

## What is 'Easy-Radio'?



Easy-Radio modules combine low power radio transmitters, receivers or transceivers with on-board microcontrollers to produce 'intelligent' RF modules that provide simple to use wireless data links. These links can be used for On/Off control tasks or for sending and receiving serial data, in standard formats, to and from host systems.

The 'firmware' within the microcontroller is optimised to suit the exact characteristics of the radio device and there is no need to understand complex RF module parameters and all their implications. The task of encoding and decoding data in a suitable format for sending over a radio link is handled entirely within the device, as is error checking, that ensures the integrity of the messages.

Within the more sophisticated transceivers this firmware relieves the user of the complex mathematical task of programming synthesiser transmit or receive frequencies and also provides the complete management of the 'frequency hopping' versions of transceivers.

In addition key parameters such as frequency of operation, data rate and power output can be programmed during product manufacture/assembly and prior to final test. This allows modules to be 'tailored' for specific applications or regional markets (e.g. Europe/USA).

Easy-Radio is a complete set of services that is constantly being developed to simplify the use and application of low power radio links. Standard products provide for most requirements but the software can be customised for medium (100) to high volume (1000 +) customers.

This guide describes 'Easy-Radio' embedded software, the Easy-Radio Demonstration Board and its use with the device programming software. It should be read in conjunction with the data sheets on the specific devices (below) which are available from the Sales Office.

### Devices Covered

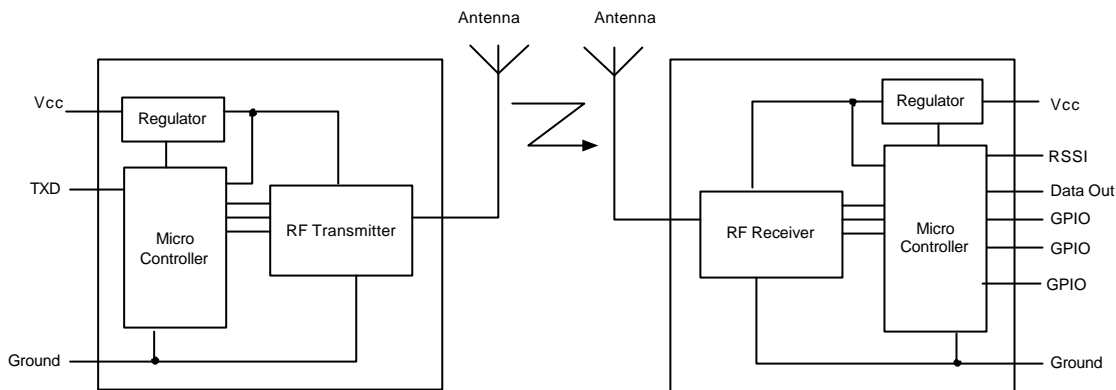
|            |                               |
|------------|-------------------------------|
| ERx00TS    | Transmitter                   |
| ERx00RS    | Receiver                      |
| ERx00TRS   | Transceiver                   |
| ERx00FHTRS | Frequency Hopping Transceiver |

Where 'x00' denotes frequency of operation. E.g. 400 = 433MHz

### 'One Way' Easy-Radio for Transmitters & Receivers

'One Way' Easy-Radio modules use FM (FSK) transmitters and receivers in combination with on-board microcontrollers and a voltage regulator to provide simplex (one way) wireless data links. Figure 1 shows the block diagram for these modules.

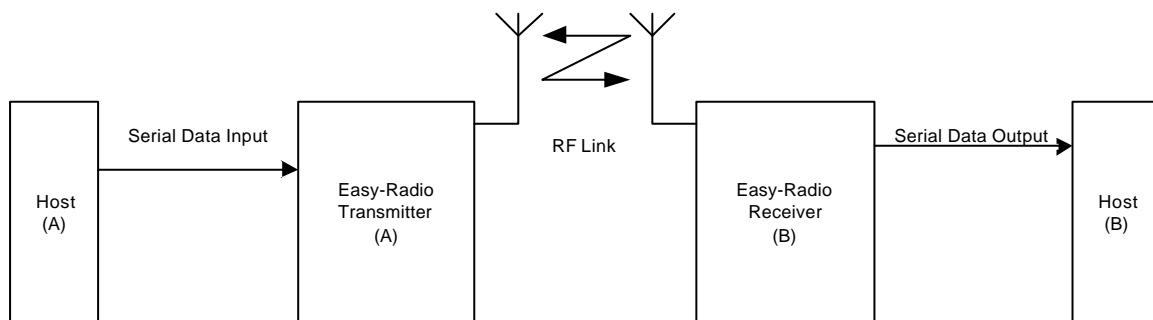
In addition to the usual supply (Vcc) and the 0 Volt (Ground) pins the transmitter has a Transmit Data (TXD) input and the receiver has a Data Out pin, a Received Signal Strength Indicator (RSSI) output and three configurable General Purpose Input/Output (GPIO) pins. Pin out details (numbering etc.) are provided in the specific data sheets.



**Figure 1 Transmitter & Receiver Block Diagram**

### Operation – 'RS232' Serial Data In & Out

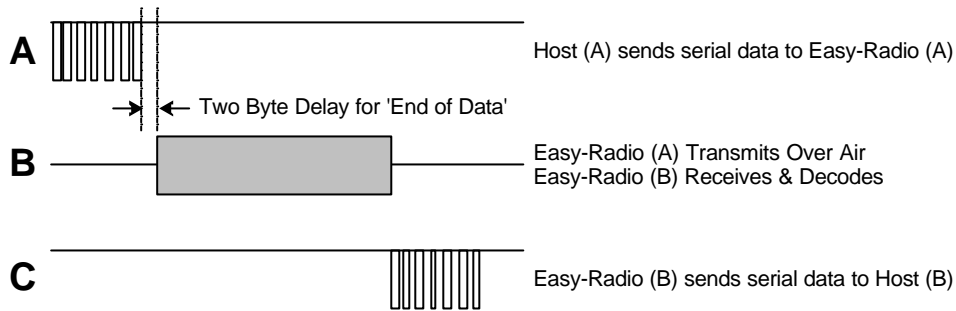
Figure 2 shows a typical system block diagram comprising hosts (user's application) connected to Easy-Radio Transmitters and Receivers. The host (A) will be monitoring (collecting data) and the host (B) will be sending data to some real world application.



**Figure 2 Host Connection**

The Host (A) should provide the serial data input (up to 128 characters per packet) to the Easy-Radio transmitter. The data should be sent in 'bursts' therefore allowing adequate time for transmission and reception over the RF link (See Figure 3). The receiver, upon reception and decoding of the RF transmission immediately sends serial data to the Host B.

Data is sent and received in standard 'RS232' serial format (logic level only) and there is no restriction on the characters that may be sent.



**Figure 3 Serial Data**

- A. Host (A) sends serial data to the Easy-Radio Transmitter (A). The data must be continuously streamed at the selected baud rate and it fills an internal transmit buffer until either 128 bytes have been received or a gap of two bytes is detected.
- B. After detecting either the 'End of Data' gap or the 'Buffer Full' condition the controller enables RF transmit and sends a fixed preamble followed by the data in the buffer which is Manchester encoded at 19,200 Baud for efficient transmission across the RF link. Any Easy-Radio receivers within range that 'hear' the transmission will simultaneously lock onto the preamble, decode the data and place it into their receive buffers.
- C. After checking the data for integrity the Data within the receive buffer of Easy-Radio Receiver (B) is then sent continuously to the host at the selected baud rate.

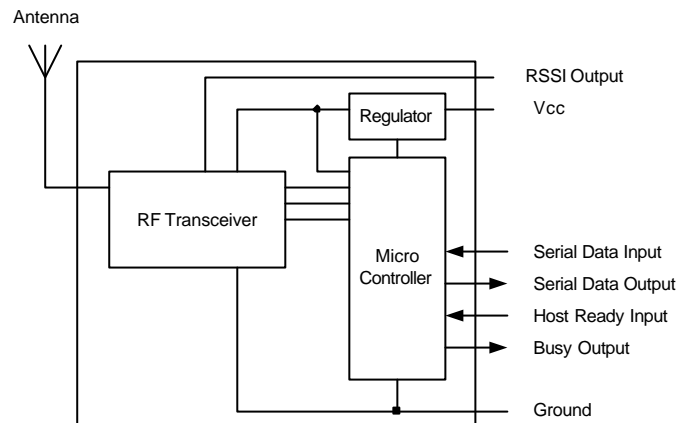
There is no 'handshaking' provided at either the transmitter or receiver. The user should therefore ensure that sufficient time is allowed for the completion of transmission and reception of data. The Timing Specifications detail these requirements<sup>1</sup>. Transmitter Host (A) must allow time for the 'Over Air' transmission and for the receiving Host (B) to unload (and process) the data before sending any more new data. The receiver Host (B) must always be 'ready and waiting' for data to arrive. It should be possible to use

With such a 'one-way' (simplex) system there is no confirmation of the satisfactory reception of the data and for added reliability it is recommended that the data be sent, perhaps, repetitively several times. For increased reliability the use of transceivers (which can acknowledge packet reception) is recommended. Easy-Radio services do not provide automatic acknowledgement (or re-tries) but these can be provided by the users application.

<sup>1</sup> For Timing Specifications see Page 6

## 'Two Way' Easy-Radio for Transceivers

Easy-Radio Transceivers are complete sub-systems that combine a high performance very low power RF FSK (FM) transceiver, a microcontroller and a voltage regulator. (Figure 2) The microcontroller programmes the functions of the RF transceiver and provides the interface to the host system via a serial input/output. It also contains programmable EEPROM memory that can hold configuration data for the various transceiver operating modes. A Received Signal Strength Indicator (RSSI) output can be optionally used to measure received signal levels.



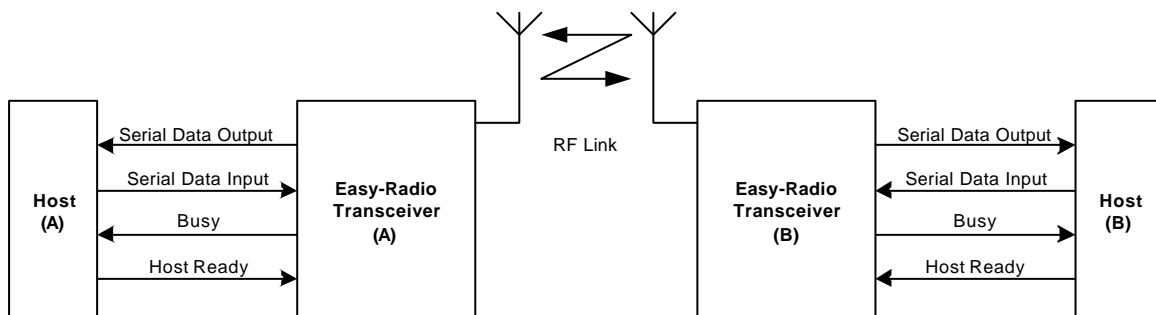
**Figure 4 Easy-Radio Transceiver Block Diagram**

The Serial Data Input and Serial Data Output operate at programmable standard Baud rates and the two handshake lines provide optional flow control to and from the host. Easy-Radio Transceivers can accept and transmit up to 128 bytes of data which they buffer internally before transmitting in an efficient over-air code format.

Any other Easy-Radio Transceivers, within range, that 'hear' the transmission will decode the message and place the recovered data within a receive buffer that can then be unloaded to the receiving host for processing and interpretation. Transmission and reception are bi-directional half duplex i.e. transmit OR receive but not simultaneously.

## Application & Operation

Figure 5 shows a typical system block diagram comprising hosts (user's application) connected to Easy-Radio Transceivers. The hosts (A & B) will be monitoring (collecting data) and/or controlling (sending data) to some real world application.

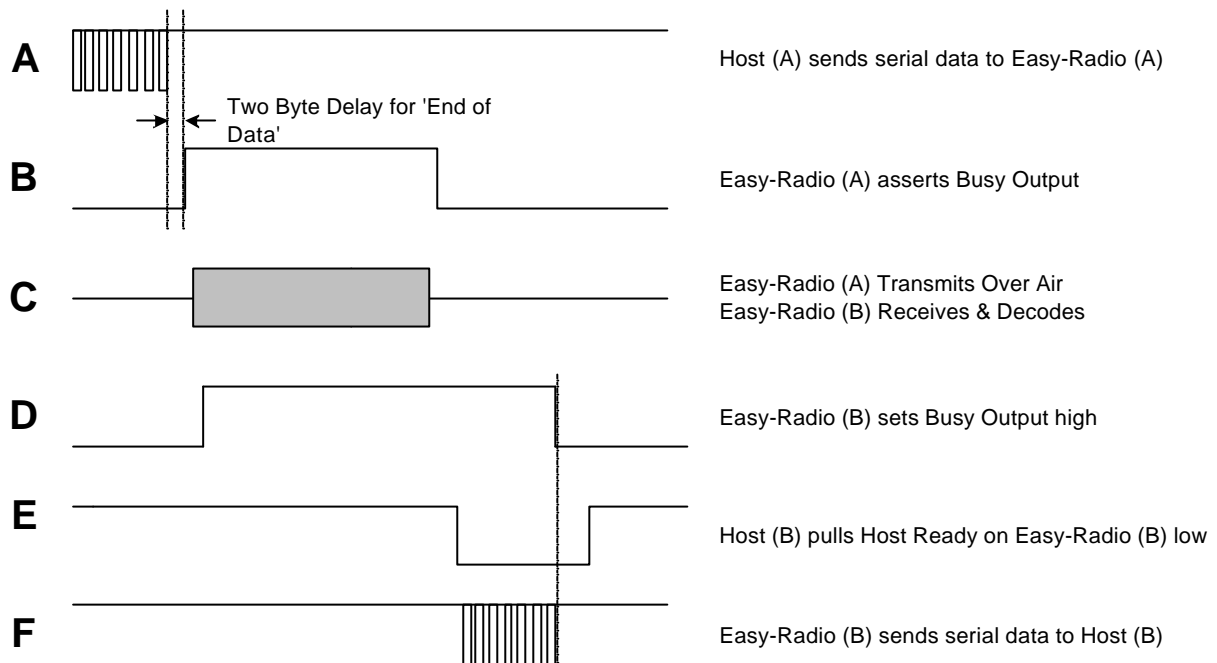


**Figure 5 Typical System Block Diagram**

The hosts provide serial data input and output lines and two 'handshaking' lines that control the flow of data to and from the Easy-Radio Transceivers. The 'Busy' output line, when active, indicates that the transceiver is undertaking an internal task and is not ready to receive serial data. The 'Host Ready' input is used to indicate that the host is ready to receive the data held in the buffer of the Easy-Radio Transceiver.

The host should check before sending data that the 'Busy' line is not high as this would indicate that the transceiver is either transmitting or receiving data over the radio link. It should also pull the 'Host Ready' line low and check that there no data appears on the Serial Data Output line. Figure 6 provides a more detailed explanation of flow control.

### Timing & Data Flow Description



**Figure 6 Timing Diagram**

- Host (A) sends serial data to the Easy-Radio Transceiver (A). Before doing so the host should check that the 'Busy Output' handshake line is low (Clear to Send data). The data must be continuously streamed at the selected baud rate and it fills an internal transmit buffer until either 128 bytes have been received or a gap of two bytes is detected.
- After detecting either the 'End of Data' gap or the 'Buffer Full' condition Easy-Radio Transceiver (A) sets the 'Busy' output handshake line high. It then enables the RF transmit section of the transceiver and sends a 5mS preamble followed by the data in the buffer which is Manchester encoded at 19,200 Baud for efficient transmission across the RF link.
- Any Easy-Radio Transceivers within range that 'hear' the transmission will simultaneously lock onto the preamble, decode the data and place it into their receive buffers.
- The 'Busy Output' goes high during the decoding process and will remain high until the receive buffer is empty. The host should not send new data to the transceiver if this line is high, should it attempt to do so the data will be ignored and therefore lost.
- Host (B) should indicate to Easy-Radio Transceiver (B) that it is ready to receive the data by setting the 'Host Ready' line low. When there is data in the buffer it will appear on the Serial Data Output. When the buffer is empty the Busy Output will go low.
- Data within the buffer then flows from the Easy-Radio Transceiver (B) to the host. The host can control the flow of data at any time by raising the Host Ready line to stop the data and lowering the line to continue the flow of data. After the host has received all the data (detected by either no more data or a minimum two byte gap in the data) it should then return

**Timing Specifications** – Applies to all Easy-Radio Modules.

| Parameters               | Min  | Typical | Max   | Units   | Notes |
|--------------------------|------|---------|-------|---------|-------|
| Host Serial Input/Output | 2400 | 19200   | 38400 | baud    | 1     |
| Host Character Time      | 4.16 | 0.52    | 0.26  | mS/Byte | 2     |
| Preamble Time            |      | 5       |       | mS      | 3     |
| End of Data Delay        | 8.32 | 2.08    | 0.52  | mS      | 4     |
| RF Transmit Time         |      | 800     |       | uS/Byte | 5     |
| Buffer Size              | 1    |         | 128   | Bytes   | 6     |

### Notes

1. Data is inverted i.e. Start Bit is logic low. The inputs are intended for direct connection to a microcontroller UART or to RS232 inputs and outputs via an RS232 Level translator such as a Maxim MAX232 which invert the logic of the RS232 signals. This allows direct connection to, for example a PIC UART. The data rate is user programmable (Default 19200 baud) and may differ between individual units within a system. (See Application Circuit diagram for logic level to RS232 interface).
2. 1 start, 8 data, 1 stop = 10 bits @ 104uS/bit = 0.52mS/character at 19200 Baud. (Default)
3. A fixed preamble of 5mS is added to all packets.
4. The 'End of Data' delay is fixed at twice the character time.
5. 8 data bits only at 19,200 bps Manchester encoded.
6. The buffer size is limited to 128 bytes. Sending more than 128 bytes will cause loss of data.

### Message Format

The Users Data is enclosed within a 'packet' that has the following format and timing:

| Preamble | 'Sync' Byte1 | Sync Byte 2 | Byte Count | Data Bytes<br>1 to n bytes (max = 128) | CRC Byte |
|----------|--------------|-------------|------------|--|----------|
| 5mS      | 800uS        | 800uS       | 800uS      | n x 800uS                              | 800uS    |

|              |  |
|--------------|--|
| Preamble     | The preamble consists of 5mS of 50% duty cycle square wave and is used by the receiver to set up internal circuitry (e.g. Data Slicer) |
| 'Sync' Bytes | Special unique characters used to find the middle of the bit time (synchronise).   |
| Byte Count   | The number of data bytes within the packet. (1 to 128)   |
| Data         | The users data (up to 128 bytes maximum)   |
| CRC          | A Cyclic Redundancy Check on the users data used to detect errors  |

The additional data sent within a packet imposes a fixed 'overhead' on the time taken to send the users data. The 'airtime' required to send, for example, 32 bytes of User Data can be calculated as:

$$5\text{mS} + (4 \times 800\text{uS}) + (32 \times 800\text{uS}) = 33.8\text{mS total}$$

This then allows the total 'End to End' (Host to Host) message time for 32 Bytes of User Data sent at 19200 baud to be calculated:

| Operation          | Time        | Sub-Total (mS) | Notes                                |
|--------------------|-------------|----------------|--------------------------------------|
| Host (A) to Tx (A) | 32 x 0.52mS | 16.64          | Load transmit buffer                 |
| End of Data Delay  | 2 x 0.52mS  | 1.04           | Max.                                 |
| Airtime            | 33.8mS      | 33.00          | Includes packet overhead – see above |
| Rx (B) to Host (B) | 32 x 0.52mS | 16.64          | Unload Receive buffer                |
| End of Data Delay  | 2 x 0.52uS  | 1.04           | Max.                                 |
|                    | Total       | 69.16mS        |                                      |

In addition there may be variable handshake delays (dependant upon the host) whilst the hosts test the handshake lines. Note that the host to Easy-Radio module 'load' and 'unload' times are dependant upon the data rate chosen for the serial interface.

## Software Checklist

1. The data is sent from the host to the transceiver as asynchronous serial data at programmable data rates with a format of 1 start bit, 8 data bits and 1 stop bit with no parity. Typically (19200,8,N,1)
2. Transmission and reception are half duplex i.e. transmit OR receive but not simultaneously.
3. Transmitters and receivers are uni-directional (Tx > Rx) and transceivers are bi-directional (Tx < > Rx).
4. The host should always test that the 'Busy' handshake line is low before sending data to the Easy-Radio Transceiver. Data will be lost, corrupted or ignored if serial data is sent whilst the line is high.
5. If more than 128 Bytes per 'packet' are sent to the Easy-Radio transceiver the first 128 bytes will be transmitted the remainder will be lost.
6. Data is held in the receive buffer for a maximum of 2.5 seconds after reception and decoding. The receiving host should therefore repetitively pull the 'Host Ready' handshake line low and check the Serial Data Output line for data within this time period. Alternatively (and more easily) the 'Host Ready' handshake line can be permanently tied to 0 Volt (Ground). The Easy-Radio Transceiver will then immediately send data from the receive buffer to the host. The host should detect serial data present by the use of interrupts.
7. Serial data sent from the host to the Easy-Radio Transceiver whilst it is holding valid received data will be ignored. The host should therefore check that there is no received data before attempting to send data.
8. The transceiver cannot 'loop back' its own data. 'Loop back' is a term used to describe 'listening' to ones own transmitted data. The use of a single transmit/receive buffer and the internal control logic precludes this mode of operation.
9. There are no constraints on the serial data that may be sent and received. Characters may be any of the 256 combinations of 8 bit data.
10. The transmitted data is internally error checked by a 'checksum'. If the computed checksum on the received data does not match the checksum sent within the data the whole message is discarded and will not appear in the receive buffer. Generally for Short Range Devices (SRD) such as the Easy-Radio Transceivers the data will either be 100% correct or at the extremes of range, be totally corrupted. Additional, more sophisticated, error checking such as a Cyclic Redundancy Check (CRC) can be embedded within the user's messages. See notes below.

## Hardware Checklist

1. The modules operate internally from an on board low drop regulators. (Typically 3.3 Volt) The logic levels of the input/output pins are therefore between 0 Volt and the regulated voltage. (See specifications).
2. Outputs will drive logic operating at 5 Volts and inputs will be correctly driven by logic operating at 5 Volts (CMOS & TTL logic levels).
3. When power is first applied the processor performs an initial 'calibration' of the RF section that compensates for temperature and power supply voltage variations. The transceiver will then be ready to receive (default) or transmit. It would normally be left in this powered state ready to receive data. During power up calibration the Busy Output line goes high.
4. The 'Host Ready Input' and the 'Serial Data Input' have 'weak' internal pull ups enabled. These inputs should not however be left 'floating' but should be tied to either Vcc or Ground 0 Volts.
5. The serial inputs and outputs are intended for connection to a UART or similar low voltage logic device. Do not connect any of the inputs or outputs directly to an RS232 port. The transceiver module may be permanently damaged by the voltages (+/- 12V) present on RS232 signal lines. See Application Circuit (Figure X) for typical connection to an RS232 port via MAX232 interface IC.
6. The 'Host Ready Input' should be tied to 0 Volt (Ground) if not used.
7. The 'Serial Data Input' should be tied to Vcc if not used. (Receive mode only).

## Addressing

There is no hardware or software 'addressing' incorporated within the Easy-Radio Transceivers as requirements will vary dependant upon the users' application. It is however very easy to embed addressing and other information within the message.

A typical user message (often called a 'telegram') might comprise the following format:

| Destination Address | Source Address | Message Type | Retry Counter | Data Length (n) | Data (1 to n) | Checksum Or CRC | End of Message |
|---------------------|----------------|--------------|---------------|-----------------|---------------|-----------------|----------------|
|---------------------|----------------|--------------|---------------|-----------------|---------------|-----------------|----------------|

**Figure 7 Typical Message Format**

|                     |   |
|---------------------|---|
| Destination Address | Defines the eventual recipient of the message. This may be an individual or a group.  |
| Source Address      | Defines the sender of the message   |
| Message Type        | The message may require the recipient to acknowledge receipt or it may be 'broadcast' to many within a group. The message may also be assigned some 'priority' if it is urgent. |
| Retry Counter       | Indicates the number of times that the same message has been sent.  |
| Data Length         | The number of data bytes within the message.  |
| Data(1 to n)        | The data bytes  |
| Checksum or CRC     | The numerical sum of the data bytes – will not detect errors that cancel each other. A Cyclic Redundancy Check provides a more rugged method of error detection.                |
| End of Message      | A special 'marker' to indicate the end of the message.  |

There are many standard protocols (e.g. Ethernet) that are necessarily complicated by the need to provide multiple universal 'services'. Most applications do not require or cannot be burdened with the 'overhead' of these protocols and a much simplified subset will usually suffice.

The essence of 'Easy-Radio' is to provide the reliable transport of serial data over a radio link so that the user can concentrate their efforts on the application.

The Internet can provide much useful information by searching for 'Communication Protocols' or 'Data Communications' within a search engine. Topics such as CRCs are dealt with in-depth.

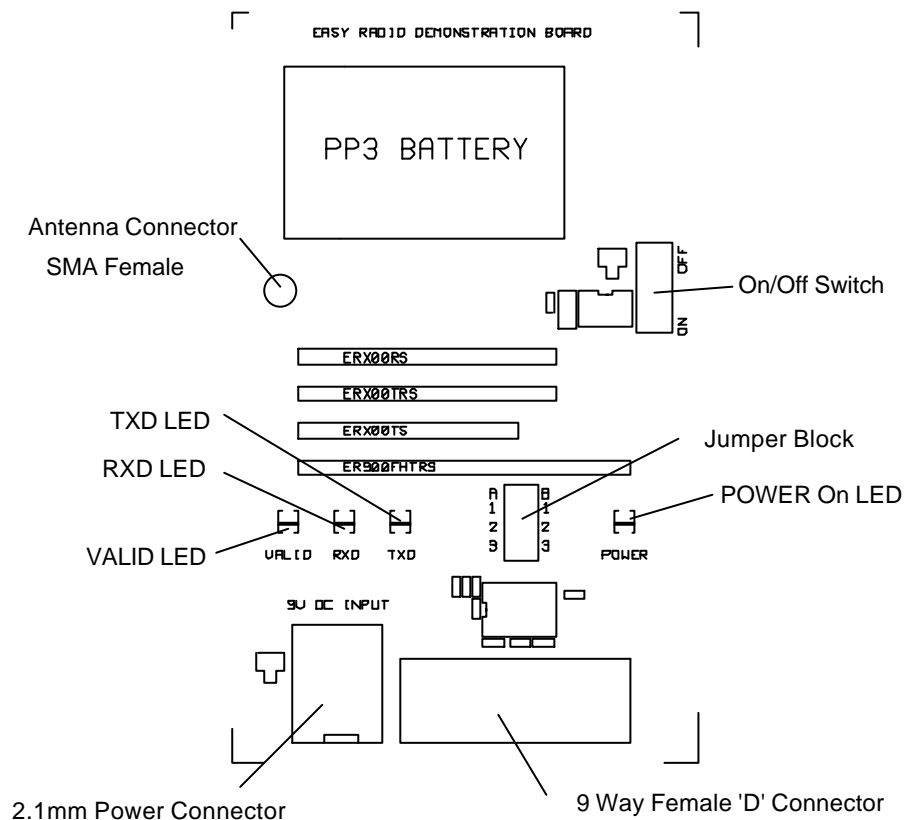
### Easy Radio Demonstration Board

The Easy-Radio Demonstration Board provides a platform for the evaluation and test of all Easy-Radio modules. Transmitters, receivers and transceivers can be plugged into the boards and are provided with an interface to allow communication with a PC serial port. The boards can be powered by either a 9 Volt PP3 battery (supplied) or a 9 Volt AC Mains adaptor (not supplied). LEDs provide status indication for transmitted data (TXD), received data (RXD), and VALID data. On board jumpers enable the use of the 'handshaking control lines and allow LED indicators to be connected to data or control lines.

The user will need to supply two cables for connection to PC Serial ports. These should be 9 way 'D' Type Male to Female, 1:1 (not cross-over type). These cables are available from most PC accessory suppliers but in case of difficulty please contact the Sales Office. Alternatively cables can be simply made up by wiring Pins 1-9 of a male connector to Pins 1-9 of a female connector.

For extended testing it may be wise to use 9 Volt DC power supplies fitted with 2.1mm connectors (+Ve centre, -Ve Ring).

### Board Layout



Schematic

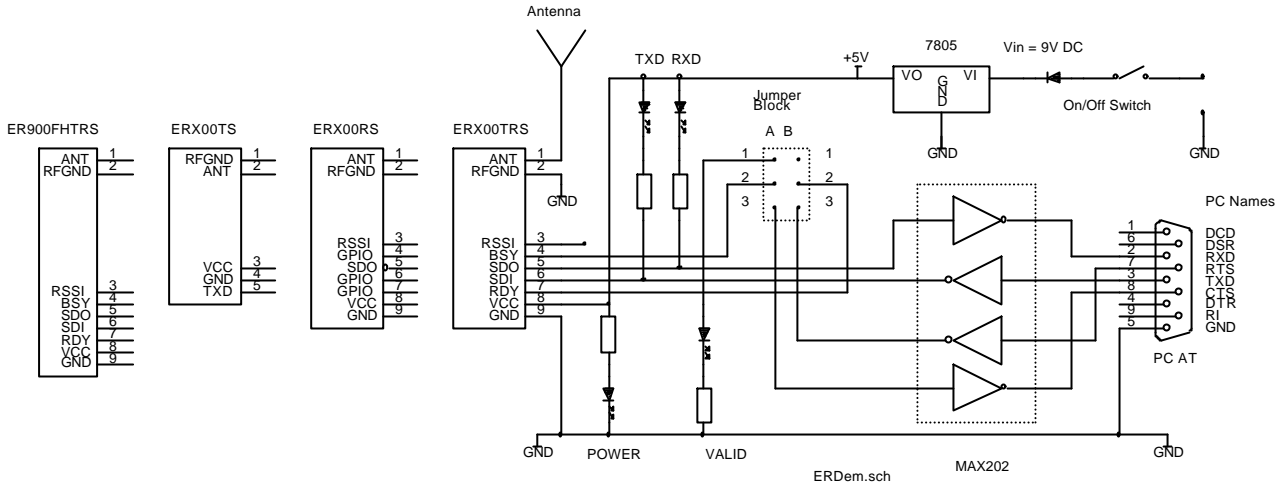
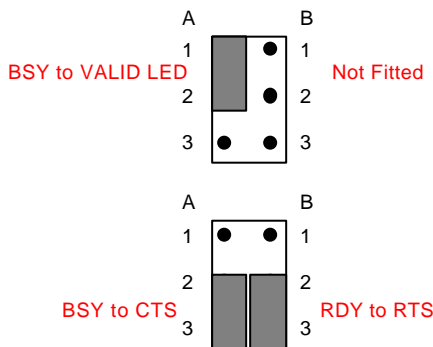


Figure 8 Schematic Diagram

The schematic is diagrammatic and does not show the detailed circuitry.

Jumper Block

A 6 pin jumper block is fitted to the board. It comprises two columns of pins, A & B and the supplied jumpers should only be fitted North-South linking either Pins 1-2 or Pin 2-3 on each column as shown below. The jumpers should not be fitted East-West as this may cause damage to the board or modules.



No Handshaking

In this configuration hardware handshaking is disabled. The VALID LED is connected to the BSY output and indicates that VALID data has been received – used for receivers only.

Handshaking Enabled

In this configuration hardware handshaking is enabled. The BSY output is connected (via logic to RS232 level translator) to CTS and RDY input is connected to RTS output.

Module Pin Descriptions – See specific Data Sheets for further description.

| Name   |        | Description   | Notes                |
|--------|--------|---|----------------------|
| ANT    | I/O    | Antenna – RF Input or Input/Output for Txcvrs   | 50 Ohm               |
| RF GND | Input  | RF Ground. Connect to ground plane and supply 0 Volt  |                      |
| RSSI   | Output | Received Signal Strength Indicator  | Analogue             |
| BSY    | Output | Low – Device is Ready to receive data from Host<br>High - Device is Not Ready to receive data from Host   | CTS function         |
| SDO    | Output | Serial Data Output from Device to Host  |                      |
| SDI    | Input  | Serial Data Input from Host to Device<br>Connect to Vcc (High) if not used.   |                      |
| RDY    | Input  | Low - Host is Ready to receive data from Device<br>High - Host is Not Ready to receive data from Device<br>Connect to 0 Volt (Ground) if not used | RTS function         |
| Vcc    | Input  | Positive supply pin   |                      |
| GND    | Input  | Connect to supply 0 Volt and ground plane   |                      |
| GPIO   | I/O    | General Purpose Input Output  | Factory configurable |

## Operation

Two PCs are required for testing. They should initially be positioned adjacent to each other so that both screens can be viewed simultaneously. Only ONE module should be fitted to the board.

- Check that 'matching' Easy-Radio modules are fitted to the Demonstration Boards. (Only one module per board)
- Position the jumpers – see diagram above.
- Connect the supplied antenna.
- Connect boards to PC serial ports using 9 way 'D' Type 1:1 connecting cable (not cross-over type).
- Power the boards from either batteries or mains power supplies.
- Switch the power switch to the 'ON' position, the POWER LED should light.
- Use a Windows terminal programme such as 'HyperTerminal' on each PC and set the comms parameters to (default - Easy-Radio settings) 19200 Bits per second , 8 Data Bits, No parity, 1 Stop Bit, and initially Flow Control – None. Ensure connection directly to the chosen Com port.



**Figure 9 Hyper Terminal Set Up**

Characters typed on one keyboard should be displayed on the other PC screen if transceiver modules are being used (Half Duplex). For transmitter and receiver modules communication will only be 'one way' i.e. Transmitter to Receiver (Simplex). This establishes that the modules are correctly connected and configured. The PCs should then be physically separated to test range and reliability. This is more convenient if one of the PCs is a laptop.

### Hardware Handshaking

If hardware handshaking is enabled (Column A: Pins 2-3 connected & Column B Pins 2-3 connected) the Flow Control setting in HyperTerminal should be modified from 'None' to 'Hardware'. (Do not select 'X on / X this mode is not supported.)

This mode allows small 'files' to be sent (using transceivers). Almost any other 'Terminal' programme can be alternatively used.

## Easy-Radio Product Programming Software

The demonstration board is supplied with a 1.44MB floppy disk or CDROM containing a file named ER400TRS.exe. This file should either be run from the floppy disk (by double clicking in Windows Explorer) or copied and run from a convenient location on the PCs used for serial communications.

There is no need for any Windows installation procedure. It has been tested on the following operating systems: Windows 95, Windows 98, Windows ME, Windows 2000 & Windows XP. The programme takes direct control of the PC Serial port (Com 1 - default) which must be present and not being used by any other application. A small file (Test.ini) is created on the Root drive.

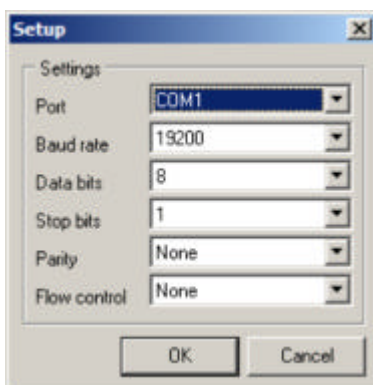
Note: The Easy radio web site [www.easy-Radio.co.uk](http://www.easy-Radio.co.uk) should be checked for updated software and registered users will be sent email notifications of available upgrades.

After opening the programme (double click on the exe file) the screen below will open and serial communications should be established with the Easy Radio module on the connected Demonstration Board. This software allows the following parameters to be configured: Serial Baud Rate, Power Output, Frequency, Power Saving (later versions), Handshake Control (later versions) and other special test modes.



Figure 10 Welcome Screen

Spend some time 'exploring' the functions and drop down box options. The functions enabled on the current version of the software.



Should there be no communication established with the module click the 'Settings' button to open the 'Setup' dialogue. Ensure module is connected to selected Com port (Com 1 – default - change if different).

Check Baud Rate is default 19200 and other settings are as shown.

Each parameter may be configured by selecting the required value from the 'drop down' boxes and then clicking the 'Update' button. The 'Terminal Window' will then display the actual text command that has been sent to the Easy-Radio module and the module will 'echo' the command that has just been sent. If the 'Auto ACK' check box is ticked (as shown) the new setting will automatically and permanently saved in non-volatile memory within the module. If the check box is not ticked the 'ACK' button should be manually clicked after each and every modification.

The Terminal window can be cleared by 'clicking' 'Clear terminal Window'.



## Notes

- Ensure that settings are 'sensible' and do not cause module operation to contravene local regulations. E.g. Excessive Power Output.
- Make a note of settings programmed into each module particularly if the default comms baud rate is changed.
- Invalid settings for particular modules are at present not checked. Later versions will identify the module and 'lock out' bad settings.
- Some commands may perform a 'hard reset' on the module and cause it to enter an extended 'calibration' mode and/or send spurious characters from the input/output buffers.
- Test Modes are used during manufacture final test and allow the Carrier to be permanently switched on. They should be used with caution.

## Easy-Radio Configuration Command Set

The programming software sends 'text messages' to the modules and this action can be performed by the terminal software using the following list of commands:

| Command    | Function                         | Value  | Notes           |
|------------|----------------------------------|--------|-----------------|
| ER_CMD#U1  | UART Data Rate                   | 2400   |                 |
| ER_CMD#U2  |                                  | 4800   |                 |
| ER_CMD#U3  |                                  | 9600   |                 |
| ER_CMD#U4  |                                  | 19200  |                 |
| ER_CMD#U5  |                                  | 38400  |                 |
| ER_CMD#P1  | RF Power Output                  | 1mW    |                 |
| ER_CMD#P2  |                                  | 2mW    |                 |
| ER_CMD#P3  |                                  | 3mW    |                 |
| ER_CMD#P4  |                                  | 4mW    |                 |
| ER_CMD#P5  |                                  | 5mW    |                 |
| ER_CMD#P6  |                                  | 6mW    |                 |
| ER_CMD#P7  |                                  | 7mW    |                 |
| ER_CMD#P8  |                                  | 8mW    |                 |
| ER_CMD#P9  |                                  | 9mW    |                 |
| ER_CMD#P10 |                                  | 10mW   |                 |
| ER_CMD#C0  | Channel 0                        | 433.23 | MHz             |
| ER_CMD#C1  | Channel 1                        | 433.30 | MHz             |
| ER_CMD#C2  | Channel 2                        | 433.45 | MHz             |
| ER_CMD#C3  | Channel 3                        | 433.55 | MHz             |
| ER_CMD#C4  | Channel 4                        | 433.68 | MHz             |
| ER_CMD#C5  | Channel 5                        | 433.83 | MHz             |
| ER_CMD#C6  | Channel 6                        | 433.88 | MHz             |
| ER_CMD#C7  | Channel 7                        | 434.00 | MHz             |
| ER_CMD#C8  | Channel 8                        | 434.15 | MHz             |
| ER_CMD#C9  | Channel 9                        | 434.35 | MHz             |
| ER_CMD#S1  | Sleep Mode                       |        |                 |
| ER_CMD#D0  | Power Saving - On/Off Duty Cycle | 100%   |                 |
| ER_CMD#D1  |                                  | 90%    |                 |
| ER_CMD#D2  |                                  | 80%    |                 |
| ER_CMD#D3  |                                  | 70%    |                 |
| ER_CMD#D4  |                                  | 60%    |                 |
| ER_CMD#D5  |                                  | 50%    |                 |
| ER_CMD#D6  |                                  | 40%    |                 |
| ER_CMD#D7  |                                  | 30%    |                 |
| ER_CMD#D8  |                                  | 20%    |                 |
| ER_CMD#D9  |                                  | 10%    | Off             |
| ER_CMD#H1  | Handshake On                     |        |                 |
| ER_CMD#H2  | Handshake Off                    |        |                 |
| ER_CMD#R1  | Default Settings                 |        | Factory Default |
| ER_CMD#T1  | Carrier Only                     | XX     | Test Mode 1     |
| ER_CMD#T2  | Constant Preamble                | XX     | Test Mode 2     |

The commands should be sent exactly as shown (case sensitive) with no spaces between characters. The ACK command is sent as three Ascii characters ACK in sequence.

## Custom Versions

Please contact the sales office for availability and other variants of the standard product. The software interface can be customised to specific requirements for medium to high volume applications.

## Document History

| Issue | Date         | Revision                                       |
|-------|--------------|--|
| 1-0   | July 2002    | Preliminary                                    |
| 1-1   | January 2003 | Updated for Demo Boards & Programming Software |
| 1-2   | May 2003     | Pictures added and slight text amendments      |
| 1-3   | June 2003    | Look up table modified for commands 8 & 9      |

## Copyright

The information contained in this data sheet is the property of Low Power Radio Solutions Ltd and copyright is vested in them with all rights reserved. Under copyright law this documentation may not be copied, photocopied, reproduced, translated or reduced to any electronic medium or machine readable form in whole or in part without the written consent of Low Power Radio Solutions Ltd.

The circuitry and design of the modules are also protected by copyright law.

## Disclaimer

Low Power Radio Solutions Ltd has an on going policy to improve the performance and reliability of their products; we therefore reserve the right to make changes without notice. The information contained in this data sheet is believed to be accurate however we do not assume any responsibility for errors or any liability arising from the application or use of any product or circuit described herein. This data sheet neither states nor implies warranty of any kind, including fitness for any particular application.



For further information or technical assistance please contact:

Tel: +44 (0)1993 709418  
 Fax: +44 (0)1993 708575  
 Web: <http://www.lprs.co.uk>  
 Email: [info@lprs.co.uk](mailto:info@lprs.co.uk)

### Low Power Radio Solutions Ltd

Two Rivers Industrial Estate  
 Station Lane  
 Witney  
 Oxon  
 OX28 4BH  
 England



Web: <http://www.easy-radio.co.uk>

The above address is a dedicated web site for Easy-Radio

End of Data Sheet