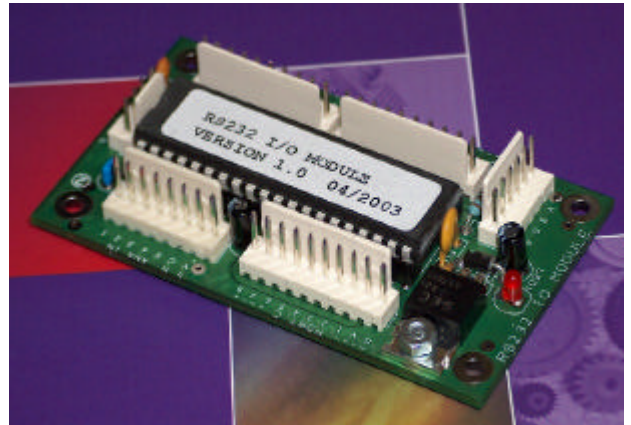


**RS232IO** is a general-purpose input and output module. It is connected to RS232 port of a computer. It features 24 digital programmable input/output lines and 5 analogue input lines (2.5V fully range and 8 bit A/D resolution). The I/O lines are organised into three 8-bit port, B, C and D. Each port can be configured as input (all lines are input) or output (all lines are output). The system uses 3 wires of RS232 port to communicate with host PC (TX, RX and Ground).



The system accepts various commands from the host through its RS232 port: These commands include: (1) to configure 24 input and output lines and to read back the configuration, (2) to output data or to read data into the computer, (3) to read analogue voltages into the host computer.

## Specification

Number of digital lines	24 lines (arranged in three 8-bit width ports: Port B, C and D) Each 8-bit port can be configured as input or output After Power-up, all digital I/O lines are configured as inputs
Voltage level	TTL ( <b>Low voltage inputs</b> )
Sink and source capability of I/O	25mA to sink and to source
Number of analogue input channels:	5 analogue voltage inputs
Input voltage range:	0 - 2.50V
Resolution:	8-bit (0.00977V)
Accuracy:	+/- 0.002V
Communication:	RS232 (TX, RX and Ground, no handshake)
Data packet format:	9600, n, 8, 1 or 9600, n, 8, 2
Command set	Setup ( <b>S</b> ), Get configuration ( <b>X</b> ), Get analogue voltages ( <b>A</b> ) Output data to digital ports ( <b>BOxxx</b> , <b>COxxx</b> , <b>DOxxx</b> ) Input data from digital ports ( <b>BI</b> , <b>CI</b> , <b>DI</b> ).
Input supply voltage:	7 to 12V DC
Power consumption active:	6.0mA maximum @ 7V
On-board fuse protection	500mA (resetable fuse)

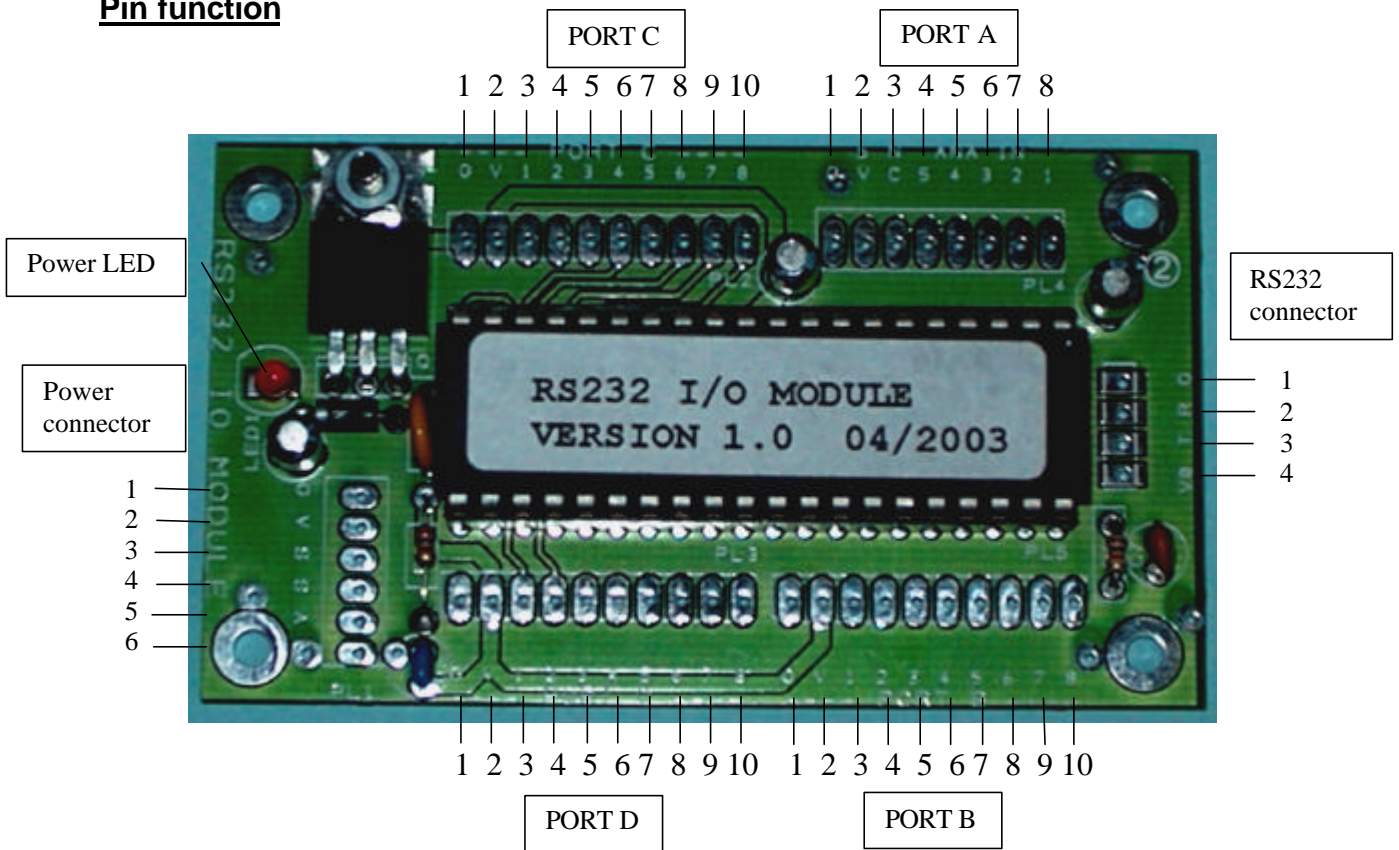
## Important Note on power-up

After power up of the module, all outputs are configured as INPUTS. This is to prevent any I/O conflict after reset. If you require a specific I/O setting, you should set up the I/O lines using the "S" command.

## Notice

There is no opto isolation on the RS232 port. Applying high voltage to the module may cause permanent damage to the module and to the host computer. It may also result in electrical shock to users. Only TTL level signals is allowed. The supplier is not responsible for any damage caused to the card and host PC.

## Pin function



PORT A Analogue Input		PORT B Digital I/O		PORT C Digital I/O		PORT D Digital I/O	
		10	Input/Output (DB7)	10	Input/Output (DB7)	10	Input/Output (DB7)
		9	Input/Output (DB6)	9	Input/Output (DB6)	9	Input/Output (DB6)
8	Analogue input #1	8	Input/Output (DB5)	8	Input/Output (DB5)	8	Input/Output (DB5)
7	Analogue input #2	7	Input/Output (DB4)	7	Input/Output (DB4)	7	Input/Output (DB4)
6	Analogue input #3	6	Input/Output (DB3)	6	Input/Output (DB3)	6	Input/Output (DB3)
5	Analogue input #4	5	Input/Output (DB2)	5	Input/Output (DB2)	5	Input/Output (DB2)
4	Analogue input #5	4	Input/Output (DB1)	4	Input/Output (DB1)	4	Input/Output (DB1)
3	Ground	3	Input/Output (DB0)	3	Input/Output (DB0)	3	Input/Output (DB0)
2	+5V	2	+5V	2	+5V	2	+5V
1	Ground	1	Ground	1	Ground	1	Ground

RS232 CONNECTOR		POWER CONNECTOR	
		6	LED -
		5	LED +
4	No connection	4	Connected to power switch
3	Transmit data to PC (Pin 2 of RS232 connector on PC, Orange cable)	3	Connected to power switch
2	Receive data from PC (Pin 3 of RS232 connector on PC, Red cable)	2	Supply input (7-12V DC)
1	Ground (Pin 5 of RS232 connector on PC) Black or gray cable	1	Ground

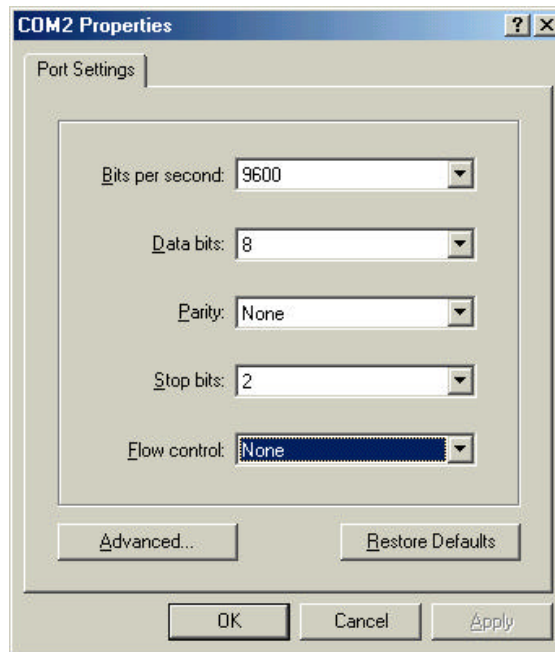
## Command set

The RS232IO module accepts the following commands. Once a full command is transmitted into the micro-controller, the micro will response to the command.

Command set	Descriptions
<b>S</b> X <sub>B</sub> X <sub>C</sub> X <sub>D</sub> X=0 to set as Output X=I to set as Input  Examples: S000, S00I, SIII	<b>(4 character command)</b> X <sub>B</sub> X <sub>C</sub> X <sub>D</sub> are either "O" or "I" to set the corresponding port B,C or D as output or input. If the setup command is accepted by the module, it will send back the configuration confirmation to the host computer. For example S0II will configure Port B as output, Port C as input and Port D as input. Reply from the module will be PB=O PC=I PD=I<return><new line>
<b>X</b>	<b>(1 character command)</b> To check the current I/O configuration of Port B, Port C and Port D. Reply from the module will be PB=O PC=I PD=I<return><new line>
<b>BOXXX</b> <b>COXXX</b> <b>DOXXX</b> (XXX=000 to 255)  Examples: BO000 CO001 DO012 BO255 CO128 DO255	<b>(5 character command)</b> If a port is configured as output, you can use these commands to output a byte defined by XXX (000 to 255). "B,C and D" in the command stand for Port B, Port C and Port D. "O" stands for Output. For example, BO123 will output byte 123 to Port B. Reply from the module will be 123<return><new line>  Following BO command, you must put 3 numbers such as 000, 001, 008, 011, 099, even if the value sent to the port has one or two digits.  If a port is configured as output, you can still read data from these ports using the following commands.  If a port is configured as input, you can not output data to that port.
<b>BI</b> <b>CI</b> <b>DI</b>	<b>(2 character command)</b> If a port is configured as input (or output), you can use these commands to input a byte. "B, C and D" in the command stand for Port B, Port C and Port D. "I" stands for input. For example, BI will input a byte from Port B. The input value will be a value ended with <return><new line> such as 123<return><new line>
<b>A</b>	<b>(1 character command)</b> This command reads in 5 analogue inputs in mV. Five mV values are separated by space " " and is ended with <return> <new line> such as 1500 1500 0 0 0<return> <new line>

## Appendix A Control the card using HyperTerminal

**Step 1:** The Baud rate, data bits, parity, stop bits and flow control are set as follows. Stop bits can be set to 2 for more reliable operations.



**Step 2:** Examples of commands (echo is turned on)

```

*****RS 232 CONTROLLER*****
S00I
PB=0 PC=0 PD=I
X
PB=0 PC=0 PD=I
A
0 1577 1323 0 0
B0255
255
C0128
128
DI
0
BI
255
                    
```

**Power up message**

Configure PB as output, PC as output and PD as input. The module replies with  
PB=0 PB=0  
PD=I<return><new line>

**Get I/O configuration**

Read 5 analogue inputs. The module replies with 5 voltages in mV. Each separated by a space and ended with <return><new line>

**Output 255 to Port B.**  
The module replies with  
255<return><new line>

**Output 128 to Port C.**  
The module replies with  
128<return><new line>

**Input from Port D, the module replies with**  
0<return><new line>

**Input from Port B, the module replies with**  
255<return><new line>

**Appendix B**  
**Control the card using VB**

If you use a programming language to control the module, please configure the RS232 port to the following: 9600,n,8,2 (2 stop bits).

The following example in VB will write a byte 255 into port D using the command DO255. A command line `MSComm1.Output = "DO255"` is used to do this. After this the program checks the echo sent back by the module. The DO...LOOP check the 3 bytes sent back from the module.

Program example list:

```
MSComm1.InBufferCount = 0
MSComm1.Output = "DO255"
Do
    DoEvents
Loop Until MSComm1.InBufferCount >= 3
```

Explanation:

```
`reset the InBufferCount
`output a command to the module
`receive 3 bytes from the module
```