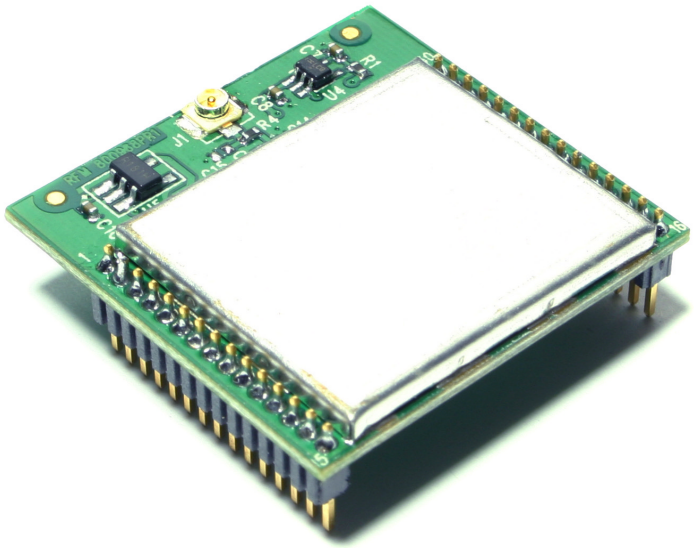
WSN802G Series

802.11g Wireless Sensor Network Modules



Integration Guide

Important Regulatory Information

RFM Product FCC ID: HSW-WSN802G
IC 4492A-WSN802G

Note: This unit has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at their expense.

The WSN802G has been designed to operate with any dipole antenna of up to 9 dBi of gain, or any patch antenna of up to 12 dBi gain.

See Section 3.10 of this manual for regulatory notices and labeling requirements. Changes or modifications to a WSN802G not expressly approved by RFM may void the user's authority to operate the module.

Important Export Information

ECCN: 5A002.a.1 ENC
CCATS: G073573

The WSN802G products are classified under ECCN codes as 5A002.a.1 class devices and carry and ENC encryption exception. RFM has had the WSN802G products pre-screened which allows exports to countries listed in Supplement 3 to Part 740 of the Export Administration Requirements *License Exception ENC Favorable Treatment Countries*. When exporting products containing any of the WSN802G products to this list of countries, CCATS G073573 should be referenced.

Export of the WSN802G products or products containing any of the WSN802G products to other countries will require additional review and may be prohibited.

This information is provided as a guide to exporting WSN802G-based products but it is the responsibility of the entity exporting WSN802G products or devices to determine their actual requirements.

Table of Contents

1.0 WSN802G Introduction

The WSN802G transceiver module is a low cost, robust solution for 802.11b/g/n sensor networks. The WSN802G is unique in that it is able to sleep while still remaining a member of an 802.11b/g/n network. The WSN802G's low active current and very low sleep current makes long life battery operation practical. The WSN802G module includes analog, digital and serial I/O, providing the flexibility and versatility needed to serve a wide range of sensor network applications. The WSN802G module is easy to integrate and is compatible with standard 802.11b/g/n routers.

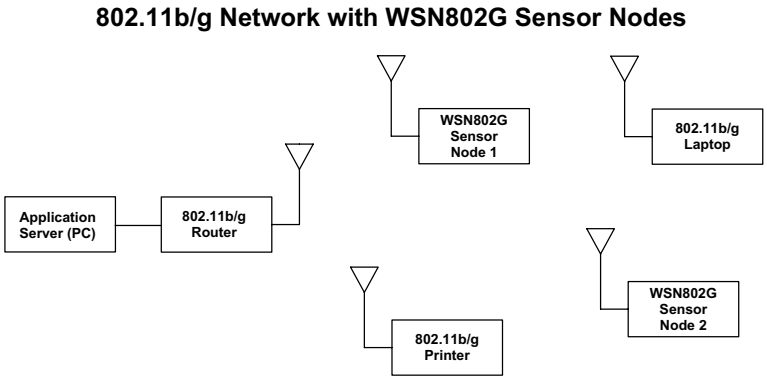


Figure 1.0.1

An example 802.11b/g/n network with WSN802G sensor nodes is shown in Figure 1.0.1. A sensor network application running on a server or PC communicates with one or more WSN802G sensor nodes through a commercial 802.11b/g/n router. WSN802G sensor nodes can be used with 802.11b/g/n routers that are also serving other applications.

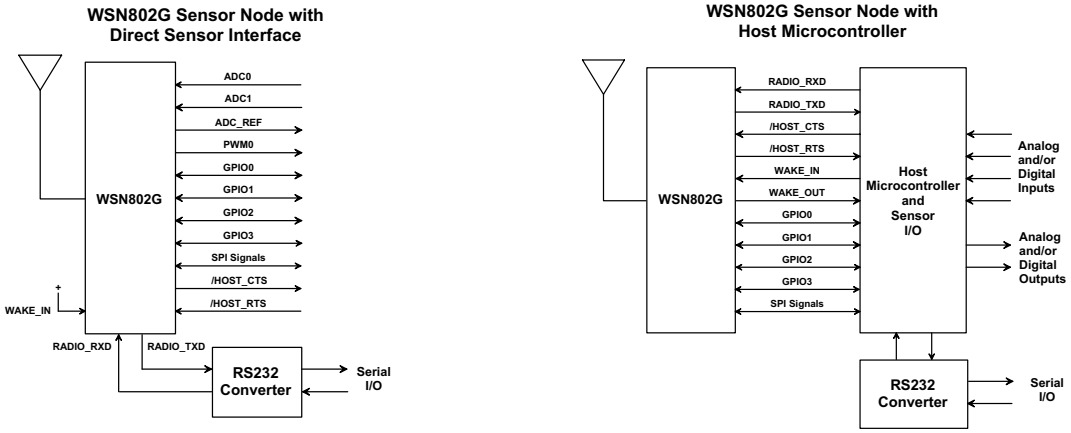


Figure 1.0.2

A WSN802G module is integrated with other components to create a complete sensor node. These components include a host circuit board, a power supply (battery), sensor I/O electronics and/or a host microcontroller, and a housing (external antenna also required on some WSN802G models). Two common configurations are show in Figure 1.0.2. Serial sensor data communication between a WSN802G module and its host microcontroller requires no protocol formatting. The WSN802G formats data received from its host into UDP packets for RF transmission, and delivers the payload data from received UDP packets to its host. The sensor network application on the server or PC uses a simple protocol to send and receive

data from WSN802G sensor nodes, as detailed in Section 4. WSN802G modules can receive configuration commands through either their serial port or over-the-air in UDP packets carrying SNMP commands.

1.1 Features

WSN802G modules provide a unique set of features for wireless sensor network applications:

- Compatibility with commercial and industrial 802.11b/g/n routers
- Low power consumption for long life battery operation including sleep mode
- Full -40 to +85 ºC industrial temperature range operation
- Analog and digital I/O, plus two serial and one SPI port
- Separate data and diagnostic ports
- System/application set up using just two Management Information Blocks (MIBs)
- Full 14 channel 802.11b/g coverage for world wide operation
- FCC, Canadian IC and European ETSI certifications
- Four module configurations:

  WSN802GC - solder reflow mounting with RF connector for external antenna connection

  WSN802GCA - solder reflow mounting with integral chip antenna

  WSN802GP - plug in connector mounting with RF connector for external antenna connection

  WSN802GPA - plug in connector mounting with integral chip antenna
- Automatic or manual I/O data reporting

1.2 Applications

WSN802G sensor networks are well suited to applications where IEEE 802.11b/g/n router compatibility, industrial temperature range operation and long battery life are important. Many applications match these criteria, including:

- Energy Monitoring and Management
- Physical Asset Management
- Cold Chain Data Logging and Food Safety
- Security and Access Control Systems
- Environmental Monitoring
- Many More

2.0 WSN802G Operation

WSN802G operation is designed to support long battery life by allowing the module to stay in sleep mode to the maximum extent possible. Compared to 802.11b/g cards used in notebook and handheld computers, the WSN802G's active current is also very low.

2.1 Active and Sleep Modes

Once the SNMP Server IP address has been set, the default state of the WSN802G is sleep mode. The WSN802G has a dedicated input to switch it from sleep to active mode, WAKE_IN (Pin 26). There are five events that will wake the WSN802G from sleep mode:

- Applying a logic high signal on the WAKE_IN pin
- Expiration of the *AutoReport* timer
- Expiration of Linkup trap timer
- Expiration of the Config trap timer
- Module's Primary SNMP Server IP address has not been set (this will not wake it, it prevents it from sleeping)

The WAKE_IN pin and *AutoReport* can be enabled/disabled. The Linkup timer sends a keep alive packet to the router every 60 seconds by default. The Config timer cannot be disabled and will generate a Config trap every 10 seconds by default. Once a module has been initially configured, the Config timer is typically set to a longer interval.

When the module wakes to an active state due to either the WAKE_IN pin or the *AutoReport* timer, it remains awake for a time period controlled by the *WakeTimeout* timer. The module returns to sleep mode when the *WakeTimeout* timer expires, subject to the conditions listed below. The *WakeTimeout* timer is held in reset and the module remains in active mode when any of the following events occur:

- A logic high signal is held on the WAKE_IN pin
- A serial byte is received
- An RF packet is sent or received
- Module's SNMP Server IP address has not been set

The *WakeTimeout* feature is used to support scenarios such as a server application parsing the I/O report and sending back a serial string or I/O output change command, or a host processor sending a serial string and waiting for a response.

As discussed in Section 5, the SNMP Server IP address can be set in a short period of time, allowing the module to switch to sleep mode for battery conservation. The SNMP server IP address only needs to be set one time.

A WSN802G module that has an SNMP Server IP Address but is not linked to an 802.11b/g/n router will cycle between sleep and active mode under the control of the scanning algorithm, even if none of the wake events discussed above are present.

Whenever the module is in active mode, a logic high is asserted on WAKE_OUT (Pin 27). WAKE_OUT can be used to signal an external processor. When the WSN802G is in sleep mode, WAKE_OUT is set to logic low.

2.2 Automatic I/O Reporting

The WSN802G sends an I/O report when one of the following events occur:

- A logic high signal is applied to the WAKE_IN pin
- The *AutoReport* timer fires (module in either active or sleep mode)

2.3 Data Serial Port

The data serial port on the WSN802G modules supports baud rates from 1.2 to 921.6 kbps. The following serial port configurations are supported:

- 5, 6, 7 and 8-bit character lengths
- 1 or 2 stop bits
- Even, odd, mark, space or no parity

The default serial port configuration is 9.6 kbps, 8, N,1. See Sections 4.14 and 7.2 for serial port configuration details. Serial port operation is full-duplex. Data is sent and received on the serial port transparently, with configuration information sent and received as UDP datagrams. The WSN802G includes an acknowledgement and retry mechanism to minimize data loss on RF transmissions. However, the UDP/IP protocol being carried by the RF transmissions does not provide guaranteed end-to-end delivery. The user must make provisions for detecting and resending data lost on an end-to-end transmission. The WSN802G data port is supported with optional /HOST_RTS and /HOST_CTS flow control signals.

2.4 Diagnostic Serial Port

The diagnostic serial port on the WSN802G modules supports baud rates from 1.2 to 115.2 kbps. The default diagnostic port configuration is 9.6 kbps, 8, N,1. Diagnostic port operation is full-duplex, three-wire, without flow control support. Diagnostic data is sent and received using a simple string protocol. Contact RFM module technical support for diagnostic port data format information.

2.5 Serial Peripheral Interface (SPI) Port

The WSN802G SPI port provides master mode functionality at data rates configurable up to 11 Mbps. SPI port operation is full duplex in the sense that a single clock signal simultaneously shifts data into and out of the SPI port.

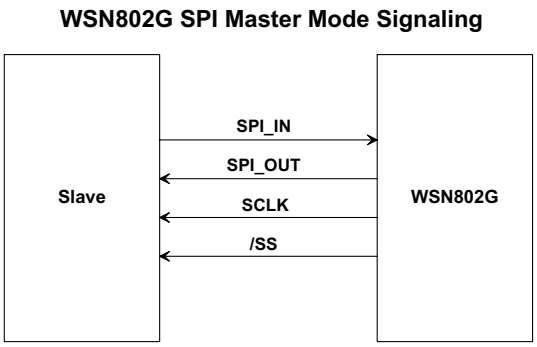


Figure 2.5.1

Figure 2.5.1 shows the signals a WSN802G uses in SPI Master mode. The AutoReport function triggers the WSN802G module to clock out a configurable command string, *SPI_MasterCmdStr*, to collect data from a Slave peripheral. The collected data is then transmitted as a data message. Alternatively, a host connected to the base can transmit an SPI command as a data message to the remote. The WSN802G will clock the command into its Slave peripheral and transmit back the Slave's response, as show in Figure 2.5.2. In either case, data strings are limited to 256 bytes.

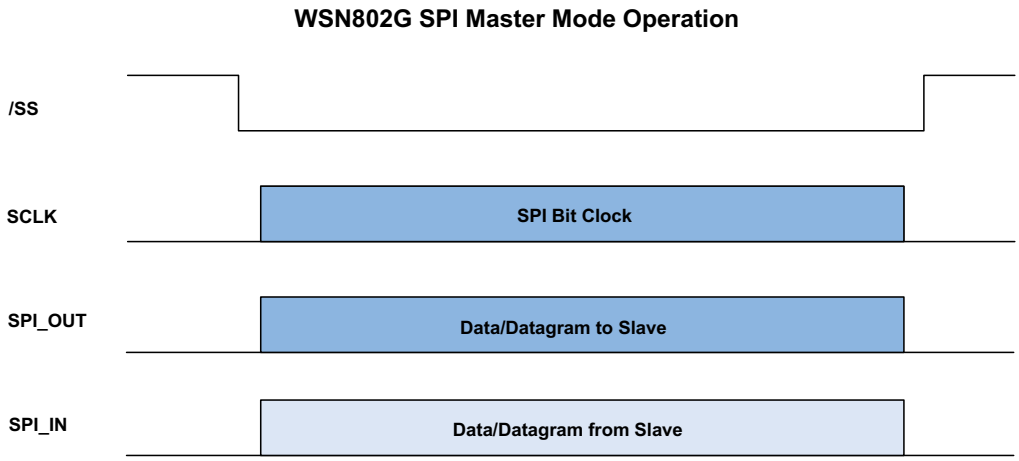


Figure 2.5.2

2.6 Analog I/O

The WSN802G includes two 10-bit ADC inputs, ADC0 (Pin 18) and ADC1 (Pin 19). Pin 25 provides a full-scale reference voltage to support ratiometric ADC measurements. ADC measurements are triggered and added to the automatic I/O report when a logic high signal is first applied to the WAKE_ IN pin or the *AutoReport* timer fires, as discussed in Section 2.2. An ADC reading is also made on the internal buss voltage of the WSN802G and included in the automatic I/O report. These readings can also be retrieved anytime the WSN802G is in active mode using the IO_REPORT application protocol command as discussed in Section 4.1.

The WSN802G also includes a 16-bit pulse width modulated output, PWM0 (Pin 9). The PWM output is low-pass filtered to provide an analog output voltage with ripple suppressed to 7 bits. External low-pass filtering can be added to further suppress ripple. The full-scale PWM output is referenced to the regulated

supply voltage (Pin 24). The PWM output is set using the IO_ WRITE_PWM application protocol command, as discussed in Section 4.4.

2.7 Digital I/O

The WSN802G includes four general purpose input/output (GPIO) ports, GPIO0 (Pin 4), GPIO1 (Pin 11), GPIO2 (Pin 10) and GPIO3 (Pin 12). When programmed as inputs, GPIO pins include an internal weak pull-up. The states of pins configured as inputs are captured as part of the automatic I/O report when a logic high signal is applied to the WAKE_IN pin or the *AutoReport* timer fires, as discussed in Section 2.2. These readings can also be retrieved anytime the WSN802G is in active mode using the IO_READ_REQUEST application protocol command as discussed in Section 4.1. The states of GPIO pins configured as outputs are set using the IO_WRITE_GPIO application protocol command as discussed in Section 4.3.

3.0 WSN802G Hardware

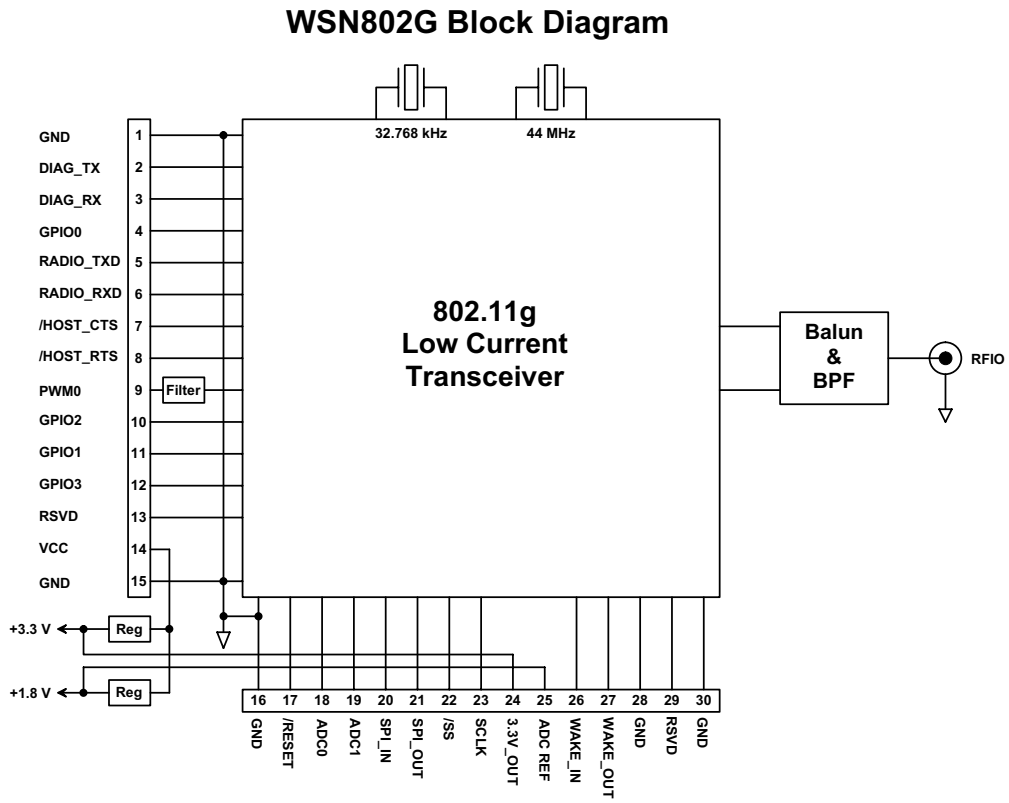


Figure 3.0.1

WSN802G modules operate in the international 2.4 GHz ISM band over the frequency range of 2401-2474 MHz, with a nominal RF output power of 10 mW. WSN802G modules support four standard 802.11g RF data rates, 1, 2, 5.5 and 11 Mbps. WSN802G modules provide a variety of hardware interfaces. There are two serial interfaces, one for data and a second for diagnostics. The data port supports standard serial baud rates from 1.2 to 921.6 kbps, and the diagnostic port supports standard serial baud rates from 1.2 to 115.2 kbps. Optional hardware flow control is provided for the data serial port. There is also a serial peripheral interface (SPI), which can operate in either master or slave mode. SPI slave mode supports data rates up to 2 Mbps, and master mode supports data rates up to 11 Mbps.

WSN802G modules includes two 10-bit ADC inputs, a 16-bit PWM (DAC) output, and four general purpose input/output (GPIO) ports. WSN802G modules are available with either RF connectors for external antennas, or with integral chip antennas (A suffix). WSN802G modules are available in two mounting configurations. The WSN802GC and WSN802GCA are designed for solder reflow mounting, and the WSN802GP and WSN802GPA are designed for plug-in connector mounting.

3.1 Absolute Maximum Ratings

Rating	Sym	Value	Units
Input/Output Pins Except ADC Inputs		-0.5 to +3.63	V
ADC Input Pins		-0.5 to 1.98	V
Non-Operating Ambient Temperature Range		-40 to +85	^{o}C

Table 3.1.1

3.2 Specifications

Characteristic	Sym	Minimum	Typical	Maximum	Units
Operating Frequency Range		2401		2474	MHz
Spread Spectrum Method			Direct Sequence		
RF Chip Rate			11		Mcps
RF Data Rates			1, 2, 5.5, 11		Mbps
Modulation Type			BPSK at 1 Mbps, QPSK at 2 Mbps CCK at 5.5 and 11 Mbps		
Number of RF Channels			11		
RF Channel Spacing			5		MHz
Receiver Sensitivity, 8% PER:					
1 Mbps RF Data Rate			-92		dBm
2 Mbps RF Data Rate			-90		dBm
5.5 Mbps RF Data Rate			-84		
11 Mbps RF Data Rate			-81		
RF Transmit Power			10		mW
WSN802GC and WSN802GP RF Connector			U.FL Coaxial Connector		
Optimum External Antenna Impedance			50		Ω
WSN802GCA and WSN802GPA Antenna			Integral Chip		
ADC Input Range		0		1.8	V
ADC Input Resolution			10		bits
ADC Input Impedance		1			MΩ
PWM Output Resolution				16	bits
Data Serial Port Baud Rates			1.2, 2.4, 4.8, 9.6 (default), 19.2, 28.8, 38.4, 57.6, 76.8, 115.2, 230.4, 460.8, 921.6		kbps
Diagnostic Serial Port Baud Rate			1.2, 2.4, 4.8, 9.6 (default), 19.2, 28.8, 38.4, 57.6, 76.8, 115.2		kbps
Serial Peripheral Interface (SPI) Data Rate, Master Mode				11	Mbps
Serial Peripheral Interface (SPI) Data Rate, Slave Mode				2	Mbps
Digital I/O:					
Logic Low Input Level		-0.3		0.7	V
Logic High Input Level		2.24		V_{CC}	V
Input Pull-up Resistor		50		1000	KΩ
Logic Low Output Level		0		0.4	V
Logic High Output Level		2.4		V_{CC}	V
Power Supply Voltage Range	V_{CC}	+3		+3.63	Vdc
Power Supply Voltage Ripple				10	mV_{P-P}
Receive Mode Current				150	mA
Transmit Mode Current				200	mA
Sleep Mode Current			7.5		µA
WSN802GC and WSN802GCA Mounting			Reflow Soldering		
WSN802GP and WSN802GPA Mounting			Socket		
Operating Temperature Range		-40		85	°C
Operating Relative Humidity Range, Non-condensing		10		90	%

Table 3.2.1

3.3 Module Interface

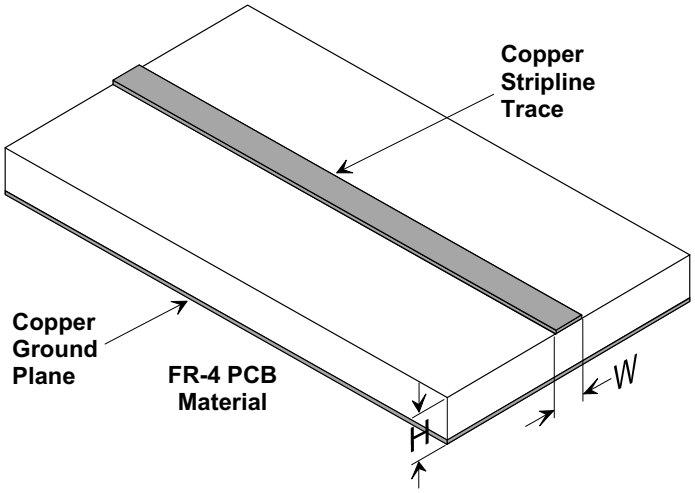
Pin	Name	I/O	Description
1	GND	-	Power supply and signal ground. Connect to the host circuit board ground.
2	DIAG_TX	O	Diagnostic serial port output.
3	DIAG_RX	I	Diagnostic serial port input.
4	GPIO0	I/O	Configurable digital I/O port 0. An internal weak pull-up is provided when configured as an input.
5	RADIO_TXD	O	Serial data output from the radio.
6	RADIO_RXD	I	Serial data input to the radio.
7	/HOST_CTS	O	UART/SPI flow control output. The module sets this line low when it is ready to accept data from the host on the RADIO_RXD or MOSI input. When the line goes high, the host must stop sending data.
8	/HOST_RTS	I	UART flow control input. The host sets this line low to allow data to flow from the module on the RADIO_TXD pin. When the host sets this line high, the module will stop sending data to the host.
9	PWM0	O	16-bit pulse-width modulated output with internal low-pass filter. Filter is first-order, with a 159 Hz 3 dB bandwidth, 10K output resistance.
10	GPIO2	I/O	Configurable digital I/O port 2. An internal weak pull-up is provided when configured as an input.
11	GPIO1	I/O	Configurable digital I/O port 1. An internal weak pull-up is provided when configured as an input.
12	GPIO3	I/O	Configurable digital I/O port 3. An internal weak pull-up is provided when configured as an input.
13	RSVD	-	Reserved pin. Leave unconnected.
14	VCC	I	Power supply input, +3.0 to +3.63 Vdc.
15	GND	-	Power supply and signal ground. Connect to the host circuit board ground.
16	GND	-	Power supply and signal ground. Connect to the host circuit board ground.
17	/RESET	I	Active low module hardware reset.
18	ADC0	I	10-bit ADC input 0. ADC full scale reading can be referenced to the module's +1.8 V regulated supply.
19	ADC1	I	10-bit ADC input 1. ADC full scale reading can be referenced to the module's +1.8 V regulated supply.
20	SPI_IN	I/O	This pin is the SPI master mode input.
21	SPI_OUT	I/O	This pin is the SPI master mode output.
22	/SS	I/O	SPI active low slave select. This pin is an output when the module is operating as a master, and an input when it is operating as a slave.
23	SCLK	I/O	SPI clock signal. This pin is an output when operating as a master, and an input when operating as a slave.
24	3.3V_OUT	O	Module's +3.3 V regulated supply, available to power external sensor circuits. Current drain on this output should be no greater than 50 mA.
25	ADC_REF	O	Module's +1.8 V regulated supply, used for ratiometric ADC readings. Current drain on this output should be no greater than 10 mA.
26	WAKE_IN	I	Active high interrupt input to wake the module from timer sleep. Can be used to wake module on event, etc.
27	WAKE_OUT	O	Active high output asserted when module wakes from timer sleep. Can be used to wake an external device.
28	GND	-	Connect to the host circuit board ground plane.
29	RSVD	-	Reserved pin. Leave unconnected.
30	GND	-	Connect to the host circuit board ground plane.

Table 3.3.1

3.4 WSN802GC and WSN802GP Antenna Connector

A U.FL miniature coaxial connector is provided on the WSN802GC and WSN802GP modules for connection to the RFIO port. A short U.FL coaxial cable can be used to connect the RFIO port directly to an antenna. In this case the antenna should be mounted firmly to avoid stressing the U.FL coaxial cable due to antenna mounting flexure. Alternately, a U.FL coaxial jumper cable can be used to connect the WSN802G module to a U.FL connector on the host circuit board. The connection between the host circuit board U.FL connector and the antenna or antenna connector on the host circuit board should be implemented as a 50 ohm stripline. Referring to Figure 3.4.1, the width of this stripline depends on the thickness of the circuit board between the stripline and the groundplane. For FR-4 type circuit board materials (dielectric constant of 4.7), the width of the stripline is equal to 1.75 times the thickness of the circuit board. Note that other circuit board traces should be spaced away from the stripline to prevent signal coupling, as shown in Table 3.4.1. The stripline trace should be kept short to minimize its insertion loss.

Circuit Board Stripline Trace Detail



For 50 ohm impedance W = 1.75 * H

Figure 3.4.1

Trace Separation from 50 ohm Microstrip	Length of Trace Run Parallel to Microstrip
100 mil	125 mill
150 mil	200 mil
200 mil	290 mil
250 mil	450 mil
300 mil	650 mil

Table 3.4.1

3.5 Input Voltage

WSN802G radio modules can operated from an unregulated DC input (Pin 14) in the range of 3.0 V (trough) to 3.63 V (peak) over the temperature range of -40 to 85° C. *Applying AC, reverse DC, or a DC voltage outside the range given above can cause damage and/or create a fire and safety hazard. Further, care must be taken so logic inputs applied to the radio stay within the voltage range of 0 to 3.3 V. Signals applied to the analog inputs must be in the range of 0 to ADC_REF (Pin 25). Applying a voltage to a logic or analog input outside of its operating range can damage the WSN802G module.*

3.6 ESD and Transient Protection

WSN802G circuit boards are electrostatic discharge (ESD) sensitive. ESD precautions must be observed when handling and installing these components. Installations must be protected from electrical transients on the power supply and I/O lines. This is especially important in outdoor installations, and/or where connections are made to sensors with long leads. *Inadequate transient protection can result in damage and/or create a fire and safety hazard.*

3.7 Interfacing to 5 V Logic System

All logic signals including the serial ports on the WSN802G are 3.3 V signals. To interface to 5 V signals, the resistor divider network shown in Figure 3.7.1 below must be placed between the 5 V signal outputs and the WSN802G signal inputs. The output voltage swing of the WSN802G 3.3 V signals is sufficient to drive 5 V logic inputs.

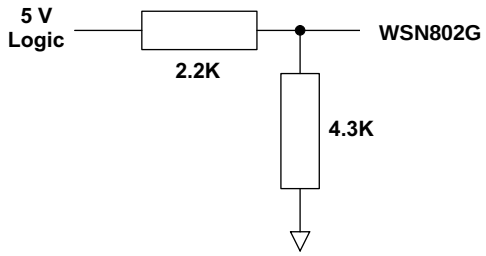


Figure 3.7.1

3.8 Power-On Reset Requirements

When applying power to the WSN802G, the /RESET pin should be held low until the power supply voltage reaches 3.3 volts for 100 milliseconds.

3.9 Mounting and Enclosures

WSN802GC radio modules are mounted by reflow soldering them to a host circuit board. WSN802GP modules are mounted by plugging their pins into a set of mating connectors on the host circuit board. Refer to Section 10.3 and/or the WSN802G Data Sheet for mounting details.

WSN802G enclosures must be made of plastics or other materials with low RF attenuation to avoid compromising antenna performance where antennas are internal to the enclosure. Metal enclosures are not suitable for use with internal antennas as they will block antenna radiation and reception. Outdoor enclosures must be water tight, such as a NEMA 4X enclosure.

3.10 Labeling and Notices

WSN802G FCC Certification - The WSN802G hardware has been certified for operation under FCC Part 15 Rules, Section 15.247. *The antenna(s) used for this transmitter must be installed to provide a separation distance of at least 20 cm from all persons and must not be co-located or operating in conjunction with any other antenna or transmitter.*

WSN802G FCC Notices and Labels - *This device complies with Part 15 of the FCC rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.*

A clearly visible label is required on the outside of the user's (OEM) enclosure stating "Contains FCC ID: HSW-WSN802G."

WARNING: This device operates under Part 15 of the FCC rules. Any modification to this device, not expressly authorized by RFM, Inc., may void the user's authority to operate this device. Canadian Department of Communications Industry Notice - IC: 4492A-WSN802G

This apparatus complies with Health Canada's Safety Code 6 / IC RSS 210.

ICES-003

This digital apparatus does not exceed the Class B limits for radio noise emissions from digital apparatus as set out in the radio interference regulations of Industry Canada.

Le present appareil numerique n'emet pas de bruits radioelectriques depassant les limites applicables aux appareils numeriques de Classe B prescrites dans le reglement sur le brouillage radioelectrique edicte par Industrie Canada.

ETSI EN 300 328

The WSN802G module has passed ETSI EN 300 328 testing conducted by an independent test laboratory.

4.0 Application Protocol

In most applications, the WSN802G auto-reporting function will be used to routinely send data from a module to the application. The WSN802G supports this type of operation through use of the auto-reporting function and the *AutoReport* and *WakeTimeout* timers.

There are three ways to read and write data and configuration parameters to and from the WSN802G module: (1) over the air using SNMP traps - configuration only; (2) over the air using UDP to send the WSN protocol; and (3) through the module's UART or SPI port using the WSN protocol. The SNMP protocol is distinct from the UDP and serial port protocol. The UDP and serial port protocol only differs in the UPD/IP packet header data.

Modules will request SNMP configuration changes using *Config Traps*. The frequency of checking for changes is configured through the Config trap timer. Details of the SNMP commands and operation is provided in Section 7 of this manual.

Modules must be in active mode (awake) to receive API commands through either over the air UDP commands or through the module's UART or SPI port. In addition to sending data, auto-report transmissions signal the application that the module is awake. Setting the *WakeTimeout* timer to 2 seconds will keep the module awake, giving the application 2 seconds to send API commands to the module. The module will remain awake past the 2 seconds if commands are being received or processed. Once the commands are completed the module will return to sleep immediately if the *WakeTimeout* time has elapsed.

The format of all of the WSN API commands and responses are given in Sections 4.1 through 4.13 below. API commands and responses sent through a wireless server (access point, etc.) are formatted as UDP/IP packets. The IPv4 UDP/IP packet format is shown in Figure 4.0.1 below. API commands and responses are carried in the UDP datagram payload area. In the text below, API commands and responses will be referred to as datagrams with the understanding they are the payload of a UDP datagram when sent through a wireless server. Automatically generated I/O reports from the WSN802G module due to timeouts or event interrupts take the form of the IO_READ - IO_REPORT datagram shown in Section 4.2. The IO_READ – IO_REPORT message is only available over the air and not through the module's serial ports.

<table>
<tr><th colspan="2">Byte 0</th><th>Byte 1</th><th colspan="2">Byte 2</th><th>Byte 3</th></tr>
<tr><td>IP Version</td><td>Header Length</td><td>Type of Service</td><td colspan="3">Total Length</td></tr>
<tr><td colspan="3">ID</td><td>Flags</td><td colspan="2">Fragment Offset</td></tr>
<tr><td colspan="2">Time to Live</td><td>Protocol</td><td colspan="3">Header Checksum</td></tr>
<tr><td colspan="6">Source IP Address</td></tr>
<tr><td colspan="6">Destination IP Address</td></tr>
<tr><td colspan="3">Source Port</td><td colspan="3">Destination Port</td></tr>
<tr><td colspan="3">UDP Length</td><td colspan="3">Checksum</td></tr>
<tr><td colspan="6">Payload (Application Command)</td></tr>
</table>

Figure 4.0.1

WSN802G modules only accept wireless application commands from and send wireless application command data/replies to the IP address of the server running their sensor application which is configured in the *SensorServerIP* parameter.

As shown in Figure 4.0.2 below, WSN802G application protocol datagrams use a standard header beginning with a protocol identifier to discriminate WSN802G protocol messages from other message types. Datagrams are in 32-bit, big-endian format. The standard header fields are:

Protocol Identifier: Unique identifier for all WSN messages, 0x52464D49

Opcode: Code indicating the type of command or response

Transaction ID: This is an incrementing transaction reference counter. Each end of the link must keep its own counter for transactions that it originates. The most significant bit of the transaction ID will be set for all transactions that the server originates.

The WSN802G application protocol messages are listed in Table 4.0.1 below. The port number that the module sends and receives wireless UDP application messages on is defined by the *SensorServerPortNum* parameter, as discussed in Section 7.2. A WSN802G module will accept wireless UDP messages specifically addressed to it, or that are broadcast (addressed to all modules). If a command is received through a wireless broadcast, the WSN802G module will reply with a broadcast.

The message format is the same for both the over the air UDP payload and the serial data communicated over either the module's UART or SPI ports. The Protocol Identifier serves as the start character for UART or SPI communicated packets with the Opcode used as a further validation of a API message instead of application data.

Byte 0	Byte 1	Byte 2	Byte 3
WSN802G Protocol Identifier = 0x52464D49			
Opcode		Transaction ID	
Data (variable length)			

Figure 4.0.2

Opcode	Direction	Description
0x0000	Server-to-Module	IO_READ_REQUEST
0x0001	Module-to-Server	IO_READ - IO_REPORT
0x0002	Server-to-Module	IO_WRITE_GPIO
0x0003	Server-to-Module	IO_WRITE_PWM
0x0004	Module-to-Server	IO_WRITE_REPLY
0x0005	Module-to-Server	IO_SERIAL_READ
0x0006	Server-to-Module	IO_SERIAL_WRITE
0x0007	Module-to-Server	IO_SPI_READ
0x0008	Server-to-Module	IO_SPI_WRITE
0x0010	Server-to-Module	CFG_READ
0x0011	Module-to-Server	CFG_READ_REPLY
0x0012	Server-to-Module	CFG_WRITE
0x0013	Module-to-Server	CFG_WRITE_REPLY

Table 4.0.1

4.1 I/O Read Request

The IO_READ_REQUEST datagram is used to request current I/O values, as shown in Figure 4.1.1.

Byte 0	Byte 1	Byte 2	Byte 3
WSN802G Protocol Identifier = 0x52464D49			
Opcode = 0x0000		Transaction ID = varies	

Figure 4.1.1

The module responds to an IO_READ_REQUEST with an IO_READ - IO_REPORT

4.2 I/O Read - I/O Report

The IO_READ - IO_REPORT datagram is used to report current I/O values, as shown in Figure 4.2.1.

Byte 0	Byte 1	Byte 2	Byte 3
WSN802G Protocol Identifier = 0x52464D49			
Opcode = 0x0001		Transaction ID = varies	
Timestamp [7..4]			
Timestamp [3..0]			
MAC Address Bytes [5..2]			
MAC Address Bytes [1..0] (sender)		ADC0	
ADC1		VOLT	
RSSI		GPIO	

Figure 4.2.1

The fields specific to this datagram are:

Timestamp: Timestamp of reading in 7.62939 µs timer ticks since startup

MAC Address: As an IO_READ - IO_REPORT can be sent unsolicited, the MAC address is provided to identify the sender, which can be helpful in situations where DHCP is used and the IP address is initially unknown or if the sender's IP address has been exchanged do to NAT.

ADC0: Current ADC0 reading, only the low 10 bits are significant

ADC1: Current ADC1 reading, only the low 10 bits are significant

VOLT: Current module voltage reading, 16-bit count in millivolts

RSSI: Current RSSI reading, only the low 10 bits are significant

GPIO: Current GPIO states, only GPIO lines defined as inputs are valid

The module generates an IO_READ - IO_REPORT datagram based on the *AutoReportInterval* and in response to an IO_READ_REQUEST.

4.3 I/O Write GPIO

The IO_WRITE_GPIO datagram is used to set module outputs, as shown in Figure 4.3.1.

Byte 0	Byte 1	Byte 2	Byte 3
WSN802G Protocol Identifier = 0x52464D49			
Opcode = 0x0002		Transaction ID = varies	
GPIO			

Figure 4.3.1

The field specific to this datagram is:

GPIO: States for GPIO lines defined as outputs. Setting a GPIO bit to 1 sets its output to 3.3 V, setting a bit to 0 sets its output to 0 V.

The module responds to an IO_WRITE_GPIO with an IO_WRITE_REPLY.

4.4 I/O Write PWM

The IO_WRITE_PWM datagram is to set the PWM0 output , as shown in Figure 4.4.1.

Byte 0	Byte 1	Byte 2	Byte 3
WSN802G Protocol Identifier = 0x52464D49			
Opcode = 0x0003		Transaction ID = varies	
PWM0		Reserved	

Figure 4.4.1

The field specific to this datagram is:

PWM0: PWM0 setting, 16-bit unsigned value

The module responds to an IO_WRITE_PWM with an IO_WRITE_REPLY.

4.5 I/O Write Reply

An IO_WRITE_REPLY datagram is sent by the WSN802G module to confirm receipt of an IO_WRITE_GPIO, or IO_WRITE_PWM datagram, as shown in Figure 4.5.1.

Byte 0	Byte 1	Byte 2	Byte 3
WSN802G Protocol Identifier = 0x52464D49			
Opcode = 0x0004		Transaction ID = varies	
Status			

Figure 4.5.1

The field specific to this datagram is:

Status: 0x0000 = successful, 0x0001 = failed

4.6 I/O Serial Read

The IO_SERIAL_READ datagram shown in Figure 4.6.1 is used by the module to send out data received on its serial port.

Byte 0	Byte 1	Byte 2	Byte 3
WSN802G Protocol Identifier = 0x52464D49			
Opcode = 0x0005		Transaction ID = varies	
Timestamp High Bytes			
Timestamp Low Bytes			
MAC Address Bytes [5..2]			
MAC Address Bytes [1..0] (sender)		Length	
Serial Data, 0-256 bytes			

Figure 4.6.1

The fields specific to this datagram are:

Timestamp: Timestamp of reading in 7.62939 µs timer ticks since startup

MAC Address: As an IO_SERIAL_READ can be sent unsolicited, the MAC address is provided to identify the sender, which can be helpful in situations where DHCP is used and the IP address is initially unknown

Length: Length of serial data string

Serial Data: Serial data string, 0-256 bytes

When a module receives a string on its serial port, it send an IO_SERIAL_READ message to its server. If the module receives an IO_SERIAL_WRITE message, it will output the received data on its serial port.

4.7 I/O Serial Write

The IO_SERIAL_WRITE datagram shown in Figure 4.7.1 is used to send data to the module to output on its serial port.

Byte 0	Byte 1	Byte 2	Byte 3
WSN802G Protocol Identifier = 0x52464D49			
Opcode = 0x0006		Transaction ID = varies	
Length		Serial Data, 0-256 bytes	

Figure 4.7.1

The fields specific to this datagram are:

Length: Length of serial data string

Serial Data: Serial data string, 0-256 bytes

When the module receives an IO_SERIAL_WRITE message, it will output the received data on its serial port. A serial string length of zero causes no data output, but is allowed for system testing purposes.

4.8 I/O SPI Read

The IO_SPI_READ datagram shown in Figure 4.8.1 is used by the module to send out data received on its serial peripheral interface (SPI) port.

Byte 0	Byte 1	Byte 2	Byte 3
WSN802G Protocol Identifier = 0x52464D49			
Opcode = 0x0007		Transaction ID = varies	
Timestamp High Bytes			
Timestamp Low Bytes			
MAC Address Bytes [5..2]			
MAC Address Bytes [1..0] (sender)		Length	
SPI Data, 0-256 bytes			

Figure 4.8.1

The fields specific to this datagram are:

Timestamp: Timestamp of reading in 7.62939 µs timer ticks since startup

MAC Address: As an IO_SPI_READ can be sent unsolicited, the MAC address is provided to identify the sender, which can be helpful in situations where DHCP is used and the IP address is initially unknown

Length: Length of serial data string

SPI Data: Data string, 0-256 bytes

The WSN802G SPI port operates in master mode. The auto-reporting function triggers the WSN802G SPI port to clock out a stored command string, *SPI_MasterCmdStr*, to collect data from a slave peripheral. The collected data is then transmitted as an IO_SPI_READ datagram.

4.9 I/O SPI Write

The IO_SPI_WRITE datagram shown in Figure 4.9.1 is used to send data to the module to output on its serial peripheral interface (SPI) port.

Byte 0	Byte 1	Byte 2	Byte 3
WSN802G Protocol Identifier = 0x52464D49			
Opcode = 0x0008		Transaction ID = varies	
Length		SPI Data, 0-256 bytes	

Figure 4.9.1

The fields specific to this datagram are:

Length: Length of serial data string

SPI Data: Data string, 0-256 bytes

The WSN802G SPI port operates in master mode, where the WSN802G clocks out data to its slave.

4.10 CFG Read

The CFG_READ datagram is used to read a configuration register from the module through the wireless link, serial port or SPI port. Configuration registers are sorted into banks. The register location in a bank may also be referred to as the register's offset.

Byte 0	Byte 1	Byte 2	Byte 3
WSN802G Protocol Identifier = 0x52464D49			
Opcode = 0x0010		Transaction ID = varies	
Length		Bank	
Location			

Figure 4.10.1

The fields specific to this datagram are:

Length: Total length of the following Bank and Location sections, in bytes

Bank: Target register bank number

Location: Target register location

The module responds to a CFG_READ with a CFG_READ_REPLY. Note that the module must be awake in order to receive and process this command.

4.11 CFG Read Reply

In response to a CFG_READ command, the CFG_READ_REPLY datagram is sent by the module to provide the contents of a configuration register

Byte 0	Byte 1	Byte 2	Byte 3
WSN802G Protocol Identifier = 0x52464D49			
Opcode = 0x0011		Transaction ID = varies	
Length		Bank	
Location		Value	

Figure 4.11.1

The fields specific to this datagram are:

Length: Total length of the following bank, register and value sections, in bytes

Bank: Target register bank number

Location: Target register location

Value: Value in target register

IMPORTANT NOTE: The register value is returned in Little-Endian order, least significant byte first.

4.12 CFG Write

The CFG_WRITE datagram is sent through the module's wireless link, serial port or SPI port to set a configuration register in the module. Configuration registers are sorted into banks. The register location in a bank may also be referred to as the register's offset.

Byte 0	Byte 1	Byte 2	Byte 3
WSN802G Protocol Identifier = 0x52464D49			
Opcode = 0x0012		Transaction ID = varies	
Length		Bank	
Location		Value	

Figure 4.12.1

The fields specific to this datagram are:

Length: Total length of the following bank, register and value sections, in bytes

Bank: Target register bank

Location: Target register location

Value: Target register value

IMPORTANT NOTE: The register value must be entered in Little-Endian order, least significant byte first.

The module responds to a CFG_WRITE with a CFG_WRITE_REPLY. Note that the module must be awake in order to receive and process this command.

4.13 CFG Write Reply

A CFG_WRITE_REPLY datagram is sent by the module to confirm the receipt of a CFG_WRITE datagram.

Byte 0	Byte 1	Byte 2	Byte 3
WSN802G Protocol Identifier = 0x52464D49			
Opcode = 0x0013		Transaction ID = varies	
Length		Status	

Figure 4.13.1

The fields specific to this datagram are:

Length: Length of the remainder of the packet in bytes

Status: Status code: 0 = successful, 1 = failed

4.14 Configuration Parameters

The parameters that can be accessed through the CFG series API commands are detailed below, organized by bank and location. The default values in Tables 4.14.1 through 4.14.8 are shown as they would be sent using CFG_WRITE or received using CFG_READ. All numerical values in the tables are in *Little-Endian* byte order, starting with the least significant byte on the left. ASCII strings holding a representation of a numerical value, such as the *AutoReportInterval* in Bank 1, Location 2 below, use Little-Endian byte order. Strings holding text, such as the *SensorName* in Bank 1, Location 1 below, are in reading byte order, first character on the left, last character on the right.

Bank 1 - General Module Configuration

Bank	Location	Name	R/W	Size, bytes	Range	Default
1	1	SensorName	R/W	128	ASCII String	"WSN Sensor"
1	2	AutoReportInterval	R/W	8	ASCII String	"0000000A00000000" (5 s)
1	3	SensorServerIP	R/W	4	Class A,B,C	0xC803A8C0 (192.168.3.200)
1	4	SensorServerPort	R/W	4	$1..2^{16}$ -1	0x3F200000 (8255)
1	5	WakeOutPredelay	R/W	4	$0..2^{32}$ -1	0x0A000000 (10 ms)
1	6	WakeOutPostdelay	R/W	4	$0..2^{32}$ -1	0x0A000000 (10 ms)
1	7	WakeTimeout	R/W	4	$0..2^{32}$ -1	0x00000000 (0 ms)
1	8	TxPower	R/W	4	0..7	0x00000000 (8 mW)
1	9	HardwareRevision	R	N/A	ASCII String	0x312E302E30 (1.0.0)
1	10	FirmwareRevision	R	N/A	ASCII String	0x322E302E31303236 (2.0.1026)
1	11	FirmwareBuildDate	R	4	ASCII String	unique to each build date
1	12	TxRetryLimit	R/W	4	0..15	0x04000000 (4 retries)
1	13	NetworkMode	R/W	4	0..1	0x00000000 (only UDP currently supported)

Table 4.14.1

SensorName - this parameter is a user-assignable sensor module name, for example "Utility Room Temperature Sensor". The name can contain up to 128 bytes.

AutoReportInterval - this parameter sets the interval at which the sensor will send periodic reports. The parameter scaling is in microcontroller clock cycles of 0.029802322 µs. This parameter is a 64-bit number formatted as an ASCII string of the equivalent hexadecimal value.

SensorServerIP - this parameter holds the IP address of the server for the module to send sensor data reports. The IP address is formatted as a 32-bit value.

SensorServerPort - this parameter holds the port number of the server for the module to send sensor data reports. The port number is formatted as a 32-bit value, with the lower 16 bits containing the port number and the upper 16 bits set to zero.

WakeOutPredelay - this parameter sets the duration in milliseconds the WAKE_OUT pin turns on to activate an external user circuit *prior* to the rest of the module waking up.

WakeOutPostdelay - this parameter sets the duration in milliseconds the WAKE_OUT turn on to activate an external user circuit *subsequent* to the rest the module waking up.

WakeTimeout - this parameter sets the duration of inactivity in milliseconds that triggers the module to go back to sleep after being activated.

TxPower - this parameter set the transmitter output power level. Changes to this parameter require a reboot to take effect. The parameter range is 0 to 7, with 0 the highest power setting.

HardwareRevision - this parameter holds the revision code of the module hardware. This parameter is read-only.

FirmwareRevision - this parameter holds the firmware revision code. This parameter is read-only.

FirmwareBuildDate - this parameter codes the build date and timestamp of the module firmware as an ASCII string. This parameter is read-only.

TxRetryLimit - this parameter sets the retry limit for 802.11 transmissions.

NetworkMode - this parameter specifies the IP format for sending auto-reports. Set this parameter to 0x00 for UDP format.

Bank 2 - Module I/O Configuration

Bank	Location	Name	R/W	Size, bytes	Range	Default
2	1	GPIO_In	R	4	0.. 0x0000000F	0x0C000000
2	2	GPIO_Out	R/W	4	0.. 0x0000000F	0x0C000000
2	3	ADC_Values	R/W	4	0.. 0x03FF03FF	N/A
2	4	BattRSSI_Values	R/W	4	0.. 0x03FF00FF	N/A
2	5	PWM_Values	R/W	4	0.. 0x000003FF	0x00000000 (0 V)
2	6	GPIO_Config	R/W	4	0.. 0x00888888	0x22448800
2	7	GPIO_Set	W	4	0.. 0x0000000F	N/A
2	8	GPIO_Clear	W	4	0.. 0x0000000F	N/A

Table 4.14.2

GPIO_In - this parameter reads the state of the GPIO. This parameter is read-only.

GPIO_Out - this parameter sets the states of the GPIO pins configured as outputs.

ADC_ Values - this parameter holds the concatenation of the last readings of ADC1 and ADC0. The lower two bytes of this parameter hold the right justified 10-bit ADC0 reading. The upper two bytes of this parameter hold the right justified 10-bit ADC1 reading. The module's ADC_REF output (Pin 25) provides a 1.8 V ADC full-scale reference voltage to support ratiometric sensor measurements.

BattRSSI_Values - this parameter holds the concatenation of the current module input voltage and the RSSI value of the last received 802.11 packet. The lower two bytes of this parameter hold the right justified 8-bit RSSI reading. The upper two bytes of this parameter hold the right justified 10-bit input voltage reading, with a 3.600 V input providing a full scale 0x03FF count, or 3.519 mV/count.

PWM_Values - this parameter holds the concatenation of the PWM output values. The lower two bytes of this parameter hold the 16-bit PWM0 setting. The upper two bytes are reserved. Full scale PWM outputs equal the module input voltage.

GPIO_Config - this parameter sets the GPIO direction, the pullup/pulldown configuration of each GPI configured as input, and the alternate GPIO functions. The parameter consists of a four 4-bit fields, with each GPIO, 0 through 3, having a 4-bit field to control its configuration. A 0x0 field sets a GPIO as an input, 0x2 field sets a GPIO as an input with internal pulldown, 0x3 sets a GPIO as an input with an internal pullup, 0x4 selects output, and a 0x8 value specifies an alternate function where defined.

GPIO_Set - writing to this parameter location sets GPIO output values. Setting a '1' in a bit position *sets* the corresponding GPIO output to a logic high state. Only bits corresponding to GPIOs configured as out-

puts are effected. The four bit positions in this parameter are right registered, with GPIO0 in the right-most bit position.

GPIO_Clear - writing to this parameter location clears GPIO output values. Setting a '1' in a bit position *clears* the corresponding GPIO output to a logic low state. Only bits corresponding to GPIOs set as outputs have any effect. The four bit positions in this parameter are right registered, with GPIO0 in the right-most bit position.

Bank 3 - Module Serial Configuration

Bank	Location	Name	R/W	Size, bytes	Range	Default
3	1	SerialDivisor	R/W	4	$0..2^{16}-1$	0x30000000 (9600 bps)
3	2	SerialCharFormat	R/W	4	0..3	0x03000000 (8-bit)
3	3	SerialStopBits	R/W	4	0..1	0x00000000 (1 stop bit)
3	4	SerialParity	R/W	4	0..4	0x04000000 (no parity)
3	5	SerialRxTimeout	R/W	4	$0..2^{32}-1$	0x20000000 (32 ms)
3	6	SerialFlowControl	R/W	4	0..1	0x00000000 (flow control disabled)
3	7	DiagDivisior	R/W	4	$0..2^{32}-1$	0x30000000 (9600 bps)
3	8	DiagEnable	R/W	4	0..1	0x01000000 (diagnostic port enabled)
3	9	SPI_Mode	R/W	4	0..2	0x00000000 (disabled)
3	10	SPI_MasterClock-Divisor	R/W	4	$0..2^{32}-1$	0x64000000
3	11	SPI_MasterCmd-String	R/W	4	ASCII String	“” (null string)

Table 4.14.3

SerialDivisor - this parameter sets the main serial port baud rate, equal to 460800 divided by the SerialDivisor value.

SerialCharFormat - this parameter sets the format for the main serial port as follows (Big-Endian format):

0x00000000 for 5-bit format
0x00000001 for 6-bit format
0x00000002 for 7-bit format
0x00000003 for 8-bit format (default)

SerialStopBits - this parameter sets the number of stop bits for the main serial port as follows :

0x00000000 for 1 stop bit (default)
0x00000001 for 2 stop bits

SerialParity - this parameter sets the parity configuration for the main serial port as follows:

0x00000000 for odd parity
0x00000001 for even parity
0x00000002 for mark parity
0x00000003 for space parity
0x00000004 for no parity (default)

SerialRxTimeout - this parameter sets the received message timeout for the main serial port. A received message is interpreted as complete when no additional bytes are received during a timeout interval. The SerialRxTimeout parameter is scaled in milliseconds.

SerialFlowControl - this parameter enables/disables /HOST_RTS - /HOST_CTS hardware flow control:

0x00000000 disables flow control (default)
0x00000001 for enables flow control

DiagDivisior - this parameter sets this parameter sets the diagnostic serial port baud rate, equal to 460800 divided by the DiagDivisor value.

DiagEnable - this parameter enables/disables diagnostic port operation:

0x00000000 disables diagnostic port operation
0x00000001 enables diagnostic port operation (default)

SPI_Mode - this parameter sets the SPI port mode:

0x00000000 to disable SPI port (default)
0x00000002 to enable SPI master mode

SPI_MasterClockDivisor - this parameter sets SPI master mode bit rate, equal to 11,000,000 divided by the SPI_MasterClockDivisor value.

SPI_MasterCmdString - this parameter holds the command string to clock into the peripheral SPI slave when the module is acting as an SPI master.

Bank 4 - Module WLAN Configuration

Bank	Location	Name	R/W	Size, bytes	Range	Default
4	1	Ap1_Ssid	R/W	32	ASCII String	"WSN-Default"
4	2	Ap1_Channel	R/W	4	0..11	0x0B000000
4	3	Ap1_AuthMode	R/W	4	1..8	0x03000000 (automatic authentication)
4	4	Ap1_EncryptionMode	R/W	4	1..165	0xA5000000 (all)
4	5	Ap1_PskPassphrase	R/W	32	ASCII String	"WSN-PASSWORD"
4	6	Ap1_PskKey	R/W	32	ASCII String	N/A
4	7	Ap1_WepKeyId	R/W	1	0..3	N/A
4	8	Ap1_WepKeyLength	R/W	1	5..13	N/A
4	9	Ap1_WepKeyValue	R/W	13	ASCII String	N/A
4	10	Ap2_Ssid	R/W	32	ASCII String	"WSN-Default"
4	11	Ap2_Channel	R/W	4	0..11	0x0B000000
4	12	Ap2_AuthMode	R/W	4	1..8	0x03000000 (automatic authentication)
4	13	Ap2_EncryptionMode	R/W	4	0..165	0xA5000000 (all)
4	14	Ap2_PskPassphrase	R/W	32	ASCII String	"WSN-PASSWORD"
4	15	Ap2_PskKey	R/W	32	ASCII String	N/A
4	16	Ap2_WepKeyId	R/W	1	0..3	N/A
4	17	Ap2_WepKeyLength	R/W	1	5..13	N/A
4	18	Ap2_WepKeyValue	R/W	13	ASCII String	N/A
4	19	Ap3_Ssid	R/W	32	ASCII String	"WSN-Default"
4	20	Ap3_Channel	R/W	4	0..11	0x0B000000
4	21	Ap3_AuthMode	R/W	4	1..8	0x03000000 (automatic authentication)
4	22	Ap3_EncryptionMode	R/W	4	0..165	0xA5000000 (all)
4	23	Ap3_PskPassphrase	R/W	32	ASCII String	"WSN-PASSWORD"
4	24	Ap3_PskKey	R/W	32	ASCII String	N/A
4	25	Ap3_WepKeyId	R/W	1	0..3	N/A
4	26	Ap3_WepKeyLength	R/W	1	5..13	N/A
4	27	Ap3_WepKeyValue	R/W	13	ASCII String	N/A
4	28	AdHoc_Ssid	R/W	32	ASCII String	"RFM-*MAC*" (*MAC* = modules MAC addr.)
4	29	AdHoc_Channel	R/W	4	0..11	0x0B000000
4	30	AdHoc_AuthMode	R/W	4	1..8	0x01000000 (open authentication)
4	31	AdHoc_Encryption-Mode	R/W	4	0..165	0x08000000 (none)
4	32	AdHoc_Psk-Passphrase	R/W	32	ASCII String	"" (null bytes)
4	33	AdHoc_PskKey	R/W	32	ASCII String	"" (null bytes)
4	34	AdHoc_WepKeyId	R/W	1	0..3	N/A
4	35	AdHoc_WepKey-Length	R/W	1	5..13	N/A
4	36	AdHoc_WepKeyValue	R/W	13	ASCII String	N/A

Table 4.14.4

Ap1_Ssid - this parameter holds the SSID of preferred access point 1.

Ap1_Channel - this parameter sets the channel of operation for preferred access point 1.

Ap1_AuthMode - this parameter sets the authentication mode for preferred access point 1.
as follows (Big-Endian format):

0x00000001 - open authentication, allows connection to a WLAN with any type of authentication
0x00000002 - shared authentication, allows connection to a WLAN only with WEP shared authentication; must be used to connect to a WEP shared WLAN
0x00000003 - automatic authentication, allows connection to a WLAN with any type of authentication
0x00000004 - WPA authentication, allows connection to a WLAN with WPA/WPA2 802.1x authentication; WPA2 is selected over WPA if both are present
0x00000005 - WPAPSK authentication, allows connection to a WLAN with WPA/WPA2 authentication; WPA2 is selected over WPA if both are present
0x00000007 - WPA2 authentication, allows connection to a WLAN with WPA2 802.1x authentication
0x00000008 - WPA2PSK authentication, allows connection to a WLAN with WPA2 PSK authentication

Ap1_EncryptionMode - this parameter selects the encryption mode for preferred access point 1 as follows:

0x00000001 - allows connection to a WLAN only with WEP encryption
0x00000004 - allows connection to a WLAN only with TKIP encryption, either unicast or group; it connects if either unicast cipher or group cipher is TKIP
0x00000020 - allows connection to a WLAN only with CCMP encryption, either unicast or group; it connects if either unicast cipher or group cipher is CCMP
0x00000080 - allows connection to a WLAN only with no encryption
0x000000A5 - allows connection to a WLAN with any of the above modes

Note that encryption mode 0x000000A5 is derived from logically ORing the values of the four encryption modes above it. Any combination of ORed encryption modes are allowed.

Ap1_PskPassphrase - this parameter holds the PSK passphrase for preferred access point 1.

Ap1_PskKey - this parameter holds the PSK key for preferred access point 1.

Ap1_WepKeyId - this parameter sets the WEP key ID for preferred access point 1. The range of this parameter is 0x00000000 to 0x00000003.

Ap1_WepKeyLength - this parameter sets the WEP key length in bytes for preferred access point 1. The range of this parameter is 0x00000005 to 0x0000000D (5 to 13).

Ap1_WepKeyValue - this parameter holds the WEP key ASCII string for access point 1, 5 to 13 bytes. The number of bytes must match the WEP key length.

Ap2_Ssid - this parameter holds the SSID of preferred access point 2.

Ap2_Channel - this parameter sets the channel of operation for preferred access point 2.

Ap2_AuthMode - this parameter sets the authentication mode for preferred access point 2 as follows:

0x00000001 - open authentication, allows connection to a WLAN with any type of authentication
0x00000002 - shared authentication, allows connection to a WLAN only with WEP shared authentication; must be used to connect to a WEP shared WLAN

0x00000003 - automatic authentication, allows connection to a WLAN with any type of authentication
0x00000004 - WPA authentication, allows connection to a WLAN with WPA/WPA2 802.1x authentication; WPA2 is selected over WPA if both are present
0x00000005 - WPAPSK authentication, allows connection to a WLAN with WPA/WPA2 authentication; WPA2 is selected over WPA if both are present
0x00000007 - WPA2 authentication, allows connection to a WLAN with WPA2 802.1x authentication
0x00000008 - WPA2PSK authentication, allows connection to a WLAN with WPA2 PSK authentication

Ap2_EncryptionMode - this parameter selects the encryption mode for preferred access point 2 as follows:

0x00000001 - allows connection to a WLAN only with WEP encryption
0x00000004 - allows connection to a WLAN only with TKIP encryption, either unicast or group; it connects if either unicast cipher or group cipher is TKIP
0x00000020 - allows connection to a WLAN only with CCMP encryption, either unicast or group; it connects if either unicast cipher or group cipher is CCMP
0x00000080 - allows connection to a WLAN only with no encryption
0x000000A5 - allows connection to a WLAN with any of the above modes

Note that encryption mode 0x000000A5 is derived from logically ORing the values of the four encryption modes above it. Any combination of ORed encryption modes are allowed.

Ap2_PskPassphrase - this parameter holds the PSK passphrase for preferred access point 2.

Ap2_PskKey - this parameter holds the PSK key for preferred access point 2.

Ap2_WepKeyId - this parameter sets the WEP key ID for preferred access point 2. The range of this parameter is 0 to 3.

Ap2_WepKeyLength - this parameter sets the WEP key length in bytes for preferred access point 2. The range of this parameter is 0x00000005 to 0x0000000D (5 to 13).

Ap2_WepKeyValue - this parameter holds the WEP key ASCII string for preferred access point 2, 5 to 13 bytes. The number of bytes must match the WEP key length.

Ap3_Ssid - this parameter holds the SSID of preferred access point 3.

Ap3_Channel - this parameter sets the channel of operation for preferred access point 3.

Ap3_AuthMode - this parameter sets the authentication mode for preferred access point 3 as follows:

0x00000001 - open authentication, allows connection to a WLAN with any type of authentication
0x00000002 - shared authentication, allows connection to a WLAN only with WEP shared authentication; must be used to connect to a WEP shared WLAN
0x00000003 - automatic authentication, allows connection to a WLAN with any type of authentication
0x00000004 - WPA authentication, allows connection to a WLAN with WPA/WPA2 802.1x authentication; WPA2 is selected over WPA if both are present

0x00000005 - WPAPSK authentication, allows connection to a WLAN with WPA/WPA2 authentication; WPA2 is selected over WPA if both are present
0x00000007 - WPA2 authentication, allows connection to a WLAN with WPA2 802.1x authentication
0x00000008 - WPA2PSK authentication, allows connection to a WLAN with WPA2 PSK authentication

Ap3_EncryptionMode - this parameter selects the encryption mode for preferred access point 3 as follows:

0x00000001 - allows connection to a WLAN only with WEP encryption
0x00000004 - allows connection to a WLAN only with TKIP encryption, either unicast or group; it connects if either unicast cipher or group cipher is TKIP
0x00000020 - allows connection to a WLAN only with CCMP encryption, either unicast or group; it connects if either unicast cipher or group cipher is CCMP
0x00000080 - allows connection to a WLAN only with no encryption
0x000000A5 - allows connection to a WLAN with any of the above modes

Note that encryption mode 0x000000A5 is derived from logically ORing the values of the four encryption modes above it. Any combination of ORed encryption modes are allowed.

Ap3_PskPassphrase - this parameter holds the PSK passphrase for preferred access point 3.

Ap3_PskKey - this parameter holds the PSK key for preferred access point 3.

Ap3_WepKeyId - this parameter is holds WEP key ID for preferred access point 3. The range of this parameter is 0 to 3.

Ap2_WepKeyLength - this parameter sets the WEP key length for preferred access point 3. The range of this parameter is 0x00000005 to 0x0000000D (5 to 13).

Ap2_WepKeyValue - this parameter holds the WEP key ASCII string for preferred access point 3, 5 to 13 bytes; the number of bytes must match the WEP key length.

AdHoc_Ssid - this parameter is the SSID for a fallback Ad Hoc server. Setting this parameter to null bytes disables Ad Hoc fallback.

AdHoc_Channel - this parameter sets the channel for Ad Hoc operation.

AdHoc_AuthMode - this parameter sets the authentication mode for Ad Hoc operation as follows:

0x00000001 - open authentication, allows connection to a WLAN with any type of authentication
0x00000002 - shared authentication, allows connection to a WLAN only with WEP shared authentication; must be used to connect to a WEP shared WLAN
0x00000003 - automatic authentication, allows connection to a WLAN with any type of authentication
0x00000004 - WPA authentication, allows connection to a WLAN with WPA/WPA2 802.1x authentication; WPA2 is selected over WPA if both are present
0x00000005 - WPAPSK authentication, allows connection to a WLAN with WPA/WPA2 authentication; WPA2 is selected over WPA if both are present
0x00000007 - WPA2 authentication, allows connection to a WLAN with WPA2 802.1x authentication

0x00000008 - WPA2PSK authentication, allows connection to a WLAN with WPA2 PSK authentication

AdHoc_EncryptionMode - this parameter selects the Ad Hoc encryption mode as follows:

0x00000001 - allows connection to a WLAN only with WEP encryption
0x00000004 - allows connection to a WLAN only with TKIP encryption, either unicast or group; it connects if either unicast cipher or group cipher is TKIP
0x00000020 - allows connection to a WLAN only with CCMP encryption, either unicast or group; it connects if either unicast cipher or group cipher is CCMP
0x00000080 - allows connection to a WLAN only with no encryption
0x000000A5 - allows connection to a WLAN with any of the above modes

Note that encryption mode 0x000000A5 is derived from logically ORing the values of the four encryption modes above it. Any combination of ORed encryption modes are allowed.

AdHoc_PskPassphrase - this parameter holds the PSK passphrase for Ad Hoc operation.

AdHoc_PskKey - this parameter holds the PSK key for Ad Hoc operation.

AdHoc_WepKeyId - this parameter is holds WEP key ID for Ad Hoc operation. The range of this parameter is 0 to 3.

AdHoc_WepKeyLength - this parameter sets the WEP key length for Ad Hoc operation. The range of this parameter is 0x00000005 to 0x0000000D (5 to 13).

AdHoc_WepKeyValue - this parameter holds the WEP key ASCII string for Ad Hoc operation, 5 to 13 bytes. The number of bytes must match the *AdHoc_WepKeyLength* value.

Bank 5 - Module Node Options

Bank	Location	Name	R/W	Size, bytes	Range	Default
5	1	PsPollTimer	R/W	8	ASCII String	“0000007800000000” (60 s)
5	4	RestoreFactoryCfg	W	4	0..1	N/A
5	6	RebootNode	W	4	0..1	N/A
5	10	BatteryReadFreq	R/W	4	0..1023	0x01000000 (reads on each TX)
5	12	BatteryWarning-LevelInMVolt	R/W	4	0..1023	0xFC080000 (2300 mV)
5	13	BatteryStandby-LevelInMVolt	R/W	4	0..1023	0xF6090000 (2550 mV)
5	14	BatteryRFirstBoot-StandbyLevel-InMVolt	R/W	4	0..1023	0xFC080000 (2550 mV)
5	16	DisableStdBy	W	4	0..2	N/A
5	17	SystemTime	R/W	8	ASCII String	N/A

Table 4.14.5

PsPollTimer - this parameter sets the interval that the module polls the access point to send any data the access point is holding for it. Parameter scaling is in microcontroller clock cycles of 0.029802322 µs. This parameter is a 64-bit number formatted as an ASCII string of the equivalent hexadecimal value.

RestoreFactoryCfg - writing any non-zero value to this location will restore module parameters with factory defaults to their default values. This parameter is write-only.

RebootNode - writing any non-zero value to this location performs a “battery-plugged” reboot of the module. This parameter is write-only.

BatteryReadFreq - This parameter holds the number of transmissions from one battery reading to the next.

BatteryWarningLevelInMVolt - this parameter sets the battery voltage level that triggers a low battery warning trap message. The parameter scaling is in millivolts, formatted as a 4-byte hexadecimal number.

BatteryStandbyLevelInMVolt - this parameter sets the battery voltage level that triggers the node to switch to standby. The parameter scaling is in millivolts, formatted as a 4-byte hexadecimal number. WARNING: setting the value of this parameter too high can “lock up” the module.

BatteryRFirstBootStandbyLevelInMVolt - this parameter sets the battery voltage level that triggers the node to switch to standby immediately when booted up. The parameter scaling is in millivolts, formatted as a 4-byte hexadecimal number. WARNING: setting the value of this parameter too high can “lock up” the module.

DisableStdBy - writing a non-zero value to this location will disable the module standby function. This parameter is write-only.

SystemTime - this parameter holds the time interval since the module was last booted. The parameter scaling is in microcontroller clock cycles of 0.029802322 µs. This parameter is a 64-bit number formatted as an ASCII string of the equivalent hexadecimal value.

Bank 6 - Module Scanning Authentication

Bank	Location	Name	R/W	Size, bytes	Range	Default
6	1	ScanType	R/W	4	0..1	0x00000000 (active)
6	6	EapOuterAuthType	R	4	$0..2^{32}$ -1	N/A
6	7	EapInnerAuthType	R	4	$0..2^{32}$ -1	N/A
6	8	RadiusUserName	W	15	ASCII String	N/A
6	9	RadiusPasswd	W	15	ASCII String	N/A
6	17	EaptlsProvision-CaCert	W	15	ASCII String	N/A
6	18	EaptlsProvision-ClientCert	W	15	ASCII String	N/A
6	19	EaptlsProvision-PvtKey	W	15	ASCII String	N/A

Table 4.14.6

ScanType - this parameter selects active or passive scan mode as follows (Big-Endian format):

0x00000000 for active scan mode
0x00000000 for passive scan mode

EapOuterAuthType - this parameter holds the outer authentication type used for EAP_FAST. It is formatted as a 32-bit hexadecimal value. This parameter is read-only.

EapInnerAuthType - this parameter holds the inner authentication type used for EAP_FAST. It is formatted as a 32-bit hexadecimal value. This parameter is read-only

RadiusUserName - this parameter holds the ASCII user name for authenticating with the RADIUS server. This parameter is write-only.

RadiusPasswd - this parameter hold the ASCII password for authenticating with the RADIUS server. This parameter is write-only.

EaptlsProvisionCaCert - this parameter holds the ASCII EAP-TLS Provision Certificate Authority Certificate. This parameter is write-only.

EaptlsProvisionClientCert - this parameter holds the ASCII EAP_TLS Provision Client Certificate. This parameter is write-only.

EaptlsProvisionPvtKey - this parameter holds the Provision Private Key as an ASCII string. This parameter is write-only.

Bank 7 - Module Network Configuration

Bank	Location	Name	R/W	Size, bytes	Range	Default
7	1	IpAddress	R/W	4	Class A,B,C	0x00000000 (0.0.0.0)
7	2	SubnetMask	R/W	4	Class A,B,C	0x00000000 (0.0.0.0)
7	3	GatewayIpAddress	R/W	4	Class A,B,C	0x00000000 (0.0.0.0)
7	4	PerformDhcp	R/W	4	Class A,B,C	0x00000000 (enabled)
7	5	MacAddress	R	4	OID 00:30:66	unique for each module
7	7	PrimaryDns-IpAddress	R/W	4	Class A,B,C	0x00000000 (0.0.0.0)
7	8	SecondaryDns-IpAddress	R/W	4	Class A,B,C	0x00000000 (0.0.0.0)
7	9	CurntIpAddress	R	4	Class A,B,C	N/A
7	10	CurntSubnetAddress	R	4	Class A,B,C	N/A
7	11	CurntGateway-IpAddress	R	4	Class A,B,C	N/A
7	12	CurntPrimary-DnsIpAddress	R	4	Class A,B,C	N/A
7	13	CurntSecondary-DnsIpAddress	R	4	Class A,B,C	N/A
7	14	DHCPLeaseTime	R	8	ASCII String	N/A

Table 4.14.7

IpAddress - this parameter holds the module's IP address if DHCP is disabled. The address must be a valid unicast address. The address is formatted as a 32-bit hexadecimal number.

SubnetMask - this parameter holds the subnet mask for the WLAN interface. The mask is formatted as a 32-bit hexadecimal number.

GatewayIpAddress - this parameter holds the subnet gateway IP address. The address is formatted as a 32-bit hexadecimal number.

PerformDhcp - this parameter sets the IP address mode as follows (Big-Endian format):

0x00000000 - for DHCP
0x00000001 - for static IP address

MacAddress - this parameter holds the module's unique MAC address. This parameter is read-only.

PrimaryDnsIpAddress - this parameter holds the IP address for the primary DNS server. The address is formatted as a 32-bit hexadecimal number.

SecondaryDnsIpAddress - this parameter holds the IP address for the secondary DNS server. The address is formatted as a 32-bit hexadecimal number.

CurntIpAddress - this parameter holds the current IP address assigned to the module. The address is formatted as a 32-bit hexadecimal number. This parameter is read-only.

CurntSubnetAddress - this parameter hold the IP address of the current subnet address. The address is formatted as a 32-bit hexadecimal number. This parameter is read-only.

CurntGatewayIpAddress - this parameter hold the IP address of the current subnet gateway address. The address is formatted as a 32-bit hexadecimal number. This parameter is read-only.

CurntPrimaryDnsIpAddress - this parameter hold the IP address of the current primary DNS server. The address is formatted as a 32-bit hexadecimal number. This parameter is read-only.

CurntSecondaryDnsIpAddress - this parameter hold the IP address of the current secondary DNS server. The address is formatted as a 32-bit hexadecimal number. This parameter is read-only.

DHCPLeaseTime - this parameter holds the time remaining on the current DHCP lease. The parameter scaling is in microcontroller clock cycles of 0.029802322 µs. This parameter is a 64-bit number formatted as an ASCII string of the equivalent hexadecimal value. This parameter is read-only.

Bank 8 - Module SNMP Configuration

Bank	Location	Name	R/W	Size, bytes	Range	Default
8	3	PrimarySNMPMgrIp	R/W	4	Class A,B,C	0x00000000 (0.0.0.0)
8	4	Secondary-SNMPMgrIp	R/W	4	Class A,B,C	0xC803A8C0 (192.168.3.200)
8	10	GetCommString	R/W	15	ASCII String	"GSN_GET"
8	11	SetCommString	R/W	15	ASCII String	"GSN_SET"
8	12	TrapCommString	R/W	15	ASCII String	"GSN_TRAP"
8	15	SnmpTrapSrcPort	R/W	4	$1..2^{16}-1$	0xA2000000 (162)
8	16	SnmpTrapDstPort	R/W	4	$1..2^{16}-1$	0xA1000000 (161)

Table 4.14.8

PrimarySNMPMgrIp - this parameter holds the IP address of the primary SNMP manager. The address is formatted as a 32-bit hexadecimal number.

SecondarySNMPMgrIp - this parameter holds the IP address of the secondary SNMP manager. The address is formatted as a 32-bit hexadecimal number.

GetCommString - this parameter holds the get community ASCII string.

SetCommString - this parameter holds the set community ASCII string.

TrapCommString - this parameter holds the trap community ASCII string.

*SnmpTrapSrcPor*t - this parameter holds the port number for the SNMP trap source. The parameter is formatted as a 32-bit hexadecimal number.

SnmpTrapDstPort - this parameter holds the port number for the SNMP trap destination. The parameter is formatted as a 32-bit hexadecimal number.

5.0 IP Address Discovery Protocol

The WSN802G module supports a separate UDP client port that provides a discovery protocol for wireless communications. The discovery protocol is used to find IP addresses of modules in a network when the IP addresses have been assigned by a DHCP server. The discovery protocol is also used to set the module's SNMP Server IP addresses which enables module commissioning. This protocol uses port 24776. The discovery protocol datagrams are shown in Table 5.0.1.

Opcode	Direction	Description
0x0071	Server-to-Module	IP_HUNT_QUERY
0x0072	Module-to-Server	IP_HUNT_REPLY

Table 5.0.1

Since the IP addresses of potential recipients may not be known, both query and reply messages are sent as UDP broadcasts. UDP broadcasts are not routed, so only nodes on the same network segment as the sender will respond. All nodes that hear an IP_HUNT_QUERY will respond with an IP_HUNT_ REPLY. Ordinarily these commands are only used to initially commission a module. Since a WSN802G module must be in active mode to hear a command, an un-commissioned module stays in active mode until its Primary SNMP server address has been set.

5.1 IP Hunt Query

The IP_HUNT_QUERY datagram shown in Figure 5.1.1 is broadcast by a commissioning server to discover WSN802G modules:

Byte 0	Byte 1	Byte 2	Byte 3
I	P	H	u
n	t	e	r
Opcode = 0x7100		Primary SNMP Server IP Address [31:16]	
Primary SNMP Server IP Address [15:0]		Secondary SNMP Server IP Address [31:16]	
Secondary SNMP Server IP Address [15:0]			

Figure 5.1.1

The fields specific to this datagram are:

SNMP Server Address: The Primary and Secondary SNMP server address fields inform WSN802G modules of the SNMP server addresses to solicit for configuration parameter updates (destination addresses for Config traps). If either or both server address fields are set to 0.0.0.0, a module hearing the message will retain its current SNMP server setting for the respective field(s). The default IP addresses for the Primary SNMP server is 0.0.0.0. The default IP address for the Secondary SNMP server is 192.168.3.200. The Primary SNMP Server IP address must be set to a different value to allow the module to enter normal sleep mode.

5.2 IP Hunt Reply

The IP_HUNT_REPLY datagram shown in Figure 5.2.1 is sent by a module in response to an IP_HUNT_QUERY command:

Byte 0	Byte 1	Byte 2	Byte 3
I	P	H	u
n	t	e	r
Opcode = 0x7200		MAC Address [47:32]	
MAC Address [31:0]			
IP Address			
Subnet Mask			
Device Code = 0x0102		Hardware Version	
Firmware Version			

Figure 5.2.1

The fields specific to this datagram are:

MAC Address: MAC address of the module

IP Address: IP address of the module

Subnet Mask: Subnet mask of the module

Device Code: Unique WSN802G device code - 0x0102

Hardware Version: Hardware version of the module

Firmware Version: Firmware version in the module

6.0 SSID, Channel, Encryption and Authentication

WSN802G modules support SSID, channel, encryption and authentication mode options for three access point connections plus an Ad Hoc connection. The SSID, channel, encryption and authentication mode options are summarized in Tables 4.14.1 and 4.14.6 in Section 14 above.

6.1 Connection Scanning

To establish an 802.11b/g/n connection, WSN802G modules will scan for Access Point 1, then 2, then 3 and then switch to Ad Hoc server mode and broadcast for a connection. This sequence repeats until a connection is established. This scanning sequence is automatic. Note that failure to make a quick connection will adversely affect battery operating life.

6.2 Ad Hoc Mode

WSN802G modules support Ad Hoc operation. Ad Hoc operation is an 802.11 mode where two non-access point devices (sometimes called non-infrastructure devices) communicate directly with each other in a point-to-point network. Ad Hoc mode can be used to configure a WSN802G device over a wireless link without needing an access point.

In an Ad Hoc network, there are server nodes and client nodes. Server nodes broadcast their presence and client nodes, once they have heard a server node, will request to join. Some devices, such as PCs, can operate as both a client and a server, where they broadcast their presence but also listen for other devices that are advertising their presence. The WSN802G module acts only as a server. That is, it advertises its presence and waits for requests from client devices to join.

The WSN802G module will automatically enter Ad Hoc mode if it is unable to join any of the three networks specified in the preferred SSID parameters. To operate in Ad Hoc mode, the WSN802G module must have DHCP disabled, have an IP address statically assigned as well as a subnet mask and a gateway IP address. These values can be set through the serial port in Bank 7, or through SNMP commands. Refer to Sections 4 and 7 in this manual for details. Since an Ad Hoc session will terminate if a WSN-802G module enters sleep mode, the module should be kept awake by asserting the WAKE_IN signal.

When the WSN802G module is in Ad Hoc mode it will advertise itself using the SSID “RFM-*macaddress*” where *macaddress* is an ASCII string of the module’s MAC address with the colons between values removed. A device wishing to join the WSN802G Ad Hoc network must have a statically set IP address that is on the same subnet as the WSN802G IP address. The WSN802G module Ad Hoc mode also allows the choice of an RF channel and encryption mode. A device wanting to join the WSN802G Ad Hoc network must have these parameters agree with the values set in the WSN802G module.

When operating in Ad Hoc mode, the WSN802G module will operate in the same way as when it is connected to an access point. This means the module will send Linkup and Config Traps as when in normal access point operation. A device connected to the WSN802G module can simply ignore these transmissions.

7.0 SNMP Module Configuration

A WSN802G can be configured two ways - (1) through API CFG parameter commands as discussed in Section 4 above, or (2) through SNMP maintenance commands sent over the wireless link in response to SNMP configuration requests (Config traps) from the module. WSN802G modules support two SNMP OID parameter sets, as discussed in Section 7.2 below. The first set of OIDs is very similar to the OIDs in the WSN802G modules prior to revision F. There are a few OIDs that did change - these are detailed in the document *11 Mbps Firmware Changes* which can be found on the CD in the developer's kit or on the RFM website in the Module Tech Support section.

The second set of OIDs, distinguished by a different left-hand string, includes most but not all of the OIDs in the first set. Traps, time-related parameters, and firmware upgrade related items are only found in the first set of OIDs. Where there is duplication, it does not matter which OID is used as they operate identically. The last two, non-zero, OID values of the second set of OIDs are the same as the Bank and Location values used in the serial port configuration commands. For example, to read the *AutoReportInveral* using SNMP, an SNMP Get to OID 1.3.6.1.4.1.32345.88.1.2.0 would be issued. To read the module's GPIOs through the UART or SPI port, the CFG_READ command would be issued with Bank 1, Location 2.

7.1 SNMP Traps

WSN802G modules can generate four trap messages - request to update configuration, request to maintain association, request to update time, and low battery warning. The port numbers used for SNMP are 162 for source and 161 for destination SNMP commands.

A Config trap is a message sent periodically to poll the SNMP server to see if it has any commands waiting for it. After sending the trap, the module remains awake for a period of four seconds to allow the server to send it commands. By default, the module issues Config Traps over the wireless link every 20 seconds. Config Traps are sent to the SNMP server address in the IPHunter device discovery communications. The interval between Config Traps is a configurable parameter. The less frequently Config Traps are issued by the module, the longer the battery life will be, but the longer it takes to change the module's configuration over the wireless link. Because Config trap activity requires a significant amount of energy to execute, for battery-powered deployments the user should set this interval to once an hour or a few times a day to conserve battery life. At receiving a Config trap, the server should immediately send a ConfigComplete command to indicate it has no commands to send. This allows the module to go back to sleep mode, rather than remaining in active mode for rest of the configuration window.

The Linkup trap is a message sent periodically by the module to maintain its association with its 802.11b/g/n access point. No information is conveyed, just "I'm here". The period of the Linkup trap is set by the *LinkupTrapInterval* system parameter. For compatibility with the majority of 802.11b/g/n routers, the default period is 10 seconds. Note - this parameter should not be set above two minutes without contacting RFM technical support.

The Time Sync Update trap is a message requesting the time server to send updated time. Windows XP and later Microsoft operating systems include a time server, allowing the module to get its time from a PC.

The *BatteryWarningLevelInMVolt* parameter (see Table 4.14.5) sets the battery voltage level that triggers a Low Battery Warning trap message. The default setting for a low battery warning is 2.3 V.

The WSNConfig utility included with the developer's kit is designed to operate as a commissioning utility for the WSN802G. The WSNConfig utility allows for each node to be configured independently or as a whole. Individual settings may be configured or a list of configuration parameters can be queued for transmission when the node or nodes wake up and issue the Config Trap. Alternatively, a third party SNMP server or utility may be used to serve the same function.

7.2 SNMP Parameter OIDs

Tables 7.2.1 through 7.2.7 detail the SNMP parameter OIDs supported by the WSN802G firmware Rev F or later with left-side fields of **1.3.6.1.4.1.28295.1.1**. The remaining OID fields for each parameter are shown in the left column of each table.

Shown below is the most commonly occurring sequence of SNMP messages between a module and its access point/server. The sequence of messages begins with a Config Trap from the module. In most cases the server will not need to update the module's configuration, so it will send a ConfigComplete message which allows the module to immediately return to sleep mode. The module acknowledges the message and returns to sleep.

Config Trap from Module:

```
04 08 47 53 4E 5F 54 52 41 50 A4 3A 06 09 2B 06
01 04 01 81 DD 07 01 40 04 C0 A8 03 A6 02 01 06
02 01 02 43 04 80 86 C4 6B 30 1B 30 82 00 17 06
0D 2B 06 01 04 01 81 DD 07 01 01 04 02 02 04 06
00 30 66 50 01 00
```

ConfigComplete from Access Point/Server:

```
04 07 47 53 4E 5F 53 45 54 A3 20 02 02 00 6A 02
01 00 02 01 00 30 14 30 12 06 0D 2B 06 01 04 01
81 D0 70 10 10 40 60 00 20 01 01
```

ConfigComplete ACK from Module:

```
04 07 47 53 4E 5F 53 45 54 A2 20 02 02 00 6A 02
01 00 02 01 00 30 14 30 12 06 0D 2B 06 01 04 01
81 DD 07 01 01 04 06 00 02 01 01
```

Generic Configuration Parameter (genericcfg) OIDs:
1.3.6.1.4.1.28295.1.1 + OID End

OID End	Name	R/W	Size, bytes	Range	Default
.1.6.0	gsnRebootNode	W	4	0..1	N/A
.1.10.0	gsnBatteryReadFrequency	R/W	4	0..1023	0x00000001 (read on each TX)
.1.12.0	gsnBatteryWarning-LevelInMVolt	R/W	4	0..1023	0x000008FC (2300 mV)
.1.13.0	gsnBatteryStandby-LevelInMVolt	R/W	4	0..1023	0x000009F6 (2550 mV)
.1.14.0	gsnBatteryRFirstBoot-StandbyLevelInMVolt	R/W	4	0..1023	0x000008FC (2550 mV)
.1.16.0	gsnDisableStdBy	W	4	0..2	N/A
.1.17.0	gsnSystemTime	R	8	ASCII String	N/A

Table 7.2.1

gsnRebootNode - writing any non-zero value to this location performs a “battery-plugged” reboot of the module. This parameter is write-only.

gsnBatteryReadFrequency - This parameter holds the number of transmissions from one battery reading to the next.

gsnBatteryWarningLevelInMVolt - this parameter sets the battery voltage level that triggers a low battery warning trap message. The parameter scaling is in millivolts, formatted as a 4-byte hexadecimal number.

gsnBatteryStandbyLevelInMVolt - this parameter sets the battery voltage level that triggers the node to switch to standby. The parameter scaling is in millivolts, formatted as a 4-byte hexadecimal number. WARNING: setting the value of this parameter too high can “lock up” the module.

gsnBatteryRFirstBootStandbyLevelInMVolt - this parameter sets the battery voltage level that triggers the node to switch to standby immediately when booted up. The parameter scaling is in millivolts, formatted as a 4-byte hexadecimal number. WARNING: setting the value of this parameter too high can “lock up” the module.

gsnDisableStdBy - writing a non-zero value to this location will disable the module standby function. This parameter is write-only.

gsnSystemTime - this parameter holds the time interval since the module was last booted. The parameter scaling is in microcontroller clock cycles of 0.029802322 µs. This parameter is a 64-bit number formatted as an ASCII string of the equivalent hexadecimal value. The system time parameter rolls over every 128 seconds. This parameter is read-only.

Scanning/Authentication/Association Parameter (saa802dot11) OIDs:
1.3.6.1.4.1.28295.1.1 + OID End

OID End	Name	R/W	Size, bytes	Range	Default
.2.1.0	gsnScanType	R/W	4	0..1	0x00000000 (active)
.2.6.0	gsnOuterAuthType	R	4	$0..2^{32}$ -1	N/A
.2.7.0	gsnInnerAuthType	R	4	$0..2^{32}$ -1	N/A
.2.8.0	gsnUserName	W	15	ASCII String	N/A
.2.9.0	gsnPassword	W	15	ASCII String	N/A
.2.17.0	gsnEaptlsProvisionCaCert	W	15	ASCII String	N/A
.2.18.0	gsnEaptlsProvision ClientCert	W	15	ASCII String	N/A
.2.19.0	gsnEaptlsProvision PvtKey	W	15	ASCII String	N/A

Table 7.2.2

gsnScanType - this parameter selects active or passive scan mode as follows:

0x00000000 for active scan mode
0x00000000 for passive scan mode

gsnOuterAuthType - this parameter holds the outer authentication type used for EAP_FAST. It is formatted as a 32-bit hexadecimal value. This parameter is read-only.

gsnInnerAuthType - this parameter holds the inner authentication type used for EAP_FAST. It is formatted as a 32-bit hexadecimal value. This parameter is read-only.

gsnUserName - this parameter holds the ASCII user name for authenticating with the RADIUS server. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnPassword - this parameter hold the ASCII password for authenticating with the RADIUS server. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnEaptlsProvisionCaCert - this parameter this parameter holds the ASCII EAP-TLS Provision Certificate Authority Certificate. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnEaptlsProvisionClientCert - this parameter this parameter holds the ASCII EAP_TLS Provision Client Certificate. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnEaptlsProvisionPvtKey - this parameter - this parameter holds the Provision Private Key as an ASCII string. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

Network Configuration Parameter (networkcfg) OIDs:
1.3.6.1.4.1.28295.1.1 + OID End

OID End	Name	R/W	Size, bytes	Range	Default
.3.1.0	gsnIpAddress	R/W	4	Class A,B,C	0x00000000 (0.0.0.0)
.3.2.0	gsnSubnetAddress	R/W	4	Class A,B,C	0x00000000 (0.0.0.0)
.3.3.0	gsnGatewayIpAddress	R/W	4	Class A,B,C	0x00000000 (0.0.0.0)
.3.4.0	gsnPerformDhcp	R/W	4	Class A,B,C	0x00000000 (enabled)
.3.5.0	gsnMacAddress	R	4	OID 00:30:66	unique for each module
.3.7.0	gsnPrimaryDnsIpAddress	R/W	4	Class A,B,C	0x00000000 (0.0.0.0)
.3.8.0	gsnSecondaryDnsIpAddress	R/W	4	Class A,B,C	0x00000000 (0.0.0.0)
.3.9.0	gsnCurntIpAddress	R	4	Class A,B,C	N/A
.3.10.0	gsnCurntSubnetAddress	R	4	Class A,B,C	N/A
.3.11.0	gsnCurntGatewayIpAddress	R	4	Class A,B,C	N/A
.3.12.0	gsnCurntPrimaryDns-IpAddress	R	4	Class A,B,C	N/A
.3.13.0	gsnCurntSecondaryDns-IpAddress	R	4	Class A,B,C	N/A
.3.14.0	gsnDHCPLeaseTime	R	8	ASCII String	N/A

Table 7.2.3

gsnIpAddress - this parameter holds the module's IP address if DHCP is disabled. The address must be a valid unicast address. The address is formatted as a 32-bit hexadecimal number. Setting this parameter to zero invokes DHCP.

gsnSubnetAddress - this parameter this parameter holds the subnet address for the WLAN interface. The mask is formatted as a 32-bit hexadecimal number.

gsnGatewayIpAddress - this parameter holds the subnet gateway IP address. The address is formatted as a 32-bit hexadecimal number.

gsnPerformDhcp - this parameter sets the IP address mode as follows:

0x00000000 - for DHCP
0x00000001 - for static IP address

gsnMacAddress - this parameter holds the module's unique MAC address. This parameter is read-only.

gsnPrimaryDnsIpAddress - this parameter holds the IP address for the primary DNS server. The address is formatted as a 32-bit hexadecimal number.

gsnSecondaryDnsIpAddress - this parameter holds the IP address for the secondary DNS server. The address is formatted as a 32-bit hexadecimal number.

gsnCurntIpAddress - this parameter this parameter holds the current IP address assigned to the module. The address is formatted as a 32-bit hexadecimal number. This parameter is read-only.

gsnCurntSubnetAddress - this parameter this parameter hold the IP address of the current subnet address. The address is formatted as a 32-bit hexadecimal number. This parameter is read-only.

gsnCurntGatewayIpAddress - this parameter hold the IP address of the current subnet gateway address. The address is formatted as a 32-bit hexadecimal number. This parameter is read-only.

gsnCurntPrimaryDnsIpAddress - this parameter hold the IP address of the current primary DNS server. The address is formatted as a 32-bit hexadecimal number. This parameter is read-only.

gsnCurntSecondaryDnsIpAddress - this parameter hold the IP address of the current secondary DNS server. The address is formatted as a 32-bit hexadecimal number. This parameter is read-only.

gsnDHCPLeaseTime - this parameter holds the time remaining on the current DHCP lease. The parameter scaling is in microcontroller clock cycles of 0.029802322 µs. This parameter is a 64-bit number formatted as an ASCII string of the equivalent hexadecimal value. This parameter is read-only.

System Management Configuration Parameter (sysmgmtcfg) OIDs:
1.3.6.1.4.1.28295.1.1 + OID End

OID End	Name	R/W	Size, bytes	Range	Default
.4.3.0	gsnPrimarySNMPMgrIp	R/W	4	Class A,B,C	0x00000000 (0.0.0.0)
.4.4.0	gsnSecon-darySNMPMgrIp	R/W	4	Class A,B,C	0xC0A803C8 (192.168.3.200)
.4.5.1.2.1	gsnAp1Ssid	R/W	32	ASCII String	"WSN-Default"
.4.5.1.3.1	gsnAp1Channel	R/W	4	0..11	0x0000000B
.4.5.1.4.1	gsnap1wepkeyid	W	1	0..3	N/A
.4.5.1.5.1	gsnap1wepkeylen	W	1	5..13	N/A
.4.5.1.6.1	gsnap1wepkeyval	W	13	ASCII String	N/A
.4.5.1.7.1	gsnap1pskpassphrase	W	32	ASCII String	"WSN-PASSWORD"
.4.5.1.10.1	gsnap1authmode	R/W	4	1..8	0x00000003 (automatic authorization)
.4.5.1.11.1	gsnap1encmode	R/W	4	0..165	0x000000A5 (all)
.4.5.1.12.1	gsnap1pskkey	W	32	ASCII String	N/A
.4.5.1.2.2	gsnAp2Ssid	R/W	32	ASCII String	"WSN-Default"
.4.5.1.3.2	gsnAp2Channel	R/W	4	0..11	0x0000000B
.4.5.1.4.2	gsnap2wepkeyid	W	1	0..3	N/A

.4.5.1.5.2	gsnap2wepkeylen	W	1	5..13	N/A
.4.5.1.6.2	gsnap2wepkeyval	W	13	ASCII String	N/A
.4.5.1.7.2	gsnap2pskpassphrase	W	32	ASCII String	“WSN-PASSWORD”
.4.5.1.10.2	gsnap2authmode	R/W	4	1..8	0x00000003 (automatic authorization)
.4.5.1.11.2	gsnap2encmode	R/W	4	0..165	0x000000A5 (all)
.4.5.1.12.2	gsnap2pskkey	W	32	ASCII String	N/A
.4.5.1.2.3	gsnAp3Ssid	R/W	32	ASCII String	“WSN-Default”
.4.5.1.3.3	gsnAp3Channel	R/W	4	0..11	0x0000000B
.4.5.1.4.3	gsnap3wepkeyid	W	1	0..3	N/A
.4.5.1.5.3	gsnap3wepkeylen	W	1	5..13	N/A
.4.5.1.6.3	gsnap3wepkeyval	W	13	ASCII String	N/A
.4.5.1.7.3	gsnap3pskpassphrase	W	32	ASCII String	“WSN-PASSWORD”
.4.5.1.10.3	gsnap3authmode	R/W	4	1..8	0x00000003 (automatic authorization)
.4.5.1.11.3	gsnap3encmode	R/W	4	0..165	0x000000A5 (all)
.4.5.1.12.3	gsnap3pskkey	W	32	ASCII String	N/A
.4.5.1.2.4	gsnAdHocSsid	R/W	32	ASCII String	“WSN-Default”
.4.5.1.3.4	gsnAdHocChannel	R/W	4	0..11	0x0000000B
.4.5.1.4.4	gsnAdHocwepkeyid	W	1	0..3	N/A
.4.5.1.5.4	gsnAdHocwepkeylen	W	1	5..13	N/A
.4.5.1.6.4	gsnAdHocwepkeyval	W	13	ASCII String	N/A
.4.5.1.7.4	gsnAdHocpskpassphrase	W	32	ASCII String	“WSN-PASSWORD”
.4.5.1.10.4	gsnAdHocauthmode	R/W	4	1..8	0x00000003 (automatic authorization)
.4.5.1.11.4	gsnAdHocencmode	R/W	4	0..165	0x000000A5 (all)
.4.5.1.12.4	gsnAdHocpskkey	W	32	ASCII String	N/A
.4.6.0	gsnConfigComplete	R/W	4	$1..2^{32}$ -1	0x00000001
.4.10.0	gsnGetCommString	R/W	15	ASCII String	“GSN_GET”
.4.11.0	gsnSetCommString	R/W	15	ASCII String	“GSN_SET”
.4.12.0	gsnTrapCommString	R/W	15	ASCII String	“GSN_TRAP”
.4.13.0	gsnTrapConfigIntTmr	R/W	8	ASCII String	“0000000028000000” (20 seconds)
.4.14.0	gsnTrapLinkUpIntTmr	R/W	8	ASCII String	“0000000014000000” (10 seconds)
.4.15.0	gsnSnmpTrapSrcPort	R/W	4	$1..2^{16}$ -1	0x000000A2 (162)
.4.16.0	gsnSnmpTrapDstPort	R/W	4	$1..2^{16}$ -1	0x000000A1 (161)
.4.17.0	gsnSnmpPsPollTimer	R/W	8	ASCII String	“0000000078000000” (60 s)

Table 7.2.4

gsnPrimarySNMPMgrIp - this parameter holds the IP address of the primary SNMP manager. The address is formatted as a 32-bit hexadecimal number.

gsnSecondarySNMPMgrIp - this parameter holds the IP address of the secondary SNMP manager. The address is formatted as a 32-bit hexadecimal number.

gsnAp1Ssid - this parameter holds the SSID for access point 1.

gsnAp1Channel - this parameter sets the channel of operation for access point 1.

gsnap1wepkeyid - this parameter sets the WEP key ID for access point 1. The range of this parameter is 0x00000000 to 0x00000003. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnap1wepkeylen - this parameter sets the WEP key length in bytes for access point 1. The range of this parameter is 0x00000005 to 0x0000000D (5 to 13). This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnap1wepkeyval - this parameter holds the WEP key ASCII string for access point 1, 5 to 13 bytes. The number of bytes must match the WEP key length. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnap1pskpassphrase - this parameter holds the PSK passphrase for access point 1. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnap1authmode - this parameter sets the authentication mode for access point 1 as follows:

0x00000001 - open authentication, allows connection to a WLAN with any type of authentication
0x00000002 - shared authentication, allows connection to a WLAN only with WEP shared authentication; must be used to connect to a WEP shared WLAN
0x00000003 - automatic authentication, allows connection to a WLAN with any type of authentication
0x00000004 - WPA authentication, allows connection to a WLAN with WPA/WPA2 802.1x authentication; WPA2 is selected over WPA if both are present
0x00000005 - WPAPSK authentication, allows connection to a WLAN with WPA/WPA2 authentication; WPA2 is selected over WPA if both are present
0x00000007 - WPA2 authentication, allows connection to a WLAN with WPA2 802.1x authentication
0x00000008 - WPA2PSK authentication, allows connection to a WLAN with WPA2 PSK authentication

gsnap1encmode - this parameter selects the encryption mode for access point 1 as follows:

0x00000001 - allows connection to a WLAN only with WEP encryption
0x00000004 - allows connection to a WLAN only with TKIP encryption, either unicast or group; it connects if either unicast cipher or group cipher is TKIP
0x00000020 - allows connection to a WLAN only with CCMP encryption, either unicast or group; it connects if either unicast cipher or group cipher is CCMP
0x00000080 - allows connection to a WLAN only with no encryption
0x000000A5 - allows connection to a WLAN with any of the above modes

Note that encryption mode 0x000000A5 is derived from logically ORing the values of the four encryption modes above it. Any combination of ORed encryption modes are allowed.

gsnap1pskkey - this parameter this parameter holds the PSK key for access point 1. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnAp1Ssid - this parameter holds the SSID for access point 1.

gsnAp1Channel - this parameter sets the channel of operation for access point 1.

gsnap1wepkeyid - this parameter sets the WEP key ID for access point 1. The range of this parameter is 0x00000000 to 0x00000003. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnap1wepkeylen - this parameter sets the WEP key length in bytes for access point 1. The range of this parameter is 0x00000005 to 0x0000000D (5 to 13). This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnap1wepkeyval - this parameter holds the WEP key ASCII string for access point 1, 5 to 13 bytes. The number of bytes must match the WEP key length. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnap1pskpassphrase - this parameter holds the PSK passphrase for access point 1. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnap1authmode - this parameter sets the authentication mode for access point 1 as follows:

0x00000001 - open authentication, allows connection to a WLAN with any type of authentication
0x00000002 - shared authentication, allows connection to a WLAN only with WEP shared authentication; must be used to connect to a WEP shared WLAN
0x00000003 - automatic authentication, allows connection to a WLAN with any type of authentication
0x00000004 - WPA authentication, allows connection to a WLAN with WPA/WPA2 802.1x authentication; WPA2 is selected over WPA if both are present
0x00000005 - WPAPSK authentication, allows connection to a WLAN with WPA/WPA2 authentication; WPA2 is selected over WPA if both are present
0x00000007 - WPA2 authentication, allows connection to a WLAN with WPA2 802.1x authentication
0x00000008 - WPA2PSK authentication, allows connection to a WLAN with WPA2 PSK authentication

gsnap1encmode - this parameter selects the encryption mode for access point 1 as follows:

0x00000001 - allows connection to a WLAN only with WEP encryption
0x00000004 - allows connection to a WLAN only with TKIP encryption, either unicast or group; it connects if either unicast cipher or group cipher is TKIP
0x00000020 - allows connection to a WLAN only with CCMP encryption, either unicast or group; it connects if either unicast cipher or group cipher is CCMP
0x00000080 - allows connection to a WLAN only with no encryption
0x000000A5 - allows connection to a WLAN with any of the above modes

Note that encryption mode 0x000000A5 is derived from logically ORing the values of the four encryption modes above it. Any combination of ORed encryption modes are allowed.

gsnap1pskkey - this parameter this parameter holds the PSK key for access point 1. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnAp2Ssid - this parameter holds the SSID for access point 2.

gsnAp2Channel - this parameter sets the channel of operation for access point 2.

gsnap2wepkeyid - this parameter sets the WEP key ID for access point 2. The range of this parameter is 0x00000000 to 0x00000003. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnap2wepkeylen - this parameter sets the WEP key length in bytes for access point 2. The range of this parameter is 0x00000005 to 0x0000000D (5 to 13). This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnap2wepkeyval - this parameter holds the WEP key ASCII string for access point 2, 5 to 13 bytes. The number of bytes must match the WEP key length. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnap2pskpassphrase - this parameter holds the PSK passphrase for access point 2. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnap2authmode - this parameter sets the authentication mode for access point 2 as follows:

0x00000001 - open authentication, allows connection to a WLAN with any type of authentication
0x00000002 - shared authentication, allows connection to a WLAN only with WEP shared authentication; must be used to connect to a WEP shared WLAN
0x00000003 - automatic authentication, allows connection to a WLAN with any type of authentication
0x00000004 - WPA authentication, allows connection to a WLAN with WPA/WPA2 802.1x authentication; WPA2 is selected over WPA if both are present
0x00000005 - WPAPSK authentication, allows connection to a WLAN with WPA/WPA2 authentication; WPA2 is selected over WPA if both are present
0x00000007 - WPA2 authentication, allows connection to a WLAN with WPA2 802.1x authentication
0x00000008 - WPA2PSK authentication, allows connection to a WLAN with WPA2 PSK authentication

gsnap2encmode - this parameter selects the encryption mode for access point 2 as follows:

0x00000001 - allows connection to a WLAN only with WEP encryption
0x00000004 - allows connection to a WLAN only with TKIP encryption, either unicast or group; it connects if either unicast cipher or group cipher is TKIP
0x00000020 - allows connection to a WLAN only with CCMP encryption, either unicast or group; it connects if either unicast cipher or group cipher is CCMP
0x00000080 - allows connection to a WLAN only with no encryption
0x000000A5 - allows connection to a WLAN with any of the above modes

Note that encryption mode 0x000000A5 is derived from logically ORing the values of the four encryption modes above it. Any combination of ORed encryption modes are allowed.

gsnap2pskkey - this parameter this parameter holds the PSK key for access point 2. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnAp3Ssid - this parameter holds the SSID for access point 3.

gsnAp3Channel - this parameter sets the channel of operation for access point 3.

gsnap3wepkeyid - this parameter sets the WEP key ID for access point 3. The range of this parameter is 0x00000000 to 0x00000003. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnap3wepkeylen - this parameter sets the WEP key length in bytes for access point 3. The range of this parameter is 0x00000005 to 0x0000000D (5 to 13). This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnap3wepkeyval - this parameter holds the WEP key ASCII string for access point 3, 5 to 13 bytes. The number of bytes must match the WEP key length. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnap3pskpassphrase - this parameter holds the PSK passphrase for access point 3. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnap3authmode - this parameter sets the authentication mode for access point 3 as follows:

0x00000001 - open authentication, allows connection to a WLAN with any type of authentication
0x00000002 - shared authentication, allows connection to a WLAN only with WEP shared authentication; must be used to connect to a WEP shared WLAN
0x00000003 - automatic authentication, allows connection to a WLAN with any type of authentication
0x00000004 - WPA authentication, allows connection to a WLAN with WPA/WPA2 802.1x authentication; WPA2 is selected over WPA if both are present
0x00000005 - WPAPSK authentication, allows connection to a WLAN with WPA/WPA2 authentication; WPA2 is selected over WPA if both are present
0x00000007 - WPA2 authentication, allows connection to a WLAN with WPA2 802.1x authentication
0x00000008 - WPA2PSK authentication, allows connection to a WLAN with WPA2 PSK authentication

gsnap3encmode - this parameter selects the encryption mode for access point 3 as follows:

0x00000001 - allows connection to a WLAN only with WEP encryption
0x00000004 - allows connection to a WLAN only with TKIP encryption, either unicast or group; it connects if either unicast cipher or group cipher is TKIP
0x00000020 - allows connection to a WLAN only with CCMP encryption, either unicast or group; it connects if either unicast cipher or group cipher is CCMP
0x00000080 - allows connection to a WLAN only with no encryption
0x000000A5 - allows connection to a WLAN with any of the above modes

Note that encryption mode 0x000000A5 is derived from logically ORing the values of the four encryption modes above it. Any combination of ORed encryption modes are allowed.

gsnap3pskkey - this parameter this parameter holds the PSK key for access point 3. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnAdHocSsid - this parameter is the SSID for a fallback Ad Hoc server. Setting this parameter to null bytes disables Ad Hoc fallback.

gsnAdHocChannel - this parameter sets the channel for Ad Hoc operation.

gsnapAdHocwepkeyid - this parameter sets the WEP key ID for Ad Hoc operation. The range of this parameter is 0x00000000 to 0x00000003. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnapAdHocwepkeylen - this parameter sets the WEP key length in bytes for Ad Hoc operation. The range of this parameter is 0x00000005 to 0x0000000D (5 to 13). This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnapAdHocwepkeyval - this parameter holds the WEP key ASCII string for Ad Hoc operation, 5 to 13 bytes. The number of bytes must match the gsn*AdHocwepkeylen* value. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

gsnAdHocpskpassphrase - this parameter holds the PSK passphrase for Ad Hoc operation. This parameter is read-only.

gsnapAdHocauthmode - this parameter sets the authentication mode for Ad Hoc operation as follows:

0x00000001 - open authentication, allows connection to a WLAN with any type of authentication
0x00000002 - shared authentication, allows connection to a WLAN only with WEP shared authentication; must be used to connect to a WEP shared WLAN
0x00000003 - automatic authentication, allows connection to a WLAN with any type of authentication
0x00000004 - WPA authentication, allows connection to a WLAN with WPA/WPA2 802.1x authentication; WPA2 is selected over WPA if both are present
0x00000005 - WPAPSK authentication, allows connection to a WLAN with WPA/WPA2 authentication; WPA2 is selected over WPA if both are present
0x00000007 - WPA2 authentication, allows connection to a WLAN with WPA2 802.1x authentication
0x00000008 - WPA2PSK authentication, allows connection to a WLAN with WPA2 PSK authentication

gsnapAdHocencmode - this parameter selects the Ad Hoc encryption mode as follows:

0x00000001 - allows connection to a WLAN only with WEP encryption
0x00000004 - allows connection to a WLAN only with TKIP encryption, either unicast or group; it connects if either unicast cipher or group cipher is TKIP
0x00000020 - allows connection to a WLAN only with CCMP encryption, either unicast or group; it connects if either unicast cipher or group cipher is CCMP
0x00000080 - allows connection to a WLAN only with no encryption
0x000000A5 - allows connection to a WLAN with any of the above modes

Note that encryption mode 0x000000A5 is derived from logically ORing the values of the four encryption modes above it. Any combination of ORed encryption modes are allowed.

gsnAdHocpskkey - this parameter holds the PSK key for Ad Hoc operation. This parameter is read-only.

gsnConfigComplete - writing any non-zero value to this location in the referenced information table signals that configuration updates are complete.

gsnGetCommString - this parameter holds the get community ASCII string.

gsnSetCommString - this parameter holds the set community ASCII string.

gsnTrapCommString - this parameter holds the trap community ASCII string.

gsnTrapConfigIntTmr - this parameter sets the interval for the module to transmit configuration traps. The parameter scaling is in microcontroller clock cycles of 0.029802322 µs. This parameter is a 64-bit number, formatted as an ASCII string of the equivalent hexadecimal value.

gsnTrapLinkUpIntTmr - this parameter sets the interval for the module to transmit linkup traps. The parameter scaling is in microcontroller clock cycles of 0.029802322 µs. This parameter is a 64-bit number, formatted as an ASCII string of the equivalent hexadecimal value.

gsnSnmpTrapSrcPort - this parameter holds the port number for the SNMP trap source. The parameter is formatted as a 32-bit hexadecimal number.

gsnSnmpTrapDstPort - this parameter holds the port number for the SNMP trap destination. The parameter is formatted as a 32-bit hexadecimal number.

gsnSnmpPsPollTimer - this parameter sets the interval that the module polls the access point to send any data the access point is holding for it. Parameter scaling is in microcontroller clock cycles of 0.029802322 microseconds. This parameter is a 64-bit number formatted as an ASCII string of the equivalent hexadecimal value.

Trap Parameter (traps) OIDs:
1.3.6.1.4.1.28295.1.1 + OID End

OID End	Name	R/W	Size, bytes	Range	Default
.4.2.1	Linkup	N/A	N/A	N/A	N/A
.4.2.2	Config	N/A	N/A	N/A	N/A
.4.2.4	Time (sync update)	N/A	N/A	N/A	N/A
.4.2.6	Battery Low	N/A	N/A	N/A	N/A

Table 7.2.5

Linkup - trap to maintain link association.

Config - trap to request any pending configuration updates.

Time sync update - trap to request time sync update.

Battery Low - low battery warning trap.

Time Configuration Parameter (timesynccfg) OIDs:
1.3.6.1.4.1.28295.1.1 + OID End

OID End	Name	R/W	Size, bytes	Range	Default
.5.5.0	gsnTimeSyncTmr	R/W	8	ASCII String	“00000000” (disabled)
.5.6.0	gsnTimeSyncSntpSrvrIp	R/W	4	$0..2^{32}$ -1	0xC0A803C8 (192.168.3.200)
.5.7.0	gsnTimeSyncSntpSrvr-TimeOut	R/W	4	$0..2^{32}$ -1	0x00000003 (3 seconds)

Table 7.2.6

gsnTimeSyncTmr - this parameter sets the interval that the module polls the time sync server. Parameter scaling is in microcontroller clock cycles of 0.029802322 µs. This parameter is a 64-bit number formatted as an ASCII string of the equivalent hexadecimal value. This parameter defaults to zero, disabling the

function. The time base within a WSN802G module is crystal controlled, so to conserve battery power, a sync update once a day or less is generally sufficient.

gsnTimeSyncSntpSrvrIp - this parameter holds the IP address of the time sync server. Note that if the IP address held in this parameter is not a valid IP address, the module will be forced to stay active for the full time set by the gsnTimeSyncSntpSrvrTimeOut parameter discussed below, which will consume more battery power than necessary. PCs running Microsoft Windows XP, Vista or Windows 7 include a built-in time sync server function.

gsnTimeSyncSntpSrvrTimeOut - this parameter sets the time out interval for a response from the time sync server. The default value is 3 seconds.

Firmware Update Parameter (timesynccfg) OIDs:
1.3.6.1.4.1.28295.1.1 + OID End

OID End	Name	R/W	Size, bytes	Range	Default
.6.1.0	gsnFwUpdateIp	R/W	4	$0..2^{32}$ -1	0xC0A803C8 (192.168.3.200)
.6.2.0	gsnFwUpdatePort	R/W	4	$0..2^{16}$ -1	0x000020A3 (8355)
.6.3.0	gsnFwUpgradeNeeded	R/W	4	$0..2^{32}$ -1	0x0000000A

Table 7.2.7

gsnFwUpdateIp - this parameter holds the IP address of the firmware update server.

gsnFwUpdatePort - this parameter hold the port number for the firmware update downloads

gsnFwUpgradeNeeded - setting this parameter to 0x0000000A signals the module that a firmware update is available and needed.

The following OIDs have left-side fields of **1.3.6.1.4.1.32345.88**. The remaining three right-side OID fields for each parameter are shown in the left column or each table. Note that the last two non-zero right-side OID fields match the bank and location for the serial API CFG parameters (same MIB data base).

General Module Configuration Parameter OIDs:
1.3.6.1.4.1.32345.88 + OID End

OID End	Name	R/W	Size, bytes	Range	Default
.1.1.0	SensorName	R/W	128	ASCII String	“WSN Sensor”
.1.2.0	AutoReportInterval	R/W	8	ASCII String	“000000000A000000” (5 s)
.1.3.0	SensorServerIP	R/W	4	Class A,B,C	0xC0A803C8 (192.168.3.200)
.1.4.0	SensorServerPort	R/W	4	$1..2^{16}$ -1	0x0000203F (8255)
.1.5.0	WakeOutPredelay	R/W	4	$0..2^{32}$ -1	0x0000000A (10 ms)
.1.6.0	WakeOutPostdelay	R/W	4	$0..2^{32}$ -1	0x0000000A (10 ms)
.1.7.0	WakeTimeout	R/W	4	$0..2^{32}$ -1	0x00000000 (0 ms)
.1.8.0	TxPower	R/W	4	0..7	0x00000000 (8 mW)
.1.9.0	HardwareRevision	R	N/A	ASCII String	0x312E302E30 (1.0.0)
.1.10.0	FirmwareRevision	R	N/A	ASCII String	0x322E302E31303236 (2.0.1026)
.1.11.0	FirmwareBuildDate	R	4	ASCII String	unique to each build date
.1.12.0	TxRetryLimit	R/W	4	0..15	0x00000004 (4 retries)
.1.13.0	NetworkMode	R/W	4	0..1	0x00000000 (only UDP currently supported)

Table 7.2.8

***SensorName* - this parameter is a user-assignable sensor module name, for example "Utility Room Temperature Sensor". The name can contain up to 128 bytes.**

AutoReportInterval - this parameter sets the interval at which the sensor will send periodic reports. The parameter scaling is in microcontroller clock cycles of 0.029802322 µs. This parameter is a 64-bit number formatted as an ASCII string of the equivalent hexadecimal value.

SensorServerIP - this parameter holds the IP address of the server for the module to send sensor data reports. The IP address is formatted as a 32-bit value.

SensorServerPort - this parameter holds the port number of the server for the module to send sensor data reports. The port number is formatted as a 32-bit value.

WakeOutPredelay - this parameter sets the duration in milliseconds the WAKE_OUT pin turns on to activate an external user circuit *prior* to the rest of the module waking up.

WakeOutPostdelay - this parameter sets the duration in milliseconds the WAKE_OUT turn on to activate an external user circuit *subsequent* to the rest the module waking up.

WakeTimeout - this parameter sets the duration of inactivity in milliseconds that triggers the module to go back to sleep after being activated.

TxPower - this parameter set the transmitter output power level. Changes to this parameter require a reboot to take effect. The parameter range is 0 to 7, with 0 the highest power setting.

HardwareRevision - this parameter holds the revision code of the module hardware. This parameter is read-only.

FirmwareRevision - this parameter holds the firmware revision code. This parameter is read-only.

FirmwareBuildDate - this parameter codes the build date and timestamp of the module firmware as an ASCII string.

TxRetryLimit - this parameter sets the retry limit for 802.11 transmissions.

NetworkMode - this parameter specifies the IP format for sending auto-reports. Set this parameter to 0x00 for UDP format or 0x01 for TCP format.

Module I/O Configuration Parameter OIDs
1.3.6.1.4.1.32345.88 + OID End

OID End	Name	R/W	Size, bytes	Range	Default
.2.1.0	GPIO_In	R	4	0.. 0x0000000F	0x0000000C
.2.2.0	GPIO_Out	R/W	4	0.. 0x0000000F	0x0000000C
.2.3.0	ADC_Values	R/W	4	0.. 0x03FF03FF	N/A
.2.4.0	BattRSSI_Values	R/W	4	0.. 0x03FF00FF	N/A
.2.5.0	PWM_Values	R/W	4	0.. 0x03FF03FF	0x00000000 (0 V)
.2.6.0	GPIO_Config	R/W	4	0.. 0x00888888	0x00884422
.2.7.0	GPIO_Set	W	4	0.. 0x0000000F	N/A
.2.8.0	GPIO_Clear	W	4	0.. 0x0000000F	N/A

Table 7.2.9

GPIO_In - this parameter maps the states of the GPIO pins configured as inputs.

GPIO_Out - this parameter sets the states of the GPIO pins configured as outputs.

ADC_ Values - this parameter holds the concatenation of the last readings of ADC1 and ADC0. The lower two bytes of this parameter hold the right justified 10-bit ADC0 reading. The upper two bytes of this parameter hold the right justified 10-bit ADC1 reading. The module's ADC_REF output (Pin 25) provides a 1.8 V ADC full-scale reference voltage to support ratiometric sensor measurements.

BattRSSI_Values - this parameter holds the concatenation of the current module input voltage and the RSSI value of the last received 802.11 packet. The lower two bytes of this parameter hold the right justified 8-bit RSSI reading. The upper two bytes of this parameter hold the 16-bit voltage value in millivolts, such that a value of 0x0E4C, 3660 decimal, corresponds to 3.660 volts.

PWM_Values - this parameter holds the concatenation of the PWM output values. The lower two bytes of this parameter hold the 16-bit PWM0 setting. The upper two bytes of this parameter hold the 16-bit PWM1 setting. Full scale PWM outputs equal the module input voltage.

GPIO_Config - this parameter sets the GPIO direction, the pullup/pulldown configuration of each GPI configured as input, and the alternate GPIO functions. The parameter consists of a four 4-bit fields, with each GPIO, 0 through 3, having a 4-bit field to control its configuration. A 0x0 field sets a GPIO as an input, 0x2 field sets a GPIO as an input with internal pulldown, 0x3 sets a GPIO as an input with an internal pullup, 0x4 selects output, and a 0x8 value specifies an alternate function where defined.

GPIO_Set - writing to this parameter location sets GPIO output values. Setting a '1' in a bit position *sets* the corresponding GPIO output to a logic high state. Only bits corresponding to GPIOs configured as outputs are effected. The four bit positions in this parameter are right registered, with GPIO0 in the right-most bit position. This parameter is write-only.

GPIO_Clear - writing to this parameter location clears GPIO output values. Setting a '1' in a bit position *clears* the corresponding GPIO output to a logic low state. Only bits corresponding to GPIOs set as outputs have any effect. The four bit positions in this parameter are right registered, with GPIO0 in the right-most bit position. This parameter is write-only.

Module Serial Configuration Parameter OIDs:
1.3.6.1.4.1.32345.88 + OID End

OID End	Name	R/W	Size, bytes	Range	Default
.3.1.0	SerialDivisor	R/W	4	$0..2^{32}$ -1	0x00000030 (9600 bps)
.3.2.0	SerialCharFormat	R/W	4	0..3	0x00000003 (8-bit)
.3.3.0	SerialStopBits	R/W	4	0..1	0x00000000 (1 stop bit)
.3.4.0	SerialParity	R/W	4	0..3	0x00000004 (no parity)
.3.5.0	SerialRxTimeout	R/W	4	$0..2^{32}$ -1	0x00000020 (32 ms)
.3.6.0	SerialFlowControl	R/W	4	0..1	0x00000000 (flow control disabled)
.3.7.0	DiagDivisior	R/W	4	$0..2^{32}$ -1	0x00000030 (9600 bps)
.3.8.0	DiagEnable	R/W	4	0..1	0x00000001 (diagnostic port enabled)
.3.9.0	SPI_Mode	R/W	4	0..2	0x00000000 (disabled)
.3.10.0	SPI_MasterClock-Divisor	R/W	4	$0..2^{32}$ -1	0x00000001
.3.11.0	SPI_MasterCmd-String	R/W	4	ASCII String	"" (null string)

Table 7.2.10

SerialDivisor - this parameter sets the main serial port baud rate, equal to 460800 divided by the SerialDivisor value.

SerialCharFormat - this parameter sets the format for the main serial port as follows:

0x00000000 for 5-bit format
0x00000001 for 6-bit format
0x00000002 for 7-bit format
0x00000003 for 8-bit format (default)

SerialStopBits - this parameter sets the number of stop bits for the main serial port as follows:

0x00000000 for 1 stop bit (default)
0x00000001 for 2 stop bits

SerialParity - this parameter sets the parity configuration for the main serial port as follows:

0x00000000 for odd parity
0x00000001 for even parity
0x00000002 for mark parity
0x00000003 for space parity
0x00000004 for no parity (default)

SerialRxTimeout - this parameter sets the received message timeout for the main serial port. A received message is interpreted as complete when no additional bytes are received during a timeout interval. The SerialRxTimeout parameter is scaled in milliseconds.

SerialFlowControl - this parameter enables/disables /HOST_RTS - /HOST_CTS hardware flow control:

0x00000000 disables flow control (default)
0x00000001 for enables flow control

DiagDivisior - this parameter sets this parameter sets the diagnostic serial port baud rate, equal to 460800 divided by the DiagDivisor value.

DiagEnable - this parameter enables/disables diagnostic port operation:

0x00000000 disables diagnostic port operation
0x00000001 enables diagnostic port operation (default)

SPI_Mode - this parameter sets the SPI port mode:

0x00000000 to disable SPI port (default)
0x00000002 to enable SPI master mode

SPI_MasterClockDivisor - this parameter sets SPI master mode bit rate, equal to 460800 divided by the SPI_MasterClockDivisor value.

SPI_MasterCmdString - this parameter holds the command string to clock into the peripheral SPI slave when the module is acting as an SPI master.

Module WLAN Configuration Parameter OIDs:
1.3.6.1.4.1.32345.88 + OID End

OID End	Name	R/W	Size, bytes	Range	Default
.4.1.0	Ap1_Ssid	R/W	32	ASCII String	"WSN-Default"
.4.2.0	Ap1_Channel	R/W	4	0..11	0x0000000B
.4.3.0	Ap1_AuthMode	R/W	4	1..8	0x00000003 (automatic authentication)
.4.4.0	Ap1_EncryptionMode	R/W	4	1..165	0x000000A5 (all)
.4.5.0	Ap1_PskPassphrase	W	32	ASCII String	"WSN-PASSWORD"
.4.6.0	Ap1_PskKey	W	32	ASCII String	N/A
.4.7.0	Ap1_WepKeyId	W	1	0..3	N/A
.4.8.0	Ap1_WepKeyLength	W	1	5..13	N/A
.4.9.0	Ap1_WepKeyValue	W	13	ASCII String	N/A
.4.10.0	Ap2_Ssid	R/W	32	ASCII String	"WSN-Default"
.4.11.0	Ap2_Channel	R/W	4	0..11	0x0000000B
.4.12.0	Ap2_AuthMode	R/W	4	1..8	0x00000003 (automatic authentication)
.4.13.0	Ap2_EncryptionMode	R/W	4	0..165	0x000000A5 (all)
.4.14.0	Ap2_PskPassphrase	W	32	ASCII String	"WSN-PASSWORD"
.4.15.0	Ap2_PskKey	W	32	ASCII String	N/A
.4.16.0	Ap2_WepKeyId	W	1	0..3	N/A
.4.17.0	Ap2_WepKeyLength	W	1	5..13	N/A
.4.18.0	Ap2_WepKeyValue	W	13	ASCII String	N/A
.4.19.0	Ap3_Ssid	R/W	32	ASCII String	"WSN-Default"
.4.20.0	Ap3_Channel	R/W	4	0..11	0x0000000B
.4.21.0	Ap3_AuthMode	R/W	4	1..8	0x00000003 (automatic authentication)
.4.22.0	Ap3_EncryptionMode	R/W	4	0..165	0x000000A5 (all)
.4.23.0	Ap3_PskPassphrase	W	32	ASCII String	"WSN-PASSWORD"
.4.24.0	Ap3_PskKey	W	32	ASCII String	N/A
.4.25.0	Ap3_WepKeyId	W	1	0..3	N/A
.4.26.0	Ap3_WepKeyLength	W	1	5..13	N/A
.4.27.0	Ap3_WepKeyValue	W	13	ASCII String	N/A
.4.28.0	AdHoc_Ssid	R/W	32	ASCII String	"RFM-MAC" (MAC = module's MAC addr.)
.4.29.0	AdHoc_Channel	R/W	4	0..11	0x0000000B
.4.30.0	AdHoc_AuthMode	R/W	4	1..8	0x00000001
.4.31.0	AdHoc_Encryption-Mode	R/W	4	0..165	0x00000080 (none)
.4.32.0	AdHoc_Psk-Passphrase	R	32	ASCII String	"" (null bytes)
.4.33.0	AdHoc_PskKey	R	32	ASCII String	"" (null bytes)
.4.34.0	AdHoc_WepKeyId	W	1	0..3	N/A
.4.35.0	AdHoc_WepKey-Length	W	1	5..13	N/A
.4.36.0	AdHoc_WepKeyValue	W	13	ASCII String	N/A

Table 7.2.11

Ap1_Ssid - this parameter holds the SSID of preferred access point 1.

Ap1_Channel - this parameter sets the channel of operation for preferred access point 1.

Ap1_AuthMode - this parameter sets the authentication mode for preferred access point 1. as follows:

0x00000001 - open authentication, allows connection to a WLAN with any type of authentication
0x00000002 - shared authentication, allows connection to a WLAN only with WEP shared authentication; must be used to connect to a WEP shared WLAN
0x00000003 - automatic authentication, allows connection to a WLAN with any type of authentication
0x00000004 - WPA authentication, allows connection to a WLAN with WPA/WPA2 802.1x authentication; WPA2 is selected over WPA if both are present
0x00000005 - WPAPSK authentication, allows connection to a WLAN with WPA/WPA2 authentication; WPA2 is selected over WPA if both are present
0x00000007 - WPA2 authentication, allows connection to a WLAN with WPA2 802.1x authentication
0x00000008 - WPA2PSK authentication, allows connection to a WLAN with WPA2 PSK authentication

Ap1_EncryptionMode - this parameter selects the encryption mode for preferred access point 1 as follows:

0x00000001 - allows connection to a WLAN only with WEP encryption
0x00000004 - allows connection to a WLAN only with TKIP encryption, either unicast or group; it connects if either unicast cipher or group cipher is TKIP
0x00000020 - allows connection to a WLAN only with CCMP encryption, either unicast or group; it connects if either unicast cipher or group cipher is CCMP
0x00000080 - allows connection to a WLAN only with no encryption
0x000000A5 - allows connection to a WLAN with any of the above modes

Note that encryption mode 0x000000A5 is derived from logically ORing the values of the four encryption modes above it. Any combination of ORed encryption modes are allowed.

Ap1_PskPassphrase - this parameter holds the PSK passphrase for preferred access point 1. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

Ap1_PskKey - this parameter holds the PSK key for preferred access point 1. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

Ap1_WepKeyId - this parameter sets the WEP key ID for preferred access point 1. The range of this parameter is 0x00000000 to 0x00000003. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

Ap1_WepKeyLength - this parameter sets the WEP key length in bytes for preferred access point 1. The range of this parameter is 0x00000005 to 0x0000000D (5 to 13). This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

Ap1_WepKeyValue - this parameter holds the WEP key ASCII string for access point 1, 5 to 13 bytes. The number of bytes must match the WEP key length. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

Ap2_Ssid - this parameter holds the SSID of preferred access point 2.

Ap2_Channel - this parameter sets the channel of operation for preferred access point 2.

Ap2_AuthMode - this parameter sets the authentication mode for preferred access point 2 as follows:

0x00000001 - open authentication, allows connection to a WLAN with any type of authentication
0x00000002 - shared authentication, allows connection to a WLAN only with WEP shared authentication; must be used to connect to a WEP shared WLAN
0x00000003 - automatic authentication, allows connection to a WLAN with any type of authentication
0x00000004 - WPA authentication, allows connection to a WLAN with WPA/WPA2 802.1x authentication; WPA2 is selected over WPA if both are present
0x00000005 - WPAPSK authentication, allows connection to a WLAN with WPA/WPA2 authentication; WPA2 is selected over WPA if both are present
0x00000007 - WPA2 authentication, allows connection to a WLAN with WPA2 802.1x authentication
0x00000008 - WPA2PSK authentication, allows connection to a WLAN with WPA2 PSK authentication

Ap2_EncryptionMode - this parameter selects the encryption mode for preferred access point 2 as follows:

0x00000001 - allows connection to a WLAN only with WEP encryption
0x00000004 - allows connection to a WLAN only with TKIP encryption, either unicast or group; it connects if either unicast cipher or group cipher is TKIP
0x00000020 - allows connection to a WLAN only with CCMP encryption, either unicast or group; it connects if either unicast cipher or group cipher is CCMP
0x00000080 - allows connection to a WLAN only with no encryption
0x000000A5 - allows connection to a WLAN with any of the above modes

Note that encryption mode 0x000000A5 is derived from logically ORing the values of the four encryption modes above it. Any combination of ORed encryption modes are allowed.

Ap2_PskPassphrase - this parameter holds the PSK passphrase for preferred access point 2. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

Ap2_PskKey - this parameter holds the PSK key for preferred access point 2. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

Ap2_WepKeyId - this parameter sets the WEP key ID for preferred access point 2. The range of this parameter is 0 to 3. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

Ap2_WepKeyLength - this parameter sets the WEP key length in bytes for preferred access point 2. The range of this parameter is 0x00000005 to 0x0000000D (5 to 13). This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

Ap2_WepKeyValue - this parameter holds the WEP key ASCII string for preferred access point 2, 5 to 13 bytes. The number of bytes must match the WEP key length. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

Ap3_Ssid - this parameter holds the SSID of preferred access point 3.

Ap3_Channel - this parameter sets the channel of operation for preferred access point 3.

Ap3_AuthMode - this parameter sets the authentication mode for preferred access point 3 as follows:

0x00000001 - open authentication, allows connection to a WLAN with any type of authentication
0x00000002 - shared authentication, allows connection to a WLAN only with WEP shared authentication; must be used to connect to a WEP shared WLAN
0x00000003 - automatic authentication, allows connection to a WLAN with any type of authentication
0x00000004 - WPA authentication, allows connection to a WLAN with WPA/WPA2 802.1x authentication; WPA2 is selected over WPA if both are present
0x00000005 - WPAPSK authentication, allows connection to a WLAN with WPA/WPA2 authentication; WPA2 is selected over WPA if both are present
0x00000007 - WPA2 authentication, allows connection to a WLAN with WPA2 802.1x authentication
0x00000008 - WPA2PSK authentication, allows connection to a WLAN with WPA2 PSK authentication

Ap3_EncryptionMode - this parameter selects the encryption mode for preferred access point 3 as follows:

0x00000001 - allows connection to a WLAN only with WEP encryption
0x00000004 - allows connection to a WLAN only with TKIP encryption, either unicast or group; it connects if either unicast cipher or group cipher is TKIP
0x00000020 - allows connection to a WLAN only with CCMP encryption, either unicast or group; it connects if either unicast cipher or group cipher is CCMP
0x00000080 - allows connection to a WLAN only with no encryption
0x000000A5 - allows connection to a WLAN with any of the above modes

Note that encryption mode 0x000000A5 is derived from logically ORing the values of the four encryption modes above it. Any combination of ORed encryption modes are allowed.

Ap3_PskPassphrase - this parameter holds the PSK passphrase for preferred access point 3. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

Ap3_PskKey - this parameter holds the PSK key for preferred access point 3. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

Ap3_WepKeyId - this parameter is holds WEP key ID for preferred access point 3. The range of this parameter is 0 to 3. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

Ap2_WepKeyLength - this parameter sets the WEP key length for preferred access point 3. The range of this parameter is 0x00000005 to 0x0000000D (5 to 13). This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

Ap2_WepKeyValue - this parameter holds the WEP key ASCII string for preferred access point 3, 5 to 13 bytes; the number of bytes must match the WEP key length. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

AdHoc_Ssid - this parameter is the SSID for a fallback Ad Hoc server. Setting this parameter to null bytes disables Ad Hoc fallback.

AdHoc_Channel - - this parameter sets the channel for Ad Hoc operation.

AdHoc_AuthMode - this parameter sets the authentication mode for Ad Hoc operation as follows:

0x00000001 - open authentication, allows connection to a WLAN with any type of authentication
0x00000002 - shared authentication, allows connection to a WLAN only with WEP shared authentication; must be used to connect to a WEP shared WLAN
0x00000003 - automatic authentication, allows connection to a WLAN with any type of authentication
0x00000004 - WPA authentication, allows connection to a WLAN with WPA/WPA2 802.1x authentication; WPA2 is selected over WPA if both are present
0x00000005 - WPAPSK authentication, allows connection to a WLAN with WPA/WPA2 authentication; WPA2 is selected over WPA if both are present
0x00000007 - WPA2 authentication, allows connection to a WLAN with WPA2 802.1x authentication
0x00000008 - WPA2PSK authentication, allows connection to a WLAN with WPA2 PSK authentication

AdHoc_EncryptionMode - this parameter selects the Ad Hoc encryption mode as follows:

0x00000001 - allows connection to a WLAN only with WEP encryption
0x00000004 - allows connection to a WLAN only with TKIP encryption, either unicast or group; it connects if either unicast cipher or group cipher is TKIP
0x00000020 - allows connection to a WLAN only with CCMP encryption, either unicast or group; it connects if either unicast cipher or group cipher is CCMP
0x00000080 - allows connection to a WLAN only with no encryption
0x000000A5 - allows connection to a WLAN with any of the above modes

Note that encryption mode 0x000000A5 is derived from logically ORing the values of the four encryption modes above it. Any combination of ORed encryption modes are allowed.

AdHoc_PskPassphrase - this parameter holds the PSK passphrase for Ad Hoc operation. This parameter is read-only.

AdHoc_PskKey - this parameter holds the PSK key for Ad Hoc operation. This parameter is read-only.

AdHoc_WepKeyId - this parameter is holds WEP key ID for Ad Hoc operation. The range of this parameter is 0 to 3. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

AdHoc_WepKeyLength - this parameter sets the WEP key length for Ad Hoc operation. The range of this parameter is 0x00000005 to 0x0000000D (5 to 13). This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

AdHoc_WepKeyValue - this parameter holds the WEP key ASCII string for Ad Hoc operation, 5 to 13 bytes. The number of bytes must match the *AdHoc_WepKeyLength* value. This parameter is write-only. Attempting to read a write-only SNMP parameter returns an error.

Module Node Options Parameter OIDs:
1.3.6.1.4.1.32345.88 + OID End

OID End	Name	R/W	Size, bytes	Range	Default
.5.1.0	PsPollTimer	R/W	8	ASCII String	“0000000078000000” (60 s)
.5.4.0	RestoreFactoryCfg	W	4	0..1	N/A
.5.6.0	RebootNode	W	4	0..1	N/A
.5.10.0	BatteryReadFreq	R/W	4	0..1023	0x00000001 (read on each TX)
.5.12.0	BatteryWarning-LevelInMVolt	R/W	4	0..1023	0x000008FC (2300 mV)
.5.13.0	BatteryStandby-LevelInMVolt	R/W	4	0..1023	0x000009F6 (2550 mV)
.5.14.0	BatteryRFirstBoot-StandbyLevel-InMVolt	R/W	4	0..1023	0x000008FC (2550 mV)
.5.16.0	DisableStdBy	W	4	0..2	N/A
.5.17.0	SystemTime	R/W	8	ASCII String	N/A

Table 7.2.12

PsPollTimer - this parameter sets the interval that the module polls the access point to send any data the access point is holding for it. Parameter scaling is in microcontroller clock cycles of 0.029802322 µs. This parameter is a 64-bit number formatted as an ASCII string of the equivalent hexadecimal value.

RestoreFactoryCfg - writing any non-zero value to this location will restore module parameters with factory defaults to their default values. This parameter is write-only.

RebootNode - writing any non-zero value to this location performs a “battery-plugged” reboot of the module. This parameter is write-only.

BatteryReadFreq - This parameter holds the number of transmissions from one battery reading to the next.

BatteryWarningLevelInMVolt - this parameter sets the battery voltage level that triggers a low battery warning trap message. The parameter scaling is in millivolts, formatted as a 4-byte hexadecimal number.

BatteryStandbyLevelInMVolt - this parameter sets the battery voltage level that triggers the node to switch to standby. The parameter scaling is in millivolts, formatted as a 4-byte hexadecimal number. WARNING: setting the value of this parameter too high can “lock up” the module.

BatteryRFirstBootStandbyLevelInMVolt - this parameter sets the battery voltage level that triggers the node to switch to standby immediately when booted up. The parameter scaling is in millivolts, formatted as a 4-byte hexadecimal number. WARNING: setting the value of this parameter too high can “lock up” the module.

DisableStdBy - writing a non-zero value to this location will disable the module standby function. This parameter is write-only.

SystemTime - this parameter holds the time interval since the module was last booted. The parameter scaling is in microcontroller clock cycles of 0.029802322 µs. This parameter is a 64-bit number formatted as an ASCII string of the equivalent hexadecimal value. The system time parameter rolls over every 128 seconds.

Module Scanning Authentication Parameter OIDs:
1.3.6.1.4.1.32345.88 + OID End

OID End	Name	R/W	Size, bytes	Range	Default
.6.1.0	ScanType	R/W	4	0..1	0x00000000 (active)
.6.6.0	EapOuterAuthType	R	4	$0..2^{32}$ -1	N/A
.6.7.0	EapInnerAuthType	R	4	$0..2^{32}$ -1	N/A
.6.8.0	RadiusUserName	W	15	ASCII String	N/A
.6.9.0	RadiusPasswd	W	15	ASCII String	N/A
.6.17.0	EaptlsProvision-CaCert	W	15	ASCII String	N/A
.6.18.0	EaptlsProvision-ClientCert	W	15	ASCII String	N/A
.6.19.0	EaptlsProvision-PvtKey	W	15	ASCII String	N/A

Table 7.2.13

ScanType - this parameter selects active or passive scan mode as follows:

0x00000000 for active scan mode
0x00000000 for passive scan mode

EapOuterAuthType - this parameter holds the outer authentication type used for EAP_FAST. It is formatted as a 32-bit hexadecimal value.

EapInnerAuthType - this parameter holds the inner authentication type used for EAP_FAST. It is formatted as a 32-bit hexadecimal value.

RadiusUserName - this parameter holds the ASCII user name for authenticating with the RADIUS server.

RadiusPasswd - this parameter hold the ASCII password for authenticating with the RADIUS server.

EaptlsProvisionCaCert - this parameter holds the ASCII EAP-TLS Provision Certificate Authority Certificate.

EaptlsProvisionClientCert - this parameter holds the ASCII EAP_TLS Provision Client Certificate.

EaptlsProvisionPvtKey - this parameter holds the Provision Private Key as an ASCII string.

Module Network Configuration Parameter OIDs:
1.3.6.1.4.1.32345.88 + OID End

OID End	Name	R/W	Size, bytes	Range	Default
.7.1.0	IpAddress	R/W	4	Class A,B,C	0x00000000 (0.0.0.0)
.7.2.0	SubnetMask	R/W	4	Class A,B,C	0x00000000 (0.0.0.0)
.7.3.0	GatewayIpAddress	R/W	4	Class A,B,C	0x00000000 (0.0.0.0)
.7.4.0	PerformDhcp	R/W	4	Class A,B,C	0x00000000 (enabled)
.7.5.0	MacAddress	R	4	OID 00:30:66	unique for each module
.7.7.0	PrimaryDns-IpAddress	R/W	4	Class A,B,C	0x00000000 (0.0.0.0)
.7.8.0	SecondaryDns-IpAddress	R/W	4	Class A,B,C	0x00000000 (0.0.0.0)
.7.9.0	CurntIpAddress	R	4	Class A,B,C	N/A
.7.10.0	CurntSubnetAddress	R	4	Class A,B,C	N/A
.7.11.0	CurntGateway-IpAddress	R	4	Class A,B,C	N/A
.7.12.0	CurntPrimary-DnsIpAddress	R	4	Class A,B,C	N/A
.7.13.0	CurntSecondary-DnsIpAddress	R	4	Class A,B,C	N/A
.7.14.0	DHCPLeaseTime	R	8	ASCII String	N/A

Table 7.2.14

IpAddress - this parameter holds the module's IP address if DHCP is disabled. The address must be a valid unicast address. The address is formatted as a 32-bit hexadecimal number. Setting this parameter to zero invokes DHCP.

SubnetMask - this parameter holds the subnet mask for the WLAN interface. The mask is formatted as a 32-bit hexadecimal number.

GatewayIpAddress - this parameter holds the subnet gateway IP address. The address is formatted as a 32-bit hexadecimal number.

PerformDhcp - this parameter sets the IP address mode as follows:

0x00000000 - for DHCP
0x00000001 - for static IP address

MacAddress - this parameter holds the module's unique MAC address. This parameter is read only.

PrimaryDnsIpAddress - this parameter holds the IP address for the primary DNS server. The address is formatted as a 32-bit hexadecimal number.

SecondaryDnsIpAddress - this parameter holds the IP address for the secondary DNS server. The address is formatted as a 32-bit hexadecimal number.

CurntIpAddress - this parameter holds the current IP address assigned to the module. The address is formatted as a 32-bit hexadecimal number.

CurntSubnetAddress - this parameter hold the IP address of the current subnet address. The address is formatted as a 32-bit hexadecimal number.

CurntGatewayIpAddress - this parameter hold the IP address of the current subnet gateway address. The address is formatted as a 32-bit hexadecimal number.

CurntPrimaryDnsIpAddress - this parameter hold the IP address of the current primary DNS server. The address is formatted as a 32-bit hexadecimal number.

CurntSecondaryDnsIpAddress - this parameter hold the IP address of the current secondary DNS server. The address is formatted as a 32-bit hexadecimal number.

DHCPLeaseTime - this parameter holds the time remaining on the current DHCP lease. The parameter scaling is in microcontroller clock cycles of 0.029802322 µs. This parameter is a 64-bit number formatted as an ASCII string of the equivalent hexadecimal value.

Module SNMP Configuration Parameter OIDs:
1.3.6.1.4.1.32345.88 + OID End

OID End	Name	R/W	Size, bytes	Range	Default
.8.3.0	PrimarySNMPMgrIp	R/W	4	Class A,B,C	0x00000000 (0.0.0.0)
.8.4.0	Secondary-SNMPMgrIp	R/W	4	Class A,B,C	0xC0A803C8 (192.168.3.200)
.8.10.0	GetCommString	R/W	15	ASCII String	“GSN_GET”
.8.11.0	SetCommString	R/W	15	ASCII String	“GSN_SET”
.8.12.0	TrapCommString	R/W	15	ASCII String	“GSN_TRAP”
.8.15.0	SnmpTrapSrcPort	R/W	4	$1..2^{16}$ -1	0x000000A2 (162)
.8.16.0	SnmpTrapDstPort	R/W	4	$1..2^{16}$ -1	0x000000A1 (161)

Table 7.2.15

PrimarySNMPMgrIp - this parameter holds the IP address of the primary SNMP manager. The address is formatted as a 32-bit hexadecimal number.

SecondarySNMPMgrIp - this parameter holds the IP address of the secondary SNMP manager. The address is formatted as a 32-bit hexadecimal number.

GetCommString - this parameter holds the get community ASCII string.

SetCommString - this parameter holds the set community ASCII string.

TrapCommString - this parameter holds the trap community ASCII string.

SnmpTrapSrcPort - this parameter holds the port number for the SNMP trap source. The parameter is formatted as a 32-bit hexadecimal number.

SnmpTrapDstPort - this parameter holds the port number for the SNMP trap destination. The parameter is formatted as a 32-bit hexadecimal number.

8.0 WSN802GDK/WSN802GADK and WSN802GDK-A/WSN802GADK-A Developer's Kit

Figure 8.1.1

8.1 Kit Contents

- One WSN802GP transceiver module installed in a developer board with one U.FL jumper cable or one WSN802GPA transceiver module installed in a developer board with integrated antenna
- One patch antenna and one dipole antenna with MMCX to SMA-R adaptor cable (not required for WSN802GPA transceiver module)
- One 9 V wall-plug power supply, 120/240 VAC, for developer board power, and one 9 V battery
- One RJ-45/DB-9F cable assembly, one RJ-11/DB-9F cable assembly and one A/B USB cable
- One pre-configured NETGEAR WGR614 Wireless-G Router with wall-plug power supply and Ethernet cable (WSN802GDK-A/WSN802GADK-A kits only)
- One WSN802GDK documentation and software CD

8.2 Additional Items Needed

- One PC with Microsoft Windows XP, Vista, or Windows 7 Operating System.

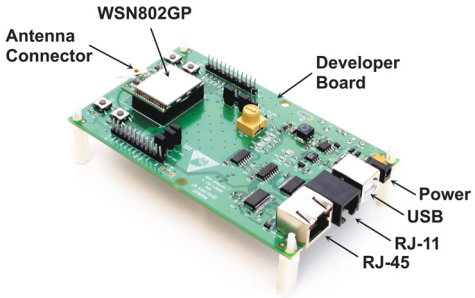


Figure 8.3.1

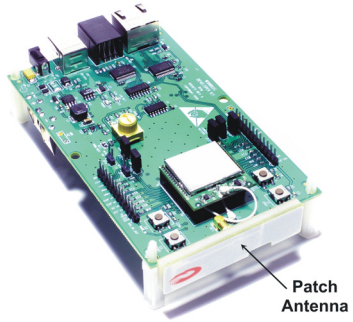


Figure 8.3.2

8.3 Developer Kit Assembly and Testing

Note: the NETGEAR WGR614 router has been preconfigured for use with the WSN802GDK-A and WSN802GADK-A developer's kits. *Do not reconfigure the router*. See the Quick Start Guide in the WSN802GDK kit for information on using a customer supplied router.

1. Observe ESD precautions when handling the WSN802GDK/WSN802GADK developer board. Referring to Figures 8.3.1 and 8.3.2, install the patch antenna on the WSN802GDK developer board antenna connector. The antenna "snaps" onto the connector with moderate pressure. The patch antenna is not required for the WSN802GADK developer board.

2. Install an AC plug on the 9 V developer board power supply. Plug the power supply cable into the developer board power connector as shown Figure 8.3.1, and plug the 9 V power supply into AC.

3. If using a PC with WiFi that supports WPA2 encryption, connect the NETGEAR wall-plug power supply cable to the NETGEAR router and plug the power supply into AC. No other connections to the router are required. Confirm the PC is configured for DHCP. Open the *Wireless Network Connection* dialog box on the PC. The NETGEAR router will be operating on channel 11 with an SSID of WSN-Default in secure mode. The PSK security passphrase to allow router access is WSN-PASSWORD. Establish a wireless connection to the router.

4. If using a PC without WiFi, or with WiFi that does not supports WPA2 encryption, connect the Ethernet cable between the PC and one of the *LAN* ports on the NETGEAR router. Confirm the PC is configured for DHCP. Connect the NETGEAR wall-plug power supply cable to the NETGEAR router and plug the power supply in.

5. Copy WSNConfig.exe and WSNApp.exe from the *Programs* folder on the kit CD to a convenient folder on the PC. These programs run using ordinary Window's resources and do not require any framework installations, registry entries, etc., to run.

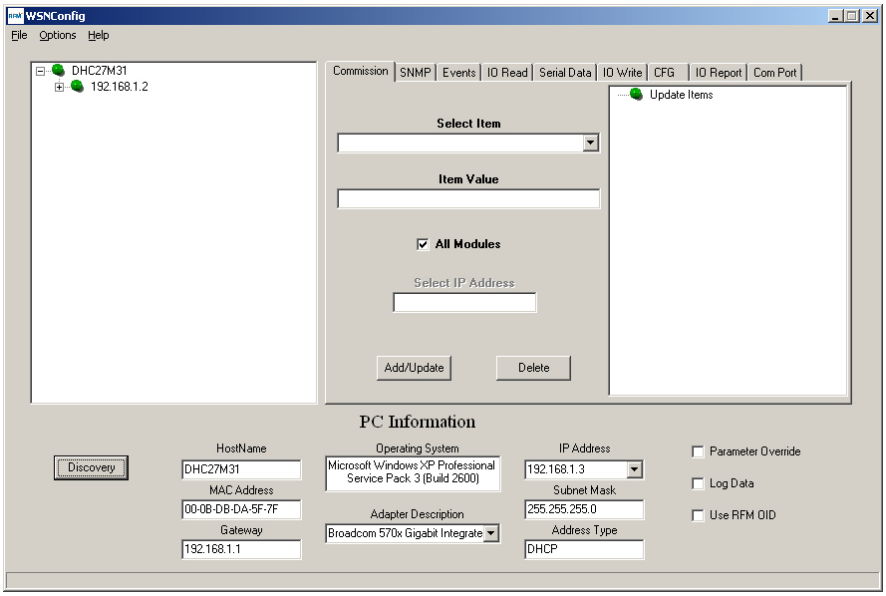


Figure 8.3.3

6. Start the WSNConfig.exe program. Click on the *Discovery* button. In a few moments the IP address of the WSN802G module will be displayed near the top of the left hand text box on the WSNConfig window, as shown in Figure 8.3.3. If the module IP address does not appear, see Section 8.4 below.

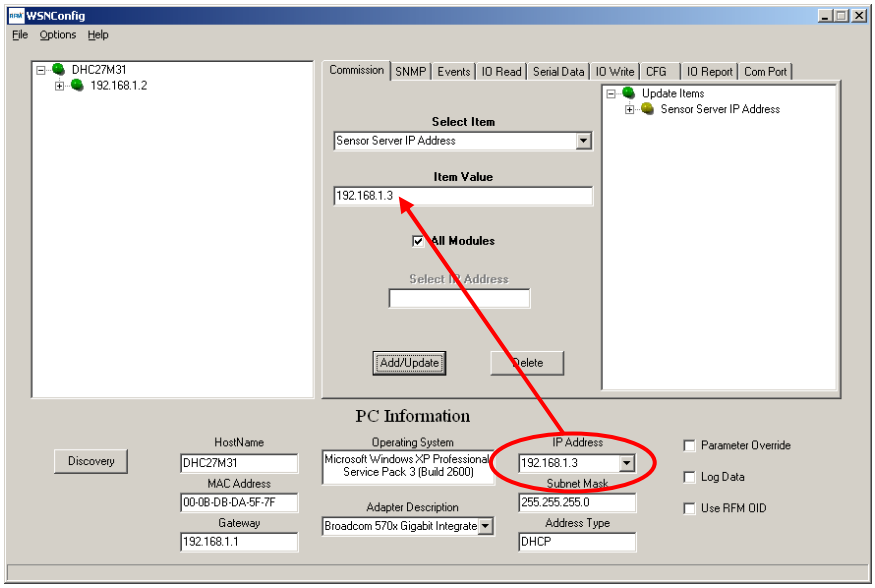


Figure 8.3.4

7. Select *Sensor Server IP Address* from the *Select Item* drop-down box on the WSNConfig.exe *Commission* tab. Enter the IP Address from the *PC Information* area in the *Item Value* text box. Then click the *Add/Update* button. See Figure 8.3.4. This action configures the WSN802G module to send its periodic I/O_READ data to the PC running WSNConfig.exe.

8. Click on the *IO Report* Tab. Data will automatically begin filling the chart at a 10 s update interval. Blowing warm air (breath) on thermistor RT1 on the developer board can be observed on the chart, as shown in Figure 8.3.5. The developer kit is now ready for use.

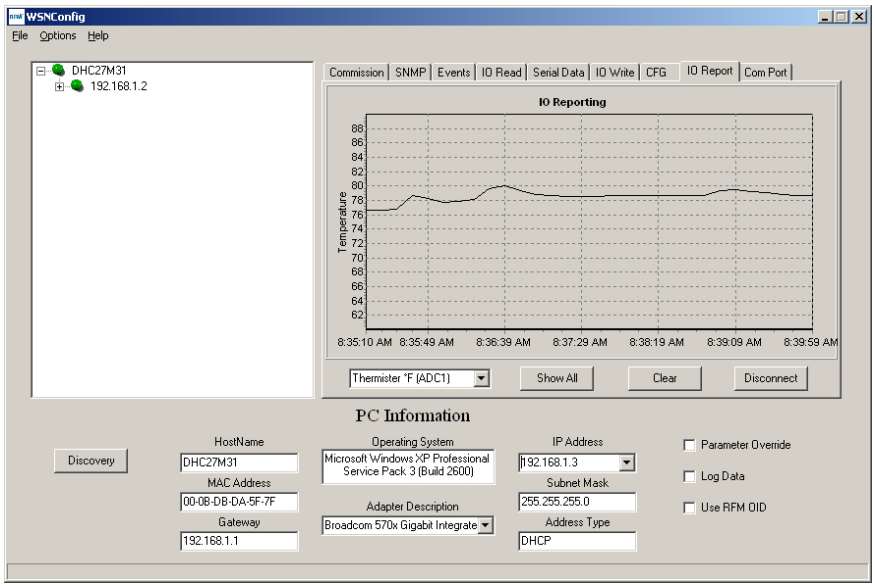


Figure 8.3.5

9. If any difficulty is encountered in setting up your kit, contact RFM's module technical support group. See Section 10.2 for contact details.

8.4 Host Configurations to Support Module Discovery

WSN802G module IP address discovery is discussed in Section 8.3, step 6. If the modules IP address does not appear in a few seconds after clicking the *Discovery* button the first time, try clicking the button several more times at intervals of about 5 seconds. If the IP address still does not appear, open your access point web browser configuration utility and check to see if the WSN802G module is associated with the access point. If the WSN802G module is not associated with the access point, check for an RF issue such as a loose antenna connection. If the WSN802G module is associated with the access point, there is something blocking module IP address discovery in the host PC. For example, if the host PC is blocking UDP broadcast messages, module IP address discovery cannot work. Some firewalls and other low-level security applications block UDP broadcast messages by default. There are two options to solving this problem. First, it may be possible to configure the security application to process UDP broadcast messages. If this is not possible or is considered to create a security problem, the host PC can be configured with the default IP address the module will recognize. This address is 192.168.3.200, using subnet mask 255.255.255.0.

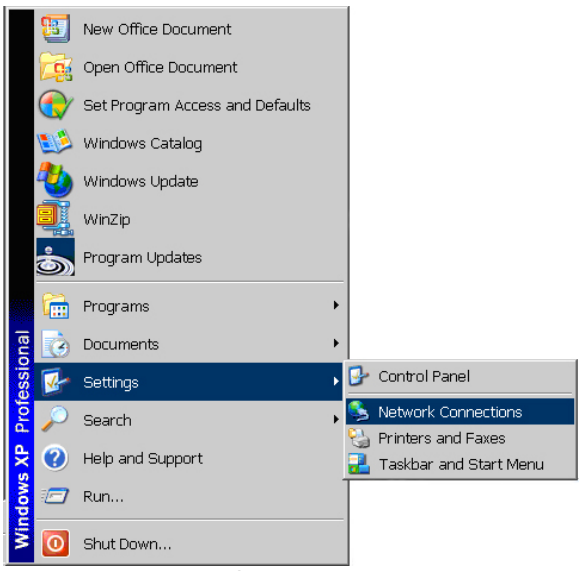


Figure 8.4.1

To configure the PC with this static address, click on the PC's *Start* button and select *Network Connections* from the *Settings* menu, as shown in Figure 8.4.1.

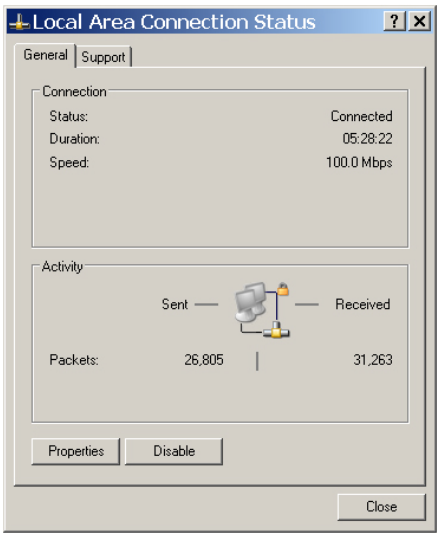


Figure 8.4.2

Double-click on the *LAN or High-speed Internet* connection being used to display its related *Local Area Connection Status* dialog box, as shown in Figure 8.4.2. Then click on *Properties* to display the *Local Area Connection Properties* dialog box, as shown in Figure 8.4.3.

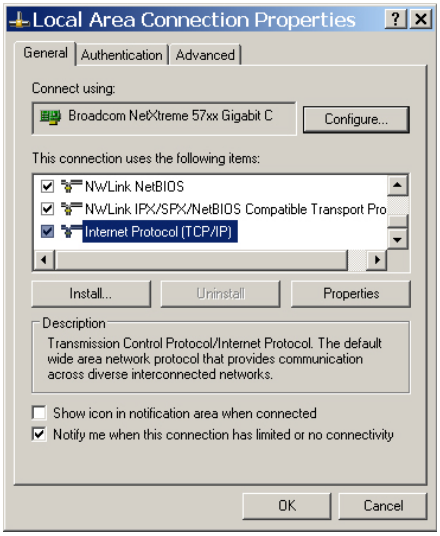


Figure 8.4.3

Select the *Internet Protocol (TCP/IP)* item as shown in Figure 8.4.3 and click on *Properties*.

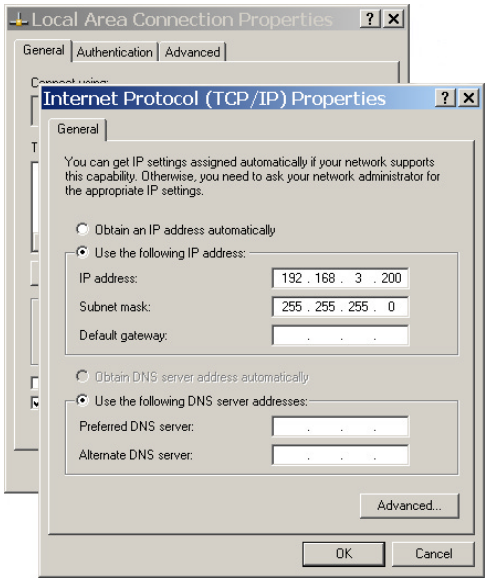


Figure 8.4.4

Click on the *Use the following IP address* radio button. Load the *IP address* text box with 192.168.3.200 and load the *Subnet Mask* with 255.255.255.0. Then click on *OK*. It should now be possible to Discover the WSN802G module's IP address by repeating Section 8.3, step 6.

8.5 Developer Board Features

A schematic of the WSN802GDK developer board is provided in the Appendix Section 10.4. The location of key components is shown in Figures 8.5.1 and 8.5.2.

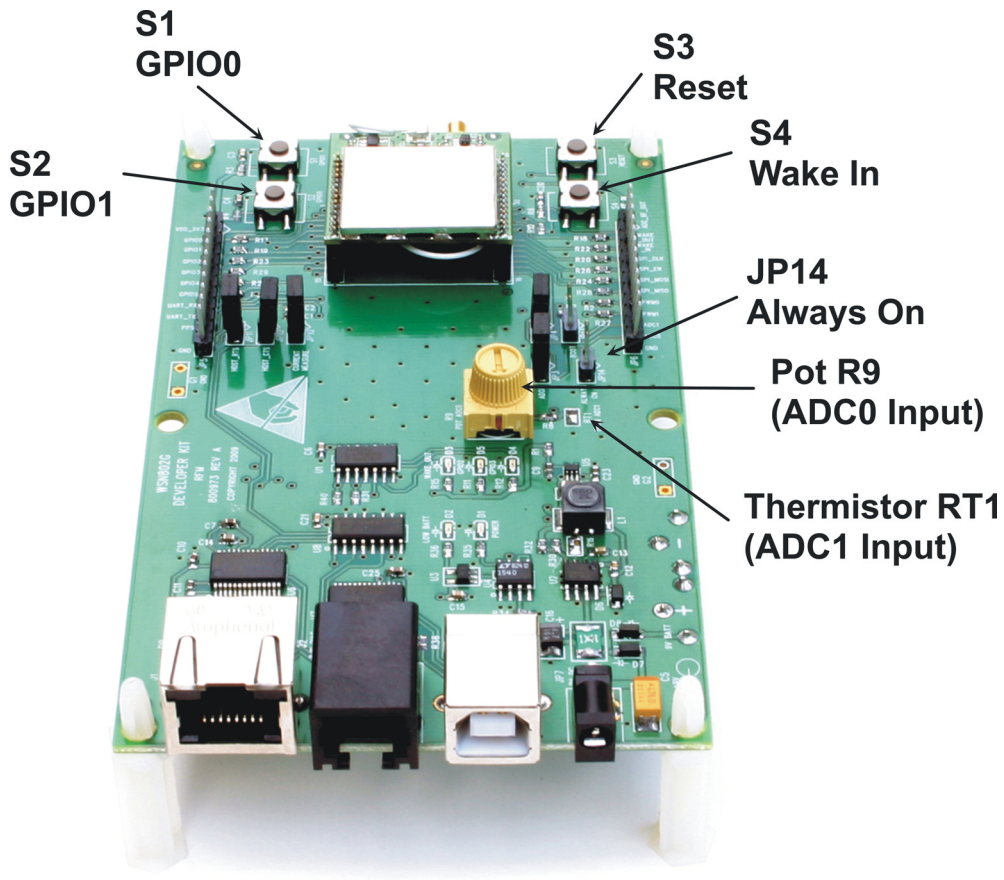


Figure 8.5.1

Switch S1 is connected to the WSN802G's GPIO0 input and switch S2 is connected to the GPIO1 input. These normally open, momentary contact switches present a logic low unless pressed, when they present a logic high. Note: the silkscreen on some developer boards have the GPIO labels reversed on switches S1 and S2. Switch S3 provides a hardware reset for the WSN802G module. Switch S4 asserts a hardware wake input to the WSN802G module.

Placing a jumper on JP14 provides a continuous wake input to the module. Pot R9 is the input to ADC0 on the WSN802G module. Thermistor RT1 is part of a voltage divider driving the ADC1 input of the WSN802G module.

LED D1 illuminates when the developer board is powered. When run from a battery, D2 will illuminate when the battery voltage declines to the minimum operating voltage. D3 illuminates when the WSN802G's WAKE_OUT line is high. D4 illuminates when GPIO3 on the module outputs a logic high. D5 illuminates when GPIO2 on the module outputs a logic high.

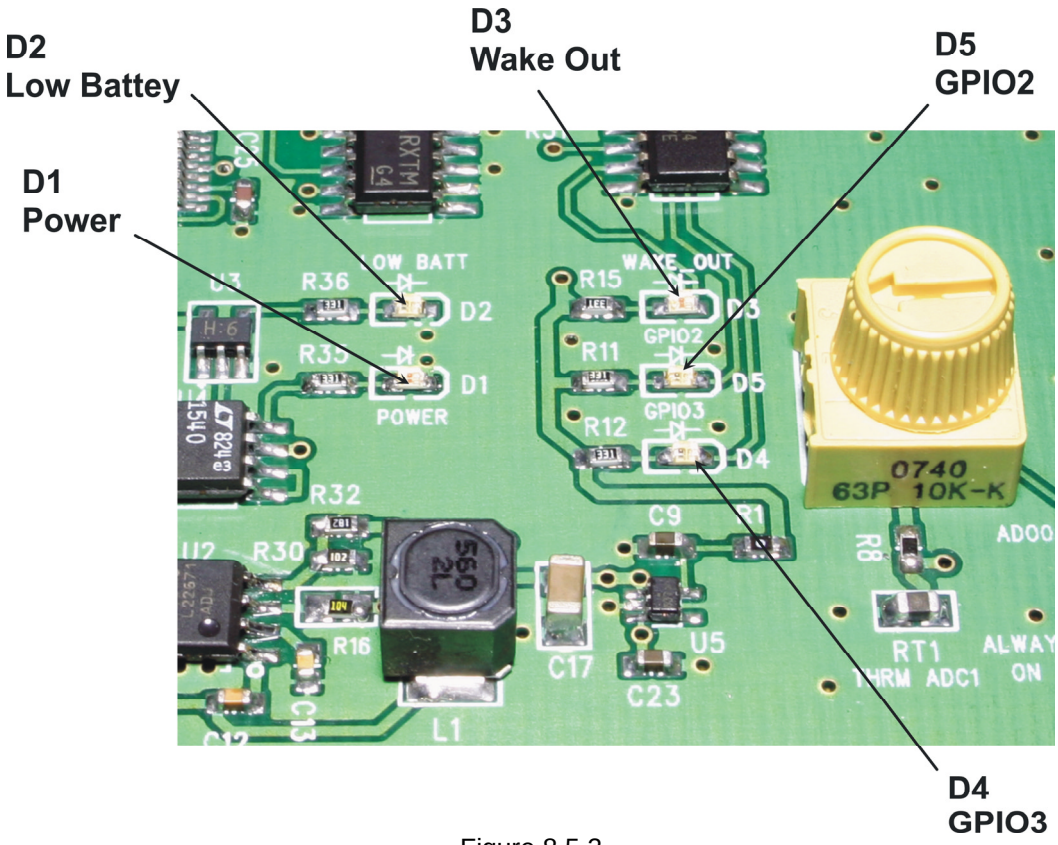


Figure 8.5.2

8.6 WSNConfig Program Operation

WSNConfig.exe provides a number of useful functions in addition to those covered above. WSN-Config.exe can be run separately or at the same time as the customer's application program to provide WSN802G module configuration support. WSNConfig.exe configuration commands run on Port 161, and WSNConfig.exe listens for SNMP commands on Port 162. The application (sensor server) runs on Port 8255 by default. When active, the customer program runs as the sensor server. Only one program at a time can run as the sensor server. WSNApp.exe on the Kit CD is an example customer application program.

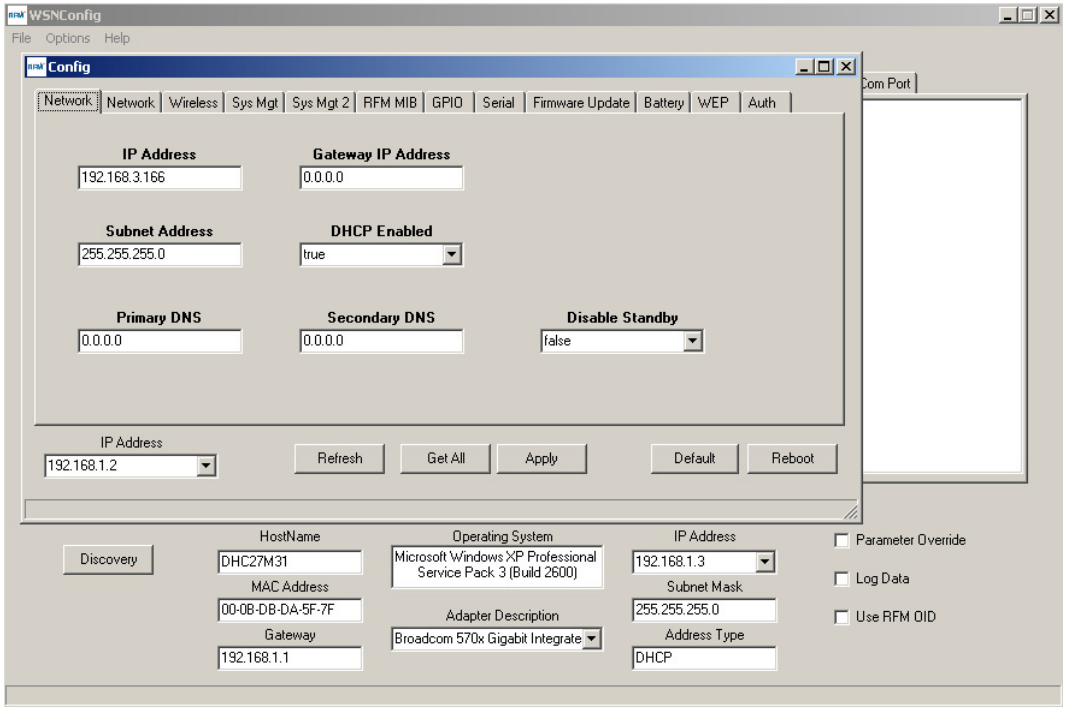


Figure 8.6.1

Double clicking on a module IP address in the left text box of the main WSNConfig frame launches a multi-tab configuration dialog box for the module, as shown in Figure 8.6.1.

All tabs in this *Config* dialog frame have *Refresh*, *Get All*, *Apply, Default* and *Reboot* buttons. WSNConfig.exe maintains a local buffer that holds a copy of all MIB configuration parameters (see Table 7.3.1). Clicking the *Refresh* button loads the configuration parameters from the local buffer into various tabs in the *Config* dialog box.

Clicking the *Get All* button queues a request to the WSN802G module to send a new copy of all its configuration parameters. How quickly the module responds depends on the *ConfigTrapInterval* system parameter and when in the trap interval the request was queued. As a new copy of the configuration parameters is received, the local buffer is updated. Clicking the *Refresh* button after the local buffer is updated will, in turn, update the data in the various tabs in the *Config* dialog box.

Clicking on the *Apply* button followed by clicking on the *Reboot* button will queue a request to the WSN802G module to modify parameter values that have been changed in a *Config* dialog tab and to reboot to use the parameter changes. Again, how quickly the module updates parameter values depends on the *ConfigTrapInterval* system parameter and when in the trap interval the request was queued.

Clicking the *Default* button queues a request to the WSN802G module to load factory default values for all configuration parameters. The *Reboot* button must be clicked following a default reload to enable the use of the default parameters. How quickly the module responds depends on the *ConfigTrapInterval* system parameter and when in the trap interval the request was queued.

The *IP Address* in the lower left of each tab in the Config dialog box is the WSN802G module IP address.

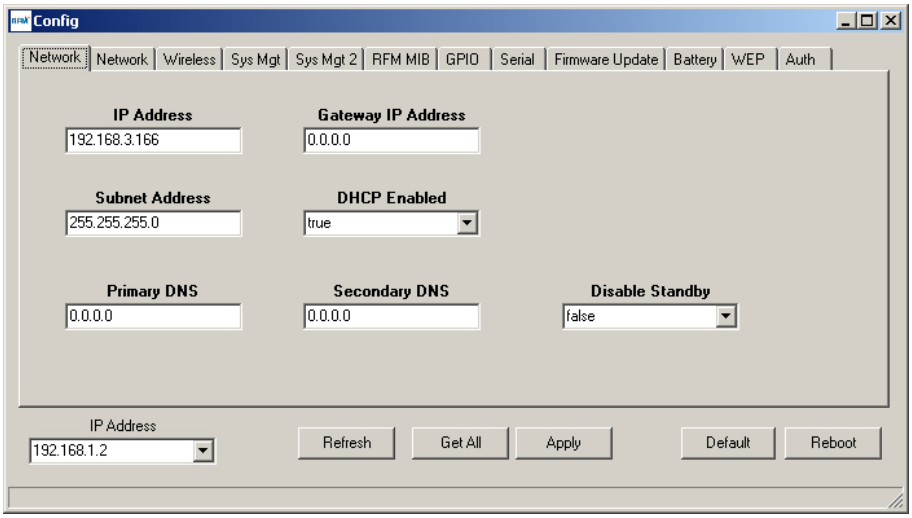


Figure 8.6.2

The first *Network* tab displays basic network parameters information. *Note: Unless you are familiar with IP networking contact RFM module technical support before making any changes on this tab. A parameter entry error on this tab can disable the WSN802G module wireless link.*

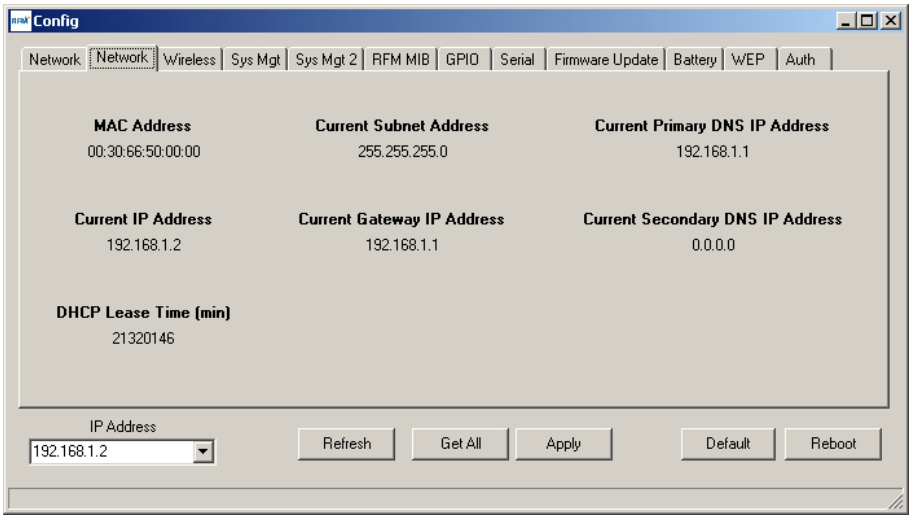


Figure 8.6.3

The second *Network* tab displays the module's fixed MAC address, plus the currently assigned IP, subnet, gateway and DNS addresses, and the DHCP lease time remaining for these assignments. The contents of this tab are read only.

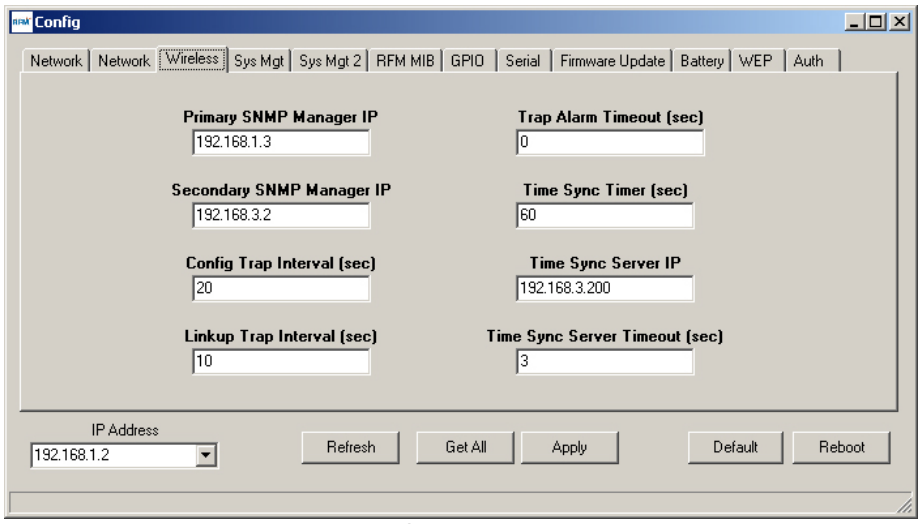


Figure 8.6.4

The *Wireless* tab accepts inputs for the *Primary* and *Secondary SNMP Manager IP* addresses, the *Config* and *Linkup Trap* intervals, the *Trap Alarm Timeout* (0 disables the alarm), plus the *IP Address*, poll *Timer* interval, and *Timeout* for the *Time Sync Server*. Clicking the *Apply* button and then the *Reboot* button queues a request to update the module with changes entered in this tab.

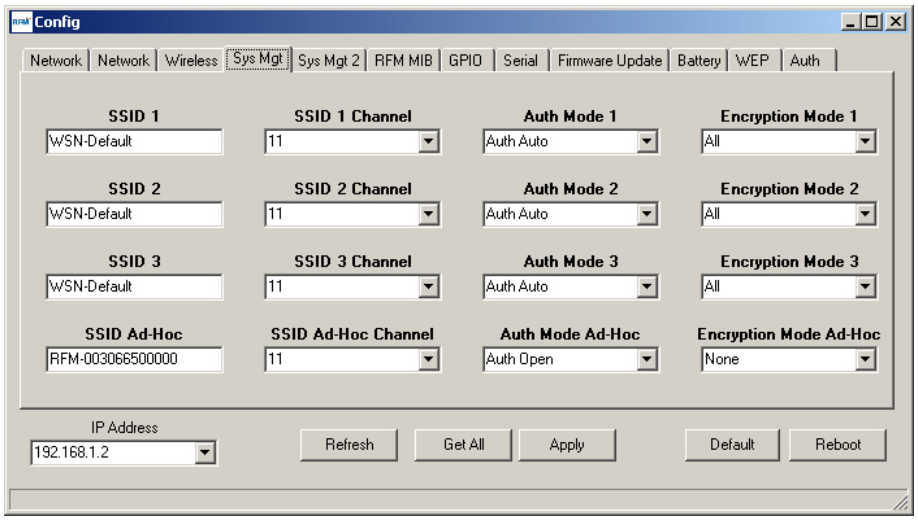


Figure 8.6.5

The first *System Management* tab displays and accepts inputs for the *SSID*, *Channel*, *Authentication Mode* and *Encryption Mode* for WLAN configurations 1, 2, 3 and Ad-Hoc.

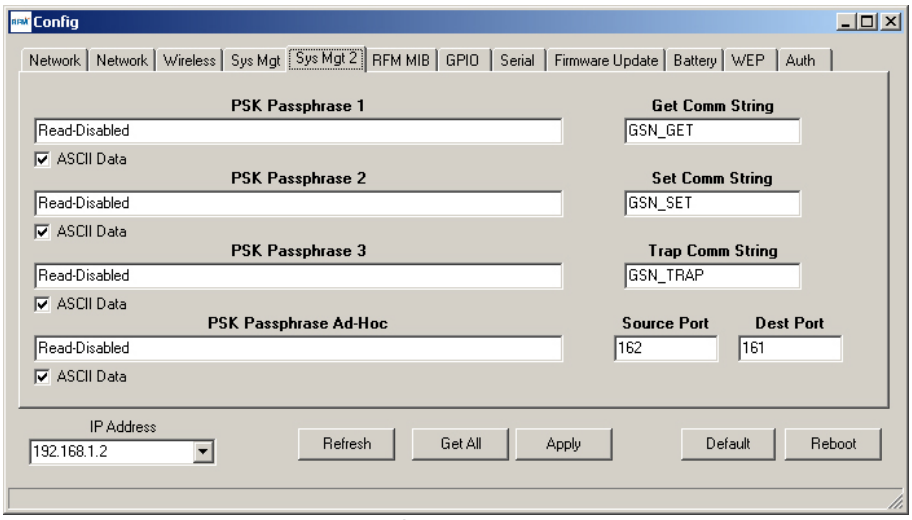


Figure 8.6.6

The second *System Management* tab displays and accepts inputs for the PSK (WPA2) Passphrases for WLAN configurations 1, 2, 3 and Ad-Hoc. The community strings for *Get*, *Set* and *Trap* are also displayed on this tab, plus the related trap *Source* and *Destination* ports.

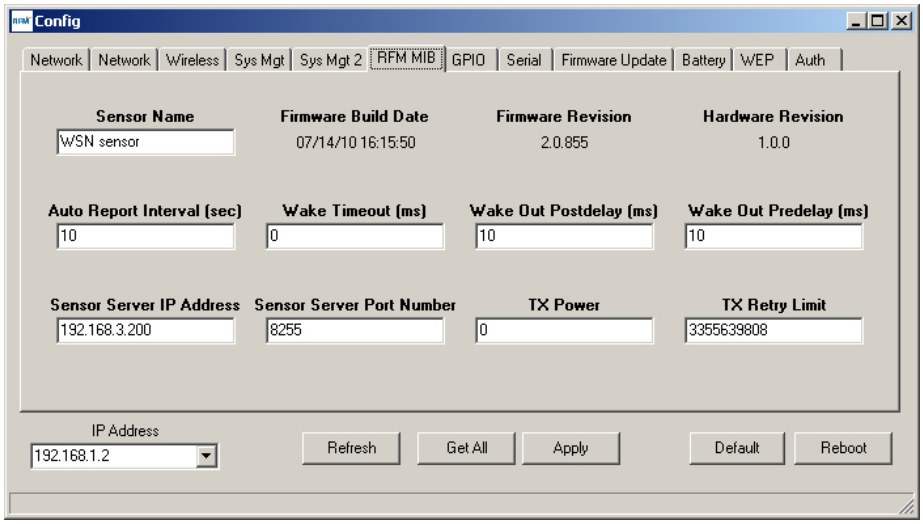


Figure 8.6.7

The *RFM MIB* tab displays and accepts inputs related to the basic MIB OID application parameters. The *Sensor Server IP Address* can be set and applied in this tab as an alternative to Step 7 in Section 8.3. The *Auto Report Interval* can be increased over the default 10 second interval to conserve battery power where the values or states of the sensor inputs change slowly.

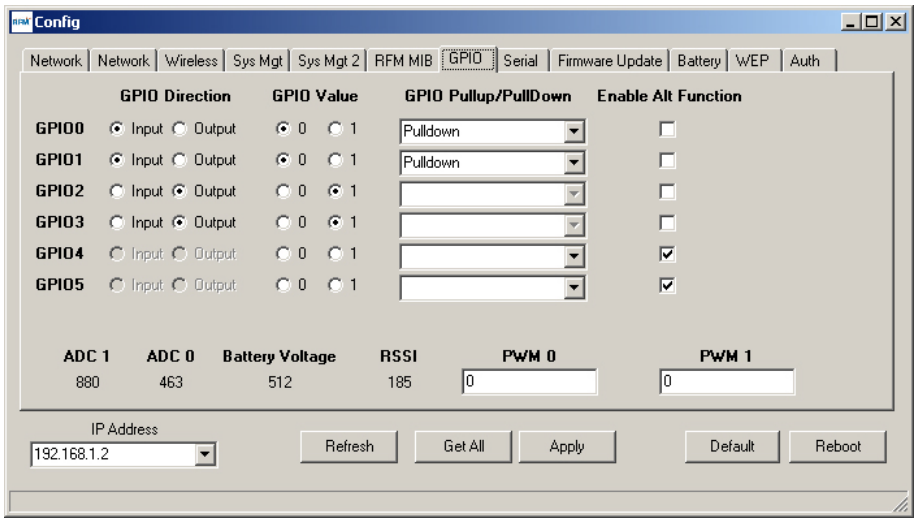


Figure 8.6.8

The *GPIO* tab displays analog and digital module inputs and displays and accepts changes in analog and digital module outputs. Numerical data is displayed and entered in decimal format. Note: the PWM 1 text box is not used by the current version of the WSN802G module.

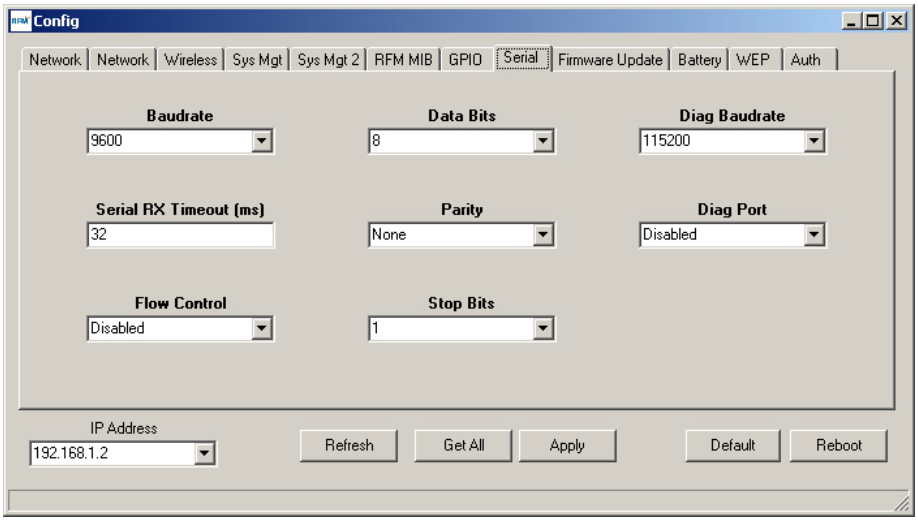


Figure 8.6.9

The *Serial* tab displays and accepts changes for parameters related to the module's serial ports. The *Diag* Port is not used for operational diagnostics in the current firmware.

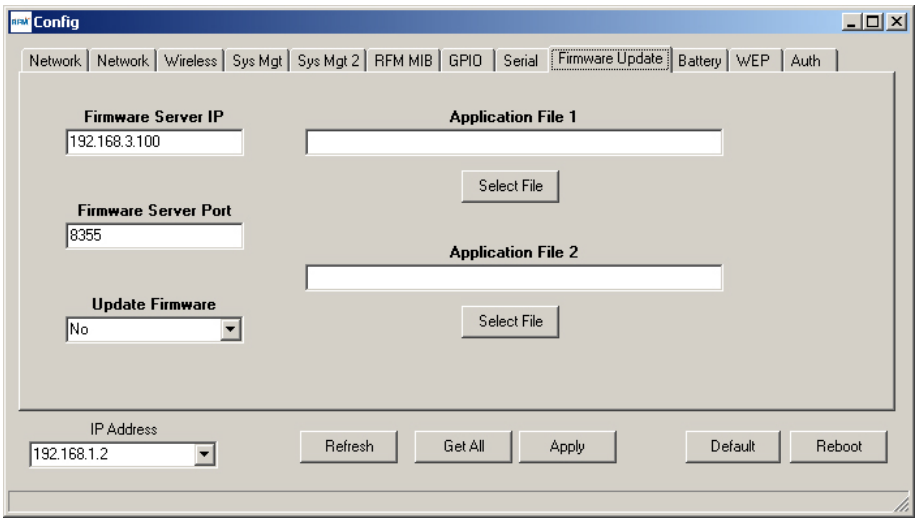


Figure 8.6.10

The *Firmware Update* tab allows new firmware to be loaded into a WSN802G module. Input the names of the two firmware *Application Files* including the paths to the files if they are not located in the same folder as WSNConfig.exe. Click on each *Select File* button. Set the *Firmware Server IP* address and *Firmware Server Port* number. Select *Yes* from the *Update Firmware* drop-down box and click on *Apply*.

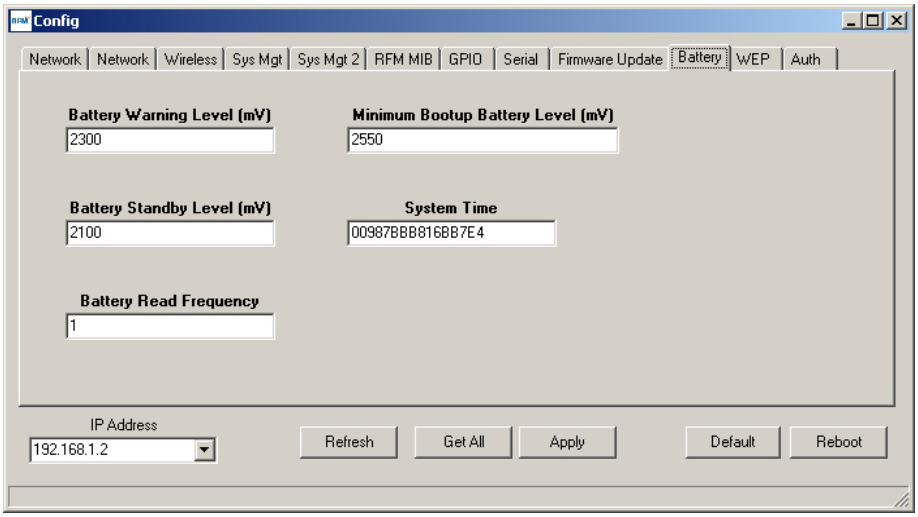


Figure 8.6.11

The *Battery* Tab allows the power supply voltage levels that send a warning, place the module in standby, or allow the module to bootup to be read and/or modified. The read frequency is in units of seconds. *Note: contact RFM module technical support before making changes to the default Warning, Standby and Bootup levels. System Time* is also provided on this Tab for reference, but its value is not require for external applications.

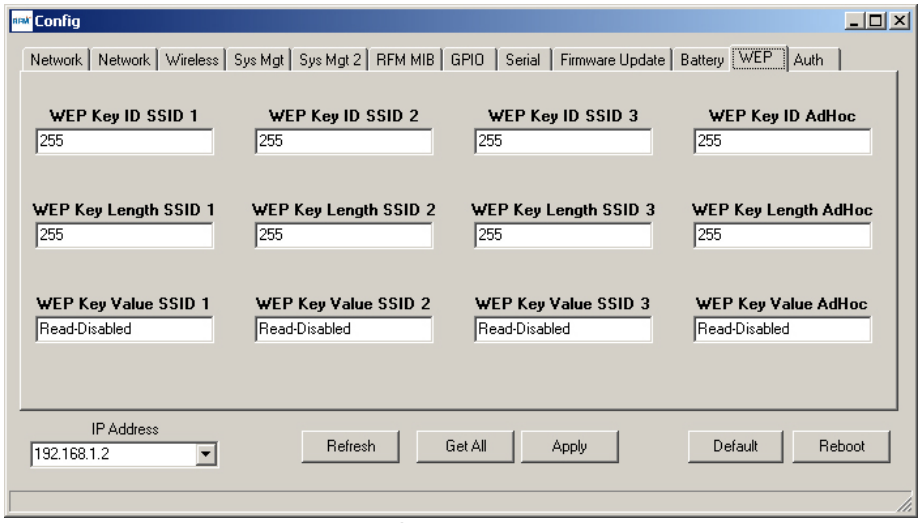


Figure 8.6.12

WEP encryption can be used instead of PSK encryption for compatibility with legacy WiFi routers. WEP encryption is not implemented in the current version of the module firmware.

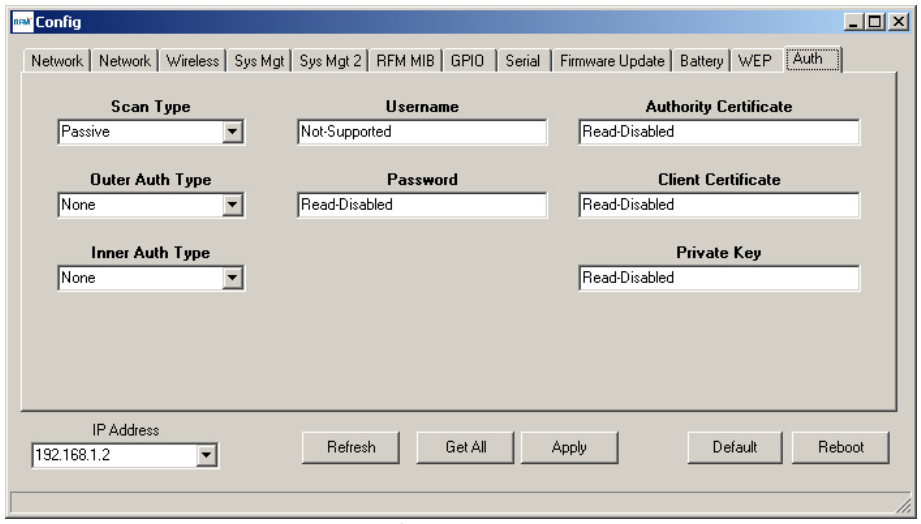


Figure 8.6.13

The *Authentication* Tab is included on the wireless link *Config* tab for the future inclusion of enterprise security. *Scan Type* is currently defaulted to *Passive* and should be left in this mode.

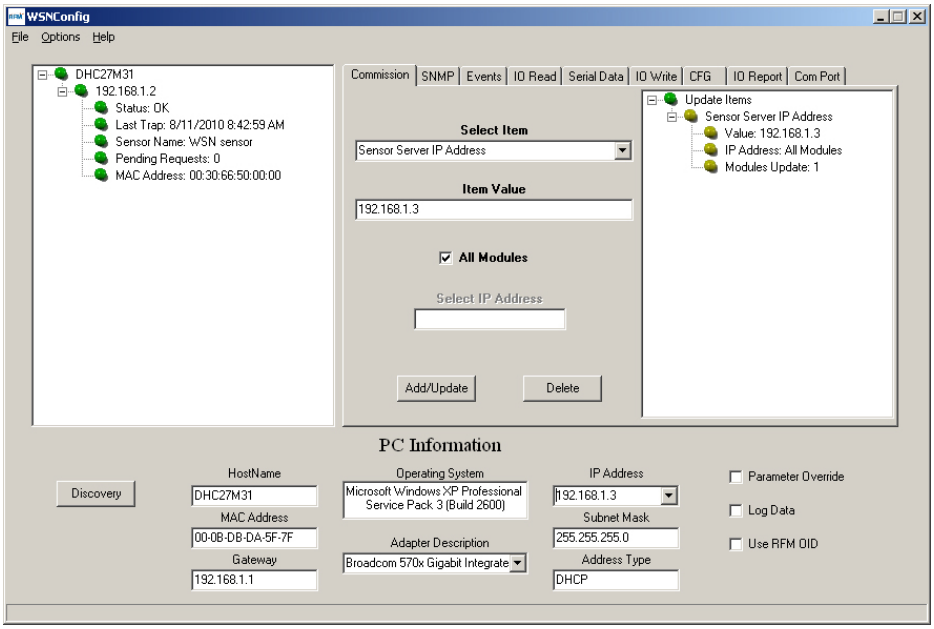


Figure 8.6.14

Figure 8.6.14 shows the same *WSNConfig* main screen as Figure 8.3.4, but with the module IP address tree expanded on the left and the Sensor Sever IP Address tree expanded on the right.

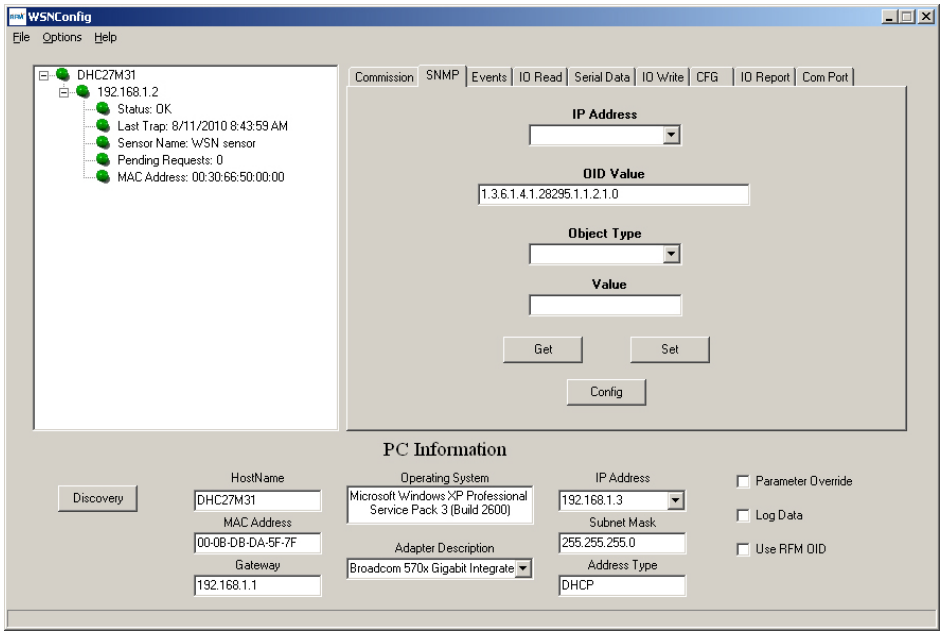


Figure 8.6.15

The *SNMP* tab allows individual parameters to be manually displayed and modified. The *OID Value*s and the *Object Types* for the two parameter sets are detailed in Sections 7.2 (GainSpan) and 7.3 (RFM). Checking the *Use RFM OID* box on the lower right corner of the main Window accesses the RFM OID parameter set. Clicking the *Get* button queues a request to retrieve the value of a parameter. Clicking the *Set* button queues a request to change the value of a parameter. How quickly the module responds to *Set* or *Get* depends on the *ConfigTrapInterval* system parameter and when in the trap interval the request was queued. Clicking the *Config* button launches the wireless link *Config* dialog box discussed above.

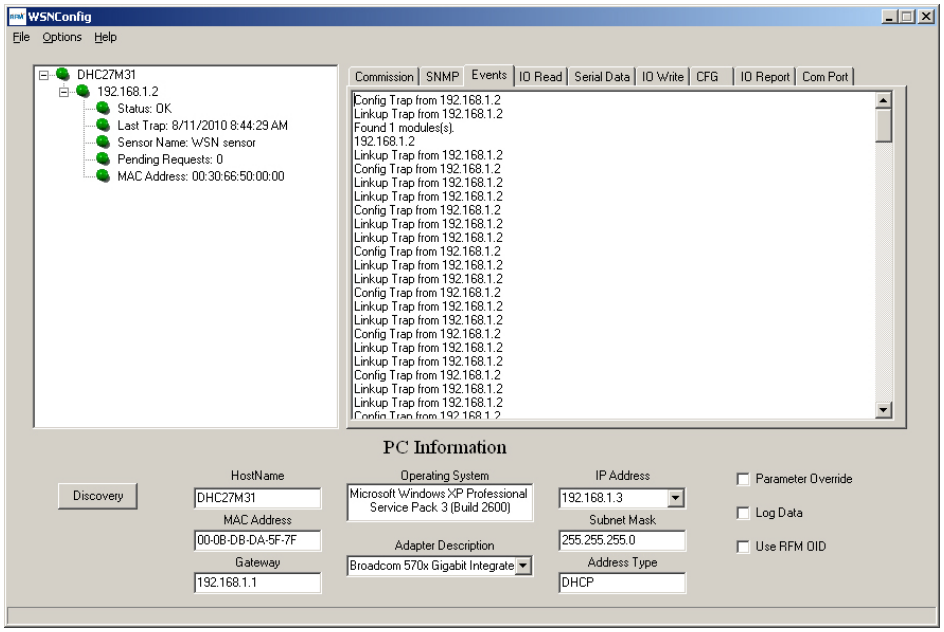


Figure 8.6.16

The *Events* tab displays a running history of events received from the WSN802G module.

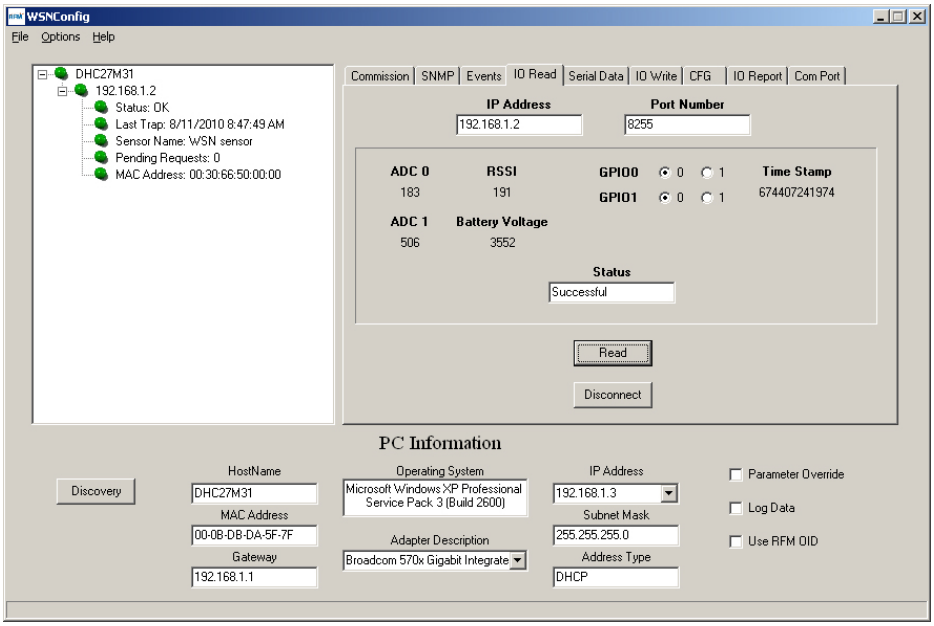


Figure 8.6.17

The WSN802G module must be awake in order to use the *IO Read* tab. To hold the module awake, place a jumper on JP14, which is labeled ALWAYS ON or push and hold the WAKE IN button. The *IP Address* and *Port Number* on the I/O Read tab refer to the WSN802G module's Sensor socket. Clicking *Read* sends an IO_READ_REQUEST to the module. Note that *ADC0* is reading the voltage from pot R9, and *ADC1* is reading the voltage from a voltage divider consisting of a fixed resistor and thermistor RT1. *GPIO0* reads the state of switch S1 and *GPIO1* reads the state of switch S2. A *GPIO* value of 1 indicates the switch is closed. Note: the silkscreen on some developer boards have the GPIO labels reversed on switches S1 and S2. A quick reply to the IO_READ_REQUEST is indicated by a *Successful Status*. A *Failure Status* is usually caused by forgetting to install a jumper on JP14.

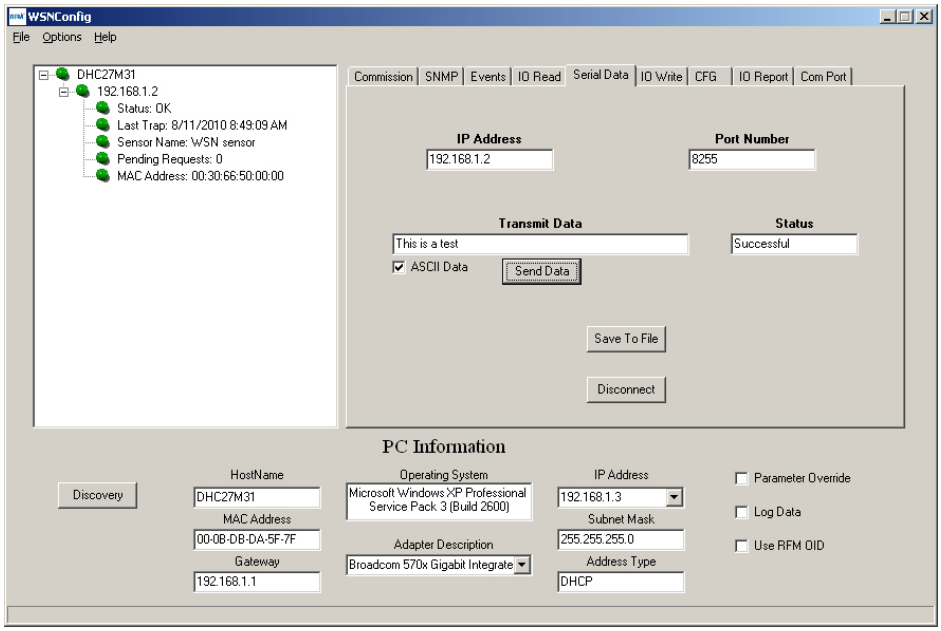


Figure 8.6.18

The WSN802G module must be awake in order to use the *Serial Data* tab. To hold the module awake, place a jumper on JP14, which is labeled ALWAYS ON or push and hold the WAKE IN button. The *IP Address* and *Port Number* on the I/O Read tab refer to the WSN802G module's Sensor socket. Clicking *Send Data* sends the string in the *Transmit Data* text box in a SERIAL_DATA command to the module. A quick reply to the command is indicated by a *Successful Status*. A *Failure Status* is usually caused by forgetting to install a jumper on JP14.

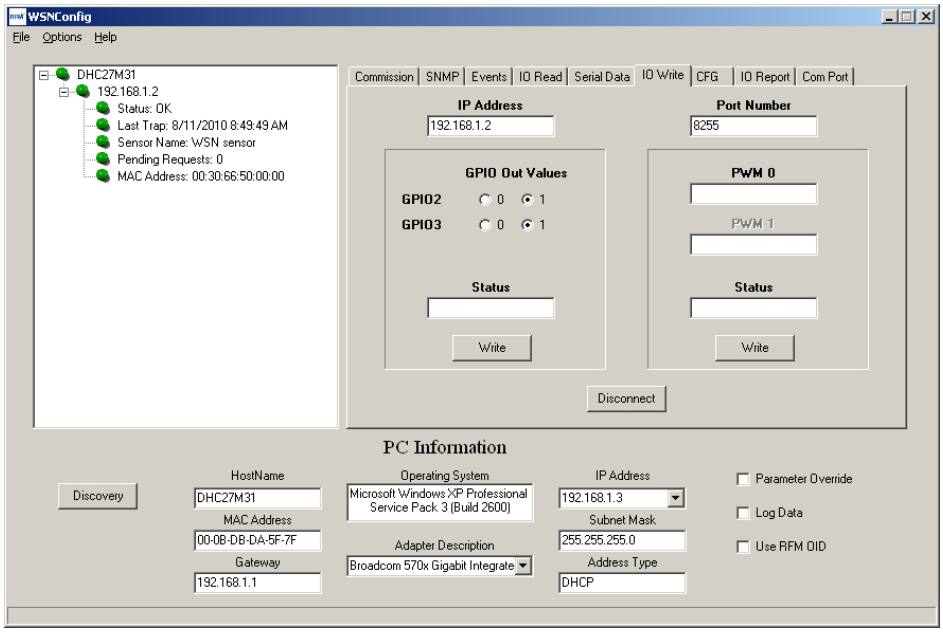


Figure 8.6.19

The WSN802G module must be awake in order to use the *IO Write* tab. To hold the module awake, place a jumper on JP14, which is labeled ALWAYS ON or push and hold the WAKE IN button. The *IP Address* and *Port Number* on the I/O Read tab refer to the WSN802G module's Sensor socket. Clicking the *Write* button under *GPIO Out Values* sends an IO_WRITE_GPIO to the module. Clicking the *Write* button under

PWM 0 sends an IO_WRITE_PWM command to the module. A quick reply to either write command is indicated by a *Successful Status*. A *Failure Status* is usually caused by forgetting to install a jumper on JP14. On the developer board, GPIO2 and 3 are connected to LEDs such that the LEDs light when the module pin is set to 1.

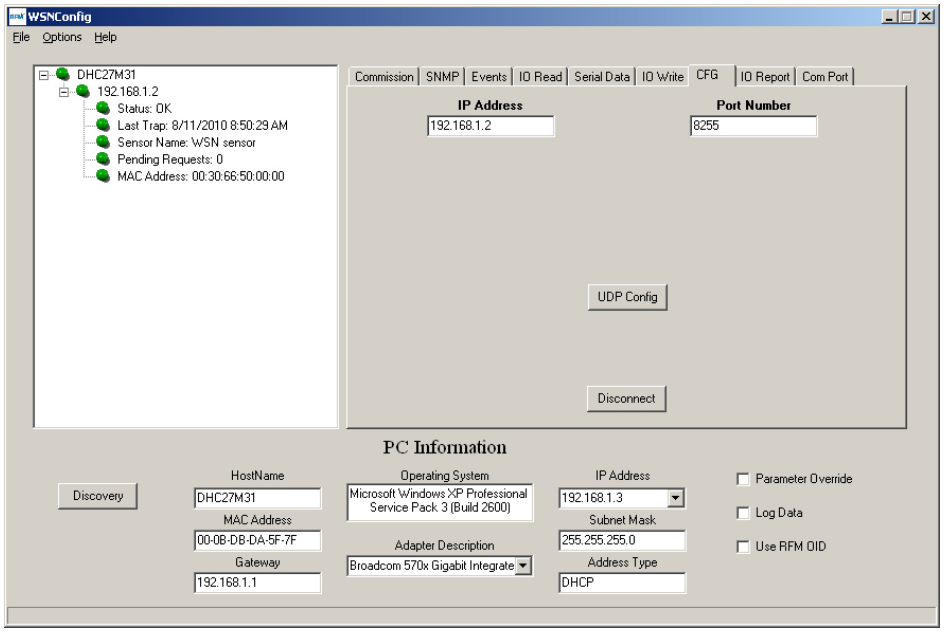


Figure 8.6.20

The *CFG* tab allows the API CFG commands to be routed in UDP packets over the wireless link.

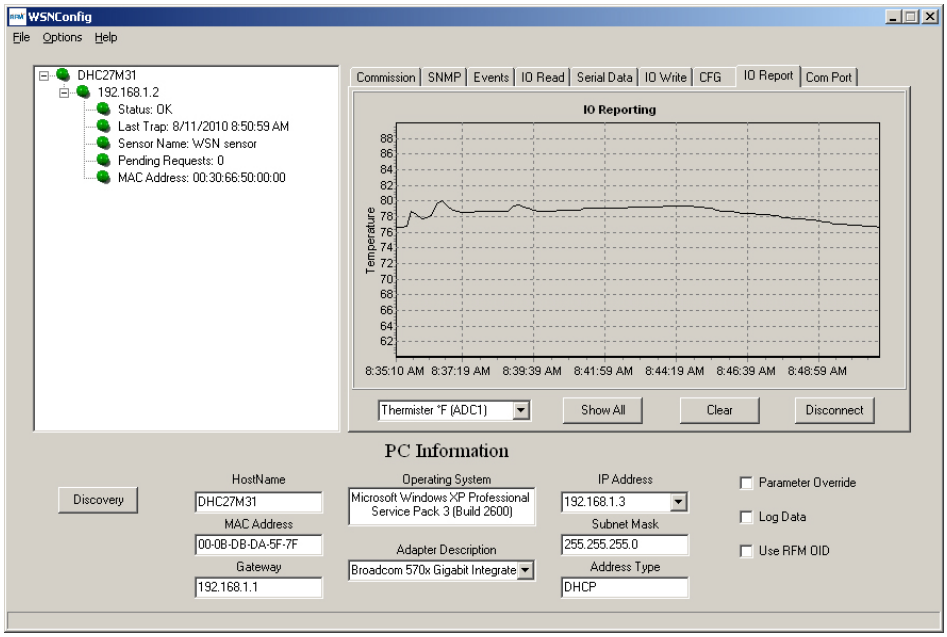


Figure 8.6.21

The WSN802G module must be awake in order to use the *IO Report* tab. To hold the module awake, place a jumper on JP14, which is labeled ALWAYS ON or push and hold the WAKE IN button. Either a single parameter can be selected for charting from the drop-down box in the lower left corner of the *IO Report* tab, or all parameters can be selected for charting by clicking on the *Show All* button. The chart area can be cleared and reset by clicking the *Clear* button.

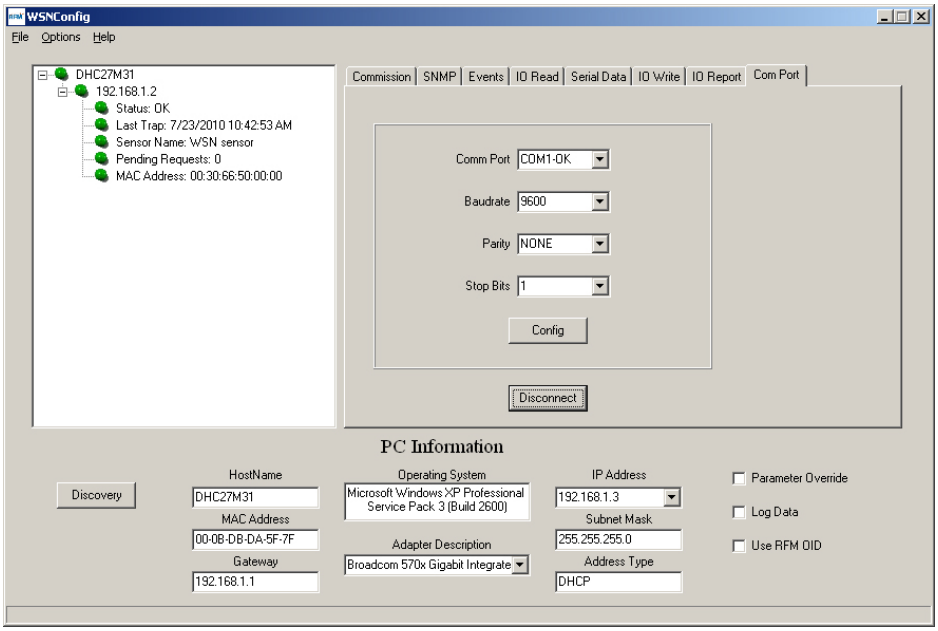


Figure 8.6.22

The WSN802G module must be awake in order to use the *COM Port* tab. To hold the module awake, place a jumper on JP14, which is labeled ALWAYS ON or push and hold the WAKE IN button. The *Com Port* tab allows access to the same parameters through the serial port that can be accessed by the SNMP server over the wireless link using the RFM OID addresses. Install a serial cable between the host PC and the WSN802G development board. Then click on the *Connect* button as shown in Figure 8.6.21 above. The *Config* button will become active. Next, click on the *Config* button to launch the multi-tab serial configuration dialog window, as shown in Figure 8.6.23.

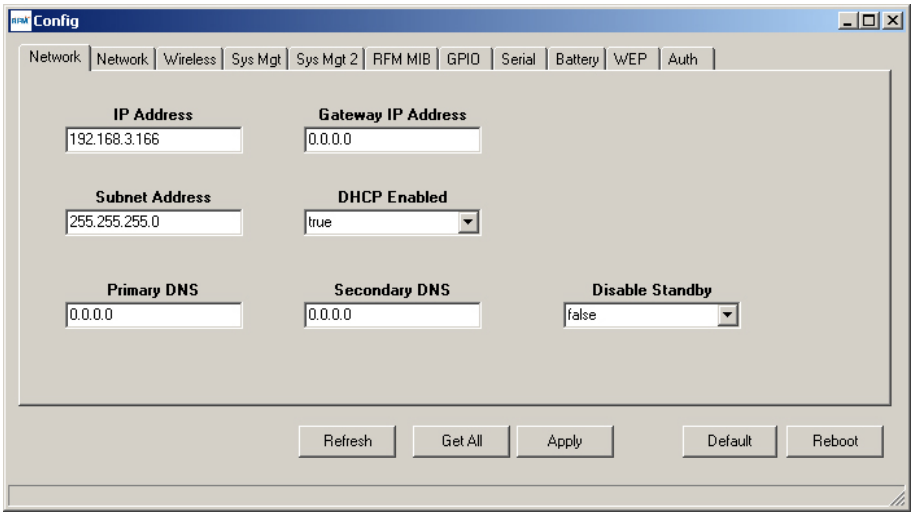


Figure 8.6.23

All tabs in the serial *Config* dialog frame have *Refresh*, *Get All*, *Apply, Default* and *Reboot* buttons. WSNConfig.exe maintains a local buffer that holds a copy of all MIB configuration parameters (see Table 7.3.1). Clicking the *Refresh* button loads the configuration parameters from the local buffer into various tabs in the *Config* dialog box.

Clicking the *Get All* button requests the WSN802G module to send a new copy of all its configuration parameters. As a new copy of the configuration parameters is received, the local buffer is updated. Clicking

the *Refresh* button after the local buffer is updated will, in turn, update the data in the various tabs in the *Config* dialog box.

Clicking on the *Apply* button followed by clicking on the *Reboot* button requests the WSN802G module to modify parameter values that have been changed in a *Config* dialog tab and to reboot to use the parameter changes.

Clicking the *Default* button requests the WSN802G module to load factory default values for all configuration parameters. The *Reboot* button must be clicked following a default reload to enable the use of the default parameters.

The first *Network* tab, as shown in Figure 8.6.24, displays and accepts inputs for basic network parameters.

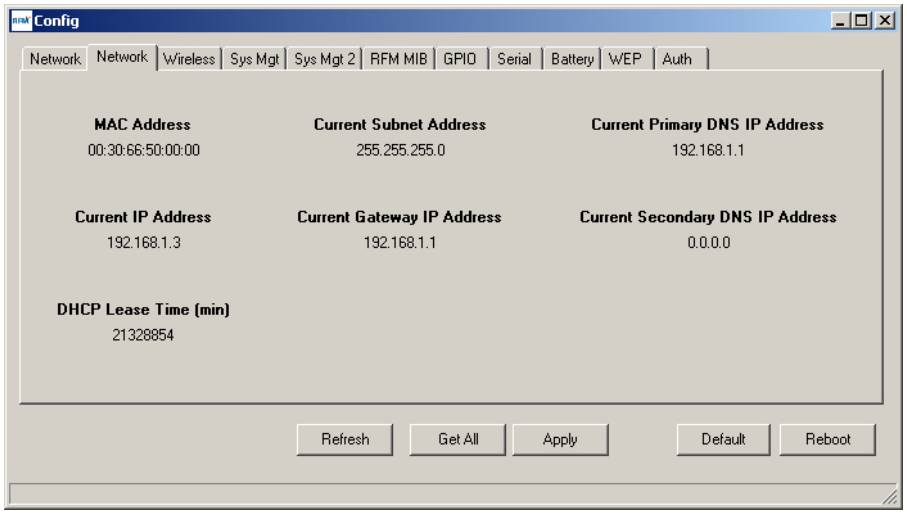


Figure 8.6.24

The second *Network* tab displays the module's fixed MAC address, plus the currently assigned IP, subnet, gateway and DNS addresses, and the DHCP lease time remaining for these assignments. The contents of this tab are read only.

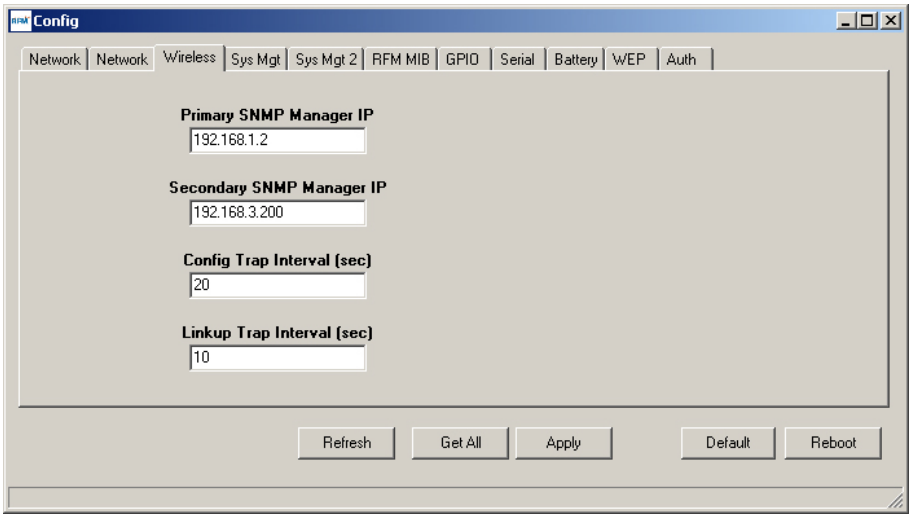


Figure 8.6.25

The *Wireless* tab accepts inputs for the *Primary* and *Secondary SNMP Manager IP* addresses, and the *Config* and *Linkup Trap* intervals. Clicking the *Apply* button and then the *Reboot* button requests the module to update the parameter changes entered in this tab.

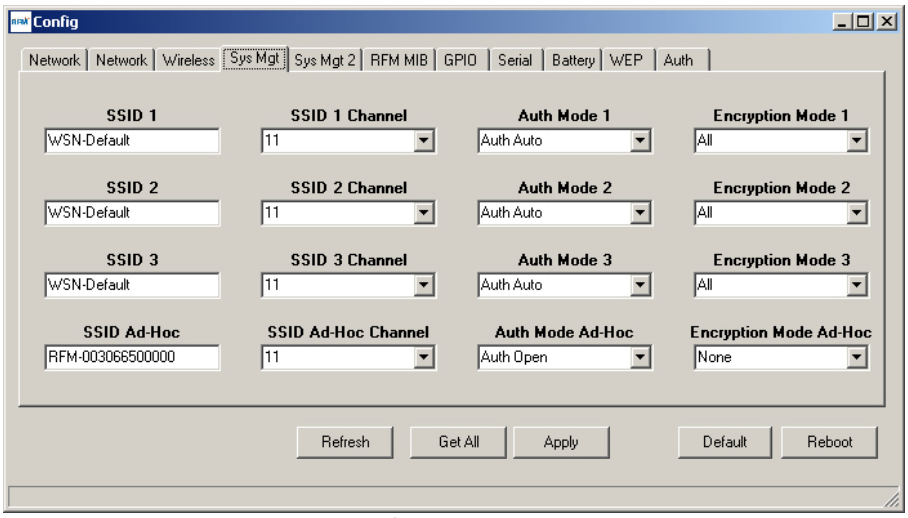


Figure 8.6.26

The first *System Management* tab displays and accepts inputs for the *SSID*, *Channel*, *Authentication Mode* and *Encryption Mode* for WLAN configurations 1, 2, 3 and Ad-Hoc.

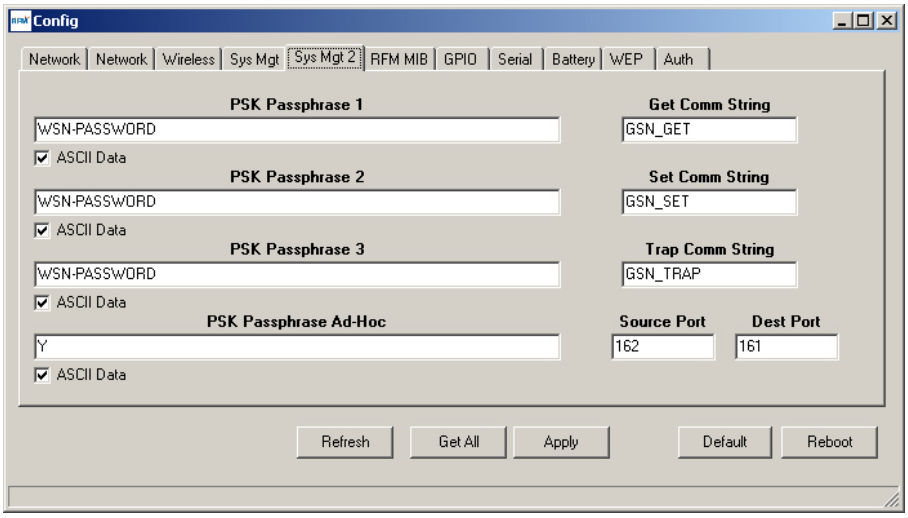


Figure 8.6.27

The second *System Management* tab displays and accepts inputs for the PSK (WPA2) Passphrases for WLAN configurations 1, 2, 3 and Ad-Hoc. The community strings for *Get*, *Set* and *Trap* are also displayed on this tab, plus the related trap *Source* and *Destination* ports.

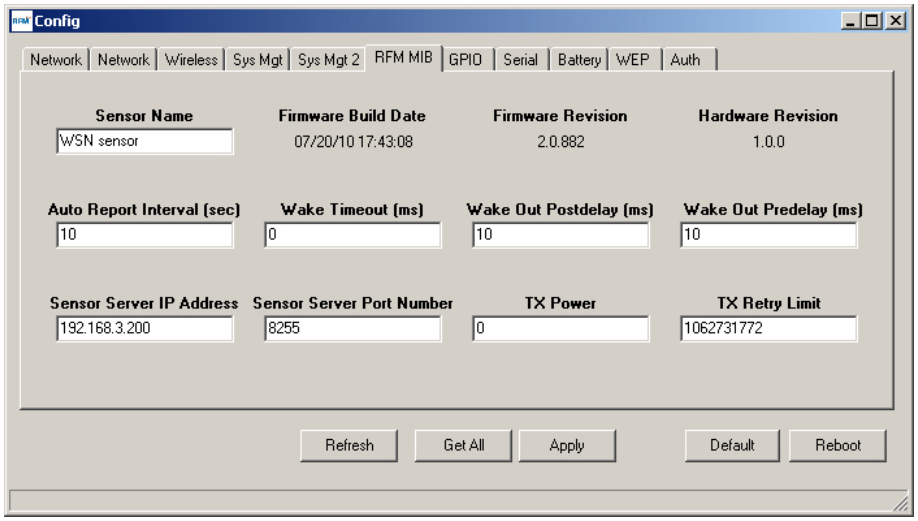


Figure 8.6.28

The *RFM MIB* tab displays and accepts inputs related to the basic MIB OID application parameters. The *Sensor Server IP Address* can be set and applied in this tab as an alternative to Step 7 in Section 8.3. The *Auto Report Interval* can be increased over the default 10 second interval to conserve battery power where the values or states of the sensor inputs change slowly.

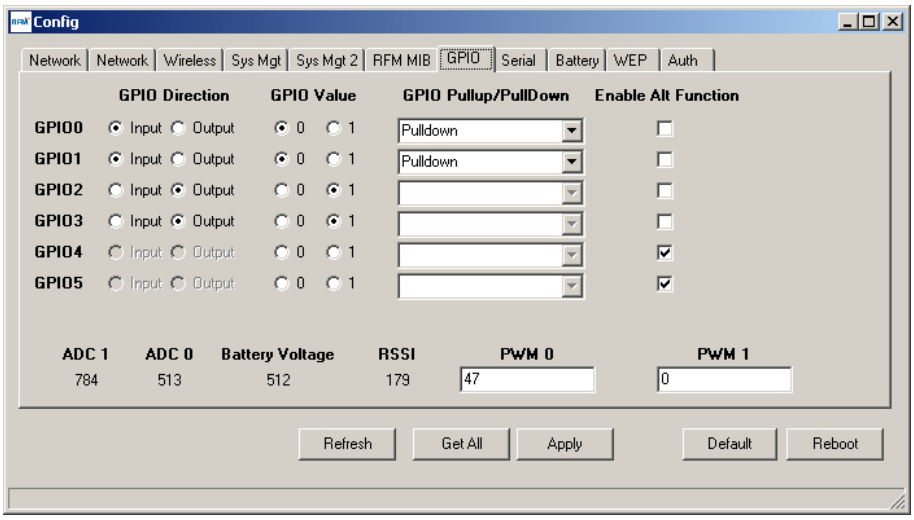


Figure 8.6.29

The *GPIO* tab displays analog and digital module inputs and displays and accepts changes in analog and digital module outputs. Numerical data is displayed and entered in decimal format. Note: the PWM 1 text box is not used by the current version of the WSN802G module.

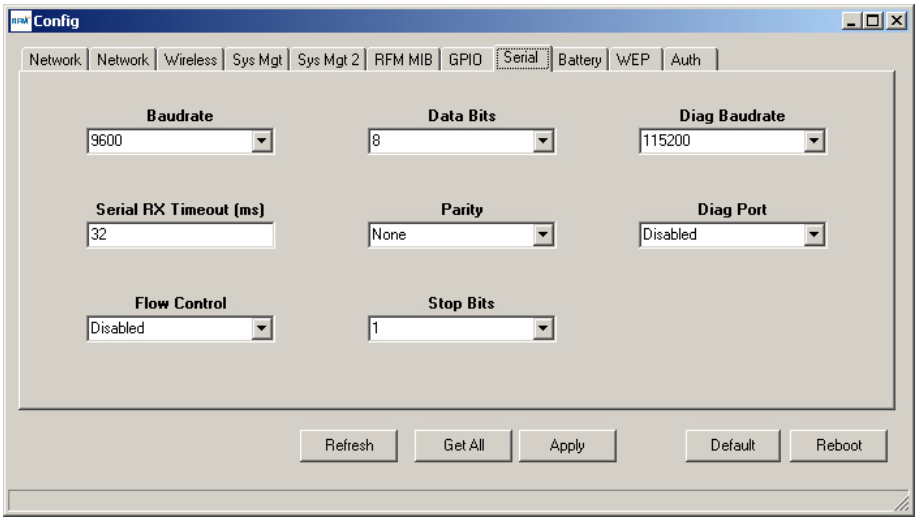


Figure 8.6.30

The *Serial* tab displays and accepts changes for parameters related to the module's serial ports. The *Diag* Port is not used for operational diagnostics in the current firmware.

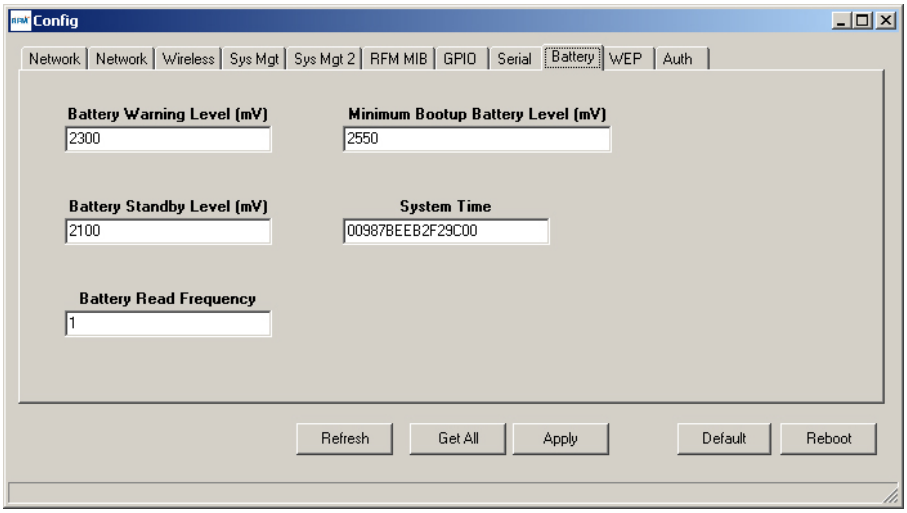


Figure 8.6.31

The *Battery* Tab allows the power supply voltage levels that send a warning, place the module in standby, or allow the module to bootup to be read and/or modified. The read frequency is in units of seconds. *Note: contact RFM module technical support before making changes to the default Warning, Standby and Bootup levels. System Time* is also provided on this Tab for reference, but its value is not require for external applications.

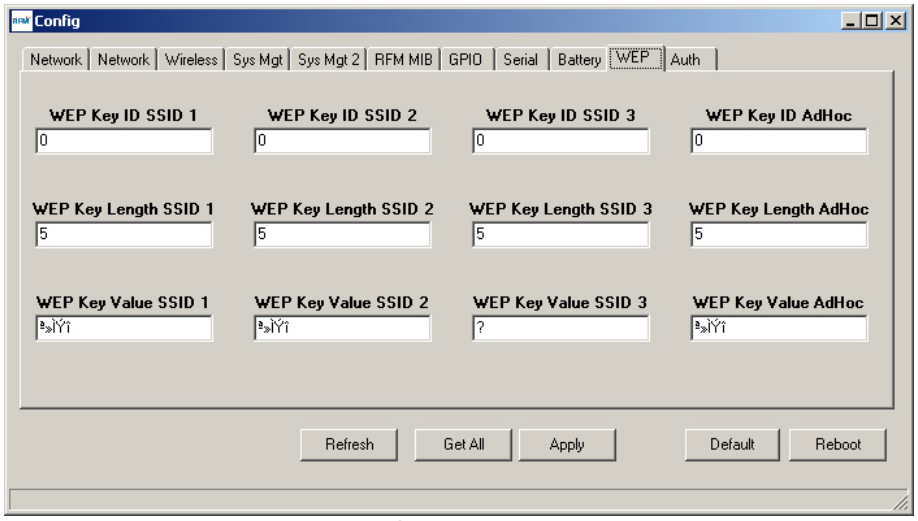


Figure 8.6.32

WEP encryption can be used instead of PSK encryption for compatibility with legacy WiFi routers. WEP encryption is not implemented in the current version of the module firmware.

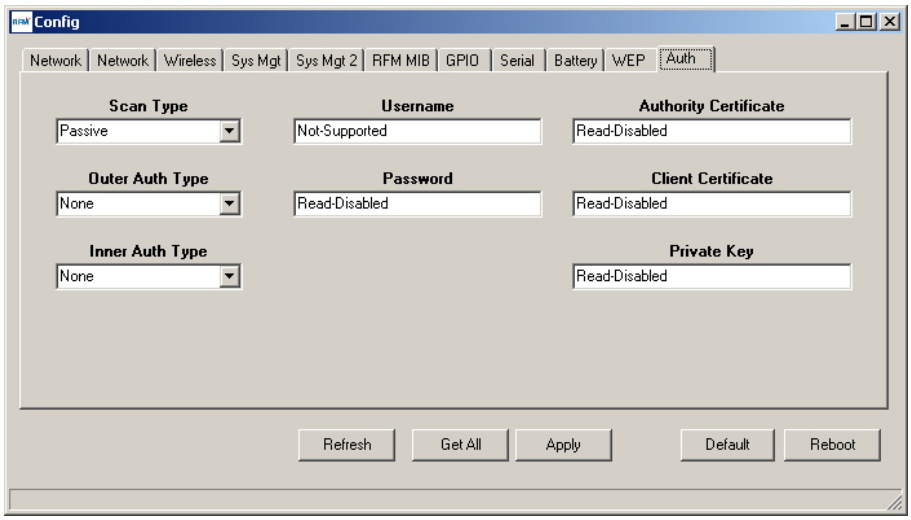


Figure 8.6.33

The *Authentication* Tab is included on the serial *Config* tab for the future inclusion of enterprise security. *Scan Type* is currently defaulted to *Passive* and should be left in this mode.

9.0 Troubleshooting

WSNConfig.exe does not communicate with a WSN802G module - The wireless router and the WiFi in the PC running WSNConfig.exe must be set to match the WSN802G Wireless configuration. The default is channel 11 with an SSID of WSN-Default (case sensitive) in secure mode. The security passphrase to allow router access is WSN-PASSWORD. Note: the NETGEAR router shipped in the WSN802GDK is preconfigured for use with the WSN802G module. Do not reconfigure the router. The WSN802G module must be connected to an antenna to work.

WSN802G will not accept sensor application commands - the module must be awake to accept application commands. Either assert a logic high on the WAKE_IN hardware line or queue the sensor application command to immediately follow an automatic (timer) I/O_REPORT. Use the *RFM MIB* tab on the *Config* dialog window in WSNConfig.exe to check for suitable *AutoReport* and *WakeTimeout* values. See Section 8.6 for additional information.

Range is extremely limited - this is usually a sign of a poor antenna connection or the wrong antenna. Check that the antenna is firmly connected. If possible, remove any obstructions near the antenna.

10.0 Appendices

10.1 Ordering Information

WSN802GC: transceiver module for solder-pad mounting with RF connector for external antenna
WSN802GCA: transceiver module for solder-pad mounting with integral chip antenna
WSN802GP: transceiver module for pin-socket mounting with RF connector for external antenna
WSN802GPA: transceiver module for pin-socket mounting with integral chip antenna

10.2 Technical Support

For WSN802G product support contact RFM's module technical support group. The phone number is +1.678.684.2000. Phone support is available from 08.30 AM to 5:30 PM US Eastern Time Zone, Monday through Friday. The e-mail address is tech_sup@rfm.com.

10.3 WSN802G Mechanical Specifications

WSN802GC Outline and Mounting Dimensions

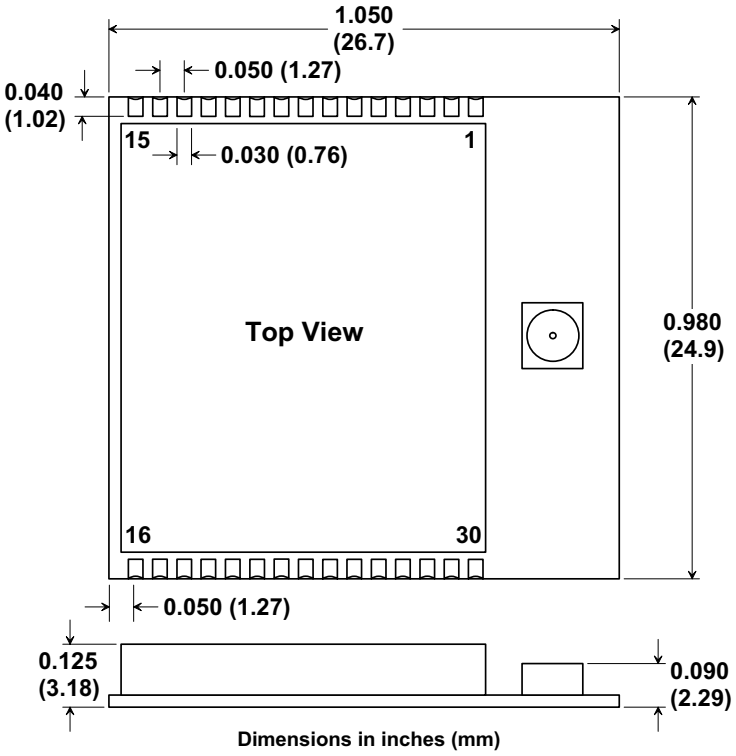


Figure 10.3.1

WSN802GC Solder Pad Dimensions

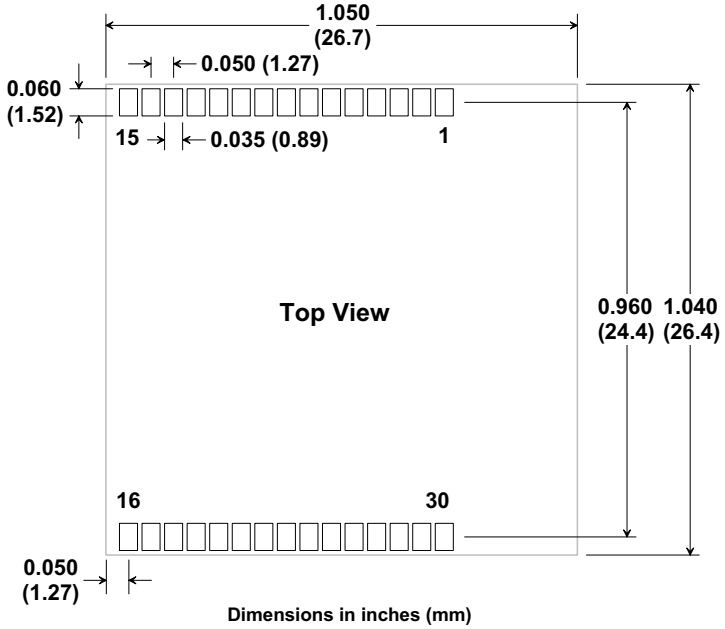


Figure 10.3.2

WSN802GP Outline and Mounting Dimensions

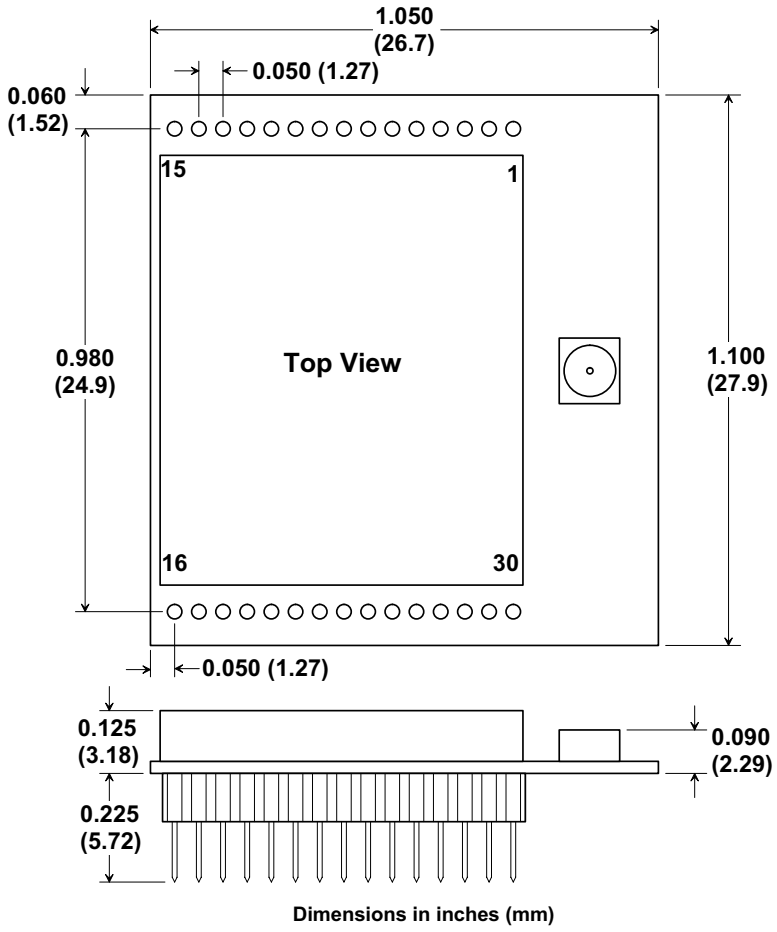


Figure 10.3.3

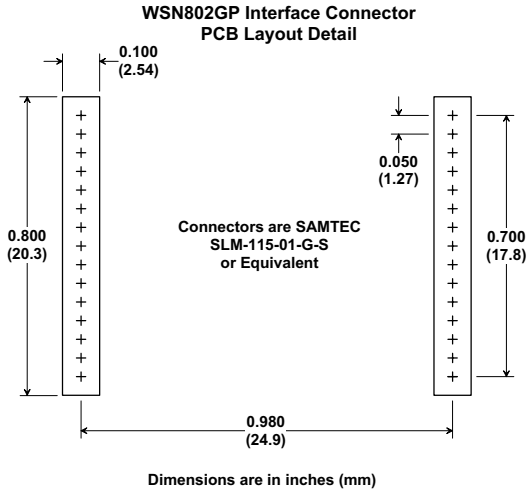


Figure 10.3.4

WSN802GCA Outline and Mounting Dimensions

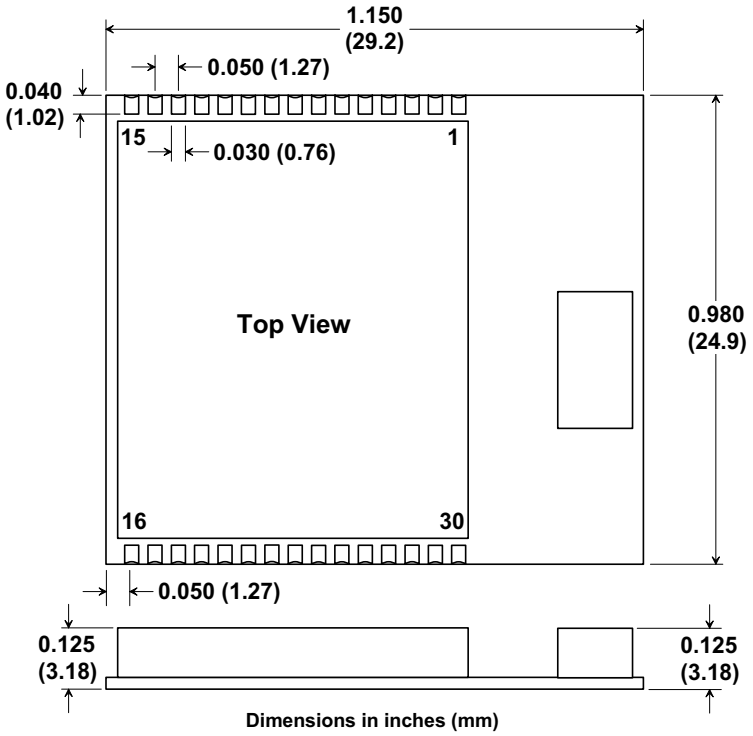


Figure 10.3.5

WSN802GCA Solder Pad Dimensions

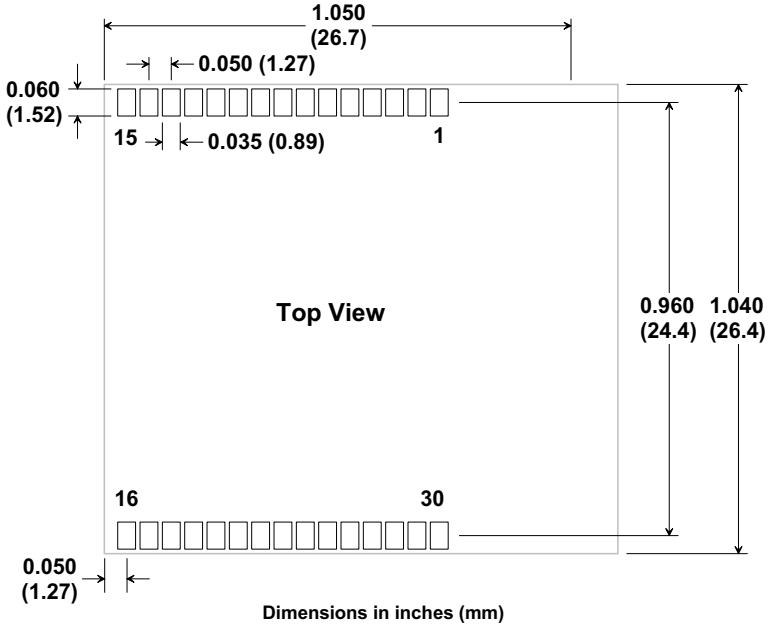


Figure 10.3.6

WSN802GPA Outline and Mounting Dimensions

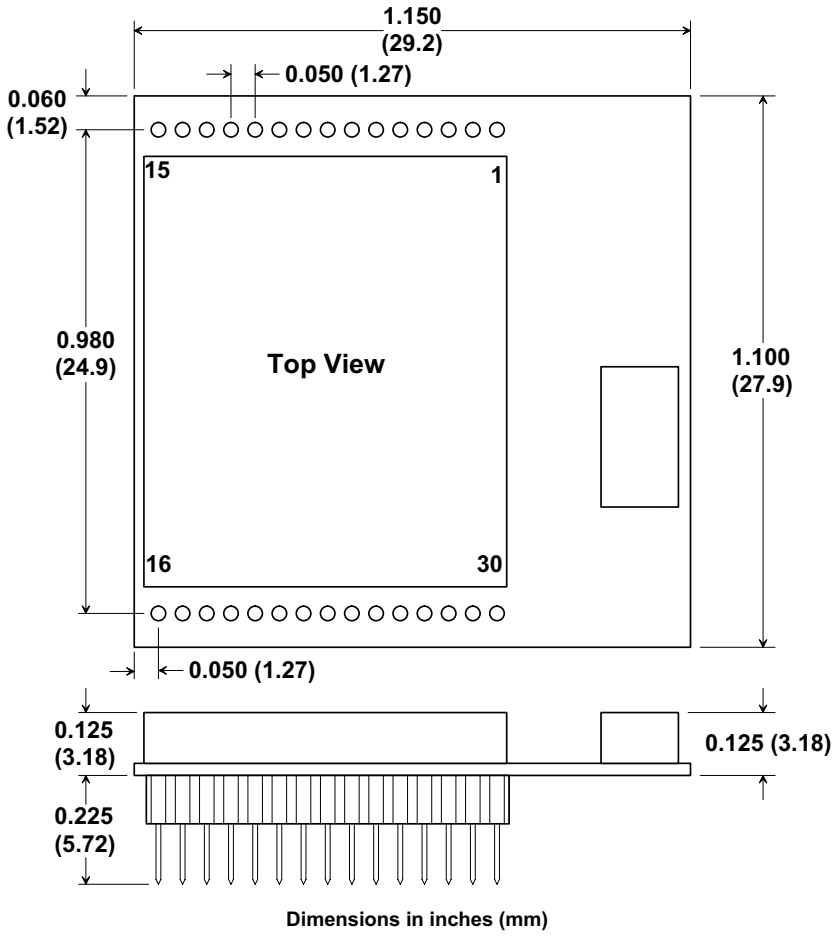


Figure 10.3.7

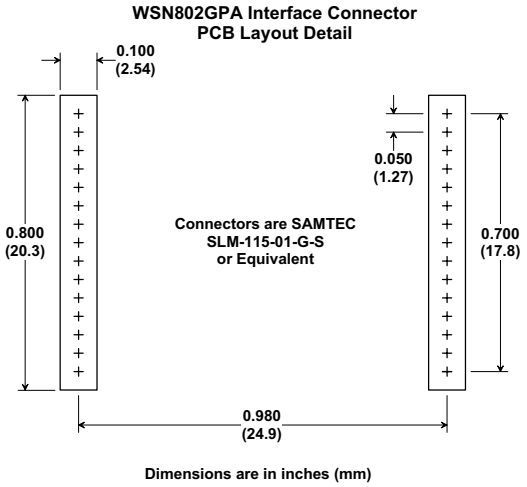
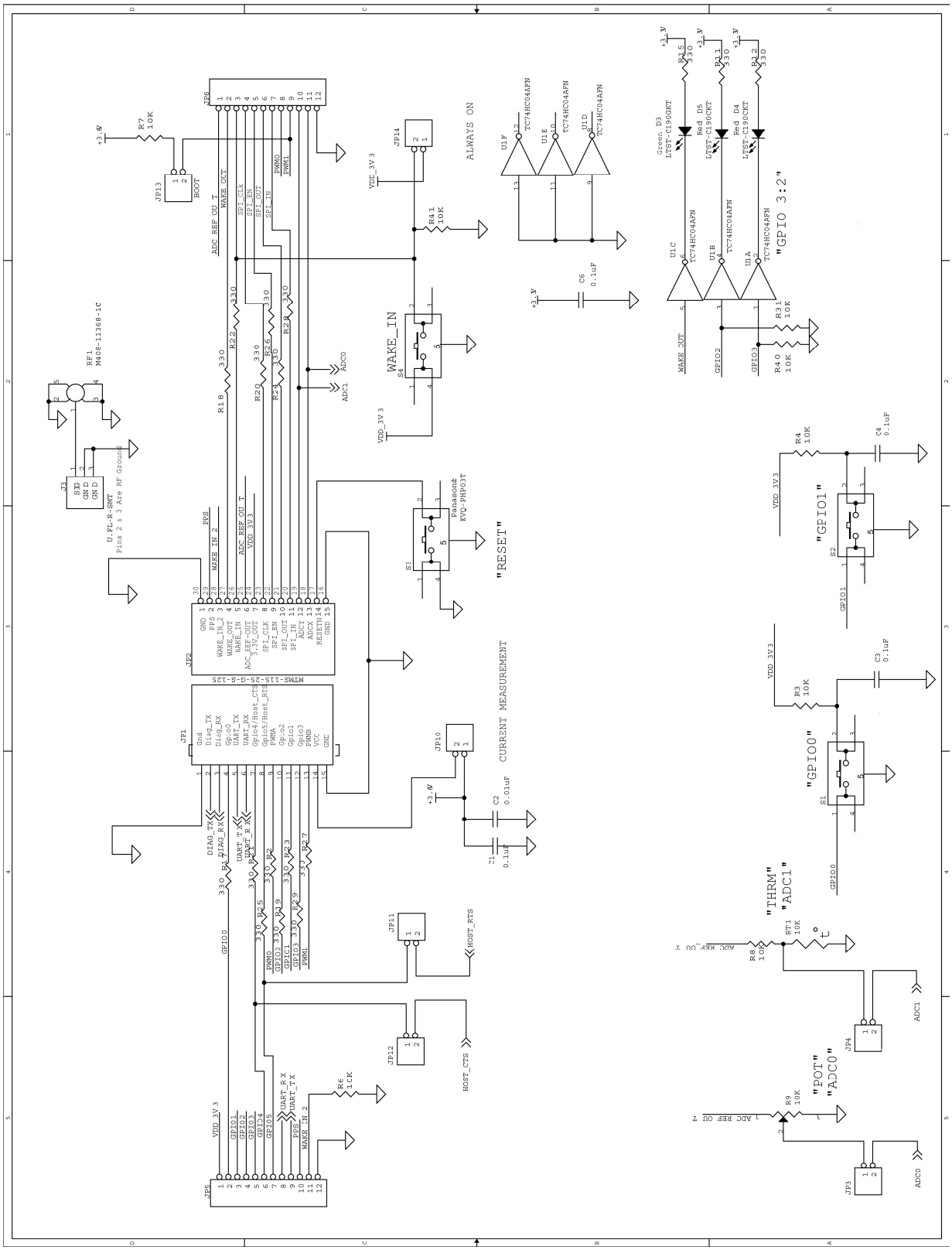
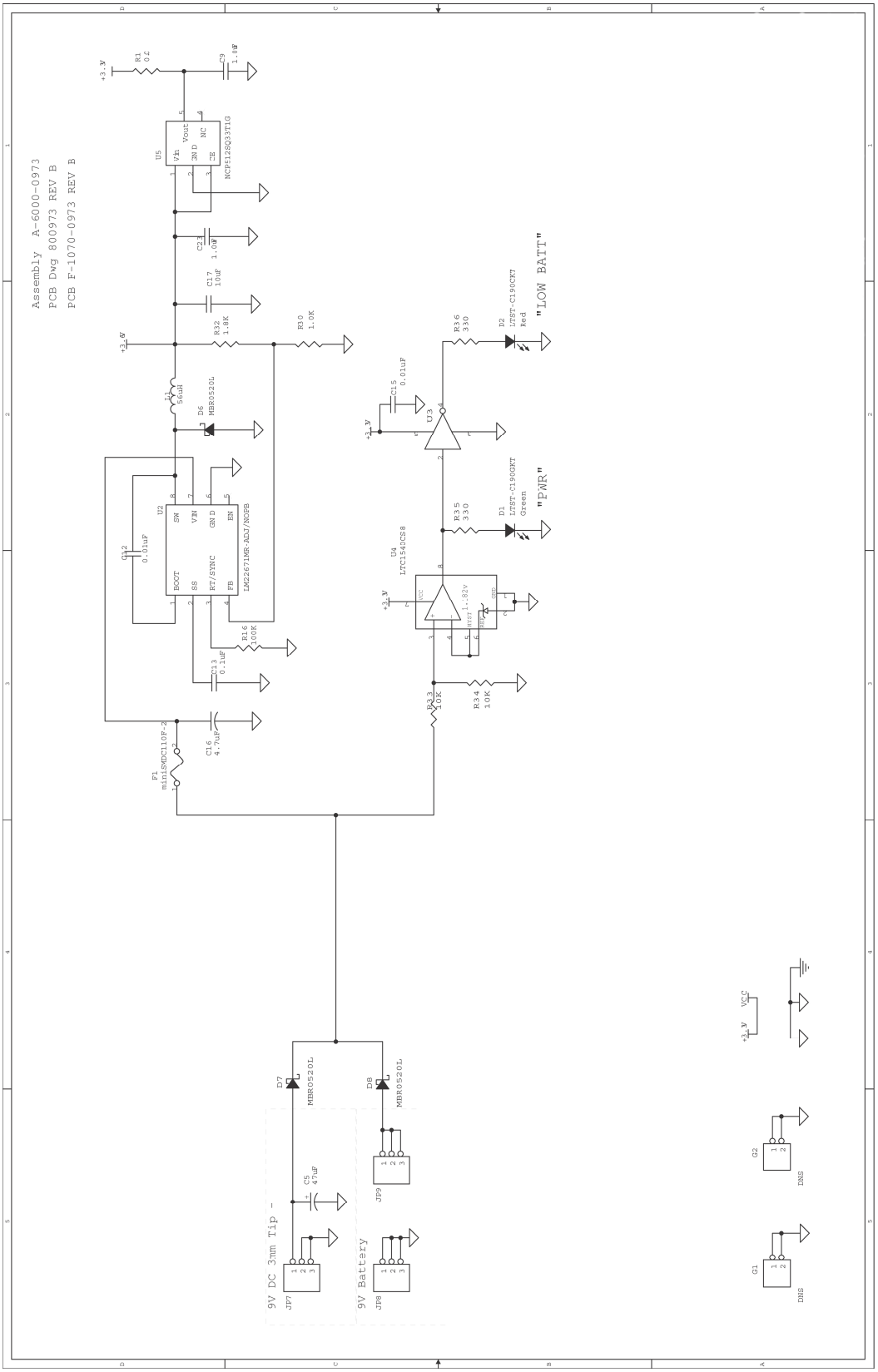


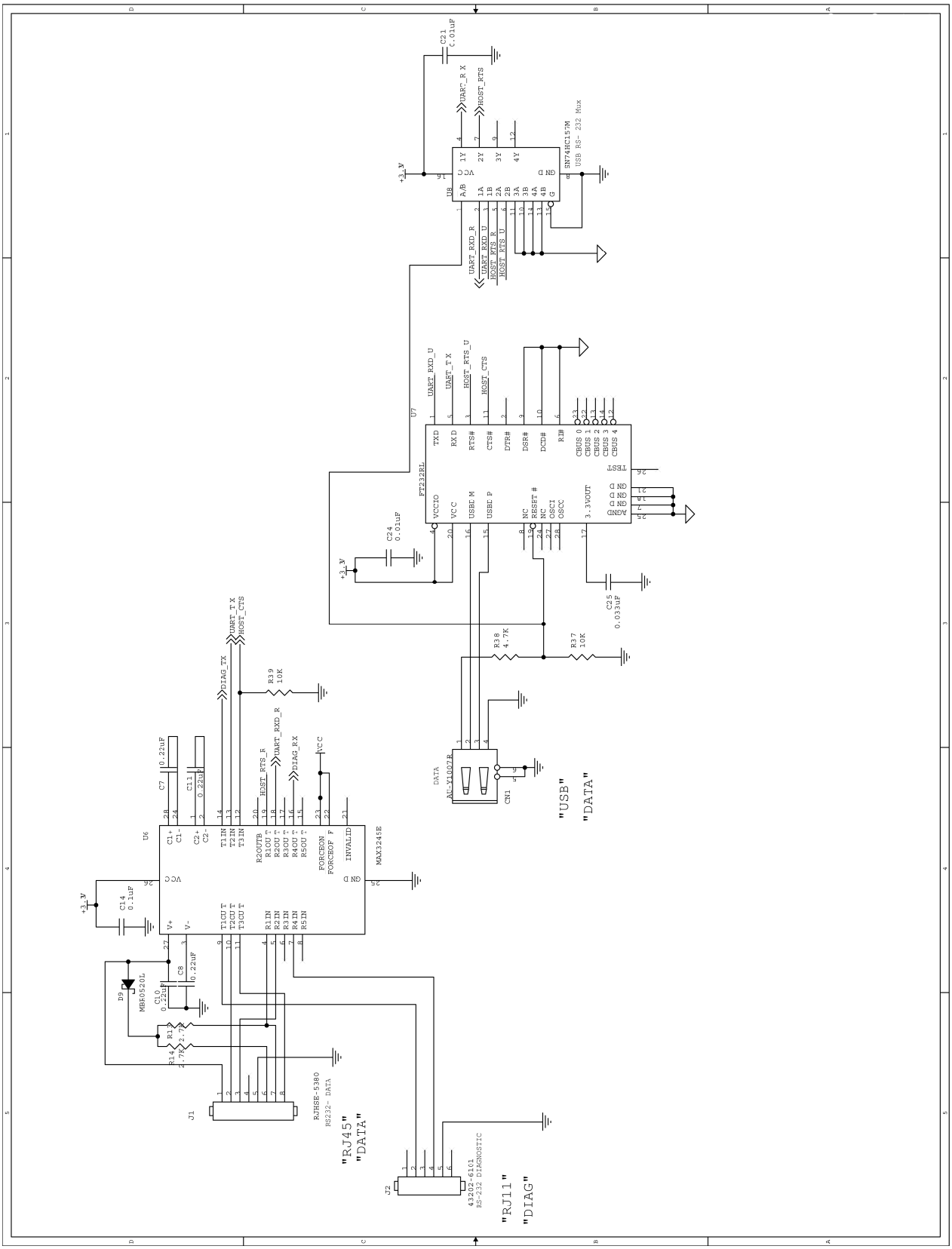
Figure 10.3.8

10.4 WSN802G Developer Board Schematic



Technical support +1.678.684.2000
E-mail: tech_sup@rfm.com


Assembly A-6000-0973
PCB Dwg 800973 REV B
PCB F-1070-0973 REV B
"LOW BATT"
"PWR"
9V DC 3mm Tip -
9V Battery
MBR0520L
LM2267-ADJ/NOPB
LTC1540CS8
NCP5142SQ33T1G
LMST-C190CK7
LMST-C190GK7
MBR0520L
DNS
DNS


"USB"
"DATA"
"RJ45"
"DATA"
"RJ11"
"DIAG"

11.0 Warranty

Seller warrants solely to Buyer that the goods delivered hereunder shall be free from defects in materials and workmanship, when given normal, proper and intended usage, for twelve (12) months from the date of delivery to Buyer. Seller agrees to repair or replace at its option and without cost to Buyer all defective goods sold hereunder, provided that Buyer has given Seller written notice of such warranty claim within such warranty period. All goods returned to Seller for repair or replacement must be sent freight prepaid to Seller's plant, provided that Buyer first obtain from Seller a Return Goods Authorization before any such return. Seller shall have no obligation to make repairs or replacements which are required by normal wear and tear, or which result, in whole or in part, from catastrophe, fault or negligence of Buyer, or from improper or unauthorized use of the goods, or use of the goods in a manner for which they are not designed, or by causes external to the goods such as, but not limited to, power failure. No suit or action shall be brought against Seller more than twelve (12) months after the related cause of action has occurred. Buyer has not relied and shall not rely on any oral representation regarding the goods sold hereunder, and any oral representation shall not bind Seller and shall not be a part of any warranty.

THE PROVISIONS OF THE FOREGOING WARRANTY ARE IN LIEU OF ANY OTHER WARRANTY, WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL (INCLUDING ANY WARRANTY OR MERCHANT ABILITY OR FITNESS FOR A PARTICULAR PURPOSE). SELLER'S LIABILITY ARISING OUT OF THE MANUFACTURE, SALE OR SUPPLYING OF THE GOODS OR THEIR USE OR DISPOSITION, WHETHER BASED UPON WARRANTY, CONTRACT, TORT OR OTHERWISE, SHALL NOT EXCEED THE ACTUAL PURCHASE PRICE PAID BY BUYER FOR THE GOODS. IN NO EVENT SHALL SELLER BE LIABLE TO BUYER OR ANY OTHER PERSON OR ENTITY FOR SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES, INCLUDING, BUT NOT LIMITED TO, LOSS OF PROFITS, LOSS OF DATA OR LOSS OF USE DAMAGES ARISING OUT OF THE MANUFACTURE, SALE OR SUPPLYING OF THE GOODS. THE FOREGOING WARRANTY EXTENDS TO BUYER ONLY AND SHALL NOT BE APPLICABLE TO ANY OTHER PERSON OR ENTITY INCLUDING, WITHOUT LIMITATION, CUSTOMERS OF BUYERS.

Part # M-0802-1002, Rev F