Radiometrix

Hartcran House, 231 Kenton Lane, Harrow, HA3 8RP, England

Tel: +44 (0) 20 8909 9595, Fax: +44 (0) 20 8909 2233

Issue A, 01 November 2004 **SP2 Eval Kit**

SpacePort Evaluation Kit

SpacePort Evaluation Kit can be used to evaluate the Radiometrix SpacePort, Radio Packet Controller modules. LED indicators are provided to show system status and to facilitate range testing and installation site surveys. Internal EEPROM values can be configured through parallel port connection to PC using RPC Development Kit Software.

Figure 1: SP2-433-160 in SpacePort Eval Kit

Features

- A pair of Evaluation PCBs to evaluate SP2, RPC and FRPC modules
- Direct interface to Parallel port
- Visual indications of operational mode and test results through LEDs
- Access to internal diagnostic/Test modes and EEPROM using RPC Development Kit Software
- All I/O are brought out with adjacent headers for developing applications and analyzing signals
- PP3 9V Battery operation makes the board portable for easy wireless evaluation

Kit contents

- 2 SP2-433-160 modules (RPC or FRPC should be ordered separately)
- 2 Evaluation boards
- 2 1/4 wavelength whip antennas (433MHz)
- 2 9V battery (PP3)
- 1 BiM2 data sheet
- 1 SP2 data sheet
- 1 SP2 Evaluation Kit manual

Optional requirement

- DB25M-DB25F parallel extension/straight through cable
- PC/laptop with ECP parallel port with MS-DOS, Win3.X, Win95/98.
- RPC Development Kit Software

The following status LEDs will be activated depending on which mode is selected:

LED	Indication
TX (Red):	Transmitter enabled
RX (Green):	Receiver enabled
Power (Green):	Evaluation Kit is switched on
SIGNAL (bright Red):	Valid preamble detected
OK (bright Yellow):	Valid packet received / Test passed

1. Standalone Operation

This mode selects the internal diagnostics modes built into the RPC, FRPC or SP2.

Set-up

- Connect the 1/4 wavelength antenna into the antenna terminal on the evaluation board. Plug the SP2 into the DIL socket marked SpacePort. RPC/FRPC should be plugged in with RPC/FRPC IC facing the Evaluation Kit and shielding can or module facing the other way.
- Connect a 7.5VDC-24VDC supply or 9V battery to the supply input terminals and slide the power switch to ON position.
- Put the DEBUG jumper on and press 'Reset'
- The Hex switch selects the required debug mode 0 to 8. A reset is not required after a mode change.

Digital Storage Oscilloscope probes can be connected to TXD/AF, TX, RX to monitor data transmission to and from RF circuit. Probes can be connected to TXR, TXA, RXR, RXA and D3:D0 data lines to monitor data upload/download from/to Host/PC.

1.1 Diagnostic Modes

Mode	Name	Function
0	RX-ON	Preamble detector ON (SIGNAL LED lit = valid preamble detected)
1	RX-PULSE	10ms on: 10ms off, preamble detector on SIGNAL LED
2	TX-ON-PRE	Preamble modulation – transmit continuous preamble
3	TX-ON-SQ	100Hz square wave modulation, for TX testing using spectrum analyser, etc.
4	TX-ON-255	random 160kbps data for eye diagram tests, sync on RXR
5	TX-PULSE	Preamble bursts (EE 01h setting): 10ms OFF, RX lock in tests
6	ECHO	Transponder mode, unit re-transmits any valid packets received
7	RADAR	Send ASCII test packet "RADIOMETRIX / TEST PACKET WITH 60 DATA BYTES / RPC32 V3.0 XX" and listen for echo. XX is packet number
8	SELF-TEST	Local loop test, TX → RX (OK on RXR). Not available in SP2

Mode 0 - Preamble Detector

In this mode, receiver circuit is continuously powered up (RX LED on) and if preamble, 80kHz or 160kbps square wave signal is detected the SIGNAL line is pulled low lighting the SIGNAL LED to indicate valid Preamble is detected. RXR will also be pulled low lighting the OK LED to indicate a pass.
If the RESET jumper is inserted, then the internal Fast Radio Packet Controller will be disable enabling the internal BiM2 equivalent receiver circuit to function independently.

Mode 1 - Pulsed Receiver

Receiver is switched on for 10ms and SP2 checks for preamble. If preamble is detected the SIGNAL line is pulled low. This will light up the SIGNAL LED. If not, the Receiver is turned off for 10ms and the process is repeated. OK LED will also light up to indicate a pass. This mode can be used to test the power up time and settling time of the receiver circuit.

Mode 2 - Transmit Preamble Modulation

Transmitter is turned on continuously and preamble (160kbps square wave) is transmitted. This complement mode can be used with Mode 0 as a pair.
If the RESET jumper is inserted, then the internal Fast Radio Packet Controller will be disable enabling the internal BiM2 equivalent transmitter circuit to function independently.

Mode 3 - Transmit 100Hz (200bps) square wave modulation

Transmitter is turned on continuously and 100Hz square wave signal is transmitted which can be used to estimate the FM deviation and power levels of the RF transmitter circuit using a spectrum analyser.

Mode 4 - Transmit Random Code

Transmitter is turned on and the carrier is modulated by a 8 bit maximal length (255) pseudo-random code at 6.25µs per bit (at 160kbps). On the receiving end, the data output AF line can be connected to an Oscilloscope to obtain an eye diagram.

An eye diagram is an oscilloscope display in which a pseudo-random data signal from AF output of a receiver is repetitively sampled and applied to the vertical input, while the data rate (RXR) on the transmitting unit is used to trigger the horizontal sweep.

System performance information can be derived by analyzing the display. The horizontal width of the lines gives the jitter (phase noise) and the rise and fall times of the data pulses can be measured from the "crossings". An open eye pattern corresponds to minimal signal distortion. Distortion of the signal waveform due to intersymbol interference and noise appears as closure of the eye pattern.

Mode 5 - Pulsed Preamble Transmitter

The transmitter is turned on and normal preamble (length used for normal data transmission) is sent. Then transmitter is turned off and waits for 10ms before another cycle. This is used to measure the lock in time of the receiver.

Mode 6 – Echo/Transponder

Receiver is turned on to checks for preamble and if a preamble is found, then receiver locks on to the data and receives the data packet. SIGNAL LED will be turned on if valid preamble is detected. Then error check is carried out and if it passes, the OK LED is turned on. Receiver waits for a Transmit to Receive Change Over Delay period. Then it retransmits (echoes back) the packet to the transmitter.

Echo or transponder mode is very useful for remote loop-back testing of user host software and for "ping-pong" range testing in conjunction with the other development board in RADAR mode.

Mode 7 – Radar

Transmitter is turned on and sends a packet **RADIOMETRIX / TEST PACKET WITH 60 DATA BYTES / RPC32 V3.0 XX** as test data where XX will be a Packet Counter. Then transmitter is turned off and receiver is turned on. Unit on this mode checks for preamble and if it finds a valid preamble, then it locks on to the data and receives the packet. Then error check is carried out and if it passes, the OK LED is turned on.

Even if a valid packet was not received, it will continue the above process but the packet counter value will be increased with each transmission.

This mode can be used along with Mode 6 (Echo Mode) to function as a ‘Pin-Pong’ system. This provides a very effective method for Range Testing and Antenna Type Evaluation. If one eval kit is set to Mode 6, then other eval kit can be set to Mode 7. By walking around the site where the final product based on the Radiometrix Modules are going to be used, the range and antenna type requirements, interference, etc could identified well in advance. The OK LED will be continuously lit with no flickering as long as the ‘Ping-Pong’ the units are within reliable radio range and the wireless link is error free.

Mode 8 – Local Loop Test

This mode is not available in SpacePort or Fast Radio Packet Controllers. This mode puts a single Radio Packet Controller into a local loop back (both transmitter and receivers are turned on). A test code pattern is continuously sent and recovered. The OK LED will light to indicate a pass. This mode is used to evaluate receiver and its Adaptive Data Slicer.

Mode F – Normal RPC Mode

DEBUG jumper should be removed and the RESET button should be depressed to exit from Debug mode to normal SpacePort operation. Therefore, Spaceport can be interfaced with Host Microcontroller or a PC to send/receive data packets.

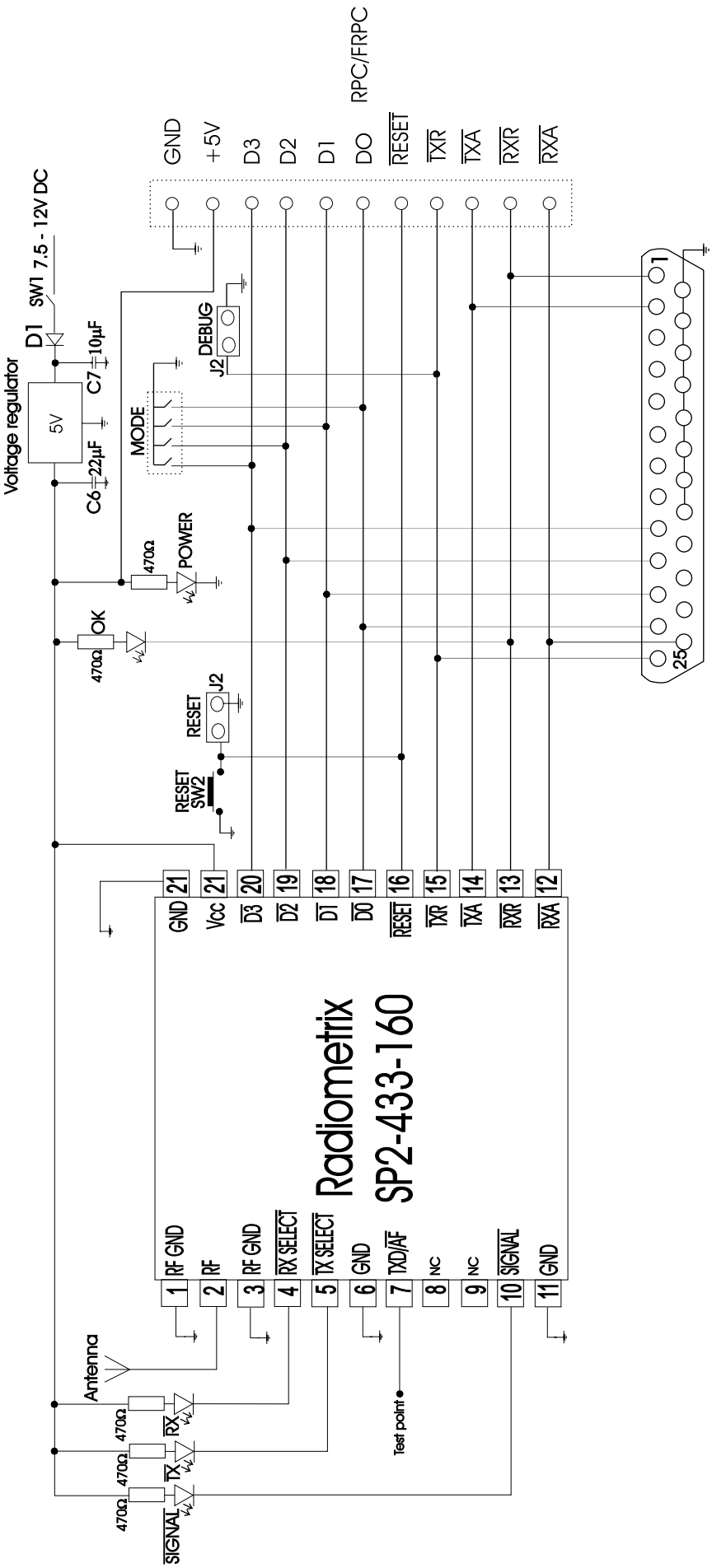

Figure 2: SpacePort Evaluation Kit circuit

2. Transceiver Operation

If RESET jumper is inserted, the internal Packet Controller IC will be RESET leaving direct access to raw RF Transceiver. Jumper across RX-GND will enable receiver circuit and jumper across TX-GND will enable transmitter circuit.

3. PC or Laptop Operation

Set-up

- Connect the antenna into the antenna terminal on the development board and also plug the SP2 into the socket.
- Connect a DC supply/9V battery to the supply input terminals and switch on.
- Connect the development board to the LPT port of a PC or Laptop with DB25M-DB25F parallel extension/straight through cable
- Remove debug jumper.
- Download the RPC Development Kit software files into a suitable directory on your hard disk. http://www.radiometrix.co.uk/products/rpceval/rpc_soft.htm

Software Overview:

The RPC Development Kit software gives immediate access to a Radiometrix SP2 module and enables simple ASCII message transmission/reception. The software will display the EEPROM memory map of the SP2 which can be changed to configure the SP2 parameters.

For Bidirectional PS/2 or Extended Capabilities Port (ECP) in new PCs

It contains the following files:

DEMO.BAT (Run this file)	Batch file to set the Byte Mode or PS/2 Mode in ECP before running SP2 software
RPC-BI.EXE	main PS/2 SP2 driver program
RPC.DAT	ASCII data file holds system information used by SP2-BI.EXE
D.A	sample ASCII test files
D.B	
D.C	
D.D	
D.E	

The main program RPC-BI.EXE is designed to run under MS-DOS 3.3 or higher on any IBM PC or compatible with printer port set to ECP mode.

Using this program it is possible gain access to the onboard EEPROM to evaluate the extended functionality provided via the Reserved Memory settings and to also read/write the User EEPROM area.

The program provides a set of commands allowing the user to operate the SP2 module. These command functions enable the user to include send and receive messages, write to EEPROM memory, send continuous messages to the SP2 and enable the 7 SP2 debug modes.

Upon start-up of the utility if an SP2 is connected to the parallel port and is working, the Reserved Memory and User Memory areas of the display will be updated.

The Outgoing and Incoming message area can be expanded or reduced as the user requires by pressing the TAB key (or alternatively entering switcH at the command prompt). This has the effect of either

hiding or showing the user memory area. In order to view the help list properly the display needs to be in the expanded mode.

Messages sent from the SP2 are displayed under the Outgoing (TX) message heading to the left of the display. Messages received by the SP2 are displayed under the Incoming (RX) heading to the right of the display.

Command set:

The following list has been taken from the RPC demonstration program and details the commands which are available for evaluating the SP2.

COMMAND	*DESCRIPTION*
Reset [n]	Reset the SP2; Test mode (n = 0 - 8)
rea**D** address	Read from memory address (Addr = 00 - 3F)
Send [$] string	Transmit string via SP2; $ selects preamble
Write address data	Write data to SP2 memory address (00 - 3F)
Clear	Clear the display output window
switc**H** or <TAB>	toggle memory display window On/Off
File [delay] file [file...]	Send a file(s) to the SP2 delay = delay between files (0.25s increments)
Test	execute the file send test list
sto**P** or <^X>	to stop repeating file send (Test & File)
Help	Display this help information
e**X**it or <F3>	Exit from this demo program
<ESC>	Erase current command line
<TAB>	Switch display between memory and output

note: 1. The capital letter in each command may be used in place of the full word.
2. All values required by specific commands should be entered in hex.

Commands in Detail:

Note: Square brackets [] means the argument(s) are optional.
Angle brackets <> means the argument(s) are required.
The capital letter in the command represents an abbreviation of that command.

Reset [0-8]
send a RESET to the SP2.
sets the SP2 into the specified demo mode.
- *example:* >R reset SP2
  >R7 reset SP2 into debug mode 7 (RADAR)

rea**D** <address>
read the EEPROM memory at location <address>
the address should be in the range of 0x00 - 0x3F
- *e.g.:* >read 20

Send [$] <string>
Transmit the given string via the SP2.
Preceding the string with a $ sign will enable extended preamble to be used when transmitting the packet. If more than 27 bytes of data are entered on the command line, the string will be broken into 2 SP2 packets and transmitted.
- *example:* **>S THIS IS A TEST send 'THIS IS A TEST'**
  >S$Is any body out there ? ; extended preamble

Write <addr> <data>
Write data to the specified SP2 EEPROM location.
The allowable memory range is from 0x00 to 0x3F. Data represents a single byte between 0x00 - 0xFF.
- *example:* >W 00 4 writes 04h into SWITCHES (00h)
  >W 08 80 will set PS1 on reset

e**X**it or <F3>	Typing either EXIT or X, or pressing F3 will exit from the demonstration program back to the command prompt.
Clear	Clear the display output window.
sto**P** or <^X>	Stop the repeating file send (Test & File commands).
Help	Display this help information as shown in section SP2DEMO commands.
<TAB>	Switches the display between the memory display and the expanded output display.
<ESC>	Clear the current command line

File [delay] file [file...] Send a file to the SP2. A maximum of 3 files can be given on the command line. The file names should contain only alpha characters (e.g. fred1.txt <- not allowed. freda.txt <- allowed). The [dly] enables a delay of between 0x00 and 0xFF seconds between files being sent. Using the delay will enable the files to be repeated continuously using the delay value between them. Without the delay value the file(s) will only be sent once.
This is NOT a file transfer function. i.e.; it will not copy the file to the destination.
If any lines in the file contain more than 27 bytes, the line will be broken into multiple blocks of 27 bytes. They will NOT be reassembled into full lines by the receiver.

- *e.g.* file 2 autoexec.bat config.sys
- *example:* >F SP2.DOC transmits this file

Test Execute the test file command line from the SP2.DAT file.
See SP2 Configuration File details later for a description of SP2.DAT command line used with this command.

- *example:* >T repetitively sends the test files.

SP2 Configuration File:

The demonstration program requires a configuration file. This file is called SP2.DAT. Following is an example SP2.DAT file:

```
PORT = 378
COLOUR = 1
FILE = 8 d.a d.b d.c d.d d.e
```

PORT = <xxx>
This entry determines the base address of the PC printer port.
The value is entered as a hex value.

COLOUR = <0 | 1>
This is a Boolean of either 0 or 1.
Setting this entry to 0 disables the colour display. i.e.; all output will be in black and white.
This is suitable for a monochrome display device such as a laptop.
Setting this entry to 1 will enable the coloured output.

FILE = <delay> <file1> <file2> <file3>
This entry has the same format as the FILE command.
It is the command line used when the TEST command is entered.

Appendix A ***Using a printer port to drive the SP2.***

For New PCs: Bi-directional Port (PS/2)

Port requirement 8 bit bi-directional PS/2
(PS/2 or ECP set to PS/2 Mode / Byte Mode)

In PS/2 Mode, Status Lines are used for Control line input from SP2 (RXR, TXA) and Printer Port Control Lines are used to output the SP2 Control signals (RXA & TXR). In Bidirectional PS/2 mode, Printer port data lines can be used as SP2 data lines in bidirectional mode.

Most of the PCs come with Extended Capabilities Port (ECP). ECP can be set to operate in PS/2 compatible bidirectional mode. Program supplied with bidirectional version will automatically change the mode from ECP to PS/2 and change it back to ECP when the RPC Dev Kit software is closed.

SP2 name	End pin		Bidirectional pin	Port Register bit	Port End pin labels
GND	1	—	18 to 25		Ground
D0	2	↔	2	D0	Data 0
D1	3	↔	3	D1	Data 1
D2	4	↔	4	D2	Data 2
D3	5	↔	5	D3	Data 3
TXR	6	←	1	C0	$\overline{\text{Strobe}}$
TXA	7	→	12	S5	Paper Out
RXR	8	→	13	S4	Printer Selected
RXA	9	←	14	C1	$\overline{\text{Auto Linefeed}}$
RES	10	←	16	C2	$\overline{\text{Initialise Printer}}$
5V	11	←	+5V supply		
GND	12	←	0V supply		
+ve interrupt		→	10	S6	$\overline{\text{Acknowledge}}$

PC Printer port registers
(addresses given for base address of 0378h)

0378	data register	b7	b6	b5	b4	b3	b2	b1	b0
		-	-	-	-	D3	D2	D1	D0

0379	status register	b7	b6	b5	b4	b3	b2	b1	b0
			int	TXA	RXR				

037A	control register	b7	b6	b5	b4	b3	b2	b1	b0
		-	-	dir	Ien	-	RES	RXA	TXR

For Extended Capabilities Port (ECP) only

077A	Extended Control Register (ECR)	b7	b6	b5	b4	b3	b2	b1	b0
		0	0	1	-	-	-	-	-

Printer Port can be configured to operate in ECP mode by changing the Printer Port setting in BIOS from SPP or EPP to ECP. BIOS setup can be accessed by pressing DEL key for AWARD BIOS or F1 for AMI BIOS when booting the computer. Parallel Port settings can be changed in the Integrated Peripherals section of the BIOS setup.

However, it may be necessary to change it back to SPP or EPP mode for some printers to operate properly.

int +ve transition interrupt bit, see "interrupt drive"
Ien bit is internal interrupt enable, not used if polling used
1 = interrupt enable, 0 = disabled (polled operation)

note RXA and TXR pins are inverted drives from the register ie a 1 in the control register gives a 0 on the pin all other registers / bits are true.

Appendix B: *Sample SP2 Driver subroutines for a PIC Host*

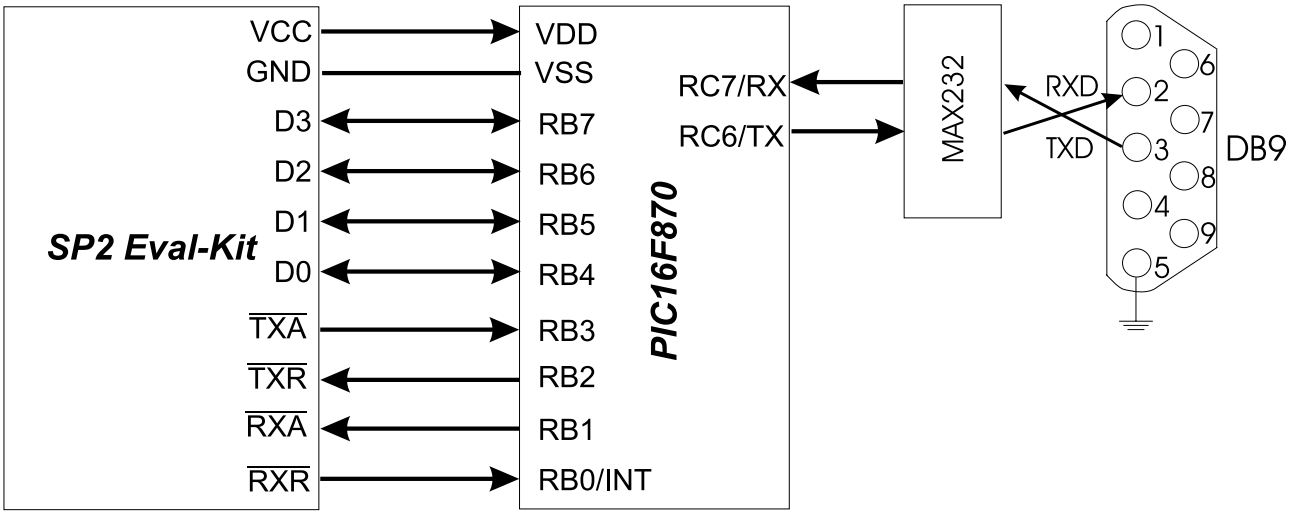

Figure 3: SpacePort Evaluation Kit interfaced to a PIC16F870 host microcontroller

SP2 Evaluation Kit can be interfaced to a Host PIC microcontroller as shown above. PIC16F870 has PortB with 8 I/O pins which can be assigned to communicate with SP2 Eval Kit. UART in PortC can be interfaced to a Serial COM Port via an RS232 driver.

The following subroutines may by used by a PIC16F870 host microcontroller to upload serial data it received from its serial port to SP2 and download the data packet from SP2 and send it out via its serial port to a PC.

OUT_BYTE & IN_BYTE
Additionally LISTEN_BUS is called on completion of a packet transfer to the SP2 to return the data bus to high impedance input mode (default state).

```
            list  p=16F870    r=hex
;
; STANDARD EQUATES - dedicated data file locations - PAGE 0
;
INDF         EQU     0            ; INDIRECT CALL , OPCODE WILL USE FSR
                                  (4H) AS FILE POINTER
STATUS       EQU     3            ; STATUS BITS
RP0          EQU     5            ; file page 0=PAGE 0 , 1=PAGE 1 e.g.
                                  DDR's etc.
FSR          EQU     4            ; file pointer (indirect file address
                                  register)
;
PORTA        EQU     5            ; i/o port A - 5 Bits , Free for HOST
                                  applications program use.
;
SP2          EQU     6            ; USE PORT B ON PIC FOR SP2 INTERFACE

; Bit assignments for SP2 PORT
D7           EQU     7            ; Bi-Dir data , D3
D6           EQU     6            ; Bi-Dir data , D2
D5           EQU     5            ; Bi-Dir data , D1
D4           EQU     4            ; Bi-Dir data , D0
TXA          EQU     3            ; INPUT  , active low TX accept from SP2
TXR          EQU     2            ; OUTPUT , active low TX request to SP2
RXA          EQU     1            ; OUTPUT , active low RX accept to SP2
RXR          EQU     0            ; INPUT  , active low RX request from
                                  SP2,(interrupt if required)
;
;

; STANDARD EQUATES - dedicated data file locations - PAGE 1
;
TRISA        EQU     85           ; I/O direction reg ,portA 1= i/p  0= o/p
SP2_DDR      EQU     86           ; Data direction register for portB (SP2)
```

```
            ORG     0
ONRESET     GOTO    START         ; jump to main program
;
; Initialise PORT B to drive SP2.
;
START       BSF     STATUS,RP0    ; select Bank 1
            MOVLW   B'11111001'   ; TXR & RXA O/P, Rest as inputs
            MOVWF   SP2_DDR
            BCF     STATUS,RP0    ; select bank 0
;

;  SUBROUTINE - IN_BYTE
;
;    IN_BYTE  - READ A BYTE FROM THE SP2 INTO FILE (REGISTER) POINTED TO BY FSR
;               W IS DESTROYED
;
;          NOTE - THIS ROUTINE WILL HANG THE HOST UNTIL THE HOST
;                 COMPLETES THE TRANSFER OF TWO NIBBLES
;
;             - THIS SUBROUTINE CAN BE CONFIGURED TO RUN
;               AS PART OF AN INTERUPT HANDLER IF THE RXR
;               LINE FROM THE SP2 IS USED TO TRIGGER A HOST INTERUPT
;

IN_BYTE     BTFSC   SP2,RXR       ; WE GOT A RX REQUEST YET ?
            GOTO    IN_BYTE       ; NO , SO LOOP BACK AND WAIT
;
;              READ THE LS NIBBLE FROM THE SP2
            BCF     SP2,RXA       ; ACCEPT THE REQUEST (SET ACCEPT LOW)
;
AWAITDATA   BTFSS   SP2,RXR       ; HAS REQUEST GONE UP ? i.e. data is
                                  present
            GOTO    AWAITDATA     ; LOOP BACK TILL IT DOES
;
            NOP                   ; TIME DELAY TO ENSURE DATA STABLE BEFOR
                                  READ
            MOVF    SP2,W         ; READ THE LS NIBBLE FROM THE BUS
            BSF     SP2,RXA       ; TELL SP2 WE GOT NIBBLE (ACCEPT = 1)
            ANDLW   B'11110000'   ; JUST THE DATA
            MOVWF   INDF          ; SAVE LS NIBBLE IN TARGET FILE (VIA FSR)
            SWAPF   INDF          ; MOVE THE NIBBLE TO LS POSITION
;
;              NOW GET MS NIBBLE FROM THE SP2
INNIBBLE    BTFSC   SP2,RXR       ; WE GOT NEXT RX REQUEST YET ?
            GOTO    INNIBBLE      ; NO , SO LOOP BACK AND WAIT
;
            BCF     SP2,RXA       ; ACCEPT REQUEST (SET ACCEPT LOW)
;
AWAITD1     BTFSS   SP2,RXR       ; HAS REQUEST GONE UP ? i.e. data is
                                  present
            GOTO    AWAITD1       ; LOOP BACK TILL IT DOES
;
            NOP                   ; TIME DELAY TO ENSURE DATA STABLE BEFORE
                                  READ
            MOVF    SP2,W         ; READ THE MS NIBBLE FROM THE BUS
            BSF     SP2,RXA       ; TELL SP2 WE GOT NIBBLE (ACCEPT=1)
            ANDLW   B'11110000'   ; JUST THE DATA
            IORWF   INDF          ; COMBINE MS NIBBLE WITH LS NIBBLE
                                  ;ALREADY IN THE FILE (VIA FSR)
            RETURN
;      A BYTE HAS BEEN READ FROM THE SP2 INTO ADDRESS POINTED AT BY FSR
;-----------------------------------------------------------------------
```

```
; SUBROUTINE - OUT_BYTE
;
;      OUT_BYTE  - WRITE A BYTE FROM FILE POINTED TO BY FSR TO SP2
;                  W IS DESTROYED
;
;      NOTE - THIS ROUTINE WILL HANG THE HOST UNTIL THE SP2
;             ACCEPTS THE TRANSFER OF TWO NIBBLES
;
;      WARNING - OUT_BYTE WILL SET THE DATA BUS TO DRIVE AFTER DETECTING
;                A TXA FROM THE SP2.
;                THE CALLING ROUTINE MUST SET 4 DATA LINES BACK TO I/P
;                ON COMPLETION OF PACKET TRANSFER (i.e. call LISTENBUS)
;
OUT_BYTE     SWAPF   INDF,W       ; GET LS NIBBLE FROM FILE (VIA FSR) INTO
                                  ; BITS 4 to 7 of W
             ANDLW   B'11110000'  ; JUST THE NIBBLE
             IORLW   B'00000010'  ; SET TXR LOW, LEAVE RXA HIGH
             MOVWF   SP2          ; SET TXR LOW , OUTPUT NIBBLE
WACCEPT      BTFSC   SP2,TXA      ; WE GOT A TX ACCEPT BACK YET ?
             GOTO    WACCEPT      ; NO , SO LOOP BACK AND WAIT
;
;       WE GOT ACCEPTANCE SO IT'S OK TO DRIVE BUS
             BSF     STATUS,RP0   ; SELECT PAGE 1
             MOVLW   B'00001001'  ; DRIVE BUS
             MOVWF   SP2_DDR
             BCF     STATUS,RP0   ; SELECT PAGE 0 BUS IS NOW DRIVING
;
             BSF     SP2,TXR      ; REMOVE REQUEST, DATA IS ON BUS
WDUN         BTFSS   SP2,TXA      ; HAS DATA BEEN READ ?
             GOTO    WDUN         ; WAIT TILL SP2 REMOVES ACCEPT
;

;        LS NIBBLE OF (FSR) IS SENT , NOW DO MS NIBBLE
             MOVF    INDF,W       ; GET MS NIBBLE FROM FILE (VIA FSR)
             ANDLW   B'11110000'  ; JUST THE MS NIBBLE
             IORLW   B'00000010'  ; SET TXR LOW (BIT 2), RXA STAYS HIGH
             MOVWF   SP2          ; OUTPUT NIBBLE + TXR LOW

WACCEPT1     BTFSC   SP2,TXA      ; WE GOT A TX ACCEPT BACK YET ?
             GOTO    WACCEPT1     ; NO , SO LOOP BACK AND WAIT
             BSF     SP2,TXR      ; REMOVE REQUEST, DATA IS ON BUS

WDUN1        BTFSS   SP2,TXA      ; HAS DATA BEEN READ ?
             GOTO    WDUN1        ; WAIT TILL SP2 REMOVES ACCEPT
;
             RETURN               ;    BYTE IS SENT TO SP2
;---------------------------------------------------------------------
; SUBROUTINE - LISTEN_BUS , SET DATA BUS TO INPUT
;
LISTEN_BUS   BSF     STATUS,RP0   ; SELECT PAGE 1
             MOVLW   B'11111001'  ; BUS TO INPUT
             MOVWF   SP2_DDR
             BCF     STATUS,RP0   ; SELECT PAGE 0
             RETURN
;               BUS IS LISTENING TO SP2
;---------------------------------------------------------------------
             END
```

Radiometrix Ltd

Hartcran House
231 Kenton Lane
Harrow, Middlesex
HA3 8RP
ENGLAND
Tel: +44 (0) 20 8909 9595
Fax: +44 (0) 20 8909 2233
sales@radiometrix.com
www.radiometrix.com

Copyright notice

Limitation of liability

The information furnished by Radiometrix Ltd is believed to be accurate and reliable. Radiometrix Ltd reserves the right to make changes or improvements in the design, specification or manufacture of its subassembly products without notice. Radiometrix Ltd does not assume any liability arising from the application or use of any product or circuit described herein, nor for any infringements of patents or other rights of third parties which may result from the use of its products. This data sheet neither states nor implies warranty of any kind, including fitness for any particular application. These radio devices may be subject to radio interference and may not function as intended if interference is present. We do NOT recommend their use for life critical applications.
The Intrastat commodity code for all our modules is: 8542 6000.

R&TTE Directive

After 7 April 2001 the manufacturer can only place finished product on the market under the provisions of the R&TTE Directive. Equipment within the scope of the R&TTE Directive may demonstrate compliance to the essential requirements specified in Article 3 of the Directive, as appropriate to the particular equipment.
Further details are available on The Office of Communications (Ofcom) web site:
http://www.ofcom.org.uk/licensing_numbering/radiocomms/licensing/licensing_policy_manual/

Information Requests
Ofcom
Riverside House
2a Southwark Bridge Road
London SE1 9HA
Tel: +44 (*0)845 456 3000 or 020 7981 3040*
Fax: +44 (0)*20 7783 4033*
information.requests@ofcom.org.uk

European Radiocommunications Office (ERO)
Peblingehus
Nansensgade 19
DK 1366 Copenhagen
Tel. +45 33896300
Fax +45 33896330
ero@ero.dk
www.ero.dk