

**Revision**

**1.1.0**



# **WI.232FHSS-25-R and WI.232FHSS-250-R DATASHEET**

RADIOTRONIX, INC.

# WI.232FHSS-25-R/ WI.232FHSS-250-R DATASHEET

---

© Radiotronics  
905 Messenger Lane  
Moore, Oklahoma 73160  
Phone 405.794.7730 • Fax 405.794.7477  
[www.radiotronics.com](http://www.radiotronics.com)

## Document Control

	SIGNED	DATE
CREATED BY	CR	9/29/05
ENGINEERING REVIEW		
APPROVED- ENG.		

## Revision History

REVISION	SIGNED	DATE	DESCRIPTION
0.9.0	CR	9/9/2005	Document Created from Rev G Specification
0.9.1	TRM	1/31/2006	Updated register documentation, added addressing description, exception description, updated all mechanical drawings and block diagrams. Corrected and added various specifications
0.9.5	TRM / GWH	1/22/2007	Update document for 250-R and v1.0.4 25mW firmware
1.0.0	LMK / TRM / GWH	5/8/2007	Add new register definitions. Add additional application information.
1.1.0	TRM/GWH	4/27/2009	Added additional background on hopping engine operation, added documentation for new features in 1.1.0, addition of RSSI data, various corrections

# Table of Contents

<b>1. INTRODUCTION .....</b>	<b>8</b>
1.1. MODULE OVERVIEW .....	8
1.2. FEATURES .....	8
1.3. APPLICATIONS.....	9
<b>2. THEORY OF OPERATION .....</b>	<b>10</b>
2.1. GENERAL .....	10
2.2. OPERATING STATES .....	12
2.3. LOW-POWER STATES .....	13
2.3.1. <i>Standby</i> .....	14
2.3.2. <i>Sleep</i> .....	14
2.3.3. <i>Deep Sleep</i> .....	14
2.4. ADDRESSING MODES.....	14
2.4.1. <i>GUID/MAC Mode</i> .....	15
2.4.2. <i>User Addressing Mode</i> .....	15
2.4.3. <i>Extended User Addressing Mode</i> .....	16
2.4.4. <i>Assured Delivery (Acknowledgement)</i> .....	16
2.5. EXCEPTION ENGINE.....	17
2.5.1. <i>Exception Codes</i> .....	17
2.5.2. <i>Exception Masking</i> .....	17
2.6. RESETTING MODULE TO FACTORY DEFAULTS .....	18
2.7. COMPATIBILITY MODE.....	18
2.8. AUTOMATIC GAIN CONTROL AND MANUAL GAIN CONTROL .....	18
<b>3. APPLICATION INFORMATION .....</b>	<b>19</b>
3.1. PIN-OUT DIAGRAM.....	19
3.1.1. <i>WI.232FHSS-25-R Pin Definitions</i> .....	19
3.1.2. <i>WI.232FHSS-25-R Pin Descriptions</i> .....	20
3.1.3. <i>WI.232FHSS-250-R Pin Definitions</i> .....	20
3.1.4. <i>Wi.232FHSS-250-R Pin Descriptions</i> .....	21
3.2. MECHANICAL SPECIFICATIONS .....	22
3.2.1. <i>WI.232FHSS-25-R Mechanical Drawings</i> .....	22
3.2.2. <i>WI.232FHSS-250-R Mechanical Drawings</i> .....	23
3.3. EXAMPLE CIRCUITS .....	25
3.3.1. <i>WI.232FHSS-25-R Evaluation Module Circuit</i> .....	25
3.3.2. <i>WI.232FHSS-250-R Evaluation Module Circuit</i> .....	26
3.4. POWER SUPPLY .....	27
3.5. UART INTERFACE .....	27
3.6. ADDITIONAL MODULE PINS.....	28
3.6.1. <i>Buffer Empty (BE)</i> .....	28
3.6.2. <i>Exception (EX)</i> .....	28
3.6.3. <i>RF Processing Incoming Packet (PROC_PKT)</i> .....	28
3.6.4. <i>Command Response (CMD_RSP)</i> .....	28
3.6.5. <i>Reset (C2CK/RST)</i> .....	29
3.6.6. <i>Receive Signal Strength Indication (RSSI)</i> .....	30
3.6.7. <i>No Connects (NC)</i> .....	31
3.7. ANTENNA .....	32
3.8. LINK BUDGET, TRANSMIT POWER, AND RANGE PERFORMANCE .....	32

<b>4.</b>	<b>MODULE CONFIGURATION</b>	<b>34</b>
4.1.	REGISTER DESCRIPTIONS	34
4.1.1.	CRC Error Count (regCRCERRCOUNT)	34
4.1.2.	Channel Hop Table (regHOPTABLE)	34
4.1.3.	Power Mode (regPWRMODE)	35
4.1.4.	UART Data Rate (regUARTDATARATE)	35
4.1.5.	Network Mode (regNETWORKMODE)	36
4.1.6.	Transmit Wait Timeout (regTXTO)	37
4.1.7.	Maximum Transmit Retries (regMAXTXRETRY)	37
4.1.8.	CRC Checking (regUSECRC)	38
4.1.9.	UART Minimum Transmission Unit (regUARTMTU)	38
4.1.10.	Verbose Mode (regSHOWVER)	39
4.1.11.	CSMA Enable (regCSMAMODE)	39
4.1.12.	Operating Mode (regOPMODE)	40
4.1.13.	UART Acknowledge on Wake (regACKONWAKE)	41
4.1.14.	User Destination ID[3] (regUSERDESTID[3])	41
4.1.15.	User Destination ID[2] (regUSERDESTID[2])	41
4.1.16.	User Destination ID[1] (regUSERDESTID[1])	41
4.1.17.	User Destination ID[0] (regUSERDESTID[0])	42
4.1.18.	User Source ID[3] (regUSERSRCID[3])	42
4.1.19.	User Source ID[2] (regUSERSRCID[2])	43
4.1.20.	User Source ID[1] (regUSERSRCID[1])	43
4.1.21.	User Source ID[0] (regUSERSRCID[0])	43
4.1.22.	User ID Mask[3] (regUSERIDMASK[3])	44
4.1.23.	User ID Mask[2] (regUSERIDMASK[2])	44
4.1.24.	User ID Mask[1] (regUSERIDMASK[1])	44
4.1.25.	User ID Mask[0] (regUSERIDMASK[0])	45
4.1.26.	Destination GUID[3] (regDESTGUID[3])	45
4.1.27.	Destination GUID[2] (regDESTGUID[2])	45
4.1.28.	Destination GUID[1] (regDESTGUID[1])	46
4.1.29.	Destination GUID[0] (regDESTGUID[0])	46
4.1.30.	Exception Mask (regEXCEPTIONMASK)	46
4.1.31.	CMD Halts Traffic (regCMDHALT)	47
4.1.32.	Receiver LNA Mode (regLNAMODE)	47
4.1.33.	Compatibility Mode (regCOMPATMODE)	48
4.1.34.	Auto Address Mode (regAUTADDMODE)	48
4.1.35.	My GUID[3] (regMYGUID[3])	49
4.1.36.	My GUID[2] (regMYGUID[2])	49
4.1.37.	My GUID[1] (regMYGUID[1])	50
4.1.38.	My GUID[0] (regMYGUID[0])	50
4.1.39.	Release Number (regRELEASENUM)	50
4.1.40.	Exception (regEXCEPTION)	51
4.1.41.	Last Good Packet RSSI (regLGPRSSI)	51
4.1.42.	Immediate RSSI (regIMMEDRSSI)	51
4.2.	REGISTER SUMMARY	53
4.2.1.	Volatile Read/ Write Registers	53
4.2.2.	Volatile Read-Only Registers	54
4.2.3.	Non-Volatile Read-Only Registers	54
4.2.4.	Non-Volatile Read/ Write Registers	55
<b>5.</b>	<b>USING CONFIGURATION REGISTERS</b>	<b>56</b>
5.1.	CMD PIN	56
5.2.	CMD_RSP PIN	56
5.3.	COMMAND FORMATTING	56
<b>5.4.</b>	<b>WRITING TO REGISTERS</b>	<b>57</b>

5.5.	READING FROM REGISTERS .....	58
<b>6.</b>	<b>ELECTRICAL SPECIFICATIONS.....</b>	<b>59</b>
6.1.	ABSOLUTE MAXIMUM RATINGS FOR WI.232FHSS-25-R.....	59
6.2.	DETAILED ELECTRICAL SPECIFICATIONS FOR WI.232FHSS-25-R.....	60
6.2.1.	<i>AC Specifications- Rx for WI.232FHSS-25-R</i> .....	60
6.2.2.	<i>AC Specifications- Tx for WI.232FHSS-25-R</i> .....	60
6.2.3.	<i>DC Specifications for WI.232FHSS-25-R</i> .....	61
6.3.	ABSOLUTE MAXIMUM RATINGS FOR WI.232FHSS-250-R.....	61
6.4.	DETAILED ELECTRICAL SPECIFICATIONS FOR WI.232FHSS-250-R.....	62
6.4.1.	<i>AC Specifications- Rx for WI.232FHSS-250-R</i> .....	62
6.4.2.	<i>AC Specifications- Tx for WI.232FHSS-250-R</i> .....	62
6.4.3.	<i>DC Specifications for WI.232FHSS-250-R</i> .....	63
6.5.	FLASH SPECIFICATIONS (NON-VOLATILE REGISTERS).....	63
<b>7.</b>	<b>CUSTOM APPLICATIONS .....</b>	<b>64</b>
<b>8.</b>	<b>ORDERING INFORMATION .....</b>	<b>65</b>
<b>9.</b>	<b>CONTACT INFORMATION.....</b>	<b>66</b>
9.1.	TECHNICAL SUPPORT .....	66
9.2.	SALES SUPPORT .....	66

## Index of Tables

Table 1, Wi.232FHSS Low-Power Pin States .....	13
Table 2, MAC Addressing Examples.....	15
Table 3, User Addressing Examples .....	16
Table 4, Extended User Addressing Examples.....	16
Table 5, Exception Codes .....	17
Table 6, Example Exception Masks .....	18
Table 7, WI.232FHSS-25-R Pin Descriptions .....	20
Table 8, WI.232FHSS-250-R Pin Descriptions .....	21
Table 9, Wi.232FHSS-25-R and WI.232FHSS-250-R UART Interface Lines .....	27
Table 10, Wi.232FHSS Reset Circuit Specifications.....	30
Table 11, Power Mode Register Settings.....	35
Table 12, Data Rate Register Settings.....	35
Table 13, Network Mode Register Settings.....	36
Table 14, Minimum TXTO Values .....	37
Table 15, Per-Baud rate ACK timeout values .....	38
Table 16, Verbose Mode Settings .....	39
Table 17, Operating Mode Register Settings .....	40
Table 18, LNA Mode Settings .....	47
Tables 19, AutoAddress Register Settings .....	49
Tables 20 and 21, Release Number Register Settings .....	50
Table 22, Exception Codes .....	51
Table 23, Volatile Read/ Write Register Summary.....	53
Table 24, Volatile Read-Only Register Summary .....	54
Table 25, Non-Volatile Read-Only Register Summary.....	54
Table 26, Non-Volatile Read/ Write Register Summary.....	55
Table 27, Write Register Command, value to be written is less than 128 (0x80).....	58
Table 28, Write Register Command, value to be written is greater than or equal to 128 (0x80).....	58
Table 29, Read Register Command.....	58
Table 30, Read Register Module Response for a Valid Register .....	58
Table 31, Absolute Maximum Ratings for WI.232FHSS-25-R .....	59
Table 32, AC Specifications- Rx for WI.232FHSS-25-R .....	60
Table 33, AC Specifications- Tx for WI.232FHSS-25-R.....	60
Table 34, DC Specifications for WI.232FHSS-25-R.....	61

Table 35, Absolute Maximum Ratings for WI.232FHSS-250-R .....	61
Table 36, AC Specifications- Rx for WI.232FHSS-250-R .....	62
Table 37, AC Specifications- Tx for WI.232FHSS-250-R.....	62
Table 38, DC Specifications for WI.232FHSS-250-R.....	63
Table 39, Flash Specifications (Non-Volatile Registers).....	63
Table 40, Ordering Information .....	65

## Table of Figures

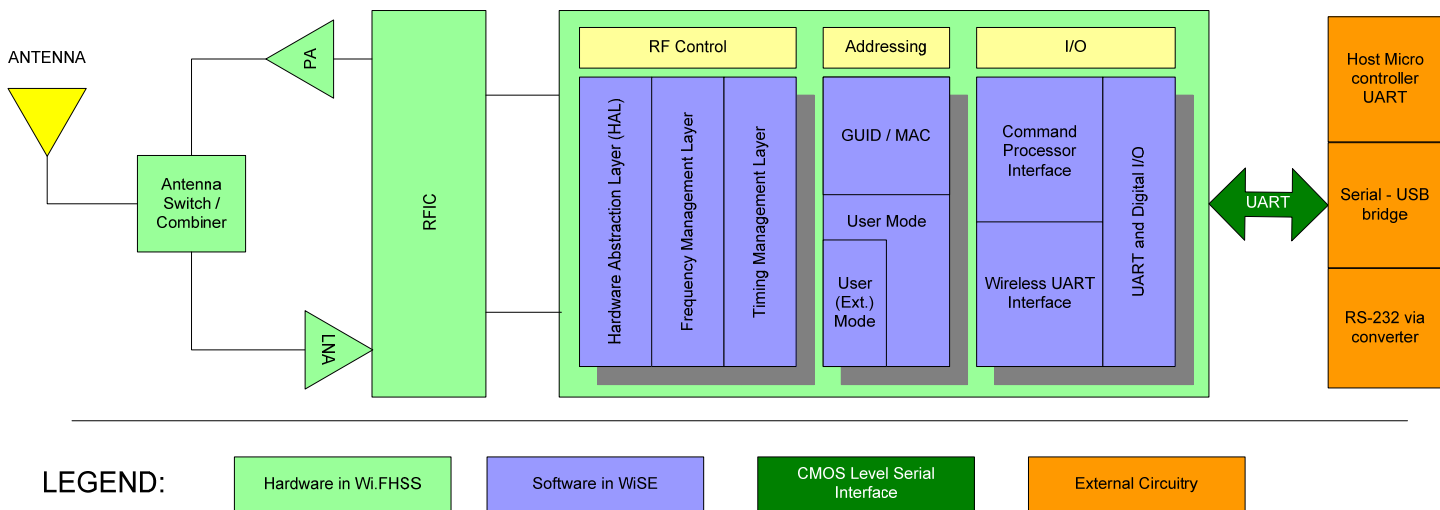
Figure 1: Wi.232FHSS Block Diagram.....	8
Figure 2: WiSE Block Diagram.....	10
Figure 3: Wi.232 Networking Concept.....	11
Figure 4: WI.232FHSS-25-R Pin Definitions .....	19
Figure 5: WI.232FHSS-250-R Pin Definitions .....	20
Figure 6: WI.232FHSS-25-R Mechanical Drawings .....	22
Figure 7: WI.232FHSS-25-R, Suggested Footprint.....	22
Figure 8: WI.232FHSS-250-R Mechanical Drawing.....	23
Figure 9: WI.232FHSS-250-R, Suggested Footprint.....	24
Figure 10: WI.232FHSS-25-R Evaluation Module Circuit .....	25
Figure 11: WI.232FHSS-250-R Evaluation Module Circuit .....	26
Figure 12: Wi.232FHSS Power-on and $V_{DD}$ Monitor Reset Time and Pin State.....	30
Figure 13: Typical Analog RSSI response (Wi.232FHSS-250-R).....	31
Figure 14: Typical Analog RSSI response (Wi.232FHSS-25-R).....	32
Figure 15: Command and CMD Pin Timing .....	56
Figure 16: Command Conversion Code.....	57



# Chapter 1

## 1. Introduction

### 1.1. Module Overview



**Figure 1: Wi.232FHSS Block Diagram**

### 1.2. Features

- True UART to Antenna Solution
- 16-bit CRC Error Checking
- 153.6 kbit/ sec Maximum RF Data Rate
- 6 Hop Sequences
- GUID/MAC Addressing Mode
- Flexible User Addressing Mode
- Link Layer Supports Assured Delivery
- Small Size (0.80" x 0.935" x 0.12") for the WI.232FHSS-25-R
- Small Size (1.20" x 1.20" x 0.20") for the WI.232FHSS-250-R
- Low Power Standby, Sleep, and Deep Sleep Modes
- PHY, MAC, and Link Layer Protocol Built-in

- CSMA Medium Access Control
- 122.4 dB Link Budget for the WI.232FHSS-25-R
- 132.9 dB Link Budget for the WI.232FHSS-250-R
- Command mode for the Volatile and Non-Volatile Configuration
- 32-bit Unique GUID/MAC Address
- 5 Volt Tolerant I/O
- 902 – 928 MHz ISM Band Requires No License
- Adjustable Output Power
- Automatic Gain Control for the WI.232FHSS-250-R Only

### **1.3. Applications**

- Direct RS-232/422/485 Wire Replacement (requires external conversion circuitry)
- Asset Tracking
- Automated Meter Reading (AMR)
- Traffic and Display Signs
- Mass-Transit Communications
- Industrial and/ or Home Automation
- RFID
- Wireless Sensors
- Remote Data Logging
- Oil and Gas Sensing
- Fleet Management
- Vehicle Tracking
- Toys
- Long-range Data Links

# Chapter 2

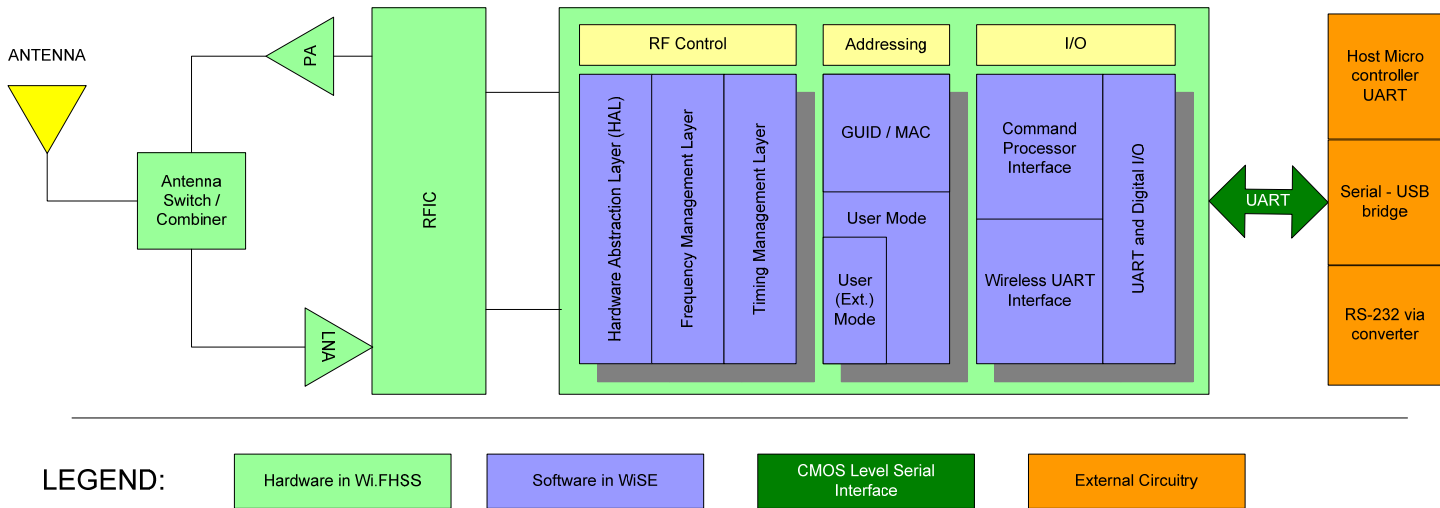
## 2. Theory of Operation

### 2.1. General

The Wi.232FHSS module is one of a family of WiSE™ (Wireless Serial Engine) modules. A WiSE™ module combines a state-of-the-art FSK data transceiver and a high-performance protocol controller to create a complete embedded wireless communications link in a small IC-style package.

The 250mW RF transceiver is built around the Analog Devices ADF7020 IC, whereas the 25mW version uses the Semtech XE1203F transceiver IC. Both transceivers are designed with the components necessary to facilitate operation in the 902-928MHz US ISM band.

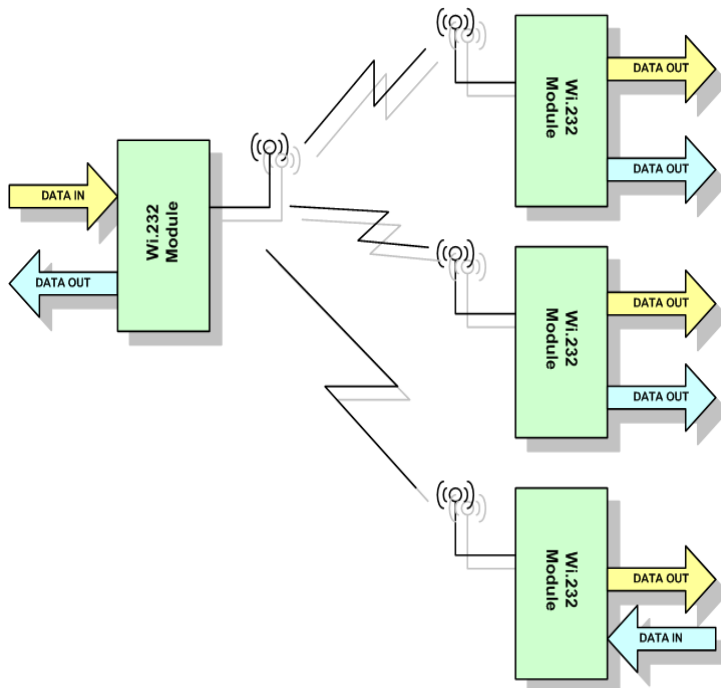
### Wireless Serial Engine (WiSE™) FHSS



**Figure 2: WiSE Block Diagram**

The Wi.232FHSS module has a UART-type serial interface and contains special application software to create a transparent UART-to-antenna wireless solution capable of direct wire replacement in most embedded RS-232/422/485 applications.

**Note:** Although the module is capable of supporting the typical serial communications required by RS-232, RS-422, and RS-485 networks, it is not compatible with the electrical interfaces for these types of networks. The module has CMOS inputs and outputs and would require an appropriate converter for the particular type of network it is connected to.



**Figure 3: Wi.232 Networking Concept**

The module is designed to interface directly to a host UART. Three signals are used to transfer data between the module and the host UART: **TXD**, **RXD**, and **CTS**. **TXD** is the data output from the module. **RXD** is the data input to the module. **CTS** is an output that indicates the status of the module's data interface. If **CTS** is low, the module is ready to accept data. If **CTS** is high, the module is busy and the host UART should not send any further data. The UART interface is capable of operating in full duplex at baud rates from 2.4 to 115.2 kbps. The UART interface expects 1 start bit, 8 data bits (lsb first), and 1 stop bit per byte with no parity (8-N-1).

Internally, the module has a 256 byte buffer for incoming characters from the host UART. The module can be programmed to automatically transmit when the buffer reaches a programmed limit, set by **regUARTMTU**. The module can also be programmed to transmit based on a delay between characters, set by **regTXTO** (set in 1mSec increments). These registers allow the designer to optimize performance of the module for fixed length and variable length data. The module supports streaming data, as well. To optimize the module for streaming data, **regUARTMTU** should be set to 128. Additionally, **regTXTO** should be set to a value greater than 1 UART byte time (10 bit times rounded up) at the current UART data rate, or 2, whichever is greater. If the UART receive/RF transmit buffer becomes nearly full (about 224 bytes), the module will assert **CTS** high, indicating that the host should not send any more data. Data sent by the host while **CTS** is high could be lost. When there is data in the UART receive/RF transmit buffer, the BE pin is low; when this buffer is empty, BE is high.

When the MAC layer has a packet to send, it will optionally use a **carrier-sense-multiple-access (CSMA)** protocol to determine if another module is already transmitting. If another module is transmitting, the module will receive that data before attempting to transmit its data again. If, during this process, the UART receive buffer gets full, the **CTS** line will go high to prevent the host UART from over-running the receive buffer. The CSMA mechanism introduces a variable delay to the transmission channel. This delay is the sum of a random period and a weighted period that is dependent on the number of times that the module has tried and failed to acquire the channel. For applications that guarantee that only one module will be transmitting at any given time, the CSMA mechanism can be turned off to avoid this delay.

Initially, the transmission of the packet begins on a random hop index within the current hop sequence, and follows the hop sequence thereafter until synchronization is lost. Synchronization is lost whenever there is no more data to transfer and the module has detected two consecutive hop indices without data present. New data is not sent within the last 5% of the hop sequence, but data which is already in the process of being sent is processed normally.

The MAC layer prefixes the data with a packet header and appends to the data a 16-bit CRC. The CRC-16 packet validation can be disabled to allow the application to do its own error checking of the payload.

The Link layer provides three distinct addressing modes: MAC, User, and User Extended. Each of these addressing modes can be configured to utilize assured (acknowledged) or best-effort (not acknowledged) delivery. MAC addressing mode is best suited for point-to-point or broadcast transmissions. User and User Extended modes, with their address masks, allow for the creation of subnets.

The Wi.232FHSS is very flexible because of its extensive configuration options. However, modules that are not configured in the same way will not be able to communicate reliably, causing poor performance or outright failure of the wireless link. **All modules in a network must have compatible configurations to ensure interoperability.**

In TRANSMIT mode, the transceiver has a complete packet queued for transmission. The module uses the same CSMA mechanism to arbitrate access to the channel. Once the module gains access to the channel, the packet is transmitted on the current channel without delay. If BIT\_ACK of [regNETMODE](#) is set, the module starts TO\_ACKWAIT and begins waiting for an ACK from the other side. If the ACK is not received, a transmission retry is attempted. If the number of transmission retries exceeds [REG\\_MAXTXRETRY](#), an exception ([EX\\_NORFACK](#)) is raised.

Once the packet is sent, the transmitter will deactivate, but remain tuned to the current channel until its hop time has expired. If another packet is queued for transmission, the module will transmit this packet once the CSMA mechanism allows access to the channel. Once the hop timer has expired, the module will then hop to the next channel (both transmit and receive channels). The module will remain synchronized until it dwells for one full hop time without transmitting or receiving a packet. The module will then return to scan mode.

Certain features of the module are controlled through programmable registers. Registers are accessed by bringing **CMD** low. When **CMD** is low, all data transfers from the host UART are considered to be register access commands. When **CMD** is high, all data transfers from the host UART are considered to be raw data that needs to be transparently transmitted across the wireless link. The module maintains two copies of each register: one in flash and one in RAM. On reset, the module loads the RAM registers from the values in the flash registers. The module is operated out of the RAM registers. Applications that need to change parameters of the module often would simply modify the RAM registers. By putting default settings in the flash registers, the module will always power up in a preconfigured state, which is useful for applications that do not have external microcontrollers, such as RS-232 adapters.

The FHSS module has 32 channels spaced on 750 KHz boundaries with guard bands on either side. These channels are pseudo-randomly arranged into six unique hopping tables comprised of 26 channels. The order of these tables is chosen so that cross-correlation is minimized, allowing multiple networks to operate in proximity with minimal interference.

## 2.2. Operating States

The primary active state is the RX SCAN state. When the module is not actively transmitting or receiving packets, it is in this state. It is cycling from one channel to another throughout the hop sequence looking for a synchronizing packet. If the module detects a pre-amble, it will stall the next hop to wait for the start-code and packet header. If the packet is addressed to it, the module will process the packet and present it

to the UART for transmission to the host microprocessor. If not, it will resume scanning for another master. If the packet received is a synchronizing packet, synchronization will be dropped if the packet address validation fails.

Once the UART has buffered enough data to send (either [regUARTMTU](#) bytes received or [regTXTO](#) has expired), it will schedule a transmission with the RF layer. The RF layer will make a best-effort attempt to keep the data in at least [regUARTMTU](#)-sized packets, but will split data to better fill the communications channel if the hop time allows. If the module is not synchronized at the time of the transmission, it votes itself to master status, and sends a synchronizing preamble. Following a hop, the module that sends the first transmission will vote itself to master status, send a synchronizing preamble, and communications will resume.

## 2.3. Low-Power States

The module supports three power saving modes: Standby, Sleep, and Deep Sleep. Standby and Sleep are included primarily for legacy compatibility with Wi.232DTS™ and Wi.232EUR™ products. The hardware required to support these two low-power modes fully is not present in the Wi.232FHSS modules. As a result, the current consumption in these two modes is considerably higher than their Wi.232DTS™/Wi.232EUR™ counterparts. It is recommended that applications utilize the Deep Sleep mode for power savings.

In the SLEEP and DEEP SLEEP modes, the transceiver is powered down and will not synchronize with other modules. SLEEP mode draws more current than DEEP SLEEP mode. In DEEP SLEEP mode the module draws the least current. To wake the module up from this mode the C2CK/RST pin must be held low for at least 20µs and then returned to high. Modules do not monitor the receive channel in either SLEEP or DEEP SLEEP mode. Therefore, it is impossible to remotely wake a sleeping module via the RF interface.

If [regACKONWAKE](#) is enabled, the module will transmit a 0x06 character out the TxD pin of the UART once awakened from a low-power mode or power-off state. This indicates that the module is ready to resume operations.

The following table indicates the states of the module pins while in a low-power state.

Pin Name 250-R	Pin Number 250-R	Pin Number 25-R	Pin State
PROC_PKT	1	N/A	Driven Low
TxD	2	6	Input with weak pull-up
NC	3	N/A	Input with weak pull-up
NC	4	N/A	Floating
NC	5	N/A	Floating
C2CK/RST	6	11	Input with weak pull-up
C2D	7	10	Input with weak pull-up
NC	8	N/A	Floating
CMD_RSP	9	8	Input with weak pull-up
EX	10	2	Driven Low
RSSI	14	9	Driven Low
CMD	15	4	Input with weak pull-up
BE	16	3	Input with weak pull-up
NC	17	N/A	Floating
RTS	18	N/A	Input with weak pull-up
CTS	19	7	In Standby, Sleep: Driven Low, In Deep Sleep: Driven High
RxD	20	5	Input with weak pull-up

**Table 1, Wi.232FHSS Low-Power Pin States**

### 2.3.1. Standby

Standby is selected by writing a 0x02 to [regOPMODE](#). In this mode, the internal oscillator of the module's protocol controller is lowered to its slowest setting. The transmitter and receiver hardware is in power-down, but the radio IC's oscillator is enabled and running. The Wi.232FHSS-250-R™ module equipped with version 1.1.0 or newer firmware wakes from standby in less than 6ms. The Wi.232FHSS modules will wake from standby in 8ms. A negative-going pulse on RxD wakes the module. This negative-going pulse should be at least 1 bit-time in duration, so sending any byte to the UART will wake it due to the negative-going start bit. Because the module's oscillator is not capable at running at ultra-low speeds, use of this mode is not recommended for new applications. The RAM contents are preserved during standby. If the RAM fails an integrity check, the module will issue itself a software reset to force reinitialization.

### 2.3.2. Sleep

Sleep is selected by writing a 0x01 to [regOPMODE](#). The internal oscillator of the module's protocol controller is lowered to its slowest setting, and all radio IC services are stopped (receiver, transmitter, oscillator, etc.). The Wi.232FHSS-250-R™ equipped with version 1.1.0 or newer firmware wakes from sleep in less than 6ms. Older versions of Wi.232FHSS-250-R™ and Wi.232FHSS-25™ wake from sleep in 8ms. A negative-going pulse on RxD wakes the module. This negative-going pulse should be at least 1 bit-time in duration, so sending any byte to the UART will wake it due to the negative-going start bit. Because the module's oscillator is not capable at running at ultra-low speeds, use of this mode is not recommended for new applications. The RAM contents are preserved during sleep. If the RAM fails an integrity check, the module will issue itself a software reset to force reinitialization.

### 2.3.3. Deep Sleep

Deep sleep is selected by writing a 0x03 to [regOPMODE](#). When the module is put into deep sleep, the CTS pin is brought high to indicate that the module is not ready to accept UART data. The radio IC is placed in its lowest power mode and all services are stopped. The protocol controller's oscillator is also stopped and all non-essential portions of the chip are turned off. While powered, this mode consumes the least amount of current. The Wi.232FHSS-250-R™ equipped with version 1.1.0 or newer firmware wakes from deep sleep in less than 6ms. Older versions of Wi.232FHSS-250-R™ and Wi.232FHSS-25™ wake from deep sleep in 8ms. A negative-going pulse of at least 20µs on C2CK/RST starts the waking process, but the module doesn't begin executing wake instructions until the C2CK/RST pin is returned high. As with the other low-power modes, the RAM contents are preserved. If the RAM fails an integrity check, the module will issue itself a software reset to force reinitialization. Note that, if your program changes the volatile data rate register during the host application initialization ([regUARTDatarate](#)), the reinitialization will return the module to the value indicated by the non-volatile counterpart ([regNVUSERDatarate](#)).

## 2.4. Addressing Modes

The Wi.232FHSS module has a very flexible addressing scheme incorporated into its firmware. The addressing modes can be dynamically changed as the module operates. Selection of the addressing mode is made through the [regNVNETWORKMODE](#) and [regNETWORKMODE](#) registers. Selection of an addressing mode forces the transmitter to address packets according to this configuration. When receiving, a module will receive and process all addressing types, regardless of [regNETWORKMODE](#) configuration. If the received message matches the addressing criteria, it will be passed through to the UART for transmission to the host processor.

## 2.4.1. GUID/MAC Mode

GUID/MAC mode supports point-to-point and broadcast communications. In this mode, endpoint addressing is accomplished through the [regNVDESTGUID\[3-0\]](#) and [regDESTGUID\[3-0\]](#) registers. The module's source GUID ([regMYGUID\[3-0\]](#)) is fixed and programmed at the factory. MAC addressing mode is selected by writing either 0x04 (MAC) or 0x14 (MAC w/ acknowledgement) to the [regNETWORKMODE](#) register. A destination GUID of 0xFFFFFFFF causes the module to send broadcast messages; all modules within range will receive and process these messages. A module will only process a MAC message if the message's destination address is 0xFFFFFFFF or it exactly matches its [regMYGUID\[3-0\]](#) address. Acknowledgement is enabled by setting bit 4 of [regNETWORKMODE](#). The GUID/MAC network mode includes 14 bytes of overhead per RF packet. The following table lists some examples of how MAC addressing works.

Sender			Receiver	
NetworkMode	MyGUID	DestinationGUID	MyGUID	Response
0x04	0x00001000	0xFFFFFFFF	0x00002000	Data sent to UART. No RF ACK sent by either module.
			0x00003000	
0x14	0x00001000	0xFFFFFFFF	0x00002000	Data sent to UART. No ACK sent by either module. This configuration will cause transmission problems.
			0x00003000	
0x14	0x00001000	0x00003000	0x00002000	Not processed- discarded.
			0x00003000	Data sent to UART. RF ACK sent to 0x00001000.
0x04	0x00001000	0x00002000	0x00002000	Data sent to UART. No RF ACK sent.
			0x00003000	Not processed- discarded.

**Table 2, MAC Addressing Examples**

## 2.4.2. User Addressing Mode

When User network mode is selected, transmitted packets locate endpoints using the customer ID and destination User ID (specified in [regCUSTID\[1-0\]](#) and [regUSERDESTID\[1-0\]](#), respectively). An RF packet sent with User Network mode enabled carries 16 bytes of overhead. On the receiving side, each module has a User ID mask ([regUSERIDMASK\[1..0\]](#)) that it applies to both its own User ID ([regUSERSRCID\[1-0\]](#)) and the incoming destination User ID. Once both are masked, if the results are equal, the receiving module will pass the payload data to the application for presentation to the host. Additionally, if the incoming address, once masked, equals the mask itself, its payload will be presented to the host. If an acknowledgement is requested, the receiving module will respond only if the unmasked User IDs are equal. When using user network mode to send packets to multiple users (mask not equal to 0xFFFF), assured delivery must be disabled. Failure to do so could cause extreme delays in transmission and loss of data. The following table shows some examples of user addressing at work.

Sender			Receiver		
Network Mode	User SRCID	User DESTID	User SRCID	User IDMASK	Response
0x06	0x1000	0xFFFF	0x2000	0xFFFF	Data sent to UART. No RF ACK sent by either module.
			0x3000	0xFFFF	
0x16	0x1000	0xFFFF	0x2000	0xFFFF	Data sent to UART in both modules. No ACK sent by either module. This configuration will cause transmission problems.
			0x3000	0xFFFF	
0x16	0x1000	0x3000	0x2000	0xE000	Data sent to UART. No RF ACK sent.



			0x3000	0xE000	Data sent to UART. RF ACK sent to 0x1000 by GUID.
0x06	0x1000	0x3000	0x2000	0xF000	Not processed- discarded.
			0x3000	0xF000	Data sent to UART. No RF ACK sent.

**Table 3, User Addressing Examples**

### 2.4.3. Extended User Addressing Mode

When Extended User Network mode is selected, transmitted packets locate endpoints using the customer ID and destination User ID (specified in [regCUSTID\[1-0\]](#) and [regUSERDESTID\[3-0\]](#), respectively). An RF packet sent with Extended User Network mode enabled carries 20 bytes of overhead. On the receiving side, each module has a User ID mask ([regUSERIDMASK\[3-0\]](#)) that it applies to both its own User ID ([regUSERSRCID\[3-0\]](#)) and the incoming destination User ID. Once both are masked, if the results are equal, the receiving module will pass the payload data to the application for presentation to the host. Additionally, if the incoming address, once masked, equals the mask itself, its payload will be presented to the host. If an acknowledgement is requested, the receiving module will respond only if the unmasked User IDs are equal. When using extended user network mode to send packets to multiple users (mask not equal to 0xFFFFFFFF), assured delivery must be disabled. Failure to do so could cause extreme delays in transmission and loss of data. The following table shows some examples of extended user addressing at work.

Sender			Receiver		
Network Mode	User SRCID	User DESTID	User SRCID	User IDMASK	Response
0x07	0x10000000	0xFFFFFFFF	0x20000001	0xFFFFFFFF	Data sent to UART. No RF ACK sent by either module.
			0x20000002	0xFFFFFFFF	
0x17	0x10000000	0xFFFFFFFF	0x20000001	0xFFFFFFFF	Data sent to UART in both modules. No ACK sent by either module. This configuration will cause transmission problems.
			0x20000002	0xFFFFFFFF	
0x17	0x10000000	0x30000001	0x20000001	0xE0000000	Data sent to UART. No RF ACK sent.
			0x30000001	0xE0000000	Data sent to UART. RF ACK sent to 0x10000000 by GUID.
0x07	0x10000000	0x30000002	0x20000001	0xF0000000	Not processed- discarded.
			0x30000001	0xF0000000	Data sent to UART. No RF ACK sent.

**Table 4, Extended User Addressing Examples**

### 2.4.4. Assured Delivery (Acknowledgement)

While not an addressing mode on its own, assured delivery can be enabled for each of the above addressing modes. When a module transmits with assured delivery enabled, it obligates the receiving module to return an acknowledgement packet. The transmitting module will wait for this acknowledgement for a preset amount of time based on the data rate. If acknowledgement is not received, it will retransmit the current packet. If the receiver receives more than one of the same packet, it will discard the packet contents but send an acknowledgment. This way, duplicate data is not presented to the receiver's UART. It is extremely important that assured delivery be used only when the unmasked user/extended user Destination ID or Destination GUID points to a specific module. Failure to specifically address a valid endpoint could cause the module to appear slow or unresponsive due to repeated retransmissions. This will also serve to congest the

network, impeding valid communications. Assured delivery is enabled by setting bit 4 of [regNETWORKMODE](#).

Should the address information match exactly, the receiving module will immediately send an RF ACK packet. This packet lets the sending module know that the message has been received. An RF ACK packet is made up of 7 bytes of overhead, and is sent immediately following reception; CSMA delay is not applied to RF ACK packets. When the sending module receives the RF ACK packet, it will mark the current block of data as completed and update its buffer pointers. If this is the last message in the queue, the sending module will assert the BE pin to indicate the state of the incoming buffer.

**Troubleshooting Hint:** Communications Problems. If you are unable to communicate with another module, it is most likely caused by one of the following. First, check to make sure that you are using the same data rate. Modules programmed with different data rates will not communicate nor share the RF channel with one another. Second, ensure that your network mode and addressing is configured to properly access the module of interest. Also, ensure that you are addressing a specific module when using acknowledgment. Failure to do so will cause large delays and loss of data.

## 2.5. Exception Engine

Wi.232FHSS modules are equipped with an internal exception engine. When errors occur during module operation, an exception is raised. Exception codes are stored in the [regEXCEPTION](#) register when they occur, and cleared once they are read. If an exception code is already present in [regEXCEPTION](#) following an error, the new exception code will overwrite the old value.

### 2.5.1. Exception Codes

Exception codes are organized by type for ease of masking (see Exception Masking). The following table lists the exception codes and their meanings. All other values are reserved.

Exception Code	Exception Name	Description
0x08	EX_BUFOVFL	Internal UART buffers overflowed
0x09	EX_RFOVFL	Internal RF packet 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	Unsupported frame type specified

**Table 5, Exception Codes**

### 2.5.2. Exception Masking

Wi.232FHSS modules have an external pin, EX, that can be asserted to indicate to the host that an error has occurred. The exception mask provides a simple method of discriminating which errors cause the EX pin to toggle. If the exception code, once ANDed with the exception mask ([regEXCEPTIONMASK](#)), is non-zero, the EX pin is asserted. Once the EX pin is asserted, the [regEXCEPTION](#) register must be read to return it to low. The following table lists some example exception masks.

It is important to note that the exception mask has no effect on the exceptions stored in the exception register. The exception mask only controls which exceptions affect the EX pin.

Exception Mask	Exception Name
0x08	Allows only EX_BUFOVFL and EX_RFOVFL to trigger the EX pin
0x10	Allows only EX_WRITEREGFAILED to trigger the EX pin
0x20	Allows only EX_NORFACK to trigger the EX pin
0x40	Allows only EX_BADCRC, EX_BADHEADER, EX_BADSEQID, and EX_BADFRAMETYPE exceptions to trigger the EX pin
0x60	Allows EX_BADCRC, EX_BADHEADER, EX_BADSEQID, EX_BADFRAMETYPE, and EX_NORFACK exceptions to trigger the EX pin
0xFF	Allows all exceptions to trigger the EX pin

**Table 6, Example Exception Masks**

## 2.6. Resetting Module to Factory Defaults

It may be necessary to reset the non-volatile registers to their factory defaults. To reset the module to factory defaults, hold the command line low and cycle power to the module. The command line must remain low for a minimum of 600ms after resetting the module. Once the command line is released, the module's non-volatile registers will be reset to factory defaults.

## 2.7. Compatibility Mode

The Wi.232FHSS-250-R operates at a much narrower receive bandwidth (200kHz) than the Wi.232FHSS-25 (600kHz). When the Wi.232FHSS-25 transmits, it deviates well outside the receiver bandwidth of the Wi.232FHSS-250-R. *To address this problem, v1.0.5 and later Wi.232FHSS-25 firmware and all Wi.232FHSS-250-R's support a compatibility mode.* This allows both modules to communicate effectively with each other.

When enabled, compatibility mode reduces the maximum RF data rate to 76.8kbps, reduces the Wi.232FHSS-25's deviation to 80kHz, and reduces the Wi.232FHSS-25's receiving bandwidth to 200kHz. All UART baud rates are supported, although the RF data rates associated with baud rates 31250, 38400, 57600, and 115200 will be diminished.

## 2.8. Automatic Gain Control and Manual Gain Control

Automatic gain control is a feature exclusive to the Wi.232FHSS-250-R module. For both modules, the gain setting of the receiver low noise amplifier (LNA) is adjustable.

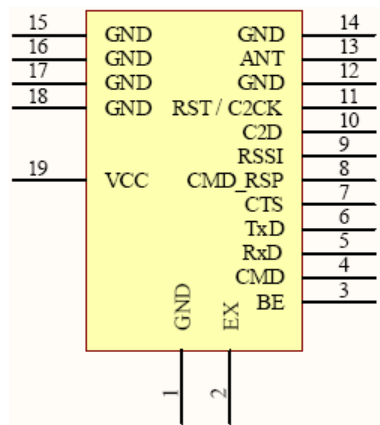
By default, the Wi.232FHSS-25 module is configured for maximum receiver sensitivity. Similarly, the Wi.232FHSS-250-R is factory-configured to use its internal automatic gain control (AGC) circuit to manage receiver sensitivity. Reducing the gain increases the linearity of the receiver, but reduces maximum sensitivity; increasing the gain does the opposite. Generally speaking, higher linearity (increased third order input intercept point, IIP3) will give improved performance in high-interference environments; high gain yields better performance in low-interference environments.

The Wi.232FHSS-250-R contains an AGC circuit that manages these settings automatically, and it should be used whenever possible. However, when attempting to make **analog** RSSI measurements, fixing the LNA gain will produce more meaningful results. Digital RSSI readings are internally compensated and may be taken with AGC enabled.

### 3. Application Information

#### 3.1. Pin-out Diagram

##### 3.1.1. WI.232FHSS-25-R Pin Definitions



**Figure 4: WI.232FHSS-25-R Pin Definitions**

### 3.1.2. WI.232FHSS-25-R Pin Descriptions

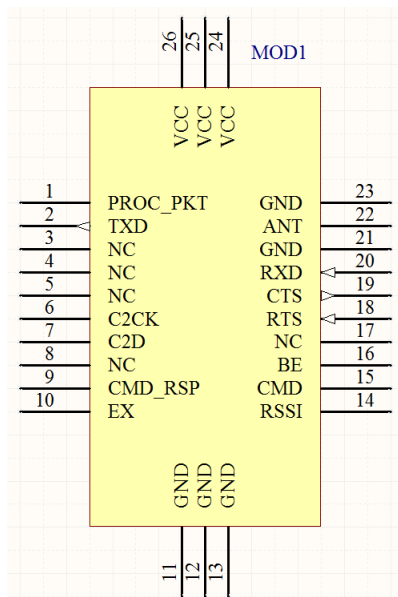
No.	Description
1	Ground
2	Exception Output
3	Buffer Empty
4	Command input – active low
5	UART receive input
6	UART transmit output
7	UART clear to send output – active low
8	Command Response indication
9	Analog RSSI
10	Reserved – ISP pin
11	ISP pin/Wake from DEEP SLEEP/Module Reset
12	Ground
13	Antenna port – 50 ohm
14	Ground
15	Ground
16	Ground
17	Ground
18	Ground
19	VCC – 2.7 to 3.6 VDC

**Table 7, WI.232FHSS-25-R Pin Descriptions**

#### Legend

	Signals that are used in this implementation
	Signals used for in-system programming

### 3.1.3. WI.232FHSS-250-R Pin Definitions



**Figure 5: WI.232FHSS-250-R Pin Definitions**

### 3.1.4. Wi.232FHSS-250-R Pin Descriptions

No.	Description
1	No connect – reserved
2	UART Transmit output
3	Processing Packet signal
4	No connect – reserved
5	No connect – reserved
6	ISP pin / Wake from deep sleep/Module Reset
7	Reserved – ISP pin
8	No connect – reserved
9	Command Response indication – low when UART output is a command response
10	Exception Output, maskable. Cleared on exception read.
11	Ground
12	Ground
13	Ground
14	RSSI
15	Command Mode select – active low
16	Buffer Empty – high when input buffer is empty
17	N/C
18	UART Request To Send input – not currently implemented (reserved)
19	UART Clear To Send output – active low
20	UART Receive input
21	Ground
22	Antenna Port – 50 ohms
23	Ground
24	VCC
25	VCC
26	VCC

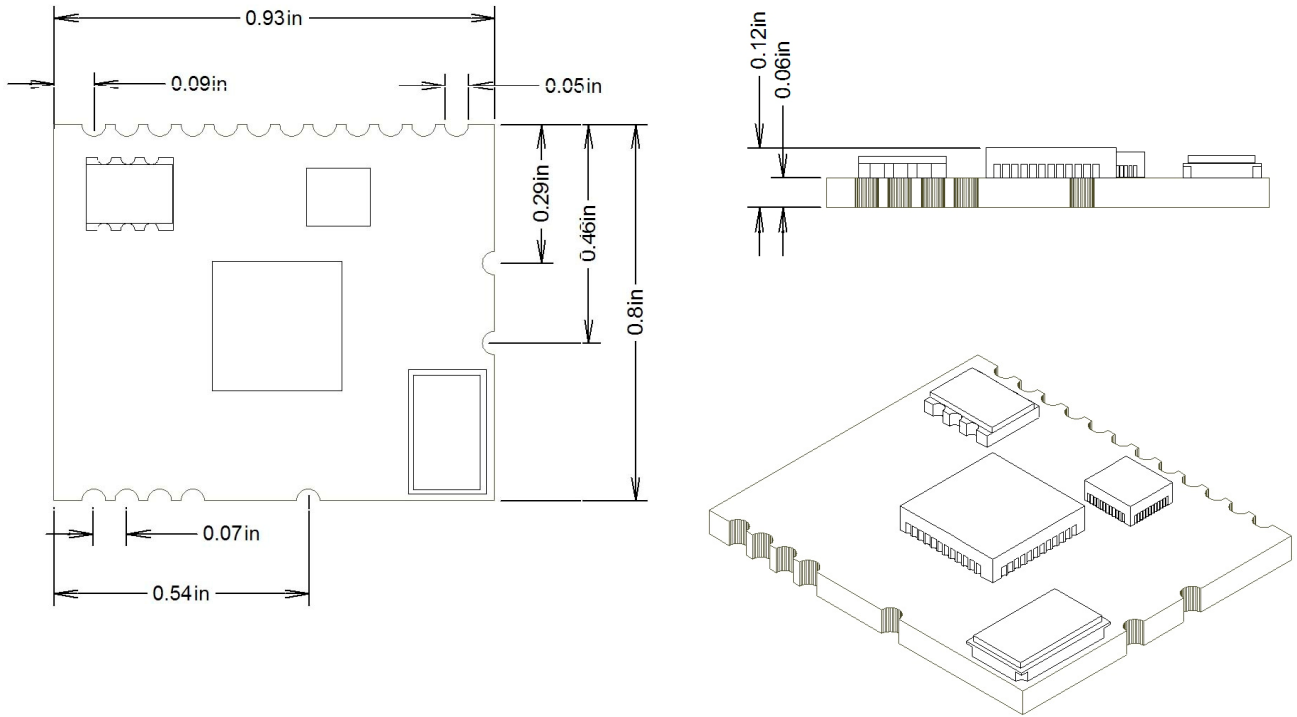
**Table 8, WI.232FHSS-250-R Pin Descriptions**

#### Legend

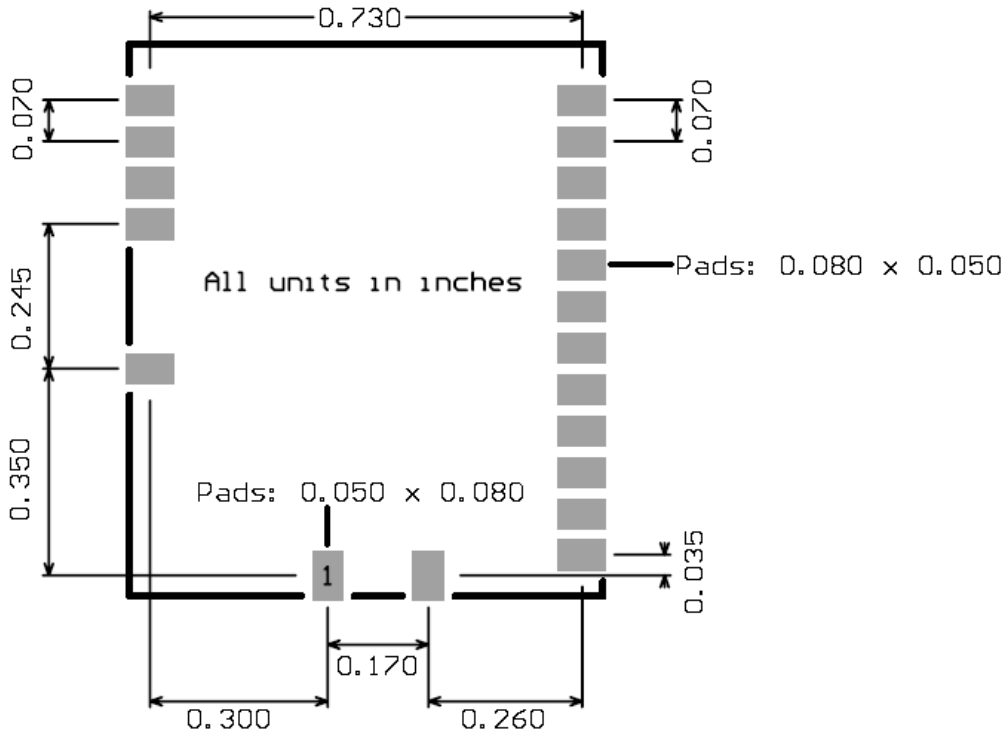
	Signals that are used in this implementation
	Signals used for in-system programming

## 3.2. Mechanical Specifications

### 3.2.1. WI.232FHSS-25-R Mechanical Drawings

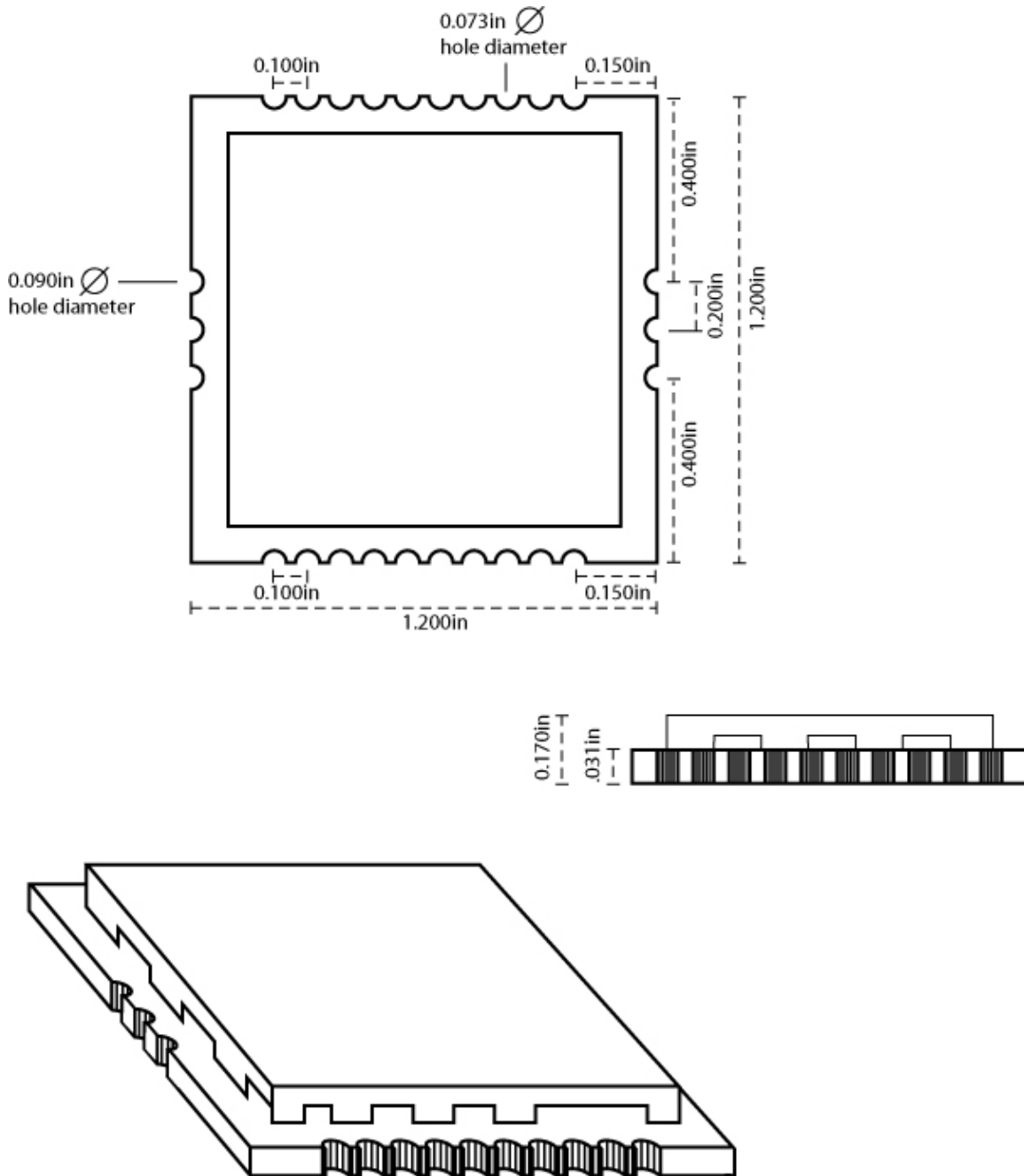


**Figure 6: Wi.232FHSS-25-R Mechanical Drawings**



Revision 1.1.0

### 3.2.2. WI.232FHSS-250-R Mechanical Drawings



**Figure 8: WI.232FHSS-250-R Mechanical Drawing**





### 3.3. Example Circuits

#### 3.3.1. WI.232FHSS-25-R Evaluation Module Circuit

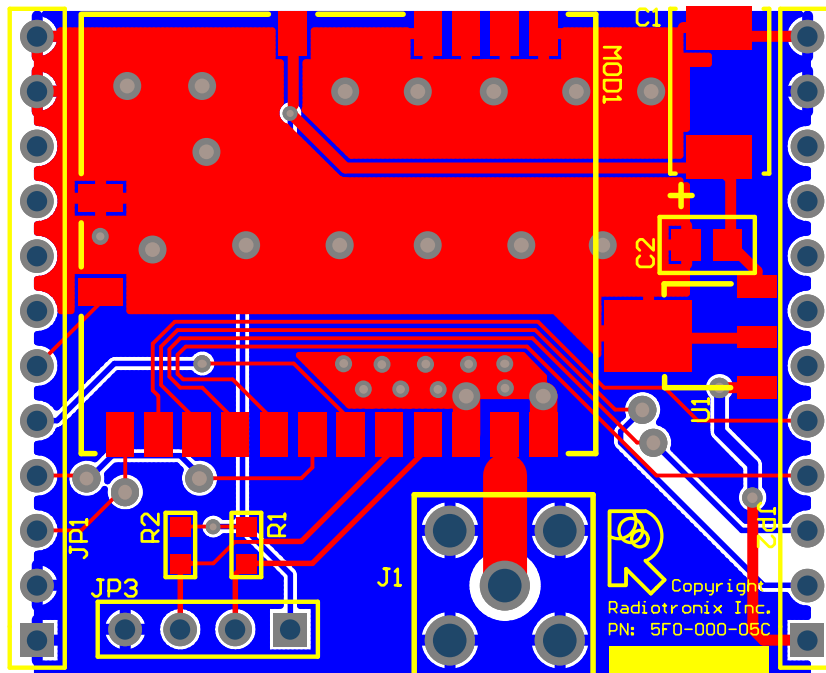
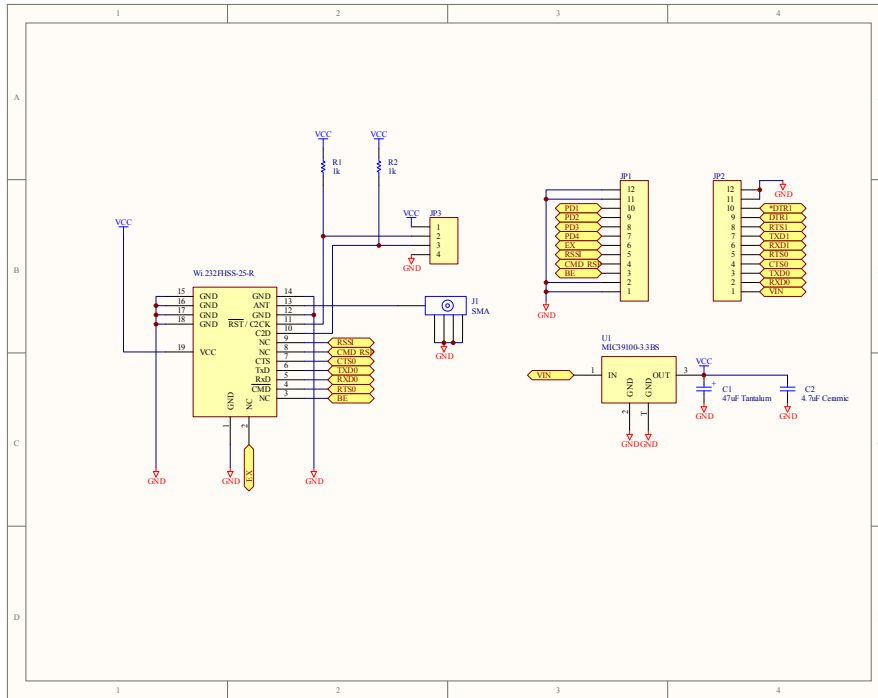


Figure 10: WI.232FHSS-25-R Evaluation Module Circuit

### 3.3.2. WI.232FHSS-250-R Evaluation Module Circuit

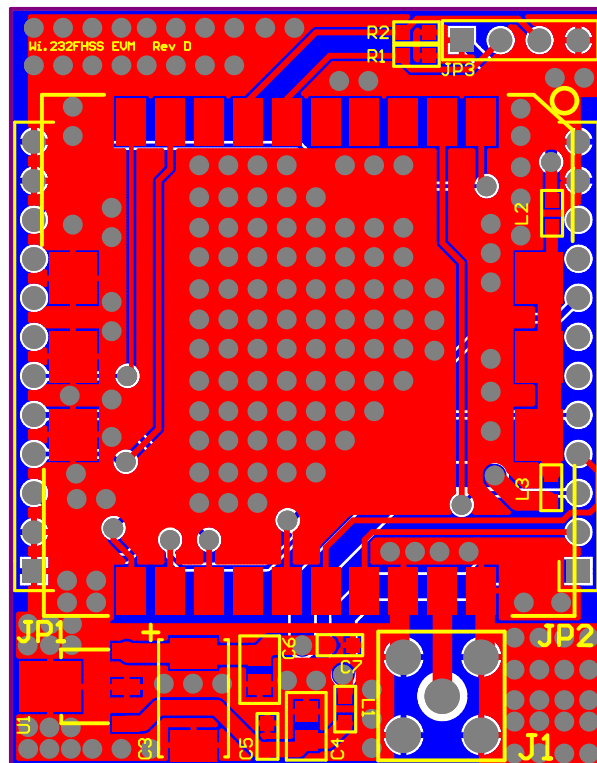
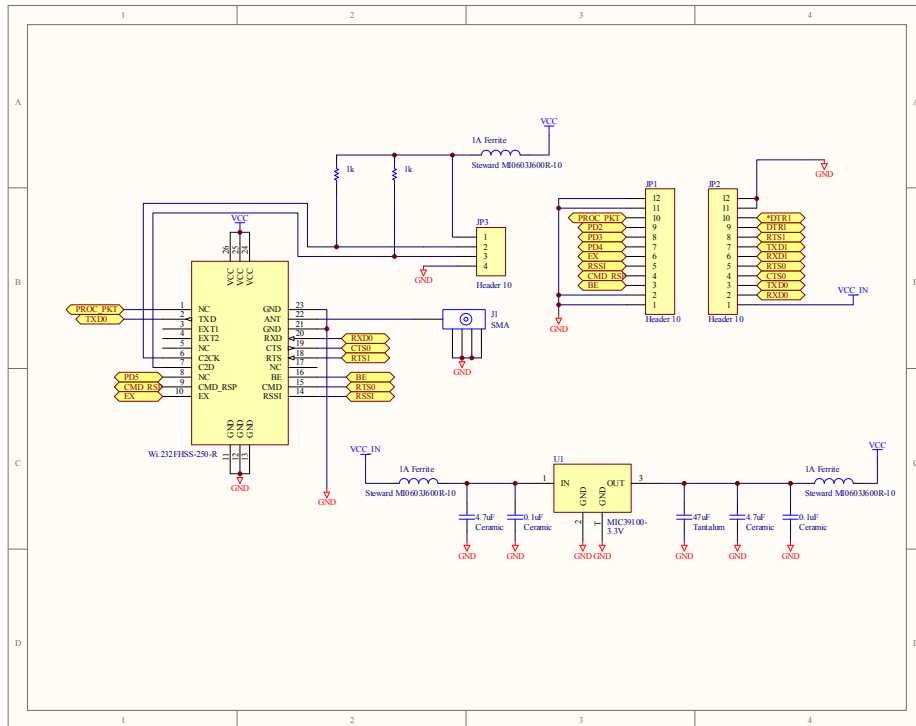


Figure 11: WI.232FHSS-250-R Evaluation Module Circuit

### 3.4. Power Supply

Although the Wi.232FHSS module is very easy to use, care must be given to the design of the power supply circuit. It is important for the power supply to be free of digital noise generated by other parts of the application circuit, such as the RS-232 converter.

Figure 11 shows the schematic for our evaluation module circuit for the Wi.232FHSS-25-R and Wi.232FHSS-250-R modules. The EVM includes an on-board power supply and antenna connector. These evaluation circuits were used to measure the performance of the Wi.232FHSS module, and should be used as a reference for Wi.232FHSS based designs.

If noise is a problem, it can usually be eliminated by using a dedicated LDO regulator for the module and/or by separating the grounds for the module and the other circuits.

Additionally, power supply rise time is **extremely important**. The power supply presented to the module must rise from  $V_{SS}$  to 2.7V in less than 1ms. If this specification cannot be met, an external reset supervisor circuit must be used to hold the module in reset until the power supply stabilizes. **Failure to ensure adequate power supply rise time can result in loss of important module configuration information.**

The Wi.232FHSS-250-R module power supply should be bypassed as close to the module as possible for maximum noise immunity (C1 and C2 in figure 11). Also, an inductor or ferrite choke may be placed close to the module, in series with the supply line to further reduce any noise being conducted back onto the supply from the module. Noise on the power supply can degrade receiver performance and cause other instabilities.

### 3.5. UART Interface

The UART interface is very simple; it is comprised of four CMOS compatible digital lines.

Pin	Direction	Description
CTS	Out	Clear to send – this pin indicates to the host micro when it is ok to send data. When CTS is high, the host micro should stop sending data to the module until CTS returns to the low state.
CMD	In	Command – the host micro will bring this pin low to put the module in command mode. Command mode is used to set and read the internal registers that control the operation of the module. When CMD is high, the module will transparently transfer data to and from other modules on the same channel. <b>Note:</b> If this pin is low when the module comes out of reset, the non-volatile registers will be reset to their factory programmed defaults. It is important to ensure that CMD is held high or left floating during power-up under normal conditions.
RxD	In	Receive data input. UART data should be sent to the module with no parity, 1 start bit, 1 stop bit, 8 data bits, least-significant bit first.
TxD	Out	Transmit data output. UART data will be sent by the module with no parity, 1 start bit, 1 stop bit, 8 data bits, least-significant bit first.
RTS	In	Currently unimplemented, this pin is reserved for future use/reassignment.

**Table 9, Wi.232FHSS-25-R and WI.232FHSS-250-R UART Interface Lines**

## 3.6. Additional Module Pins

Additional module pins indicate the operational status of different functional blocks, or provide additional control. There is no requirement that these pins be connected for normal operation, however they can provide additional insight into the module. They may prove useful in the optimization of the end-application.

### 3.6.1. Buffer Empty (BE)

The BE pin indicates the state of the incoming UART buffer. When the module receives data in the RxD pin, and the CMD pin is high, the BE pin is lowered until all data in this buffer has been processed by the RF engine. If acknowledgement is not enabled, the BE pin will be raised as soon as the RF engine processes the outgoing packets. If acknowledgement is enabled, the buffer will not be updated until either the data transmissions are acknowledged by the remote end, or the maximum number of retries has been exceeded. When the BE pin returns high, the EX pin may be sampled, or the [regEXCEPTION](#) register polled to determine if an error occurred during transmission.

### 3.6.2. Exception (EX)

The EX pin indicates whether a module exception has occurred. The pin is normally low. When an exception occurs that passes masking (see [Exception Engine section](#) for details), the EX pin is raised. When the [regEXCEPTION](#) register is read, the exception is cleared and the EX pin will return low. If more than one exception occurs before the [regEXCEPTION](#) register is read, the old exception will be overwritten by the new one.

### 3.6.3. RF Processing Incoming Packet (PROC\_PKT)

Available on 250-R modules with firmware release 1.1.0 or later, the PROC\_PKT pin indicates whether the RF engine has determined there to be either valid or potentially valid data incoming. The line is normally low (not processing). When awake and not transmitting, the RF engine is constantly searching for an incoming signal. When scoring indicates that a potential packet is inbound, this line is raised until the scoring falls below a given threshold or the complete packet is received. It is possible that the packet scoring will fall below the PROC\_PKT threshold during reception, causing the line to be lowered. Such an instance can occur when the module hops to a channel late in the transmitter's extended preamble. Since there aren't a large number of valid bits to score, the line may fall low during the packet start sequence. Once this sequence arrives, the PROC\_PKT signal will re-raise and latch for the duration of the packet reception.

### 3.6.4. Command Response (CMD\_RSP)

The CMD\_RSP pin is normally high. When the module's command processor responds to a command, such as a register read, the CMD\_RSP pin lowers as the characters are transmitted out of the TxD pin. Some host processors cannot react quickly enough to this signal, and may not be able to separate the command responses from incoming data. In this case, the [regCMDHALT](#) register may be of use. This register controls the behavior of the TxD line when the CMD pin is low. If this register is set to 0x01 and the CMD pin is low, the module stops the flow of incoming RF data to the TxD pin, internally buffering it. Once the CMD pin is raised, the buffered incoming RF data resumes transport out the TxD pin.

### 3.6.5. Reset (C2CK/RST)

The C2CK/RST pin is normally high. It is an open-drain input/output pin with an integrated weak pull-up. Because it periodically operates as an output, care must be taken when interfacing to this pin. When operating as an input, it has different functions depending on the state the module is in.

#### 3.6.5.1. Hardware Reset (Input)

During normal operation, the pin functions as an active-low hardware reset input. When the module is awake and running, bringing this pin low for at least 15 $\mu$ s forces the module's processor into hardware reset. While the pin is low, execution of module operations are suspended and all module I/O pins revert to open-drain inputs with weak pull-ups. This behavior can be exploited during power-up if the  $V_{DD}$  ramp time exceeds 1ms. By suspending execution, the dangers associated with slow  $V_{DD}$  ramp are nullified.

#### 3.6.5.2. Wake from Deep Sleep (Input)

When the module is in deep sleep, all execution is suspended in the protocol controller, and the radio is its lowest power mode. To wake the module the module's C2CK/RST pin must be lowered for at least 15 $\mu$ s. When the C2CK/RST line is raised, execution begins in the protocol controller. The module maintains its state engine while asleep. Because of this, it can detect whether the hardware reset was intended to cause a hard reset or wake the module. The protocol controller's RAM is preserved during deep sleep. The RAM is checked prior to entering deep sleep, and examined upon waking. If the RAM contents are corrupted upon wake, the module will issue itself a software reset to reinitialize the module.

#### 3.6.5.3. Hardware Reset Indicator (Output)

When the module starts from power-off, or is reset by the internal  $V_{DD}$  monitor circuitry, the C2CK/RST pin is driven low to indicate the reset state. During power-on reset, assuming  $V_{DD}$  ramp time is valid, C2CK/RST is driven low from the time that  $V_{DD}$  reaches approximately 1V until  $V_{DD}$  reaches  $V_{RST} + T_{PORDelay}$ .  $T_{PORDelay}$  is the power-on reset delay imposed by the protocol controller's hardware.

The other event that will drive the C2CK/RST pin low is a low-voltage or brown-out condition. In this case, the  $V_{DD}$  monitor will hold the device in reset, thus driving the C2CK/RST pin low. It will remain low until the power drops below the operating threshold for that circuit (becoming indeterminate), or until the module power supply returns to  $V_{RST}$ . The figure below illustrates the operation of C2CK/RST as an output.

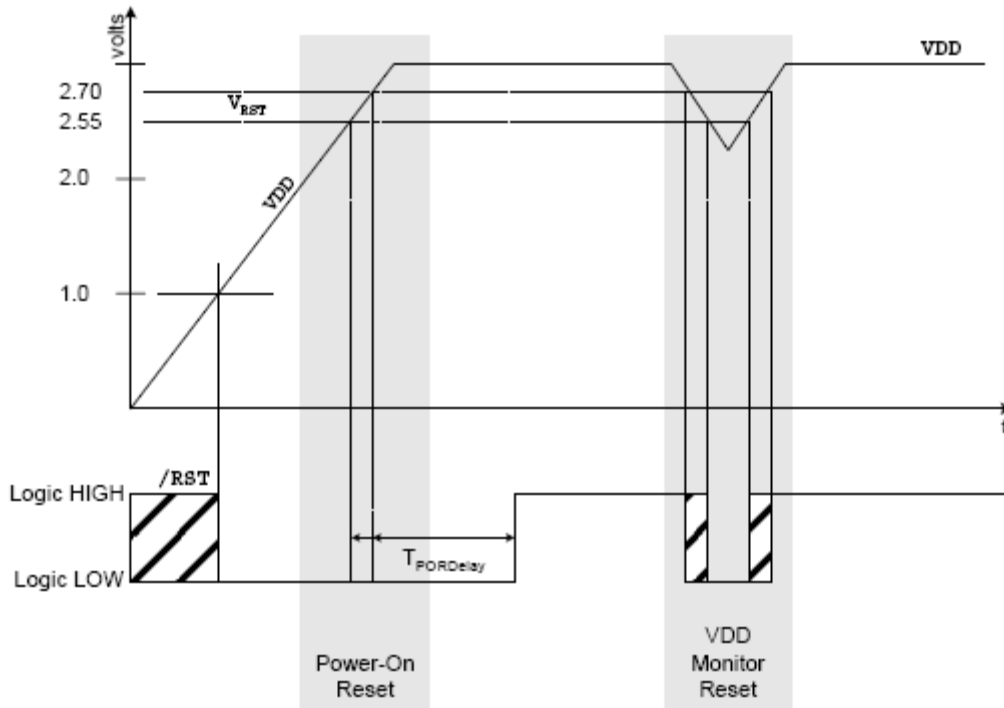


Diagram courtesy of Silicon Laboratories Inc.

**Figure 12: Wi.232FHSS Power-on and V<sub>DD</sub> Monitor Reset Time and Pin State**

Parameter	Min	Typ	Max	Units	Notes
C2CK/RST Output Low Voltage			0.6	V	V <sub>DD</sub> = 2.7 – 3.6V
C2CK/RST Input Pull-up Current		25	40	µA	C2CK/RST = 0.0V
V <sub>DD</sub> Monitor Threshold (V <sub>RST</sub> )	2.40	2.55	2.70	V	
Minimum C2CK/RST Low Time to Generate a Hardware Reset	15			µs	
Power-on Reset Delay (T <sub>PORDelay</sub> )		< 300		µs	VDD Ramp Time is Valid
Allowed/Valid VDD Ramp Time			1	ms	

**Table 10, Wi.232FHSS Reset Circuit Specifications**

### 3.6.6. Receive Signal Strength Indication (RSSI)

The RSSI pin contains an analog voltage proportional to the signal strength present on the channel at the time. In normal operation, the module is hopping rapidly from channel to channel looking for valid data. In this case, the RSSI value does not provide much in the way of useful information. However, it can be used to keep a module awake. For instance, if you are preparing to put the module to sleep, you may sample the RSSI pin to determine if it is processing a packet.

The Wi.232FHSS-250-R module has an internal, digital RSSI indication of [ambient/immediate](#) and of the [last good packet received](#). Additionally, with v1.1.0 and newer, the [PROC\\_PKT](#) pin can be used to indicate the state of the packet engine.

RSSI level is dependent on the power of the signal received at the antenna port and the mode the LNA is in. [regLNAMODE](#) controls the mode of the internal LNA, and has different values/meanings for the

Wi.232FHSS-250-R and Wi.232FHSS-25-R. Figure 13 and Figure 14 below indicate typical traces of RSSI voltage versus signal strength.

### 3.6.7. No Connects (NC)

These pins are currently unused and should be left unconnected or floating.

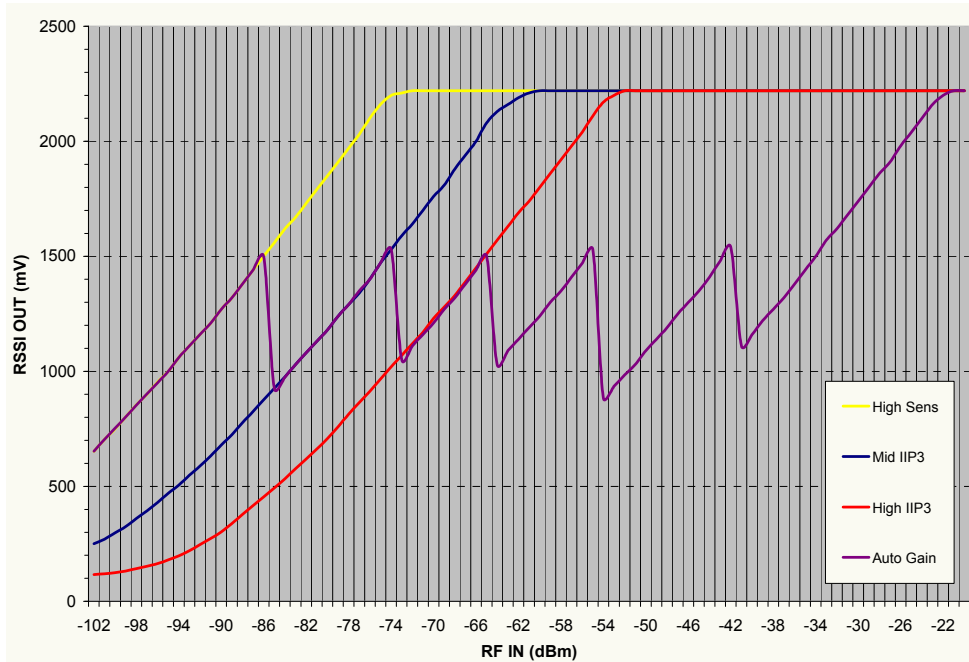
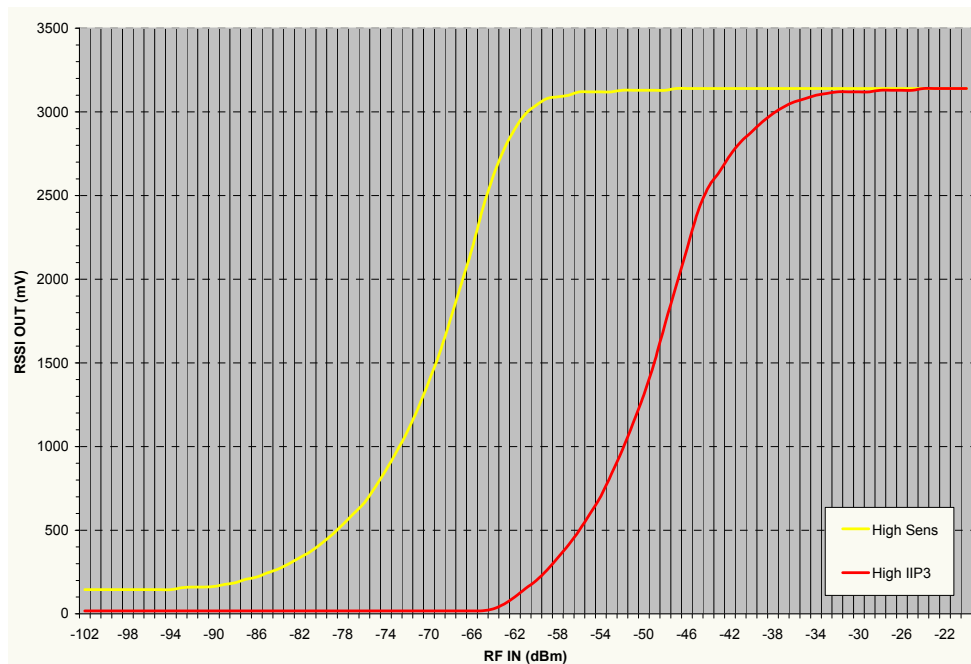


Figure 13: Typical Analog RSSI response (Wi.232FHSS-250-R)





**Figure 14: Typical Analog RSSI response (Wi.232FHSS-25-R)**

### 3.7. Antenna

The module is designed to work with any 50-ohm antenna, including PCB trace antennas.

We are often asked to recommend the best antenna to use with our modules. As a rule, either a  $\frac{1}{4}$  wave whip or  $\frac{1}{2}$  wave dipole antenna paired with a good, solid ground plane are good choices (Radiotronix part numbers ANT-915-03A/ANT-915-02A for  $\frac{1}{4}$  wave whip, SMA and RP-SMA, respectively and ANT-915-09A/ANT-915-06A for  $\frac{1}{2}$  wave dipole, SMA and RP-SMA, respectively). If the application has a small ground plane, the  $\frac{1}{2}$  wave dipole is a better choice. However, many embedded applications cannot support an externally mounted antenna. If this is the case, a PCB antenna must be used. The designer can either use an off-the-shelf PCB antenna, such as the Fractus FR05-S1-R-0-105, or design a trace antenna.

**Note:** Antenna design is difficult and can severely impair the module's performance if done incorrectly. As such, we strongly encourage all of our customers to use off-the-shelf antennas whenever possible. Alternatively, Radiotronix offers design services and can assist with your PCB layout.

### 3.8. Link Budget, Transmit Power, and Range Performance

A link budget is the best figure of merit for comparing wireless solutions and determining how they will perform in the field.

In general, the solution with the best link budget will deliver the best line-of-sight range performance. Improving the link budget by increasing the receiver sensitivity will result in lower power consumption while improving the link budget by increasing the transmit power will result in more robust performance in the presence of an on-channel interferer or multi-path interference.

To calculate the link budget for a wireless link, simply add the transmit power, the antenna gains, and the receiver sensitivity:

$$LB = Ptx + Gtxa - SENSrx + Grxa$$

For example, the link budget for a pair of Wi.232FHSS modules at the minimum data rate and using 2.2dBi ½ wave dipole antennas would be:

(25mW): +13dBm + 2.2dB – (-105dBm) + 2.2dB: 122.4dB

(250mW): +23.5dBm + 2.2dB – (-105dBm) + 2.2dB: 132.9dB

A link budget of 122.4 dB should easily yield a range of 1/2 mile or more outdoors. If the environment is open and the antennas are 8 to 10 feet off of the ground, the range could be more than a mile. Indoors, this link budget should yield a range of several hundred feet. Packet testing of the 250mW module at full power has yielded results in excess of 3 miles at maximum data rate.

The balance of receive sensitivity and transmit power allows good performance indoors in the presence of multi-path while keeping the overall operating current low. This makes these modules suitable for primary battery powered applications such as RFID and automated meter reading.

## 4. Module Configuration

The Wi.232FHSS modules will work right “out of the box” without any configuration. However, a great many configuration registers are exposed to allow for the custom-tailoring of the wireless link. These registers are classified as four different types: non-volatile R/W, non-volatile R/O, volatile R/W, and volatile R/O. During the power-on/reset sequence, the non-volatile read/write registers are copied into the volatile read/write registers. This allows the integrator to determine a default configuration for the wireless link that can be preset at the factory, requiring no intervention by the host processor. During operation, changes to the volatile read/write registers have an immediate effect on the Wi.232FHSS parameters. Non-volatile read-only registers provide a way for the host application to retrieve information about the module hardware, firmware, and hard-coded configuration parameters such as the MyGUID address.

### 4.1. Register Descriptions

#### 4.1.1. CRC Error Count (regCRCERRCOUNT)

					regCRCERRCOUNT (0x40)		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
B7	B6	B5	B4	B3	B2	B1	B0
7	6	5	4	3	2	1	0

The value in this register is augmented each time a packet is received that fails CRC check. Writing 0x00 to this register will initialize the count.

#### 4.1.2. Channel Hop Table (regHOPTABLE)

regNVHOPTABLE (0x00)			regHOPTABLE (0x4B)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RES	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default hop sequence: 0

The Wi.232FHSS supports 6 different hop sequences with minimal correlation. Changing the hop sequence changes the physical band utilization, much the same way that a channel does in a static transmitter. Valid values are 0-5.

### 4.1.3. Power Mode (regPWRMODE)

regNVPWRMODE (0x02)					regPWRMODE (0x4D)		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
NA	NA	NA	NA	NA	NA	PM1	PM0
7	6	5	4	3	2	1	0

Default Power Mode: High Power

The following table shows the available power settings and typical power outputs for the Wi.232FHSS-25-R and Wi.232FHSS-250-R:

PM1	PM0	WI.232FHSS-250-R	WI.232FHSS-25-R
0	0	Low (+8 dBm)	Low (-2 dBm)
0	1	Mid - Low (+13 dBm)	Mid - Low (+3 dBm)
1	0	Mid - High (+18 dBm)	Mid - High (+8 dBm)
1	1	High (+23.5 dBm)	High (+13 dBm)

**Table 11, Power Mode Register Settings**

Wi.232FHSS-25-R modules transmitting at high power in close proximity of one another may cause an increase in data loss. The Wi.232FHSS-250-R incorporates on-board AGC circuitry to mitigate this effect.

### 4.1.4. UART Data Rate (regUARTDARATE)

regNVUARTDARATE (0x03)					regUARTDARATE (0x4E)		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RES	RES	RES	RES	RES	BR2	BR1	BR0
7	6	5	4	3	2	1	0

Default UART Data Rate: 2400 baud

Changing the value in regNVUARTDARATE will change the default data rate at power-on or following a reset (wake from deep sleep is not considered a reset). Changing regUARTDARATE will change the current data rate immediately following the command acknowledgment. Valid settings are:

Baud Rate	BR2	BR1	BR0
2400	0	0	0
9600	0	0	1
19200	0	1	0
38400	0	1	1
57600	1	0	0
115200	1	0	1
10400	1	1	0
31250	1	1	1

**Table 12, Data Rate Register Settings**

**Troubleshooting Hint:** Baud Rate Problems. If you lose track of the baud rate setting of the module, it will be impossible to program the module. You can either try every possible baud rate to discover the setting, or force a power-on reset with CMD held low to set the baud rate to its default: 2.4kbit/second.

## 4.1.5. Network Mode (regNETWORKMODE)

regNVNETMODE (0x04)			regNETMODE (0x4F)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
N/A	N/A	N/A	RA0	EP0	NM2	NM1	NM0
7	6	5	4	3	2	1	0

Default Network Mode: MAC Mode

The module supports three distinct networking modes, MAC, USER, and Extended USER, configured through bits NM0 – NM2. For each of these modes, assured delivery (acknowledgement) can be either enabled or disabled. For more information, see Addressing Modes section of this manual. Valid settings are listed in the table below.

Wi.232FHSS-250-R version 1.1.0 and greater firmware support an additional flag, EP0 (force extended preamble). This flag forces the module to send an extended preamble on every RF transmission. An extended preamble is of a length that allows modules that have just awakened or have not yet synchronized to find and temporarily synchronize with the transmitting module. This can be useful in systems that require the endpoints to spend most of their time sleeping. Endpoints can awaken, receive an application-defined message from the master transmitter, and go back to sleep. This message could contain scheduling information as to when to wake again for a full bi-directional communications session. It is important to note that extended preamble RF packets are not necessarily RF synchronizing packets. Upon completing the reception of an extended preamble RF packet that is not a synchronizer, the receiving module will lose its temporary synchronization and begin looking for another extended preamble/synchronizer on another channel.

Network Mode	Meaning
0x04	MAC Address Mode
0x05	Reserved
0x06	User Address Mode
0x07	User Extended Address Mode
0x14	MAC Address Mode with Acknowledgement
0x15	Reserved
0x16	MAC Address Mode with Acknowledgement
0x17	User Extended Address Mode with Acknowledgement

**Table 13, Network Mode Register Settings**

All other network modes are reserved and may cause undesired operation.

**Troubleshooting Hint:** If you are unable to communicate with another module, it is most likely one of following problems. First, check to make sure that you are using the same data rate. Modules programmed to different data rates will not communicate nor share the RF channel with one another. Second, ensure that your network mode and addressing is configured to properly access the module of interest. Also, ensure that you are addressing a specific module when using acknowledgment. Failure to do so will cause large delays and loss of data.

## 4.1.6. Transmit Wait Timeout (regTXTO)

regNVTXTO (0x05)					regTXTO (0x50)		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default Transmit Wait Timeout: 16 (15-16ms)

When a byte is received by the UART, the module will reset the UART Tx timeout millisecond timer to **regTXTO** Underline and blue. If the timer expires before another byte is received by the UART, the data is queued for transmission. Normally, this value should be set to a value greater than one byte time at the current UART data rate (rounding up) with a minimum of 0x02. **regTXTO** should never be set to a value of 0x01 or any value less than one byte time; unpredictable results will occur. If the timeout value is set to 0x00, the transmit wait timeout will be disabled, and a minimum of [regUARTMTU](#) bytes will be required for transmission.

This register works in conjunction with the [regUARTMTU](#) register to determine when the module sends a packet. A pseudo-code logical arrangement is:

```
if (UART_incoming_buffer_size >= regUARTMTU) or (UART_last_char_received_timer >
regTXTO) then SendPacket()
```

Wi.232FHSS-25-R / Wi.232FHSS-250-R	
Baud Rate	Minimum TXTO
2400	6ms
9600	3ms
19200	2ms
38400	2ms
57600	2ms
115200	2ms

**Table 14, Minimum TXTO Values**

## 4.1.7. Maximum Transmit Retries (regMAXTXRETRY)

regNVMAXTXRETRY (0x07)					regMAXTXRETRY (0x52)		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default Maximum Transmit Retries: 26

Sets the number of transmission tries if an acknowledgement is not received. If an acknowledgement is not received, [EX\\_NORFACK](#) is raised.

Wi.232FHSS-25-R / Wi.232FHSS-250-R	
Baud Rate	EX_NORFACK Timeout
2400	170ms
9600	75ms
19200	45ms
38400	30ms
57600	30ms
115200	30ms

**Table 15, Per-Baud rate ACK timeout values.**

### 4.1.8. CRC Checking (regUSECRC)

regNVUSECRC (0x08)			regUSECRC (0x53)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
B7	B6	B5	B4	B3	B2	B1	B0
7	6	5	4	3	2	1	0

Default CRC Checking: Enabled (0x01)

Set to 0x01 to enable receiver CRC-16 validation, or 0x00 to disable. If this register is enabled (0x01), an invalid packet will not be sent to the UART interface for transmission; it will be discarded. The default CRC mode setting is enabled.

### 4.1.9. UART Minimum Transmission Unit (regUARTMTU)

regNVUARTMTU (0x09)			regUARTMTU (0x54)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
B7	B6	B5	B4	B3	B2	B1	B0
7	6	5	4	3	2	1	0

Default UART Minimum Transmission Unit: 64

This register determines the UART buffer level that will trigger the transmission of a packet. This register works in conjunction with the [regTXTO](#) register to determine when the module sends a packet. A pseudo-code logical arrangement is:

```
if (UART_incoming_buffer_size >= regUARTMTU) or (UART_last_char_received_timer > regTXTO) then SendPacket()
```

The minimum value is 1 and the maximum value is 192. The default value for this register is 64, which provides a good mix of throughput and latency. At maximum data rate, a value of 128 will optimize throughput. This register does not guarantee a particular transmission unit size; rather, it specifies the minimum desired size. If there is not enough time left in a hop, for instance, the packet engine will send as many characters as it can to fill the current hop, and send the remaining characters in the next hop.

## 4.1.10. Verbose Mode (regSHOWVER)

regNVSHOWVER (0x0A)					regSHOWVER (0x55)		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
B7	B6	B5	B4	B3	B2	B1	B0
7	6	5	4	3	2	1	0

Default Verbose Mode: Enabled (0x01)

Setting this register to 0x00 will suppress the start-up message, including firmware version, which is sent to the UART when the module is reset. A value of 0x01 will cause the message to be displayed after reset. By default, the module start-up message will be displayed. The non-volatile counterpart, regSHOWVER, has no function.

Verbose Mode	Meaning
0x00	Startup message will NOT be displayed upon reset or power-up
0x01	Startup message will be displayed upon reset or power-up. This is a blocking call, and any incoming UART data will be lost during the transmission of this message through the TxD pin. All UART commands must be sent after this message has completed.
0x02	Startup message will be displayed upon reset or power-up. This is a non-blocking call. Any incoming UART data will be buffered, and incoming UART commands will be processed. If a change of baud rate is commanded while the startup message is being output, the current byte will finish at the current baud rate, and subsequent bytes will be transmitted at the new baud rate. * Available in 250-R firmware 1.1.0 and newer only.

**Table 16, Verbose Mode Settings**

## 4.1.11. CSMA Enable (regCSMAMODE)

regNVCSMAMODE (0x0B)					regCSMAMODE (0x56)		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
B7	B6	B5	B4	B3	B2	B1	B0
7	6	5	4	3	2	1	0

Default CSMA Enable: Enabled (0x01)

Carrier-sense multiple access (CSMA) is a best-effort delivery system that listens to the channel before transmitting a message. If another Wi.232 module is already transmitting when a message is queued, the module will wait before sending its payload. This helps to eliminate RF message corruption at the expense of additional latency. Setting this register to 0x01 will enable CSMA. Setting this register to 0x00 will disable CSMA. By default, CSMA is enabled.



## 4.1.12. Operating Mode (regOPMODE)

regNVOPMODE (0x0D)					regOPMODE (0x58)		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
B7	B6	B5	B4	B3	B2	B1	B0
7	6	5	4	3	2	1	0

Default Operating mode: Awake (0x00)

Changing this register will place the register in the operating modes indicated in the table below. If the module remains properly powered, and is awakened from Sleep, Standby, or Deep Sleep properly, the volatile registers retain their values when awakened.

Awake mode is the normal operating mode. This is the only mode in which the RF circuitry is able to receive and transmit RF messages.

Sleep and Standby are modes intended for legacy Wi.232DTS™ developers. Because the Wi.232FHSS-25/250-R modules do not include the hardware to properly implement these two modes, current consumption is much higher than in the Wi.233DTS™. For this reason, use of these two modes is not recommended for new development. Standby leaves the RF oscillator circuit operating for faster wakeup, whereas Sleep does not. In order to wake the module from these two modes, send one byte of 0x0F to the module's RxD pin (CMD left high) at the current baud rate. If [regACKONWAKE](#) is enabled (default), the module will transmit a 0x06 byte through the TxD pin when the module is awake and ready to accept data or commands.

Deep Sleep mode disables all circuitry on-board the module. This is the lowest-power mode available for the module. When in Deep Sleep, the CTS pin is brought low, and both RF and microcontroller oscillators are stopped. In order to wake the module, a negative-going pulse on the C2CK/RST pin of at least 15µs is required. The module will begin the wake process once the C2CK/RST pin is returned high. If this line experiences digital noise or glitching, it can cause multiple triggers (wake from sleep, hardware reset, hardware reset, etc.). Multiple triggers can cause the reloading of volatile registers with their non-volatile values. If your circuit introduces noise onto this pin, it should be filtered or bypassed with a capacitor to ground or an RC filter.

If the volatile registers have been corrupted during sleep/standby/deep sleep, a software reset is forced and the module will reboot. Possible causes of this are power surges or brownout. The module will display the startup message (if not disabled in [regNVSHOWVER](#)), and send a 0x06 byte out the UART TxD pin (if not disabled in [regNVACKONWAKE](#)) when the module is ready to accept commands and data.

Operating Mode	B1	B0
Awake	0	0
Sleep	0	1
Standby	1	0
Deep Sleep	1	1

**Table 17, Operating Mode Register Settings**

### 4.1.13. UART Acknowledge on Wake (regACKONWAKE)

regNVACKONWAKE (0x0E)			regACKONWAKE (0x59)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default UART Acknowledge on Wake: Enabled (0x01)

When UART Acknowledge on Wake is enabled, the module will send an ACK (0x06) character out of the UART TxD pin after the module wakes. This indicates that the module is ready to accept data and command traffic. A value of 0x01 enables this feature. Setting the register to 0x00 disables it.

### 4.1.14. User Destination ID[3] (regUSERDESTID[3])

regNVUSERDESTID[3] (0x0F)			regUSERDESTID[3] (0x5A)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default User Destination ID[3]: 0xFF

This is the most significant byte (byte 3) of the 32-bit user extended destination address. When Extended User addressing is enabled, this register is used in conjunction with User Destination Address[2-0] registers to direct a transmitted packet to the proper endpoint(s). For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

### 4.1.15. User Destination ID[2] (regUSERDESTID[2])

regNVUSERDESTID[2] (0x10)			regUSERDESTID[2] (0x5B)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default User Destination ID[2]: 0xFF

This is byte 2 of the 32-bit user extended destination address. When Extended User addressing is enabled, this register is used in conjunction with User Destination ID[3] and User Destination ID[1-0] registers to direct a transmitted packet to the proper endpoint(s). For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

### 4.1.16. User Destination ID[1] (regUSERDESTID[1])

regNVUSERDESTID[1] (0x11)			regUSERDESTID[1] (0x5C)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default User Destination ID[1]: 0xFF

When user-extended addressing is selected, this is byte 1 of the 32-bit user-extended destination address; when user addressing is selected, it is the most significant byte of the user destination address. When extended-user addressing is selected, this register is used in conjunction with User Destination ID[3-2] and User Destination ID[0] registers to direct a transmitted packet to the proper endpoint(s). When user addressing is selected, this register is used in conjunction with the User Destination ID[0] register to direct a transmitted packet to the proper endpoint(s). For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

#### 4.1.17. User Destination ID[0] (regUSERDESTID[0])

regNVUSERDESTID[0] (0x12)			regUSERDESTID[0] (0x5D)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default User Destination ID[0]: 0xFF

When user-extended addressing is selected, this is the least significant byte of the 32-bit extended-user destination address; when user addressing is selected, it is the least significant byte of the user destination address. When user-extended addressing is selected, this register is used in conjunction with User Destination Address[3-1] registers to direct a transmitted packet to the proper endpoint(s). When user addressing is selected, this register is used in conjunction with the User Destination Address[1] register to direct a transmitted packet to the proper endpoint(s). For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

#### 4.1.18. User Source ID[3] (regUSERSRCID[3])

regNVUSERSRCID[3] (0x13)			regUSERSRCID[3] (0x5E)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default User Source ID[3]: 0xFF

When user-extended addressing is selected, this register holds the most significant byte (byte 3) of the 32-bit user-extended source address. When Extended User addressing is invoked, this register is used in conjunction with User Source Address[2-0] registers to make up the module's 32-bit user-extended source address. For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

### 4.1.19. User Source ID[2] (regUSERSRCID[2])

regNVUSERSRCID[2] (0x14)			regUSERSRCID[2] (0x5F)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default User Source ID[2]: 0xFF

When user-extended addressing is selected, this register holds byte 2 of the 32-bit user-extended source address. When Extended User addressing is invoked, this register is used in conjunction with User Source Address [3] and User Source Address [1-0] registers to make up the module's 32-bit extended-user source address. For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

### 4.1.20. User Source ID[1] (regUSERSRCID[1])

regNVUSERSRCID[1] (0x15)			regUSERSRCID[1] (0x60)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default User Source ID[1]: 0xFF

When user-extended addressing is selected, this register holds byte 1 of the 32-bit extended-user source address; when user addressing is selected, this register holds the most significant byte of the 16-bit user source address. When extended-user addressing is selected, this register is used in conjunction with User Source Address[3-2] and User Source Address[0] registers to make up the module's 32-bit user-extended source address. When user addressing is selected, this register is used in conjunction with the User Source Address[0] register to make up the module's 16-bit user source address. For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

### 4.1.21. User Source ID[0] (regUSERSRCID[0])

regNVUSERSRCID[0] (0x16)			regUSERSRCID[0] (0x61)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default User Source ID[0]: 0xFF

When extended-user addressing is selected, this register holds the least significant byte (byte 0) of the 32-bit user-extended source address; when user addressing is selected, this register holds the least significant byte of the 16-bit user address. When extended-user addressing is selected, this register is used in conjunction with User Source Address[3-1] registers to make up the module's 32-bit user-extended source address. When user addressing is selected, this register is used in conjunction with the User Source Address[1] register to make up the module's 16-bit user source address. For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

## 4.1.22. User ID Mask[3] (regUSERIDMASK[3])

regNVUSERIDMASK[3] (0x17)					regUSERIDMASK[3] (0x62)		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default User ID Mask[3]: 0xFF

This register holds the most significant byte of the 32-bit extended-user address mask. When Extended User addressing is invoked, this register is used in conjunction with User ID Mask[2-0] registers to make up the module's 32-bit extended-user address mask. For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

## 4.1.23. User ID Mask[2] (regUSERIDMASK[2])

regNVUSERIDMASK[2] (0x18)					regUSERIDMASK[2] (0x63)		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default User ID Mask[2]: 0xFF

This register holds byte 2 of the 32-bit extended-user address mask. When Extended User addressing is invoked, this register is used in conjunction with User ID Mask[3] and User ID Mask[1-0] registers to make up the module's 32-bit extended-user address mask. For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

## 4.1.24. User ID Mask[1] (regUSERIDMASK[1])

regNVUSERIDMASK[1] (0x19)					regUSERIDMASK[1] (0x64)		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default User ID Mask[1]: 0xFF

When extended-user addressing is invoked, this register holds byte 1 of the 32-bit extended-user address mask and is used in conjunction with User ID Mask[3-2] and User ID Mask[0] to form the complete address mask; when user addressing is invoked, this register holds the most significant byte of the 16-bit user address mask and is used in conjunction with the User ID Mask[0] register to form the complete address mask. For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

### 4.1.25. User ID Mask[0] (regUSERIDMASK[0])

regNVUSERIDMASK[0] (0x1A)					regUSERIDMASK[0] (0x65)		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default User ID Mask[0]: 0xFF

When extended-user addressing is invoked, this register holds the least significant byte of the 32-bit user-extended address mask and is used in conjunction with User ID Mask[3-1] to form the complete address mask; when user addressing is invoked, this register holds the least significant byte of the 16-bit user address mask and is used in conjunction with the User ID Mask[1] register to form the complete address mask. For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

### 4.1.26. Destination GUID[3] (regDESTGUID[3])

regNVDESTGUID[3] (0x1D)					regDESTGUID[3] (0x68)		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default Destination GUID[3]: 0xFF

When MAC addressing is invoked, this register holds the most significant byte of the 32-bit destination GUID. This register is used in conjunction with Destination GUID[2-0] registers to make up the module's 32-bit destination GUID. For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

### 4.1.27. Destination GUID[2] (regDESTGUID[2])

regNVDESTGUID[2] (0x1E)					regDESTGUID[2] (0x69)		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default Destination GUID[2]: 0xFF

When MAC addressing is invoked, this register holds byte 2 of the 32-bit destination GUID. This register is used in conjunction with Destination GUID[3] and Destination GUID[1-0] registers to make up the module's 32-bit destination GUID. For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

### 4.1.28. Destination GUID[1] (regDESTGUID[1])

regNVDESTGUID[1] (0x1F)			regDESTGUID[1] (0x6A)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default Destination GUID[1]: 0xFF

When MAC addressing is invoked, this register holds byte 1 of the 32-bit destination GUID. This register is used in conjunction with Destination GUID[3-2] and Destination GUID[1] registers to make up the module's 32-bit destination GUID. For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

### 4.1.29. Destination GUID[0] (regDESTGUID[0])

regNVDESTGUID[0] (0x20)			regDESTGUID[0] (0x6B)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default Destination GUID[0]: 0xFF

When MAC addressing is invoked, this register holds the least significant byte of the 32-bit destination GUID. This register is used in conjunction with Destination GUID[3-1] registers to make up the module's 32-bit destination GUID. For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

### 4.1.30. Exception Mask (regEXCEPTIONMASK)

regNVEXCEPTIONMASK (0x21)			regEXCEPTIONMASK (0x6C)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default Exception Mask: 0xFF

The module has a built-in exception engine that can notify the host processor of an unexpected event. When an exception occurs, this register is ANDed with the exception code. A non-zero result causes an assertion of the EX pin. Reading the [regEXCEPTION](#) register will clear the exception and return the EX pin to low. If the result is zero, the EX pin is not asserted, but the exception code is stored in the [regEXCEPTION](#) register.

### 4.1.31. CMD Halts Traffic (regCMDHALT)

regNVCMDHALT (0x23)			regCMDHALT (0x6E)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default CMD Halt Setting: Disabled (0x00)

When configuring the module's register settings, it is possible that incoming RF transmissions can intermix with the command interpreter's response, making it difficult to determine if your commands were successfully processed. Changing this register setting to 0x01 will cause the module to store incoming RF traffic (up to RF buffer overflow) while the CMD line is low. When the CMD pin is returned high, the RF engine will send all buffered data to the UART. This feature is present in all Wi.232FHSS-250-R modules and Wi.232FHSS-25(-R) modules with firmware version 1.0.4 or newer.

### 4.1.32. Receiver LNA Mode (regLNAMODE)

regNVLNAMODE (0x24)			regLNAMODE (0x6F)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default Receiver LNA Mode: High Sensitivity (Wi.232FHSS-25), AGC Enabled (Wi.232FHSS-250-R)

By default, the Wi.232FHSS-25 module is configured for maximum receiver sensitivity. Similarly, the Wi.232FHSS-250-R is factory-configured to use its internal automatic gain control (AGC) circuit to manage receiver sensitivity. Reducing the gain increases the linearity of the receiver, but reduces maximum sensitivity; increasing the gain does the opposite. Generally speaking, higher linearity (increased third order input intercept point, IIP3) will give improved performance in high-interference environments; high gain yields better performance in low-interference environments.

The Wi.232FHSS-250-R contains an AGC circuit that manages these settings automatically, and it should be used whenever possible. When reading the digital RSSI registers ([regIMMEDRSSI](#), [regLGPRSSI](#)), the internal calculation automatically compensates for the current LNA gain setting. However, when attempting to make *analog* RSSI measurements, fixing the LNA gain will produce more meaningful results.

The following table summarizes the LNA settings for each of the modules.

LNA Mode	WI.232FHSS-25-R			WI.232FHSS-250-R		
	Meaning	IIP3 Increase	Sensitivity Decrease	Meaning	IIP3 Increase	Sensitivity Decrease
0x00	High Sensitivity	Reference	Reference	AGC Enabled	Variable	Variable
0x01	High Linearity	15 dB	10 dB	High Sensitivity	Reference	Reference
0x02	Invalid	N/A	N/A	Mid Linearity	19.1 dB	6.5 dB
0x03	Invalid	N/A	N/A	High Linearity	41.8 dB	9.5 dB

**Table 18, LNA Mode Settings**



### 4.1.33. Compatibility Mode (regCOMPATMODE)

regNVCOMPATMODE (0x25)			regCOMPATMODE (0x70)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default Compatibility Mode: Disabled (0x00)

The Wi.232FHSS-250-R operates at a much narrower receive bandwidth (200kHz) than the Wi.232FHSS-25 (600kHz). When the Wi.232FHSS-25 transmits, it deviates well outside the receiver bandwidth of the Wi.232FHSS-250-R. *To address this problem, v1.0.5 and later Wi.232FHSS-25 firmware and all Wi.232FHSS-250-Rs support a compatibility mode.* This allows both modules to communicate effectively with each other.

When enabled (0x01), compatibility mode reduces the maximum RF data rate to 76.8kbps, reduces the Wi.232FHSS-25's deviation to 80kHz, and reduces the Wi.232FHSS-25's receiving bandwidth to 200kHz. All UART baud rates are supported, although the RF data rates associated with baud rates 31250, 38400, 57600, and 115200 will be diminished.

### 4.1.34. Auto Address Mode (regAUTADDMODE)

regNVAUTADDMODE (0x26)			regAUTADDMODE (0x71)				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default Auto Address Mode: Disabled (0x00)

When this register is enabled, the Wi.232FHSS-250-R module reads the Source Address from an incoming packet and auto-populates this information into the respective Destination Address register. This ensures that a subsequent transmission is sent only to the relevant module, in a point-to-point configuration.

The upper 4 bits of the volatile register contains the last-received packet type. The values of these four bits are the same as the mode settings below. For instance, set regAUTADDMODE to 0x0F (Any Auto Address), and then receive a MAC packet from another module. When you read back the regAUTADDMODE register, the register value will be 0x4F. The lower 4-bits indicate the setting of the register (0xF, Any Auto Address), and the upper 4 bits indicate the type of packet that was received (0x4, MAC Address).

The table below summarizes the allowed values for the lower 4-bits of the register. The lower 4-bits contain the value that controls the operation of this feature. The upper 4-bits are not read by the module and are only written upon a successful packet reception:

Auto Address Mode	Meaning	Action
0x00	Auto Address Mode disabled	Destination Registers not populated
0x04	MAC Auto Address Mode	Auto-populates MAC Address Destination Register Only
0x06	User Auto Address Mode	Auto-populates User Address Destination Register
0x07	Extended User Auto Address Mode	Auto-populates User Address Destination Register
0x0F	Any Auto Address Mode	Auto-populates MAC Address Destination Register
0x01-0x03, 0x05, 0x08- 0x0E	Undefined	Undefined

**Tables 19, AutoAddress Register Settings**

### 4.1.35. My GUID[3] (regMYGUID[3])

regMYGUID[3] (0x34)							
R	R	R	R	R	R	R	R
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default My GUID[3]: 0xFF

This is the most significant byte of the 32-bit factory-programmed, read-only MAC/GUID address. The complete MyGUID address is unique amongst Wi.232 modules. The MyGUID[3-0] unique address is used by all packet types as a unique origination address. This register is used in conjunction with My GUID[2-0] registers to make up the module's 32-bit source GUID. For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

### 4.1.36. My GUID[2] (regMYGUID[2])

regMYGUID[2] (0x35)							
R	R	R	R	R	R	R	R
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default My GUID[2]: 0xFF

This is byte 2 of the 32-bit factory-programmed, read-only MAC/GUID address. The complete MyGUID address is unique amongst Wi.232 modules. The MyGUID[3-0] unique address is used by all packet types as a unique origination address. This register is used in conjunction with My GUID[3] and My GUID[1-0] registers to make up the module's 32-bit source GUID. For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

### 4.1.37. My GUID[1] (regMYGUID[1])

regMYGUID[1] (0x36)							
R	R	R	R	R	R	R	R
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default My GUID[1]: 0xFF

This is byte 2 of the 32-bit factory-programmed, read-only MAC/GUID address. The complete MyGUID address is unique amongst Wi.232 modules. The MyGUID[3-0] unique address is used by all packet types as a unique origination address. This register is used in conjunction with My GUID[3-2] and My GUID[0] registers to make up the module's 32-bit source GUID. For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

### 4.1.38. My GUID[0] (regMYGUID[0])

regMYGUID[0] (0x37)							
R	R	R	R	R	R	R	R
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

Default My GUID[0]: 0xFF

This is byte 2 of the 32-bit factory-programmed, read-only MAC/GUID address. The complete MyGUID address is unique amongst Wi.232 modules. The MyGUID[3-0] unique address is used by all packet types as a unique origination address. This register is used in conjunction with My GUID[3-1] registers to make up the module's 32-bit source GUID. For more information on the operation of addressing modes, please refer to the [Addressing Modes](#) section of this document.

### 4.1.39. Release Number (regRELEASENUM)

					regRELEASENUM (0x78)		
R	R	R	R	R	R	R	R
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

This register contains a hard-coded release number corresponding to a firmware version and hardware platform. The following table lists current releases to date:

Wi.232FHSS-25-R	
Release Number	Version Number
0x08	1.0.0
0x0A	1.0.2
0x0C	1.0.4
0x0E	1.0.4a
0x0F	1.0.5

Wi.232FHSS-250-R	
Release Number	Version Number
0x0D	1.0.5
0x10	1.0.5a
0x11	1.1.0
0x14	1.1.0 (Brazil)

Tables 20 and 21, Release Number Register Settings

## 4.1.40. Exception (regEXCEPTION)

			regEXCEPTION (0x79)				
R	R	R	R	R	R	R	R
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

The module has a built-in exception engine that can notify the host processor of an unexpected event. When an exception occurs, the exception code is stored in this register. Reading from this register clears the exception and, if applicable, returns the EX pin to low. If an exception occurs before the previous exception code is read, the previous value is overwritten. The following table lists the exception codes and their values.

Exception Code	Exception Name	Description
0x08	EX_BUFOVFL	Internal UART buffers overflowed
0x09	EX_RFOVL	Internal RF packet buffer overflowed
0x13	EX_WRITEREGFAILED	Attempted write to register failed
0x20	EX_NOACK	Acknowledgment 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	Unsupported Frame type specified

**Table 22, Exception Codes**

## 4.1.41. Last Good Packet RSSI (regLGPRSSI)

			regLGPRSSI (0x7B)				
R	R	R	R	R	R	R	R
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

This register holds the receive strength indication, in dBm, of the last successful packet reception. A successful packet reception is one that causes payload data to be delivered to the UART interface. Each time a new packet is successfully processed, the value in this register is overwritten. The register value is an 8-bit signed integer representing the RSSI in dBm. It is accurate to +/-3dB and has +/-2dB linearity. The values returned take into account the LNA gain.

## 4.1.42. Immediate RSSI (regIMMEDRSSI)

			regIMMEDRSSI (0x7C)				
R	R	R	R	R	R	R	R
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

This register returns the current receive signal strength indication, in dBm. When the register is accessed, the RF interface is queried at that time. The register value is an 8-bit signed integer representing the RSSI

in dBm. It is accurate to +/-3dB and has +/-2dB linearity. The values returned take into account the LNA gain.

## 4.2. Register Summary

### 4.2.1. Volatile Read/ Write Registers

Volatile Read/ Write Registers		
Name	Address	Description
<a href="#">regCRCERRCOUNT</a>	0x40	CRC error count value
<a href="#">regHOPTABLE</a>	0x4B	Hop Table
<a href="#">regPWRMODE</a>	0x4D	Power amplifier setting
<a href="#">regUARTDATARATE</a>	0x4E	UART data rate
<a href="#">regNETWORKMODE</a>	0x4F	Sets addressing mode
<a href="#">regTXTO</a>	0x50	UART to transmitter timeout
<a href="#">regMAXTXRETRY</a>	0x52	Maximum times to retry packet transmission
<a href="#">regUSECRC</a>	0x53	Enable/Disable CRC checking
<a href="#">regUARTMTU</a>	0x54	Minimum transmission unit
<a href="#">regSHOWVERSION</a>	0x55	Non-functional – Use regNVSHOWVERSION
<a href="#">regCSMAMODE</a>	0x56	Enable/disable CSMA
<a href="#">regOPMODE</a>	0x58	Sets operating mode
<a href="#">regACKONWAKE</a>	0x59	Enable/Disable ACK sent to UART upon wake from
<a href="#">regUSERDESTID[3]</a>	0x5A	Destination Address for User Packet Type, extended
<a href="#">regUSERDESTID[2]</a>	0x5B	Destination Address for User Packet Type, extended
<a href="#">regUSERDESTID[1]</a>	0x5C	Destination Address for User Packet Type
<a href="#">regUSERDESTID[0]</a>	0x5D	Destination Address for User Packet Type
<a href="#">regUSERSRCID[3]</a>	0x5E	Source Address for User Packet Type, extended
<a href="#">regUSERSRCID[2]</a>	0x5F	Source Address for User Packet Type, extended
<a href="#">regUSERSRCID[1]</a>	0x60	Source Address for User Packet Type
<a href="#">regUSERSRCID[0]</a>	0x61	Source Address for User Packet Type
<a href="#">regUSERIDMASK[3]</a>	0x62	Address Mask for User Packet Type, extended
<a href="#">regUSERIDMASK[2]</a>	0x63	Address Mask for User Packet Type, extended
<a href="#">regUSERIDMASK[1]</a>	0x64	Address Mask for User Packet Type
<a href="#">regUSERIDMASK[0]</a>	0x65	Address Mask for User Packet Type
<a href="#">regDESGUID[3]</a>	0x68	Sets MAC Address Destination
<a href="#">regDESGUID[2]</a>	0x69	Sets MAC Address Destination
<a href="#">regDESGUID[1]</a>	0x6A	Sets MAC Address Destination
<a href="#">regDESGUID[0]</a>	0x6B	Sets MAC Address Destination
<a href="#">regEXCEPTIONMASK</a>	0x6C	Exception and Mask used to activate exception pin
<a href="#">regCMDHALT*</a>	0x6E	Halt RF traffic when CMD pin is low
<a href="#">regLNAMODE*</a>	0x6F	Receiver LNA gain/linearity setting
<a href="#">regCOMPATMODE**</a>	0x70	Compatibility mode for 25(-R) / 250-R intercommunication
<a href="#">regAUTADDMODE**</a>	0x71	Sets Auto-address mode
* supported in 25(-R) module f/w v1.0.4 and 250-R f/w v1.0.5 and newer		
** supported in 250(-R) module f/w v1.0.5 and newer		

**Table 23, Volatile Read/ Write Register Summary**

## 4.2.2. Volatile Read-Only Registers

Volatile Read-Only Registers		
Name	Address	Description
regEXCEPTION	0x79	Stores latest exception code
regLGPRSSI**	0x7B	Last Good Packet RSSI value
regIMMEDRSSI**	0x7C	Current RSSI value
<i>** supported in 250(-R) module f/w v1.0.5 and newer</i>		

**Table 24, Volatile Read-Only Register Summary**

## 4.2.3. Non-Volatile Read-Only Registers

Non-Volatile Read-Only Registers		
Name	Address	Description
<a href="#">regMYGUID[3]</a>	0x34	Factory programmed GUID used in MAC Addressing Mode
<a href="#">regMYGUID[2]</a>	0x35	Factory programmed GUID used in MAC Addressing Mode
<a href="#">regMYGUID[1]</a>	0x36	Factory programmed GUID used in MAC Addressing Mode
<a href="#">regMYGUID[0]</a>	0x37	Factory programmed GUID used in MAC Addressing Mode
regCUSTID[1]	0x39	Factory programmed customer ID, default 0xFF.
regCUSTID[0]	0x3A	Factory programmed customer ID, default 0xFF.
regRELEASENUM	0x78	Holds release number indicating h/w and f/w

**Table 25, Non-Volatile Read-Only Register Summary**

## 4.2.4. Non-Volatile Read/ Write Registers

Non-Volatile Read/ Write Registers			
Name	Address	Description	Factory Default
<a href="#">regNVHOPTABLE</a>	0x00	Hop Table	0
<a href="#">regNVPWRMODE</a>	0x02	Power amplifier setting	3 (High Power)
<a href="#">regNVUARTDatarate</a>	0x03	UART data rate	0 (2400)
<a href="#">regNVNETWORKMODE</a>	0x04	Sets addressing mode	4 (MAC/GUID)
<a href="#">regNVTXTO</a>	0x05	UART to transmitter timeout	16 (15-16ms)
<a href="#">regNVMAXTXRETRY</a>	0x07	Maximum times to retry packet transmission	26
<a href="#">regNVUSECRC</a>	0x08	Enable/Disable CRC checking	1 (Enabled)
<a href="#">regNVUARTMTU</a>	0x09	Minimum transmission unit	64 (64 bytes)
<a href="#">regNVSHOWVERSION</a>	0x0A	Enable/disable startup message	1 (Enabled)
<a href="#">regNVCSMAMODE</a>	0x0B	Enable/disable CSMA	1 (Enabled)
<a href="#">regNVOPMODE</a>	0x0D	Sets operating mode	0 (Awake)
<a href="#">regNVACKONWAKE</a>	0x0E	Enable/Disable ACK sent to UART upon wake from	1 (Enabled)
<a href="#">regNVUSERDESTID[3]</a>	0x0F	Destination Address for User Packet Type, extended	0xFF
<a href="#">regNVUSERDESTID[2]</a>	0x10	Destination Address for User Packet Type, extended	0xFF
<a href="#">regNVUSERDESTID[1]</a>	0x11	Destination Address for User Packet Type	0xFF
<a href="#">regNVUSERDESTID[0]</a>	0x12	Destination Address for User Packet Type	0xFF
<a href="#">regNVUSERSRCID[3]</a>	0x13	Source Address for User Packet Type, extended	0xFF
<a href="#">regNVUSERSRCID[2]</a>	0x14	Source Address for User Packet Type, extended	0xFF
<a href="#">regNVUSERSRCID[1]</a>	0x15	Source Address for User Packet Type	0xFF
<a href="#">regNVUSERSRCID[0]</a>	0x16	Source Address for User Packet Type	0xFF
<a href="#">regNVUSERIDMASK[3]</a>	0x17	Address Mask for User Packet Type, extended	0xFF
<a href="#">regNVUSERIDMASK[2]</a>	0x18	Address Mask for User Packet Type, extended	0xFF
<a href="#">regNVUSERIDMASK[1]</a>	0x19	Address Mask for User Packet Type	0xFF
<a href="#">regNVUSERIDMASK[0]</a>	0x1A	Address Mask for User Packet Type	0xFF
<a href="#">regNVDESTGUID[3]</a>	0x1D	Sets MAC Address Destination	0xFF
<a href="#">regNVDESTGUID[2]</a>	0x1E	Sets MAC Address Destination	0xFF
<a href="#">regNVDESTGUID[1]</a>	0x1F	Sets MAC Address Destination	0xFF
<a href="#">regNVDESTGUID[0]</a>	0x20	Sets MAC Address Destination	0xFF
<a href="#">regNVEXCEPTIONMASK</a>	0x21	Used to mask exception for EX pin.	0xFF (All)
<a href="#">regCMDHALT*</a>	0x23	Halt RF traffic when CMD pin is low	0 (Disabled)
<a href="#">regLNAMODE*</a>	0x24	Receiver LNA gain/linearity setting	0 (High Gain[25] / Auto[250])
<a href="#">regCOMPATMODE**</a>	0x25	Compatibility mode for 25(-R) / 250-R intercommunication	0 (Disabled)
<a href="#">regAUTADDMODE**</a>	0x26	Auto-address mode	0 (Disabled)

\* supported in 25(-R) module f/w v1.0.4 and 250-R f/w v1.0.5 and newer  
\*\* supported in 250(-R) module f/w v1.0.5 and newer

**Table 26, Non-Volatile Read/ Write Register Summary**

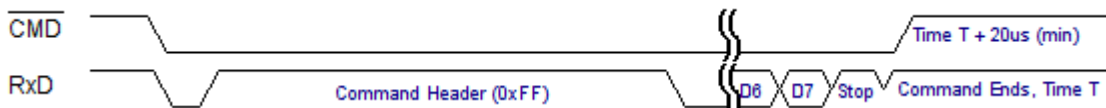


## 5. Using Configuration Registers

### 5.1. CMD Pin

The CMD pin is used to inform the module where incoming UART information should be routed. When the CMD pin is high or left floating, all incoming UART information is treated as payload data and transferred over the wireless interface. If the CMD pin is low, the incoming UART data is routed to the command parser for processing. Since the module's processor looks at UART data one byte at a time, the CMD line must be held low for the entire duration of the command plus a 20 $\mu$ s margin for processing. Leaving the CMD pin low for additional time (for example, until the ACK byte is received by your application) will not adversely affect the module. If RF packets are received while the CMD line is active, they are still processed and presented to the module's UART for transmission.

The CMD pin is also used during the module startup process to determine whether or not to reload the non-volatile registers with factory defaults. The module startup or boot process is executed when the module is powered on from a power off state, or is issued a software or hardware reset. When the module goes through this boot process, it checks the state of the CMD pin. If it is low, the module will clear the non-volatile registers and re-populate them with factory default values. Possible reset sources that could cause the module to reboot are power supply brown-out, power supply instability, noise present on C2CK/RST pin, noise/voltage spikes on digital I/O pins, issuing a reset command through the command interface, and toggling the C2CK/RST pin when not in deep sleep.



**Figure 15: Command and CMD Pin Timing**

### 5.2. CMD\_RSP Pin

The CMD\_RSP pin can be used to determine whether the outbound UART traffic is actual data or a response from the command interpreter. This pin is normally high. When the module responds to a serial command, such as a register read or write, the pin is pulled low for the duration of the outbound command response. Monitoring of this pin is optional. It should be tied to a CMOS input or left floating.

### 5.3. Command Formatting

The Wi.232FHSS module contains several volatile and non-volatile registers that control its configuration and operation. The volatile registers all have a non-volatile mirror register that is used to determine the

default configuration when power is applied to the module. During normal operation, the volatile registers are used to control the module.

Placing the module in the command mode allows these registers to be programmed. Byte values in excess of 127 (0x80 or greater) must be changed into a two-byte escape sequence of the format: 0xFE, [value - 128]. For example, the value 0x83 becomes 0xFE, 0x03. The following function will prefix a 0xFF header and size specifier to a command sequence and create escape sequences as needed. It is assumed that *\*src* is populated with either the register number to read (one byte, pass 1 into *src\_len*) or the register number and value to write (two bytes, pass 2 into *src\_len*). It is also assumed that the *\*dest* buffer has enough space for the two header characters plus, the encoded command, and the null terminator.

```
int EscapeString(char *src, char src_len, char *dest)
{
    // The following function copies and encodes the first
    // src_len characters from *src into *dest. This
    // encoding is necessary for Wi.232 command formats.
    // The resulting string is null terminated. The size
    // of this string is the function return value.
    // -----
    char src_idx, dest_idx;

    // Save space for the command header and size bytes
    // -----
    dest_idx = 2;

    // Loop through source string and copy/encode
    // -----
    for (src_idx = 0; src_idx < src_len; src_idx++)
    {
        if (src[src_idx] > 127)
        {
            dest[dest_idx++] = 0xFE;
        }/*if*/

        dest[dest_idx++] = (src[src_idx] & 0x7F);
    }/*for*/

    // Add null terminator
    // -----
    dest[dest_idx] = 0;

    // Add command header
    // -----
    dest[0] = 0xFF;
    dest[1] = dest_idx - 2;

    // Return escape string size
    // -----
    return dest_idx;
}
```

**Figure 16: Command Conversion Code**

## 5.4. Writing to Registers

Writing to a volatile register is nearly instantaneous. Writing to a non-volatile register, however, typically takes 16 ms. Because the packet size can vary based on the need for encoding, there are two possible packet structures. The following tables show the byte sequences for writing a register in each case.

**WARNING:** Be sure that the module is properly powered and remains powered for the duration of the register write. Loss of important configuration information could occur if the unit loses power during a non-volatile write cycle.

Byte 0								Byte 1								Byte 2								Byte 3							
Header								Size								Register								Value							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0xFF								0x02								0	Register							0	Value						

**Table 27, Write Register Command, value to be written is less than 128 (0x80)**

Byte 0								Byte 1								Byte 2								Byte 3								Byte 4							
Header								Size								Register								Escape								Value							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0xFF								0x03								0	Register							0xFE								0	Lower 7 bits of Value						

**Table 28, Write Register Command, value to be written is greater than or equal to 128 (0x80)**

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

## 5.5. Reading from Registers

A register read command is constructed by placing an escape character before the register number. The following table shows the byte sequence for reading a register.

Byte 0								Byte 1								Byte 2								Byte 3							
Header								Size								Escape								Register							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0xFF								0x02								0xFE								0	Register						

**Table 29, Read Register Command**

The module will respond to this command by sending an ACK (0x06) followed by the register number and register value. The register value is sent unmodified. For example, if the register value is 0x83, 0x83 is returned after the ACK (0x06). See table below for the format of the response. If the register number is invalid, it will respond with a NACK (0x15).

Byte 0								Byte 1								Byte 2							
ACK								Register								Value							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0x06								0	Register							Value							

**Table 30, Read Register Module Response for a Valid Register**

## 6. Electrical Specifications

### 6.1. Absolute Maximum Ratings for WI.232FHSS-25-R

Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational section of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability. This device is ESD sensitive. Proper precautions should be taken for handling and assembly.

Parameter	Min	Max	Units
Vdd- Power Supply	0	3.9	VDC
Voltage on any digital I/O pin	0	5	VDC
Input RF Level		10	dBm
Storage Temperature	-40	+85	°C

**Table 31, Absolute Maximum Ratings for WI.232FHSS-25-R**

## 6.2. Detailed Electrical Specifications for WI.232FHSS-25-R

### 6.2.1. AC Specifications- Rx for WI.232FHSS-25-R

Parameter	Min	Typ	Max	Units	Notes
Receive Frequency- US	902.2		927.8	MHz	At antenna pin
Hop Sequences	6				26 channels each
Channel Spacing		750		kHz	
Receiver Sensitivity		-100		dBm	152.34 kbit/ sec RF, $10^{-3}$ BER
Receiver Sensitivity		-102		dBm	38.4 kbit/ sec RF, $10^{-3}$ BER
Receiver Sensitivity		-105		dBm	9.6 kbit/ sec RF, $10^{-3}$ BER
Input IP3		-40		dBm	Flo +1 MHz & Flo +1.945 MHz
Input Impedance		50		Ohms	No matching required
LO Leakage		-65		dBm	50 Ohm termination at ANT
Adjacent Channel Rejection		-48		dBc	Fc +/- 650 kHz
IF Bandwidth		600		kHz	

**Table 32, AC Specifications- Rx for WI.232FHSS-25-R**

### 6.2.2. AC Specifications- Tx for WI.232FHSS-25-R

Parameter	Min	Typ	Max	Units	Notes
Transmit Frequency- US	902.2		927.8	MHz	
Center Frequency Error		2	4	ppm	915 MHz @ 25°C
Frequency Deviation		160		kHz	
Max. Output Power		13	15	dBm	915 MHz into 50 Ohm load
Output Impedance		50		Ohms	
Carrier Phase Noise		TBD		dBc	Into 50 Ohm load
Harmonic Output		-50		dBc	Into 50 Ohm load

**Table 33, AC Specifications- Tx for WI.232FHSS-25-R**

### 6.2.3. DC Specifications for WI.232FHSS-25-R

Parameter	Min	Typ	Max	Units	Notes
Operating Temperature	-40		+85	°C	HW Revision C and later
Supply Voltage	3.0	3.3	3.6	VDC	Operating limits
Allowed Vdd Ramp Time			1	Ms	From Vss to 2.7V
Receive Current Consumption		20		mA	Continuous operation, Vdd=3.3V, 2400 baud
Transmit Current Consumption					
(-2 dBm)		30		mA	Output into 50 Ohm load, Vdd= 3.3 VDC, 2400 baud
(+3 dBm)		35		mA	
(+8 dBm)		48		mA	
(+13 dBm)		65		mA	
Standby Current Consumption		2.1		mA	Vdd= 3.3 VDC
Sleep Current Consumption		1.4		mA	Vdd= 3.3 VDC
Deep Sleep Current Consumption		3		μA	Vdd= 3.3 VDC
Vih- Logic High Level Input	0.7*Vdd		5.0	VDC	
Vil- Logic Low Level Input	0		0.3*Vcc	VDC	
Voh- Logic High Level Output	2.5		Vcc	VDC	
Vol- Logic Low Level Output	0		0.4	VDC	

**Table 34, DC Specifications for WI.232FHSS-25-R**

### 6.3. Absolute Maximum Ratings for WI.232FHSS-250-R

Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational section of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability. This device is ESD sensitive. Proper precautions should be taken for handling and assembly.

Parameter	Min	Max	Units
Vdd- Power Supply	0	5.0	VDC
Voltage on any digital I/O pin	0	5.0	VDC
Input RF Level		12	dBm
Storage Temperature	-40	+85	°C

**Table 35, Absolute Maximum Ratings for WI.232FHSS-250-R**

## 6.4. Detailed Electrical Specifications for WI.232FHSS-250-R

### 6.4.1. AC Specifications- Rx for WI.232FHSS-250-R

Parameter	Min	Typ	Max	Units	Notes
Receive Frequency- US	902.2		927.8	MHz	At antenna pin
Hop Sequences	6				26 channels each
Channel Spacing		750		kHz	
Receiver Sensitivity		-100		dBm	153.6 kbit/ sec RF, $10^{-3}$ BER
Receiver Sensitivity		-102		dBm	38.4 kbit/ sec RF, $10^{-3}$ BER
Receiver Sensitivity		-105		dBm	9.6 kbit/ sec RF, $10^{-3}$ BER
Input IP3		-24		dBm	$P_{in} = -20$ dBm, 2 CW interferers, FRF = 915MHz, F1 = FRF + 3MHz, F2 = FRF + 6MHz, max gain, high-sensitivity
Input Impedance		50		Ohms	No matching required
Adjacent Channel Rejection		60		dB	Desired signal 3dB above input sensitivity level, CW interferer power level increased until BER = $10^{-2}$ , +/- 1MHz
IF Bandwidth		200		kHz	

**Table 36, AC Specifications- Rx for WI.232FHSS-250-R**

### 6.4.2. AC Specifications- Tx for WI.232FHSS-250-R

Parameter	Min	Typ	Max	Units	Notes
Transmit Frequency- US	902.2		927.8	MHz	
Center Frequency Error		2	4	ppm	915 MHz @ 25°C
Frequency Deviation		50		kHz	
Max. Output Power		23.5	24	dBm	915 MHz into 50 Ohm load
Output Impedance		50		Ohms	
Carrier Phase Noise		TBD		dBc	Into 50 Ohm load
Harmonic Output		-50		dBc	Into 50 Ohm load

**Table 37, AC Specifications- Tx for WI.232FHSS-250-R**

### 6.4.3. DC Specifications for WI.232FHSS-250-R

Parameter	Min	Typ	Max	Units	Notes
Operating Temperature	-40		+85	°C	HW Revision C and later
Supply Voltage	2.7	3.3	3.6	VDC	Operating limits
Allowed Vdd Ramp Time			1	ms	from Vss to 2.7 V
Receive Current Consumption		25		mA	Continuous operation, Vdd = 3.3 VDC, 2400 baud
Transmit Current Consumption					
(+8 dBm)		54		mA	Output into 50 Ohm load, Vdd= 3.3 VDC
(+13 dBm)		71		mA	
(+18 dBm)		109		mA	
(+23.5 dBm)		190		mA	
Standby Current Consumption		1.5		mA	Vdd= 3.3 VDC
Sleep Current Consumption		1.5		mA	Vdd= 3.3 VDC
Deep Sleep Current Consumption		3		μA	Vdd= 3.3 VDC
Vih- Logic High Level Input	0.7*Vdd		5.0	VDC	
Vil- Logic Low Level Input	0		0.3*Vcc	VDC	
Voh- Logic High Level Output	2.5		Vcc	VDC	
Vol- Logic Low Level Output	0		0.4	VDC	

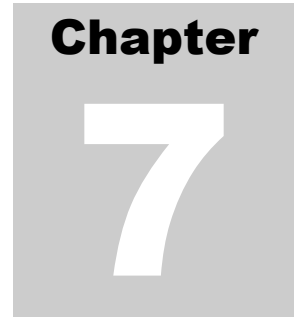
**Table 38, DC Specifications for WI.232FHSS-250-R**

### 6.5. Flash Specifications (Non-Volatile Registers)

Parameter	Min	Typ	Max	Units	Notes
Flash Write Duration		16		ms	Module stalled during operation
Flash Write Cycles	20k	100k		cycles	

**Table 39, Flash Specifications (Non-Volatile Registers)**





## 7. Custom Applications

For cost-sensitive applications, such as wireless sensing and AMR, Radiotronix can embed the application software directly into the microcontroller built into the module. For more information on this service, please contact Radiotronix.

## 8. Ordering Information

Product Part Number	Description
WI.232FHSS-25-R	Embedded Wireless Module, 25 mW (900 MHz)
Wi.232FHSS-250-R	Embedded Wireless Module, 250 mW (900 MHz)
Wi.232FHSS-25-FCC-R	Pinned, Pre-Certified Module, 25 mW (900MHz)
Wi.232FHSS-250-FCC-R	Pinned, Pre-Certified Module, 250 mW (900MHz)
RK-Wi.232FHSS-25-FCC-R	Development Kit for 25 mW FHSS modules, contains two Wi.232FHSS-25-FCC-R modules
RK-Wi.232FHSS-250-FCC-R	Development Kit for 250 mW FHSS modules, contains two Wi.232FHSS-250-FCC-R modules

**Table 40, Ordering Information**

## 9. Contact Information

Corporate Headquarters:  
905 Messenger Lane  
Moore, Oklahoma 73160  
405-794-7730

website: [www.radiotronix.com](http://www.radiotronix.com)  
support: [support@radiotronix.com](mailto:support@radiotronix.com)

### 9.1. Technical Support

Radiotronix has built a solid technical support infrastructure so that you can get answers to your questions when you need them. Our primary technical support tools are the support forum and knowledge base found on our website. We are continuously updating these tools. To find the latest information about these technical support tools, please visit <http://www.radiotronix.com/support>. Our technical support engineers are available Mon-Fri between 9:00 am and 5:00 pm central standard time. The best way to reach a technical support engineer is to submit a Webcase. Webcase submissions can be made at <http://www.radiotronix.com/support/webcase.asp>. For customers that would prefer to talk directly to a support engineer, we do offer phone support free of charge.

### 9.2. Sales Support

Our sales department can be reached via e-mail at [sales@radiotronix.com](mailto:sales@radiotronix.com) or by phone at 405-794-7730. Our sales department is available Mon-Fri between 8:30 am and 5:00 pm central standard time. Visit our web site at <http://www.radiotronix.com/corpsales.asp> for information on where to buy our products.