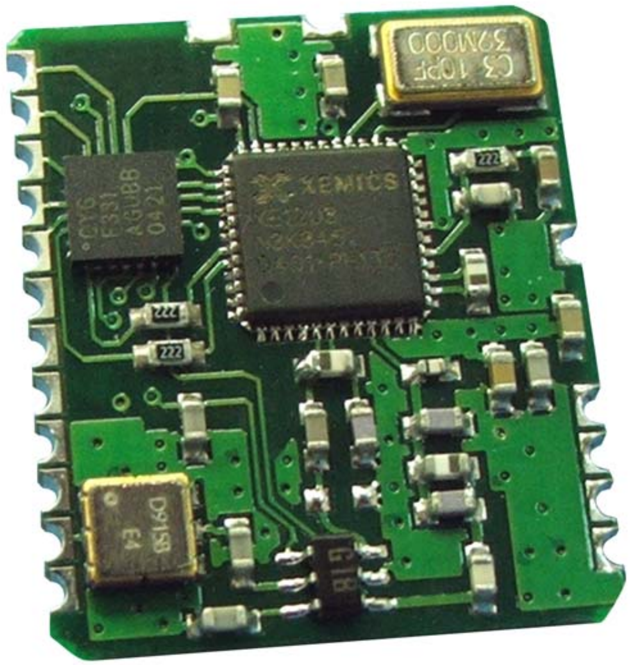

Wi.232DTS User's Manual

U.S 902-928MHz ISM Band Version

Rev 1.4.1

905 Messenger Lane
Moore, OK 73160
405-794-7730

1. Document Control

Created By	Steve Montgomery	12/9/03
Engineering Review		
Marketing Review		
Approved - Engineering		
Approved - Marketing		

Revision	Author	Date	Description
1.0	SJM	12/9/2003	Document Created
1.2	TRM / GWH	5/12/2004	Various corrections including register addresses
1.3	TRM	6/2/2004	Corrected programming omission regarding esc chars
1.3.1	TRM	6/3/2004	Corrected MAC/OUI and NVSLPMODE register errors
1.4.0	TRM	6/27/2004	Corrected inconsistencies in register tables. Altered encoding algorithm to return size and make resultant string null-terminated. Corrected error in Pin Description section. Added defaults to NV register table. Reorganized sections for better readability. Added flash and command timing information.
1.4.1	TRM	3/15/2005	Corrected error in register summary table regarding MAC address. Renamed OUI2-0/MAC2-0 to MAC5-0. Updated sales information.

2. Introduction

2.1. Module Overview

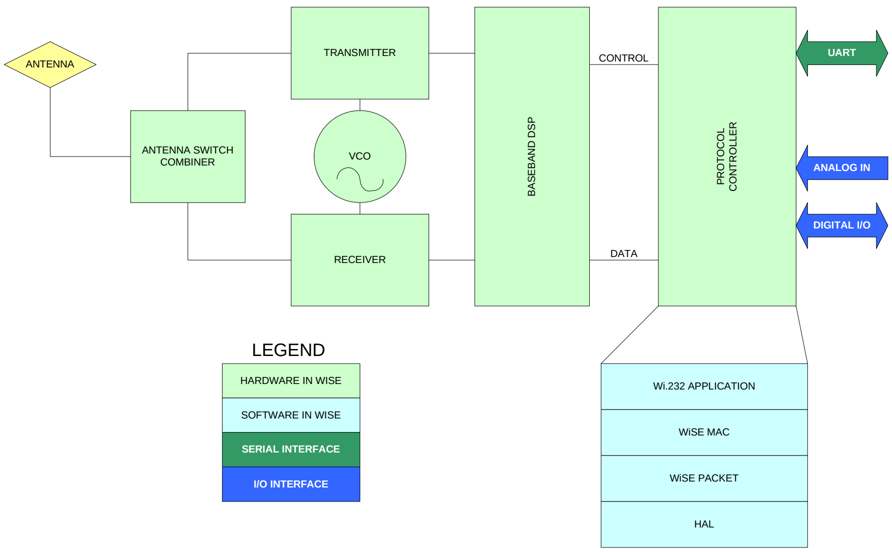

Figure 1: Wi.232DTS Block Diagram

2.2. Features

- True UART to antenna solution
- 16-bit CRC error checking
- 152.34kbit/sec maximum RF data rate
- 32 channels in DTS mode
- 84 channels in low-power mode
- Small size – .8" x .935" .08"
- Low power standby and sleep modes
- PHY and MAC layer protocol built in
- CSMA medium access control
- 115dB link budget in DTS mode
- 4 modes allow user to optimize power/range
- Command mode for volatile and non-volatile configuration
- 48-bit unique MAC address
- 5 volt tolerant I/O
- Under $20 in production quantities
- *868MHz European version available*

2.3. Applications

- Direct RS-232/422/485 wire replacement (requires external RS-232 to 3V CMOS conversion circuitry)
- Asset Tracking
- Automated Meter Reading
- Industrial/Home Automation
- RFID
- Wireless Sensors
- Remote Data Logging

3. Table of Contents

4. Table of Figures

5. Index of Tables

6. Theory of Operation

6.1. General

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

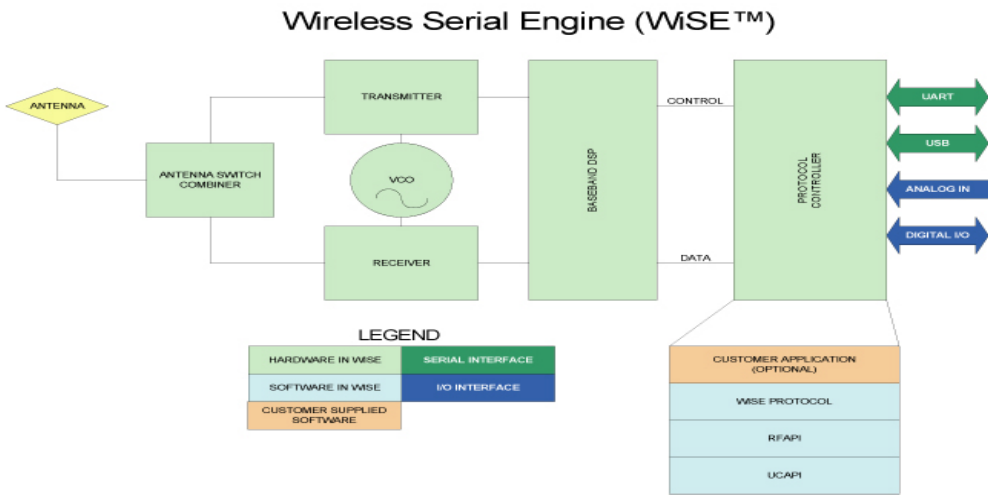

Figure 2: WiSE Block Diagram

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

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

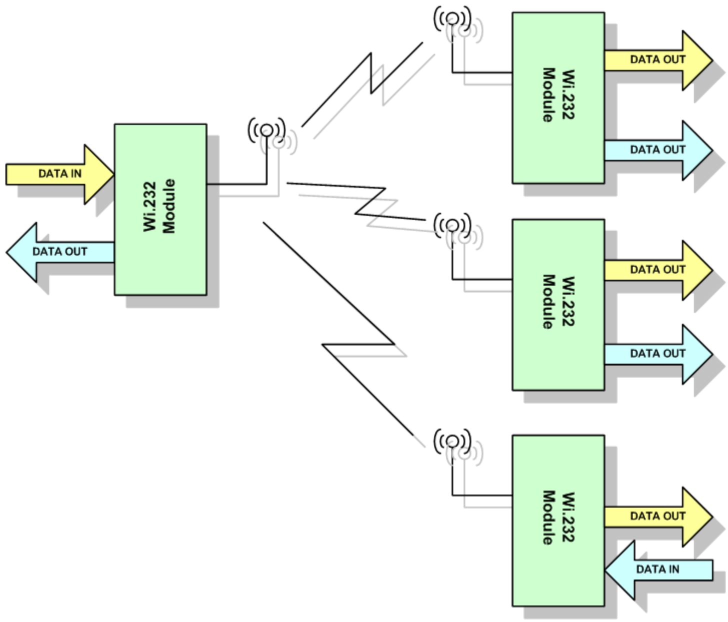

Figure 3: Wi.232DTS Networking Concept

The module is designed to interface directly to a host UART. Three signals are used to transfer data between the module and the host UART: **TXD**, **RXD**, and **CTS**. **TXD** is the data output from the module **RXD** is the data input to the module. **CTS** is an output that indicates the status of the module's data interface. If **CTS** is low, the module is ready to accept data. If **CTS** is high, the module is busy and the host UART should not send any further data.

Internally, the module has a 192 byte buffer for incoming characters from the host UART. The module can be programmed to automatically transmit when the buffer reaches a programmed limit, set by ***regUARTMTU***. The module can also be programmed to transmit based on

a delay between characters, set by ***regTXTO***(set in 1mSec increments). These registers allow the designer to optimize performance of the module for fixed length and variable length data. The module will support streaming data, as well. To optimize the module for streaming data***,*** ***regUARTMTU*** should be set to 128, and ***regTXTO*** should be set to a value equal to 1 byte time at the current UART data rate. If the buffer is full, or the timer set by ***regTXTO*** expires, and the module is in the process of sending the previous packet over the RF link, the module will assert **CTS** high, indicating that the host should not send any more data. Data sent by the host while **CTS** is high will be lost.

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

The MAC layer prefixes the data with a packet header and postfixes the data with a 16-bit CRC. The 16-bit CRC error checking can be disabled to allow the application to do its own error checking. Data is encoded using a proprietary algorithm (DirectSPREAD™) to spread the RF energy equally within the transmission bandwidth.

Modules can operate in groups. Each module can be assigned an 8-bit group ID, which is used to logically link it to other modules on the same channel. All modules on a channel will interoperate, regardless of their respective group Ids. In other words, the CSMA mechanism will prevent collisions of modules on the same channel but belonging to different groups.

Modules can also operate in two network modes: Master/Slave and Peer-to-Peer. These modes define a set of communication rules that identifies which modules can talk to any given module. In Master/Slave mode, masters can talk to slaves and other masters, slaves can talk to masters, but slaves cannot talk to other slaves. This mode is sometimes required for applications that are replacing legacy RS-485 networks. In peer-to-peer mode, any module can hear any other module. In both modes, group integrity is enforced.

When a module transmits a packet, all other modules on the same channel will receive the packet, check the packet for errors, and determine whether the received group ID matches the local group ID. If the packet is error free and the group Ids match, the module will decrypt the data if necessary, and send the error free data to its host UART for processing. The modules only implement the ISO reference network stack up to the MAC layer, so they are transparent to link layer addressing schemes. Therefore, the modules can work with any link-layer and higher protocols in existing today.

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

The UART interface is capable of operating in full duplex at baud rates from 2.4 to 115.2 kbps.

The module has six power modes: High DTS, Medium DTS, Low DTS, low power, standby, and sleep.

The Wi.232DTS module is the first module in the world to take advantage of the digital spread spectrum provision in FCC part 15 rules. Under this provision, transmitters can operate at a higher output power if the transmission bandwidth is at least 500kHz. Through an encoding technique we call DirectSPREAD™, the Wi.232DTS module is able to operate at +11dBm and meet the requirements of this provision.

In DTS mode, the module’s channel bandwidth is set to 600kHz and the transmit power is set to +11dBm. In this mode, the module can operate on 32 channels and support a maximum RF data rate of 152.34kbit/second. The receiver sensitivity at the max data rate is –100dBm typical, yielding a link budget of 111dB. This mode is an excellent alternative to frequency hopping spread spectrum. It has very fast synchronization, allowing it to operate in a duty-cycle mode for extended battery life.

In low-power mode the module’s channel bandwidth is set to 200kHz. In this mode the module can operate on 84 channels and support a maximum data rate of 38.4 kbit/second. The receiver sensitivity at the maximum data rate is –105 typical, yielding a link budget of 105dB. This mode reduces transmit current consumption, allowing use with batteries that cannot supply the pulse currents required for DTS mode. The range in this mode will be a little more than half of the range in DTS mode.

The module can be placed into sleep mode through the command mode. In sleep mode, the RF section is completely shutdown, and the protocol processor is in an idle state. Once the module has been placed in the sleep mode, it can be awakened by either cycling power, which will loose all volatile settings, or by sending a power-up sequence through the serial port. The power up sequence is a combination of four 0xFF bytes sent back-to-back at the data rate for which the module is configured.

***Note**: When in sleep mode, the module will not be able to receive data from other modules. Any data sent to the module while it is in sleep mode will be lost.*

If the current draw in sleep mode is too high for a particular application, the designer can switch power to the module through a FET to "turn-off" the module when it is not needed. If this technique is used, the volatile registers will reset to the values in their non-volatile mirrors, so any changes from the default will have to be reloaded.

The Wi.232DTS is a very flexible module because of all of the configurable parameters it supports. However, modules that are not configured in the same way will not be able to communicate reliably, causing poor performance or outright failure of the wireless link. **All modules in a network must have the same mode configuration to ensure interoperability.**

Every Wi.232 module has read-only internal registers that contain factory programmed information that includes calibration data and a 48-bit MAC address that can be used by the host application for higher level, connection oriented protocols. This MAC address can be read through the command interface.

6.2. Operating States

The primary active state is the IDLE state. When the module is not actively transmitting or receiving data, it is in this state. While in this state, the receiver is enabled and the module is continuously listening for incoming data. If the module detects a pre-amble and valid start-code, it will enter the RX_HEADER state.

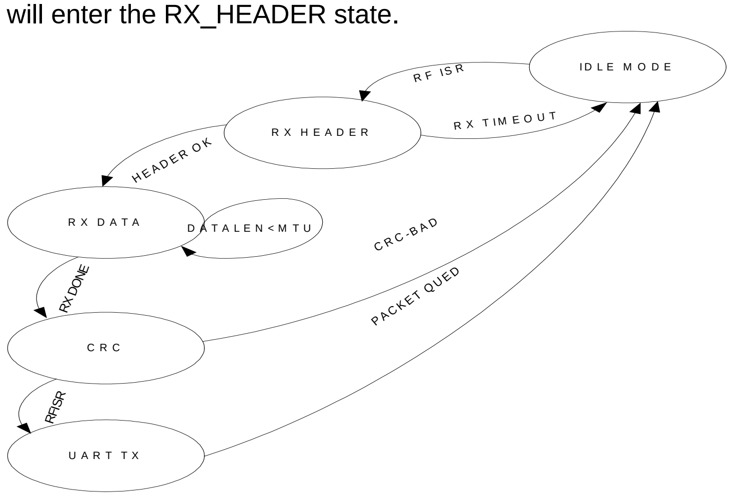

Figure 4: RX State Machine

If the module is in the IDLE state and a byte is received by the UART, it will enter the TX_WAIT state.

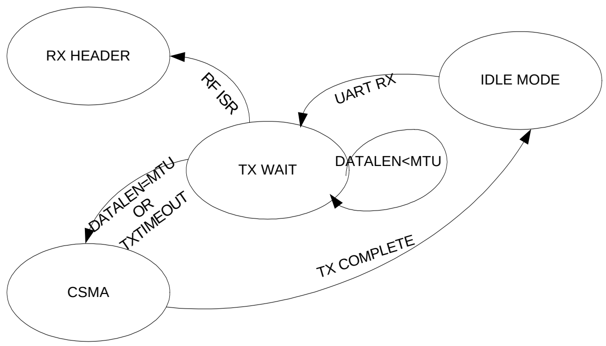

Figure 5: TX State Machine

6.3. Resetting Module to Factory Defaults

It may be necessary to reset the non-volatile registers to their factory defaults. To reset the module to factory defaults, hold the command line low and cycle power to the module. The command line must remain low for a minimum of 450ms after the resetting the module. Once the command line is released, the module's non-volatile registers will be reset to factory defaults.

7. Application Information

7.1. Pin-out Diagram

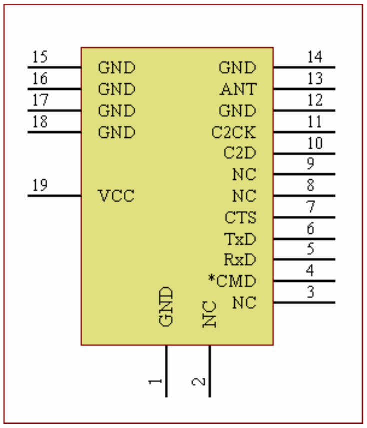

Figure 6: Pin-out diagram

7.2. Pin Description

No.	Description
1	Ground
2	No connect – reserved
3	No connect – reserved
4	Command input – active low
5	UART receive input
6	UART transmit output
7	UART clear to send output – active low
8	No connect – reserved
9	No connect – reserved
10	Reserved – ISP pin
11	Reserved – ISP pin
12	Ground
13	Antenna port – 50 ohm
14	Ground
15	Ground
16	Ground
17	Ground
18	Ground
19	VCC – 2.7 to 3.6 VDC

Table 1, Module Pin Descriptions

Legend

(yellow)	Signals that are used in this implementation
(green)	Signals not used in this implementation –do not connect
(orange)	Signals used for in-system programming

7.3. Mechanical Drawings

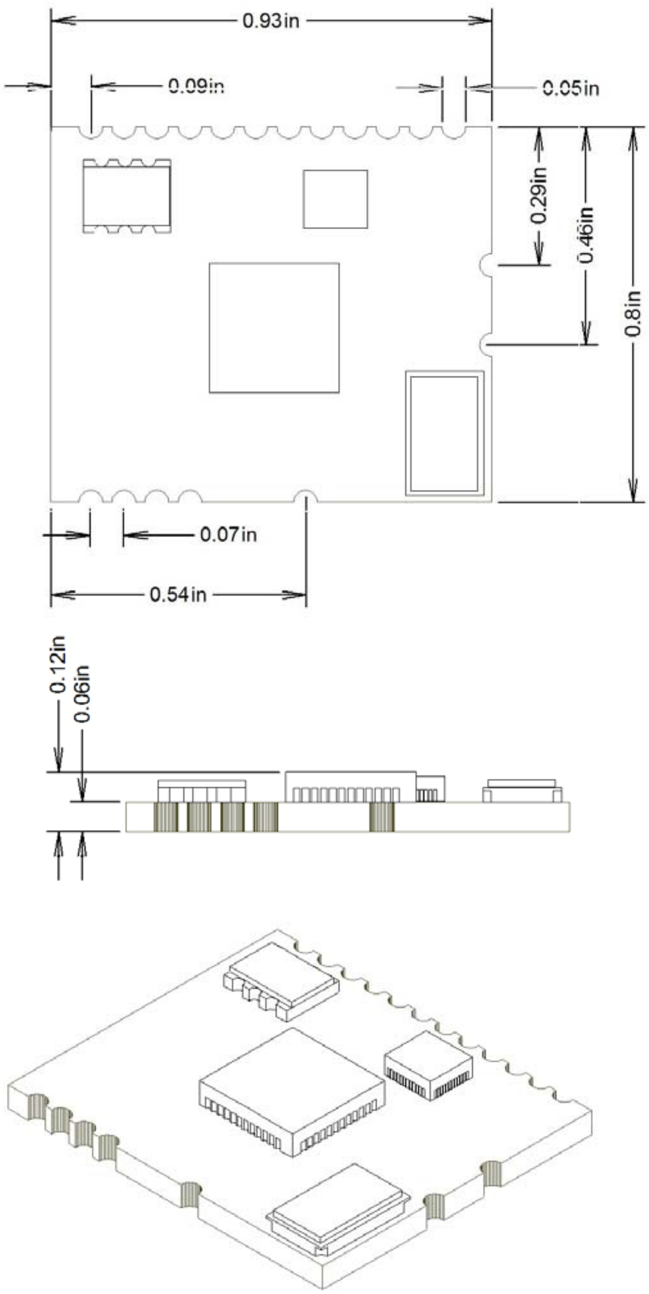

Figure 7: Module Mechanical Drawings

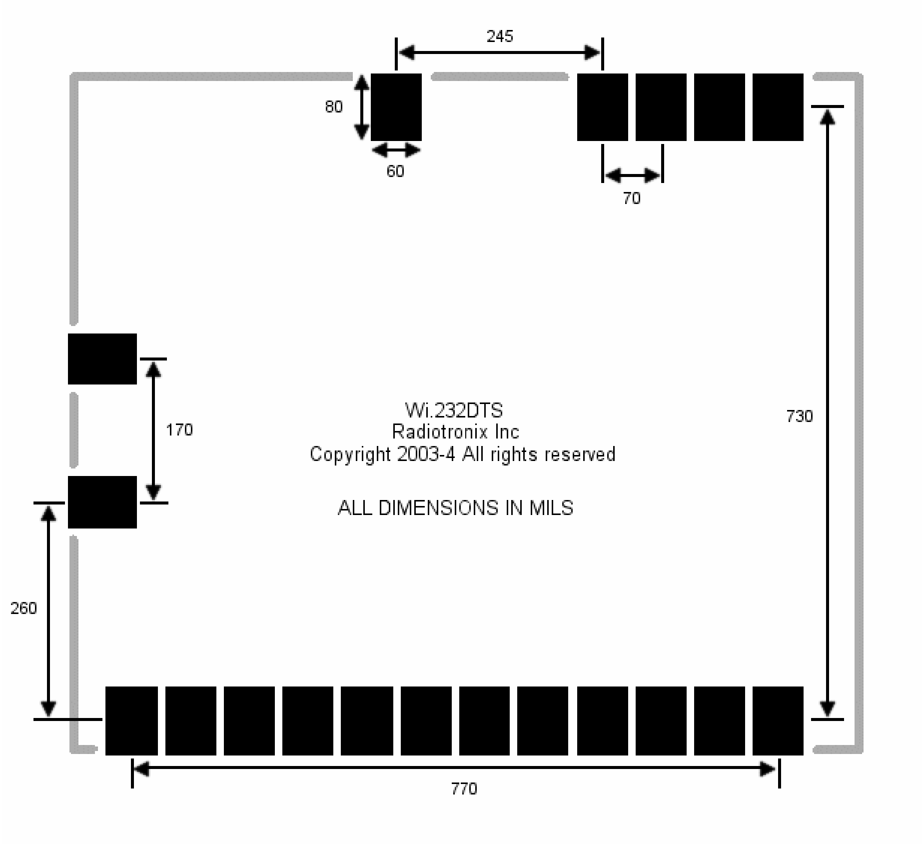

Figure 8: Wi.232DTS Suggested Footprint

7.4. Example Circuit

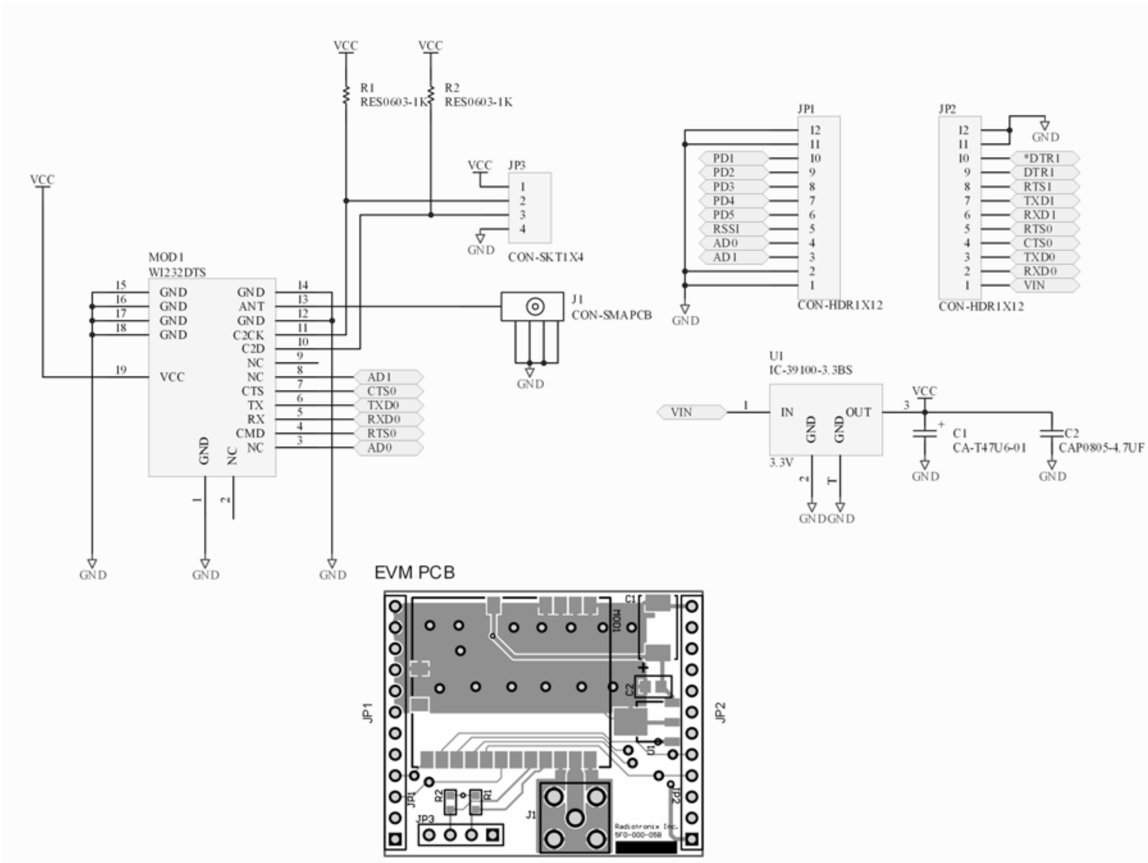

Figure 9: Evaluation Module Circuit

7.5. Power Supply

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

Figure 4 shows the schematic for our evaluation module circuit for the Wi.232DTS module. It includes an on-board power supply and antenna connector. This evaluation circuit was used to measure the performance of the Wi.232DTS module, and should be used as a reference for Wi.232DTS based designs.

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

7.6. UART Interface

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

Line	Direction	Description
CTS	Out	Clear to send – this pin indicates to the host micro when it is ok to send data. When CTS is high, the host micro should stop sending data to the module until CTS returns to the low state.
CMD	In	Command – the host micro will bring this pin low to put the module in command mode. Command mode is used to set and read the internal registers that control the operation of the module. When CMD is high, the module will transparently transfer data to and from other modules on the same channel. NOTE: If this pin is low when the module comes out of reset, the registers will be reset to their factory programmed defaults. It is important to ensure that CMD is held high during power-up under normal conditions.
RXD	In	Receive data input.
TXD	Out	Transmit data output

Table 2, Wi.232DTS UART Interface Lines

7.7. Antenna

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

We are often asked: " What is the best antenna to use with your module?" Actually, the selection of an antenna is based on a particular application, not the module used.

As a rule, a ¼ wave whip antenna with a good, solid ground plane is the best choice. However, many embedded applications cannot support an externally mounted antenna. If this is the case, a PCB antenna must be used. The designer can either use an off-of-the-shelf PCB antenna, such as the Splatch from Linx Technologies, or design a trace antenna. There are several good antenna tutorials and references on the Internet and we encourage the designer to use these resources.

***Note**: Antenna design is difficult and can be impossible without the proper test equipment. As such, we strongly encourage all of our customers to use off-of-the-shelf antennas whenever possible.*

7.8. Link budget, transmit power, and range performance

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

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

Wireless Fact: *You will never reduce the performance of a wireless link by increasing the sensitivity*.

It has been proposed that less sensitive receivers will perform better in a noisy environment. That simply is not true. It is the equivalent of saying that someone who is hard of hearing can hear better in a noisy room than in a quiet room. The real solution is to make the talker speak louder to get over the noise in the room. The same is true for a wireless link. In real-world, noisy environments, increased output power is generally the best way to improve range performance.

The transmit power on unlicensed devices is regulated by the FCC. For transmitters that are not spread-spectrum, the output power is limited to 0dBm (1mW) when a standard ¼ wave whip antenna is used. If the transmitter operates under the spread-spectrum rules, however, the transmit power can be increased; up to 1W depending on the spread-spectrum technique and antennas that are used.

Wireless Fact*: Frequency hopping spread spectrum does not effectively combat multipath in the 902-928 MHz band. It does combat in-channel interference, but at the expense of bandwidth, power consumption, and latency. Direct sequence spread spectrum, like FHSS, does not combat multipath. It does do a better job than FHSS at combating in-channel interference, but at the expensive of occupied bandwidth and power consumption. These spread spectrum techniques are generally chosen because the FCC will allow higher output power from a transmitter employing these techniques. Recently, the FCC rules changed to include a new type of spread spectrum device, call digital transmission system (DTS). This method of spread spectrum has no processing gain, but allows lower cost solutions like the Wi.232DTS to transmit with higher output power.*

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

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

For example, the link budget for a pair of Wi.232 modules in DTS mode at the maximum data rate and using 3dBi whips antennas would be:

+11dBm + 3dB – (-100dBm) + 3dB = 117dB

A link budget of 117dB should easily yield a range of ¼ mile or more outdoors. If the environment is open and the antennas are 8 to 10 feet off of the ground, the range could be a mile. Indoors, this link budget should yield a range of several hundred feet.

This is a well-balanced link budget. More than 10dB of the budget is achieved through transmit power, which will allow good performance indoors in the presence of multi-path while keeping the overall operating current low, making the module suitable for primary battery powered applications such as RFID and automated meter reading.

8. Module Configuration

8.1. Channel settings

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

regNVRXCHAN (0x01)					regRXCHAN (0x4C)		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RES	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

The Wi.232DTS supports 32 channels (0 – 31) in DTS mode and 84 channels (0 – 83) in low power mode.

Transmit and receive channels are set in ***regTXCHAN (addr 0x4B)*** and ***regRXCHAN (addr 0x4C)*** respectively.

When the module is in DTS mode, the channel registers are masked so that only the lower 6-bits determine the channel.

The following equations can be used to calculate transmit center frequency in LP and DTS modes.

$$Fc = 902.3 + chan * .3MHz(LP)$$
$$Fc = 903.0 + chan * .75MHz(DTS)$$

All modules in a network must be in the same mode (LP or DTS) and must have the same transmit and receive channels programmed in order to communicate properly.

8.2. Power Mode

The transmission and reception modes of the module are determined by the settings of the ***regPWRMODE*** register. It is important to note that a module configured to operate in LP mode cannot "hear" another module transmitting in DTS mode, or vice versa. However, a module configured to operate in any of the three DTS modes can "hear" any other module transmitting in any of the DTS modes (provided that they are within range of one another).

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

PM1	PM0	Mode
0	0	LP Mode – -5dBm power setting (typical)
0	1	DTS Mode – -1dBm power setting (typical)
1	0	DTS Mode – +2dBm power setting (typical)
1	1	DTS Mode – +11dBm power setting (typical)

Table 3, Power Mode Register Settings

8.2.1. DTS Mode

In DTS mode, the module is configured as follows:

DTS Mode Parameters	
TX Power	-1, +2, or +11 dBm
Deviation	+/-235kHz
TX Current	28 to 57mA
RX Current	20mA
RX Bandwidth	600kHz

Table 4, DTS Mode Parameters

8.2.2. Low power Mode

In low-power mode, the module is configured as follows:

LP Mode Parameters	
TX Power	-3.5dBm
Deviation	+/-50kHz
TX Current	24mA
RX Current	20mA
RX Bandwidth	200kHz

Table 5, LP Mode Parameters

8.3. UART Data Rate

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

By default, the UART data rate is set to 2.4 kbit/second at the factory. This data rate can be changed by setting the ***regDATARATE*** register. Valid settings are:

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

Table 6, Data Rate Register Settings

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

8.4. Network Mode

regNVNETMODE (0x04)					regNETMODE (0x4F)		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

The module supports two networking modes: Normal and Slave.

In normal mode, the module can talk to any other module. In slave mode, the module can talk to normal-mode modules, but cannot transmit to or receive from other slaves.

Slave mode is selected by writing 0x00 to this register. The default network mode is 0x01 (Normal Mode).

8.5. Transmit Wait Timeout

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

When a byte is received by the UART, the module will start a timer that will countdown every millisecond. The timer is restarted when each byte is received.

If the timer reaches zero before the next byte is received from the UART, the module begin transmitting the data in the buffer. Normally, this timeout value should be greater than 0x01 and greater than one byte time at the current UART data rate. If the timeout value is set to 0x00, the transmit wait timeout will not operate, and a full buffer will be required for transmission. When setup this way, the data will be sent only when a full MTU has been received through the UART. The default setting for this register is 0x10 (~16ms delay).

8.6. Network Group

regNVNETGRP (0x06)					regNETGRP (0x51)		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
B7	B6	B5	B4	B3	B2	B1	B0
7	6	5	4	3	2	1	0

Modules can be grouped into networks. Although only modules with the group ID will be able to talk to each other, modules in different groups but on the same channel will still coordinate transmissions through the CDMA mechanism. Valid values for this register are 0 to 127. The default group setting is 0.

8.7. CRC Control

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

Set to 0x01 to enable CRC mode, or 0x00 to disable CRC mode. The default CRC mode setting is enabled.

8.8. UART minimum transmission unit

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

This register determines the UART buffer level that will trigger the transmission of a packet. The minimum value is 1 and the maximum value is 128. The default value for this register is 64, which provides a good mix of throughput and latency.

8.9. Verbose mode

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

Setting this register to 0x00 will suppress the start-up message, including firmware version, that is sent to the UART when the module is reset. A value of 0x01 will cause the message to be displayed after reset. By default, the module start-up message will be displayed.

8.10. CSMA enable

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

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

8.11. Sleep control

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

Setting this register to 0x01 will place the module into sleep mode; 0x02 will place the module in standby mode. Sleep mode places the module in the lowest power inactive state (~35µA) and requires approximately 7-8ms to resume transmission or reception once awakened. Standby draws ~850µA and requires approximately 1-2ms to awaken. To wake up the module, send four 0xFF bytes to the UART in a row or perform a hard reset. If four 0xFF bytes are used to wake the module, the fifth character sent to the UART will be transmitted over the RF link. Upon awakening, the module will clear the volatile register to 0x00. The default value for this register is 0x00 (awake).

8.12. MAC Address

regMAC5 – regMAC0 (0x22 – 0x27)					N/A		
R	R	R	R	R	R	R	R
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

These registers make a unique 48-bit MAC address. These values are factory preset and cannot be altered. These address bytes are not used by the module. They are provided for customer applications as a unique address.

8.13. Register Summary

Volatile Read/Write Registers			
Name	**Address**	**Description**	
regTXCHANNEL	0x4B	Transmit channel setting	
regRXCHANNEL	0x4C	Receive channel setting	
regPWRMODE	0x4D	Operating mode settings	
regDATARATE	0x4E	UART data rate	
regNETMODE	0x4F	Network mode (Normal or Slave)	
regTXTO	0x50	Transmit wait timeout	
regNETGRP	0x51	Network group ID	
regUSECRC	0x53	Enable/Disable CRC	
regUARTMTU	0x54	Minimum transmission unit.	
regSHOWVER	0x55	Enable/disable start-up message	
regCSMAMODE	0x56	Enable/disable CSMA	
regSLPMODE	0x58	Power state of module	
Non-volatile Read Only Registers			
Name	**Address**	**Description**	
regMAC5	0x22	These registers form the unique 48-bit MAC address.	
regMAC4	0x23		
regMAC3	0x24		
regMAC2	0x25		
regMAC1	0x26		
regMAC0	0x27		
Non-volatile Registers			
Name	**Address**	**Description**	**Default**
regNVTXCHANNEL	0x00	Transmit channel setting	16
regNVRXCHANNEL	0x01	Receive channel setting	16
regNVPWRMODE	0x02	Operating mode settings	-1 dBm DTS mode
regNVDATARATE	0x03	UART data rate	2400bps
regNVNETMODE	0x04	Network mode (Normal/Slave)	Normal
regNVTXTO	0x05	Transmit wait timeout	~16ms
regNVNETGRP	0x06	Network group ID	0x00
regNVUSECRC	0x08	Enable/Disable CRC	Enabled
regNVUARTMTU	0x09	Minimum transmission unit.	64 bytes
regNVSHOWVER	0x0A	Enable/Disable start-up message	Enabled
regNVCSMAMODE	0x0B	Enable/Disable CSMA	Enabled
regNVSLPMODE	0x0D	Power state of module	Awake

Table 7, Register Summary

9. Using Configuration Registers

9.1. CMD Pin

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

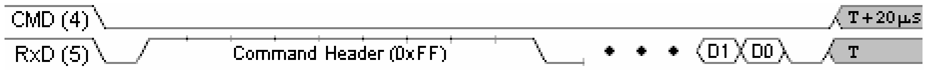

Figure 10: Command and CMD Pin Timing

9.2. Command Formatting

The Wi.232DTS module contains several volatile and non-volatile registers that control its configuration and operation. The volatile registers all have a non-volatile mirror register that is used to determine the default configuration when power is applied to the module. During normal operation, the volatile registers are used to control the module.

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

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

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

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

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

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

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

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

Figure 11: Command Conversion Code

9.3. Writing To Registers

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

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

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

Table 8, Write Register Command, value to be written is less than 128 (0x80).

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

Table 9, Write Register Command, value to be written is greater than or equal to 128 (0x80).

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

9.4. Reading From Registers

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

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

Table 10, Read Register Command

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

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

Table 11, Read Register Module Response For A Valid Register

10. Electrical Specifications

10.1. Absolute Maximum Ratings

Parameter	Min	Max	Units
VCC – Power Supply	-0.3	5.0	VDC
Voltage on any pin	-0.3	5.2	VDC
Input RF Level		15	dBm
Storage Temperature	-40	85	°C

Table 12, Absolute Maximum Ratings

10.2. Detailed Electrical Specifications

10.2.1. AC Specifications – RX

Parameter	Min	Typ	Max	Units	Notes
Receive frequency - US	902.2		927.8	MHz	At antenna pin
Channels – DTS		32			
Channels – LP Mode		84			
Channel spacing – DTS Mode		750		kHz	
Channel spacing – LP Mode		300		kHz	
Receiver sensitivity – DTS MODE		-100		dBm	152.34 kbit/sec
Receiver sensitivity – DTS MODE		-102		dBm	38.4 kbit/sec
Receiver sensitivity – DTS MODE		-104		dBm	9.6 kbit/sec
Receiver sensitivity – LP MODE		-104		dBm	38.4 kbit/sec
Receiver sensitivity – LP MODE		-105		dBm	9.6 kbit/sec
Input IP3		-40		dBm	Flo+1MHz and Flo+1.945MHz
Input Impedance		50		Ohms	No matching required
LO Leakage		-65		dBm	50-ohm termination at ANT
Adjacent channel rejection		-48		dBc dBc	Fc +/-650kHz
IF Bandwidth – DTS Mode		600		KHz	
IF Bandwidth – LP Mode		200		KHz	

Table 13, AC Specifications - Rx

10.2.2. AC Specifications – TX

Parameter	Min	Typ	Max	Units	Notes
Transmit Frequency –US	902.2		927.8	MHz	
Center frequency error		2	3	ppm	915 MHz @ 25°C
Frequency Deviation – DTS Mode		+/-235		kHz	
Frequency Deviation – LP Mode		+/-50		kHz	
Maximum Output Power – LP Mode		-5	0	dBm	915 MHz Into 50 ohm load
Maximum Output Power – DTS Mode		11	14	dBm	915 MHz Into 50 ohm load
Output Impedance		50		Ohms	
Carrier phase noise		TBD		dBc	Into 50 ohm load
Harmonic Output		-50		dBc	Into 50 ohm load

Table 14, AC Specifications - Tx

10.2.3. DC Specifications

Parameter	Min	Typ.	Max	Units	Notes
Operating Temperature	-20		+70	°C	
Supply voltage	2.7	3.0	3.6	VDC	Operating limits
Receive current consumption		20		mA	Continuous operation, Vdd = 3.3VDC
Transmit current consumption LP Mode – -5 dBm DTS Mode – -1 dBm DTS Mode – 2 dBm DTS Mode – 11 dBm		 28 32 39 57		 mA mA mA mA	 Output into 50 ohm load, Vdd = 3.3VDC
Standby current consumption		850		µA	Vdd = 3.3VDC
Sleep current consumption		35		µA	Vdd = 3.3VDC
Vih – Logic high level input Vil – Logic low level input Voh – Logic high level output Vol – Logic low level output	0.7*Vcc 0 2.5 0		5.2 0.3*Vc c Vcc .4	VDC VDC VDC VDC	

Table 15, DC Specifications

10.3. Flash Specifications (Non-Volatile Registers)

Parameter	Min	Typ.	Max	Units	Notes
Flash Write Duration		16	21	ms	Module stalled during write operation
Flash Write Cycles	20k	100k		Cycles	

Table 16, Flash Specifications (Non-Volatile Registers)

11. Custom Applications

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

12. Ordering Information

Wi.232DTS modules can be ordered on-line 24/7 from Mouser Electronics at www.mouser.com/radiotronix or Future Electronics at www.futureelectronics.com (p/n: **WI.232DTS**).

13. Contact Us

13.1. Technical Support

Radiotronix has built a solid technical support infrastructure so that you can get answers to your questions when you need them.

Our primary technical support tools are the support forum and knowledge base found on our website. We are continuously updating these tools. To find the latest information about these technical support tools, please visit http://www.radiotronix.com/support/.

Our technical support engineers are available Mon-Fri between 9:30 am and 4:30 pm central standard time. The best way to reach a technical support engineer is to send an email by visiting the Support page at http://www.radiotronix.com/support/ . E-mail support requests are given priority because we can handle them more efficiently that phone support requests.

For customers that would prefer to talk directly to a support engineer, we do offer phone support free of charge. All support requests are placed in a queue and returned in the order that they are received.

13.2. Sales Support

Our sales department can be reached via e-mail at sales@radiotronix.com or by phone at 405-794-7730.

Our sales department is available Mon-Fri between 8:30 am and 5:00 pm. Our modules can be purchased through distribution at Future Electronics (www.futureelectronics.com) or Mouser Electronics (www.mouser.com).

Radiotronix Inc.

905 Messenger Lane
Moore, OK 73160

(405) 794-7730
(405) 794-7477 (Fax)

www.radiotronix.com
sales@radiotronix.com