. The default UART data rate is 9,600bps.

If the timing on PB does not match the specified limits, the sequence is ignored. Another attempt can be made after lowering PB for at least 3 seconds.

## Using the Low Power Features

The module supports several low-power features to save current in battery-powered applications. This allows the module to be asleep most of the time, but be able to quickly wake up, send data and go back to sleep.

Taking the Power Down ( $\overline{\text{POWER\_DOWN}}$ ) line low places the module into the lowest power state. In this mode, the internal voltage regulator and all oscillators are turned off. All circuits powered from the voltage regulator are also off. The module is not functional while in this mode and current consumption drops to below 6 $\mu$ A. Taking the line high wakes the module.

When the  $\overline{\text{POWER\_DOWN}}$  line is high, the **IDLE** register determines sleep operation.

If **IDLE** is set to 1 during normal operation, the module sends an ACK byte, waits for completion of an active transmission, then goes into sleep mode. Unsent data in the incoming UART data buffer does not inhibit sleep. During sleep mode, the output lines are in the states in Figure 35.

A rising transition on the  $\overline{\text{POWER\_DOWN}}$  or CMD\_DATA\_IN lines wakes the module. If a negative-going pulse is needed to generate a rising edge, the pulse width should be greater than 1  $\mu$ s.

Other lines also wake the module but it immediately goes back to sleep. Floating inputs should be avoided since they may cause unintended transitions and cause the module to draw additional current.

HumPRO™ Series Transceiver Output Line Sleep States	
Output Line	Sleep State
EX	Unchanged
$\overline{\text{CRESP}}$	Low
LNA_EN	Low
PA_EN	Low
TXD	High
$\overline{\text{CTS}}$	High
MODE_IND	Low
BE	Unchanged

Figure 35: HumPRO™ Series Output Line Sleep States

If the volatile registers have been corrupted during sleep, a software reset is performed. This restarts the module as if power were cycled. This can be caused by power surges or brownout among other things.

After the module wakes up, it sets the **IDLE** register to 0 (active). If the **WAKEACK** register is set to 1, then the module outputs the 0x06 byte on the CMD\_DATA\_OUT line. The  $\overline{\text{CRESP}}$  line is taken high and the module then begins normal operation.

Pulsing  $\overline{\text{RESET}}$  low causes the module to restart rather than continue from sleep.



## The Command Data Interface

The HumPRO™ Series transceiver has a serial Command Data Interface (CDI) that is used to configure and control the transceiver through software commands. This interface consists of a standard UART with a serial command set. The CMD\_DATA\_IN and CMD\_DATA\_OUT lines are the interface to the module's UART. The UART is configured for 1 start bit, 1 stop bit, 8 data bits, no parity and a serial data rate set by register **UARTBAUD** (default 9,600bps). The  $\overline{\text{CMD}}$  line tells the module if the data on the UART is for configuration commands (low) or data transmission (high).

The module has a 256 byte buffer for incoming data. The module starts transmitting when the buffer reaches a specified limit or when the time since the last received byte on the UART reaches a specified value. This allows the designer to optimize the module for fixed length and variable length data.

If the buffer gets nearly full (about 224 bytes), the module pulls the  $\overline{\text{CTS}}$  line high, indicating that the host should not send any more data. Data sent by the host while the buffer is full is lost, so the  $\overline{\text{CTS}}$  line provides a warning and should be monitored. When there is outgoing data waiting to be transmitted or acknowledged the BE line is low, otherwise BE is high.

Configuration settings are stored in two types of memory inside the module. Volatile memory is quick to access, but it is lost when power is removed from the module. Non-volatile memory has a limited number of write cycles, but is retained when power is removed. When a configuration parameter has both a non-volatile and volatile register, the volatile register controls the operation unless otherwise stated. The non-volatile register holds the default value that is loaded into the volatile register on power-up.

Configuration settings are read from non-volatile memory on power up and saved in volatile memory. The volatile and non-volatile registers have different address locations, but the same read and write commands. The two locations can be changed independently.

The general serial command format for the module is:

[FF] [Length] [Command]

The Length byte is the number of bytes in the Command field. The

Command field contains the register address that is to be accessed and, in the case of a write command, the value to be written. Neither Length nor Command can contain a 0xFF byte.

Byte values of 128 (0x80) or greater can be sent as a two-byte escape sequence of the format:

0xFE, [value - 0x80]

For example, the value 0x83 becomes 0xFE, 0x03. The Length count includes the added escape bytes.

A response is returned for all valid commands. The first response byte is CMD\_ACK (0x06) or CMD\_NACK (0x15). Additional bytes may follow, as determined by the specific command.

## Reading from Registers

A register read command is constructed by placing an escape character (0xFE) before the register number. The module responds by sending an ACK (0x06) followed by the register number and register value. The register value is sent unmodified, so if the register value is 0x83, 0x83 is returned. If the register number is invalid, the module responds with a NACK (0x15). The command and response are shown in Figure 36.

HumPRO™ Series Read From Configuration Register				
<b>Command</b>				
Header	Size	Escape	Address	
0xFF	0x02	0xFE	REG	
<b>Response</b>				
ACK	Address	Value		
0x06	REG	V		
<b>Command for an Address greater than 128 (0x80)</b>				
Header	Size	Escape	Addr1	Addr2
0xFF	0x03	0xFE	0xFE	REG-80
<b>Response</b>				
ACK	Address	Value		
0x06	REG	V		

Figure 36: HumPRO™ Series Read from Configuration Register Command and Response

## Writing to Registers

To allow any byte value to be written, values of 128 (0x80) or greater can be encoded into a two-byte escape sequence of the format 0xFE, [value - 0x80]. This includes register addresses as well as values to be written to the registers. The result is that there are four possible packet structures because of the possible escape sequences. These are shown in Figure 37.

HumPRO™ Series Write to Configuration Register Command					
<b>Register and Value less than 128 (0x80)</b>					
Header	Size	Address	Value		
0xFF	0x02	REG	V		
<b>Register less than 128 (0x80) and a Value greater than or equal to 128 (0x80)</b>					
Header	Size	Address	Escape	Value	
0xFF	0x03	REG	0xFE	V-0x80	
<b>Register greater than or equal to 128 (0x80) and a Value less than 128 (0x80)</b>					
Header	Size	Escape	Address	Value	
0xFF	0x03	0xFE	REG-0x80	V	
<b>Register and Value greater than or equal to 128 (0x80)</b>					
Header	Size	Escape	Address	Escape	Value
0xFF	0x04	0xFE	REG-0x80	0xFE	V-0x80

Figure 37: HumPRO™ Series Write to Configuration Register Command

Generally, there are three steps to creating the command.

1. Determine the register address and the value to be written.
2. Encode the address and value as either the number (N) or the encoded number (0xFE, N-0x80) as appropriate.
3. Add the header (0xFF) and the size.

The module responds with an ACK (0x06). If the ACK is not received, the command should be resent. The module responds with a NACK (0x15) if a write is attempted to a read-only or invalid register.

As an example, to write 01 to register 0x83, send

FF 03 FE 03 01

**Note:** The non-volatile memory has a life expectancy of at least 18,000 write operations.

## Command Length Optimization

Some commands may be shortened by applying the following rules:

1. Escape sequences are not required for byte values 0x00 to 0xEF (besides 0xFE and 0xFF, bytes 0xF0 – 0xFD are reserved for future use).
2. An escape byte inverts bit 7 of the following data byte.
3. The 0xFE as the first byte of the Read Register Command field is an escape byte.
4. Two consecutive escape bytes cancel unless the following data byte is 0xf0-0xff.

Examples:

- FF 02 FE 02 (read nv:TXPWR) is equivalent to FF 01 82.
- FF 03 FE FE 53 (read v:PKOPT) is equivalent to FF 01 53.
- FF 03 1A FE 7F (write FF to nv:UMASK0) cannot be shortened.
- FF 03 1A FE 40 (write C0 to nv:UMASK0) is equivalent to FF 02 1A C0.

These rules are implemented in the sample code file EncodeProCmd.c, which can be downloaded from the Linx website.