

Communicating RobotBASIC With The Basic Stamp (BS2)

With the BS2 there are two ways to do serial I/O. The first way is through Pins 1 and 2. Pin 1 is the Output (from the BS2) and Pin 2 is the Input (to the BS2). These pins are usually the ones used by the programming board Like the Education board or the Professional Development Board (PDB). Even if you have the USB versions of these boards the USB still eventually go to these pins. **It is suggested you use the Professional Development Board for reasons that will become clear further down.**

Note: Pin numbers are the physical pin numbers on the chip starting with the top left pin if you are holding the chip with the notch up (**see the BS2 manual**).

The advantage of these pins is that they support the RS232 standard voltages and can be connected directly to a PC serial port. Unfortunately, you cannot use these pins for communicating to other systems if you are programming the BS2 at the same time. If you are going to use the BS2 to do communications with the PC with another program like *Hyper Terminal* or *RobotBASIC* at the same time as you are programming the BS2 or if you want to see BS2 DEBUG statements while it is running, you cannot use these pins to connect to the PC except to the PBasic programming environment.

If you are not programming the BS2 then you have these pins available for serial I/O if you wish. However, this is not normally the case since most of the time you want to repeatedly **reprogram and debug** the BS2 while testing the serial I/O programming.

Therefore, generally you will not be able to use the Com port used to connect to the BS2 by the PBasic IDE except to program the BS2 and communicate with it from the IDE. This is usually Com1 on the PC.

Therefore, how do you communicate with the BS2 from RobotBASIC while you are also programming it from the PBasic IDE?

The answer is in two parts:

- 1- You need to have **two** com ports on your PC.
If you do not have two com ports see the three notes further down.
- 2- You need to use two of the I/O pins of the BS2 (physical pins 5 to 20). These are normally referred to as P0, P1 etc up to P15.

However, there is a small problem. These pins cannot be directly connected to a serial port on the PC. They are TTL (0 to +5Volts) while the Serial port on the PC requires from -12V to +12V.

This is where the Professional Development Board comes in handy. It has two 9 pin serial connections. One is used to program the BS2 while it is sitting in the board and the other can be connected to any two I/O pins to do serial Send/Receive to a PC. This second port has the necessary circuitry to do serial voltage levels and you can connect the TTL I/O pins to it directly without any need for additional components.

If you do not have a PDB, you can still achieve a connection with other boards but you have to set them up according to the manual. In the BS2 manual Version 2.2 on page 395 in the description of the **SERIN** command there is a diagram as well as an explanation of how you can use a resistor to be able to connect a normal TTL BS2 I/O pin to the serial port of a PC. ***You must read this carefully; if you do not you can damage the BS2.*** You also need to read the description on page 417 under the **SEROUT** command as well as page 410. ***As a matter of fact you should read pages 393 to 429 carefully.***

Once you have the above knowledge you will be able to connect two I/O pins from the BS2 to the serial port of the PC.

Now, let's say you have chosen P0 as the input pin from the PC's Tx pin to the BS2. Also you have chosen P1 as the output pin from the BS2 to the Rx pin on the PC's serial port. Thus, now you can send serial data from the BS2 to the PC on P1 (remember this is actual physical pin 6 on the chip) and you can receive serial data from the PC into the BS2 on P0 (physical pin 5).

Additionally, let's assume that the serial port you are using is Com2; remember Com1 is being used by the PBasic IDE to program the BS2 and to also communicate with it when you use the DEBUG command in PBasic.

Using the above setup, a RobotBASIC program will be sending and receiving data on Com2, and a BS2 program will be sending data on P1 and receiving data on P0.

Let's write a program in RB that will sit waiting for the BS2 to send it **n** bytes and once it receives those bytes it sends out **m** bytes to the BS2 which will be waiting for them (remember the BS2 does not have a receive buffer so RB must ensure that it does not send the bytes before the BS2 is ready for them).

When the BS2 receives the **m** bytes it sends **q** bytes back to the PC which will be waiting for them but this time with a time out (just to show how it is done). The PC will then repeat the whole process and so will the BS2. Also the BS2 will print what it is doing using the DEBUG command to the BS2 Debug Window in the PBasic IDE and the RB program will print on the RB Terminal Screen any bytes that have arrived from the BS2 to show that it is getting the bytes.

Note: The demo below is just to test the connectivity setup described above. For a more in depth discussion on how to implement serial I/O that can send binary as well as text data between RobotBASIC and the BS2 with more complex examples see [RobotBASIC Serial IO.pdf](#).

To summarize:

- 1- Connect the BS2 to Com1 and make sure that the PBasic IDE is able to see it and is able to program it and communicate with it.
- 2- Connect P0 to the Tx pin and P1 to the Rx pin of the PC's Com2 using the appropriate circuit as described in the manual. **WARNING!!!! do this right!!!**
- 3- Write the RB program below and run it.
- 4- Write the BS2 program below **(V1)** and upload it to the BS2 which will also run it. If you want to rerun it just press the reset push-button on the board.
- 5- Make sure the RB program is running before you run the BS2 program. This is due to the BS2 not having a receive buffer.

Note1: You can use the two programs below without the BS2 being connected to the PBasic IDE if you do not have two Com ports on your PC. What you need to do is program the BS2 with the program below **(V2)** and **then close the PBasic IDE**. This version ensures that the BS2 will use physical pin 2 for input and 1 for output. **Also in the RB program change the number assigned to CommPortNo from 2 to 1 since now RB will be using Com1**. Then run the RB program and then reset the BS2 to start it running its program from the beginning.

Note2: If you do not have any serial ports or if you have only one, you can use the USB ports on the PC to create serial ports using the [Parallax USB to Serial \(RS-232\) Adapter](#) Part#28030 from Parallax Inc. (www.Parallax.com). With this option you can use the USB version of a programming board (e.g. Education Board) to program the BS2 and at the same time communicate with RobotBASIC using the USB to serial converter above.

This device will setup a *virtual* com port. This port is easily accessible from RobotBASIC, all you have to do is know its number and assign that number to the variable **CommPortNo** in the RB program below. The number is indicated when you install the USB converter.

Note3: Another way you can communicate with the BS2 while also programming it, is to use a wireless Bluetooth connection with the [Easy Bluetooth Module](#) Part#3085 from Parallax. With this module installed on the BS2, and a Bluetooth transceiver on the PC (e.g. D-Link's USB Bluetooth adaptor DBT-120), you can send and receive serial data between the PC and the BS2 wirelessly.

The Bluetooth adaptor on the PC will setup a *virtual* com port. This port is easily accessible from RobotBASIC, all you have to do is know its number and assign that number to the variable **CommPortNo** in the RB program below. The number is indicated when you install the USB converter or you can look it up through the Bluetooth manager on the PC.

The Easy Bluetooth Module on the BS2 communicates with the BS2 through its I/O pins and when it receives serial data from the PC it will give it to the BS2 and will take data from the BS2

and send it to the PC. The net result is as if you have a wired serial connection between the PC and BS2.

The RobotBASIC Program:

```
CommPortNo = 2 //change this number to the port you need
//The settings below reflect the settings in the BS2 program
setcommport CommPortNo,br9600,8,pNone,sbOne,fcNone
while true
  //make sure no previous data in the buffer
  clearserbuffer
  repeat
    //wait until there are 27 bytes in the buffer
    checkserbuffer n
  until n == 27
  //read the buffer and print it
  SerIn s \ print s
  delay 1000
  //above delay is to make sure the BS2 is ready for receiving.
  //The delay can be made smaller if you know for sure it is OK
  serout "Hi There" //send to the BS2 8 bytes
  SerBytesIn 28,s,n //wait for 28 bytes to arrive but with
                    //a time out just to show you how it is
                    //done. But the previous method is usually
                    //better with the BS2
  print s //print the arrived bytes
wend //repeat the whole action infinitely
```

The BS2 program V1

```
' {$STAMP BS2}
' {$PBASIC 2.5}
tt VAR Byte(8) 'input data
Do
  'display to the debug window
  DEBUG "Outputting a string to PC",CR
  'Output 27 bytes to the PC
  SEROUT 1,84,["BS2: Please Enter 8 letters"]
  'display to the debug window
  DEBUG "Waiting for input from PC",CR
  'Wait forever for 8 bytes on P0 and the 84
  `means 9600 baud, 8 bit no parity and true voltage
  SERIN 0,84,[STR tt\8]
  'display to the debug window
  DEBUG "Got 8 characters from PC here they are...",STR tt\8,CR
  'output 28 bytes to the PC on P1
  SEROUT 1,84,["BS2: Your Input is: ",STR tt\8]
LOOP 'repeat the whole action
```

The BS2 program V2

```
' {$STAMP BS2}
' {$PBASIC 2.5}
tt VAR Byte(8)   'input data
Do
  'Output 27 bytes to the PC
  SEROUT 16,84,["BS2: Please Enter 8 letters"]
  'Wait forever for 8 bytes on P0 and the 84
  'means 9600 baud, 8 bit no parity and true voltage
  SERIN 16,84,[STR tt\8]
  'output 28 bytes to the PC on P1
  SEROUT 16,84,["BS2: Your Input is: ",STR tt\8]
LOOP      'repeat the whole action
```