

Introduction

The PINK was originally sold as a network interface for the Parallax basic stamp module. However, it can be used with any microprocessor that has a UART serial interface and we now refer to this module as a "serial enabled web server".

While the serial enabled web server application can be run on any NetBurner module, it is typically used with two: the SB70 and SB70LC. Both of these module are part of the NetBurner Serial to Ethernet product line. The major differences between these two are: the SB70 has 5V tolerant UART signals, while the SB70LC is a 3.3V only device. Second, the SB70LC is lower cost than the SB70. Other than that, the same application is run on both devices.

This document will describe the hardware included with both the SB70 and SB70LC. The software description is identical for both devices.

Cautions

Be sure to observe static precautions and be sure power is off when handling and/or connecting the PINK Module to other equipment.

Do not set the PINK IP Address to a Multicast Address such as XXX.XXX.XXX.255. Valid IP Address Ranges for Subnet Mask 255.255.255.0 in Class C are 192.168.X.1 through 192.168.X.254. For Class A ranges 10.10.X.1 through 10.10.X.254 are valid.

The PINK Serial Ports cannot be connected to a PC RS-232 Port without using a line driver IC such as a MAX232. The PINK Serial Ports are TTL level non-inverted whereas the PC uses inverted RS-232 level signals. Direct connection will damage your PINK Module.

PINK (5V) / PINKLC (3.3V) Development Kit

General Description

The PINK/PINKLC (based on the NetBurner SB70/SB70LC hardware) is an Embedded Web Server which enables any microprocessor with a UART serial port to communicate over a network or even the Internet. You can send email or UDP messages to alert other devices to events or provide status. Communication with the host controller is done using a simple TTL serial interface.

Features

- Embedded Web Server with 192K Flash Memory for web pages and files
- FTP interface for uploading web pages and files
- Telnet interface for debugging serial communication
- Send/Receive UDP Packets on any port
- Send Email with or without SSL authentication
- Password protection of configuration, FTP and selected web pages and files
- Over 100 web accessible variables, allowing for dynamic web pages
- Over 20 variables can be written to flash memory for default values
- Web and serial access to IP Address, Network Mask, Default Gateway and DNS Server configuration

Application Ideas

- Remote monitoring and control of projects or applications
- Short or long distance communication between multiple networked microcontrollers

Kit Contents

- (1) PINK or PINKLC Ethernet Module
- (1) Blue CAT5 Ethernet Cable
- (1) Grey Serial Cable
- (1) USB Power Cable
- (1) Adapter Board Connector Ribbon
- (1) Adapter Board
- (6) Hookup Wires



PINK (5V) Specifications

Symbol	Quantity	Minimum	Typical	Maximum	Units
Vdd	Supply Voltage [†]	-5%	5.0	+5%	V
Temp	Operating Temperature Range	0	-	70	C

[†] Suggested Power Supply Current is 350mA

Power Connector

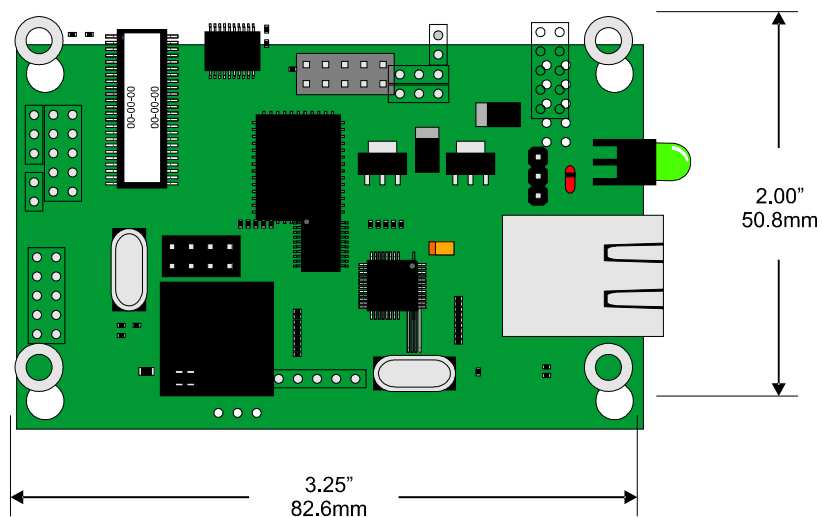
Pin	Name	Function
1	Vdd	-5%
2	Ground	0
3	Vdd	

[†] Suggested Power Supply Current is 350mA

Pin Definitions and Ratings (J5)

Pin	Name	Function
1	GND	Ground (Vss)
2	CTS1	TTL Serial Port 1 Flow Control (Debug Port)
3	TX1	TTL Serial Port 1 Transmit Data (Debug Port)
4	RX1	TTL Serial Port 1 Receive Data (Debug Port)
5	RTS1	TTL Serial Port 1 Flow Control (Debug Port)
6	RTS0	TTL Serial Port 0 Flow Control (Serial Port)
7	CTS0	TTL Serial Port 0 Flow Control (Serial Port)
8	TX0	TTL Serial Port 0 Transmit Data (Serial Port)
9	Vdd (+5V)	+5VDC Power
10	RX0	TTL Serial Port 0 Flow Control (Serial Port)

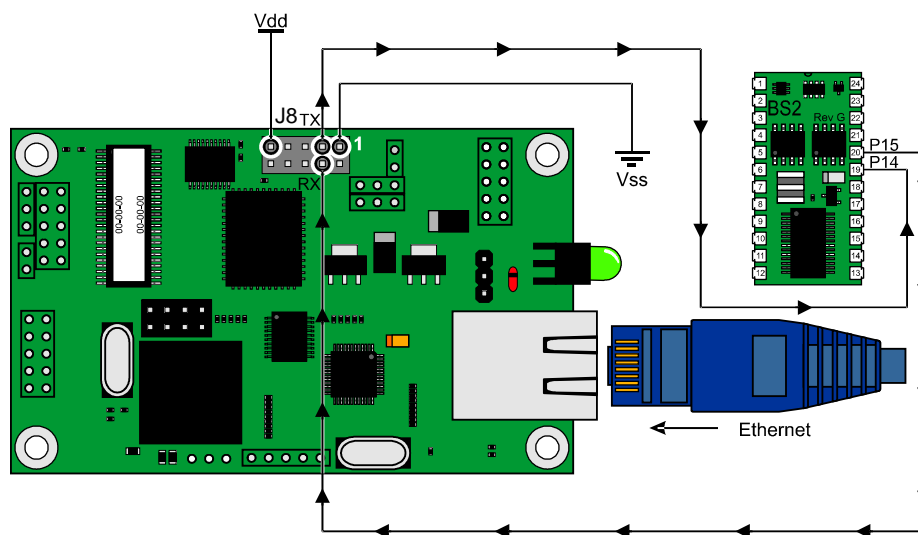
PINK Dimensions



LEDs

The large round right-angle LED is the power LED. On the RJ-45 connector LED1 is 10/100 (lit is 100) and LED2 is data activity.

PINK Quick Start Circuit



Connection Notes

Please be sure to read the following sections before using your module. The connection diagrams above show the connections used by the factory application. Be sure to observe the Serial Out (TX) and Serial In (RX) connections on the 2x5 header. Please double check all connections before applying power.

PINKLC (3.3V) Specifications

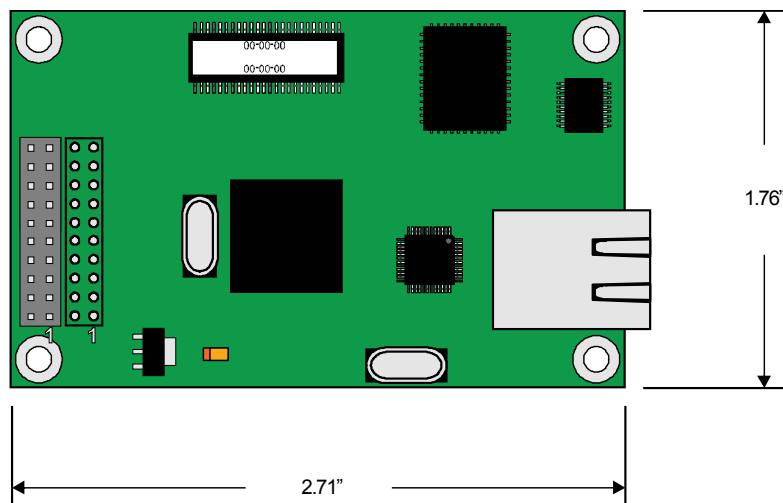
Symbol	Quantity	Minimum	Typical	Maximum	Units
Vdd	Supply Voltage [†]	-5%	3.3	+5%	V
Temp	Operating Temperature Range	-40	-	85	C

[†] Suggested Power Supply Current is 350mA\

Pin Definitions and Ratings (JP1)

Pin	Name	Function
1		
2	Vdd (+3.3V)	+3.3VDC Power
3		
4		
5	RTS0	TTL Serial Port 0 Flow Control (Serial Port)
6		
7		
8	CTS1	TTL Serial Port 1 Flow Control (Debug Port)
9	RTS1	TTL Serial Port 1 Flow Control (Debug Port)
10	TX0	TTL Serial Port 0 Transmit Data (Serial Port)
11	RX0	TTL Serial Port 0 Flow Control (Serial Port)
12	TX1	TTL Serial Port 1 Transmit Data (Debug Port)
13	RX1	TTL Serial Port 1 Receive Data (Debug Port)
14	CTS0	TTL Serial Port 0 Flow Control (Serial Port)
15		
16		
17		
18		
19		
20	GND	Ground (Vss)

Module Dimensions

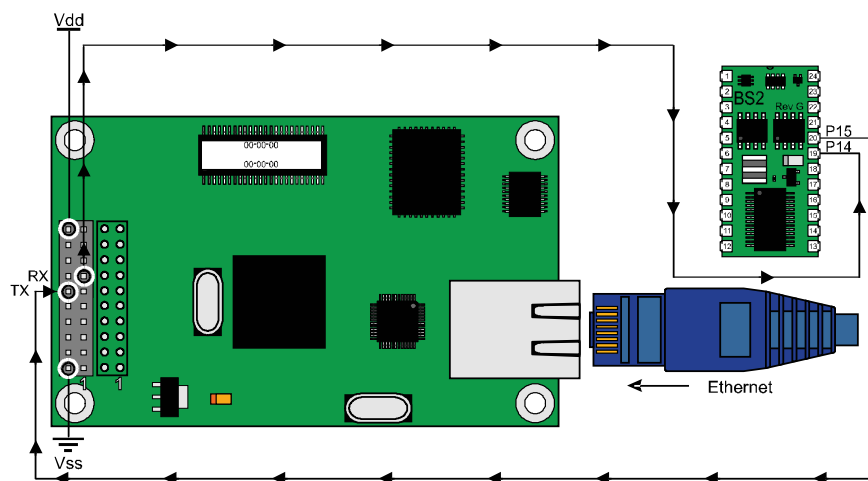


Please visit the NetBurner website SB70LC product web page for the a detailed SB70LC-100IR Mechanical Diagram.

LEDs

On the RJ-45 connector LED1 is 10/100 (lit is 100) and LED2 is data activity.

PINKLC Quick Start Circuit

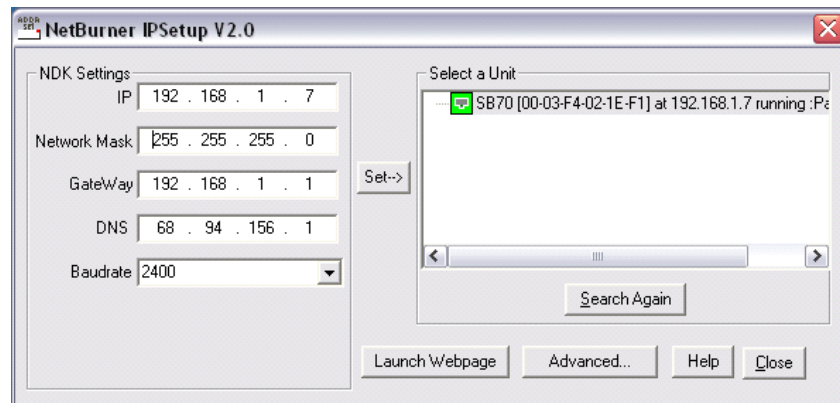


Connection Notes

Please be sure to read the following sections before using your module. The connection diagrams above show the connections used by the factory application. Be sure to observe the Serial Out (TX) and Serial In (RX) connections on the 2x5 header. Please double check all connections before applying power.

Network Setup

Connect the PINK Module to your network using the standard Ethernet cable. Once the PINK Module is connected to the same network as your PC and is powered on, run the IPSETUP program (located on the PINK product webpage at www.netburner.com). If the static IP address field is 0.0.0.0, DHCP will be used to get the IP address settings. To set a static IP address simply type in the values you wish to use and DHCP will automatically be disabled. You can also set the UART serial port baud rate to match the baud rate of the microprocessor that will be communicating with the PINK.



When you run the IPSETUP program your PINK Module should appear in the top right window (if it does not, see troubleshooting). Select your PINK Module by clicking on the SB70/SB70LC entry in the selection window. To set a Static IP, enter the IP Address, Subnet Mask, Gateway and DNS Server into the appropriate fields. For Dynamic addresses set all four fields to 0.0.0.0. Click the Set button to execute the configuration. If your PINK Module does not reappear in the list, click the Search Again button. Once configured, select the device again and click the Launch Webpage button to open a browser window to the newly assigned address of the PINK Module.

Web Configuration

When the web browser opens you should see the default Factory Page. Whenever you browse to the address of the PINK Module and there are no custom pages uploaded you should see the default factory web page. The address of this page is **XXX.XXX.XXX.XXX/nb_factory.htm** where the **XXX.XXX.XXX.XXX** is the IP Address of the PINK Module. From this page you can access the network settings, serial data port settings, variables, diagnostic and FTP functions by clicking on various links on the page.



The Factory Page lists, "5 quick things you can experiment with". Items 1 and 2 can be done from the configuration page so click on these links to see how the module can be configured from the web page interface.

5 quick things you should try out NOW!

1. Change the web server's network settings via a [web form](#).
2. Configure the [Serial Data Port Settings](#)
3. Familiarize yourself with the web server's [diagnostic functions](#).
4. Upload a file via [ftp to webserver](#)
5. [Download a file via ftp](#) using a web browser

Once you're on the configuration page be sure to set the IP Mode to *Static* or *Dynamic* matching how you set it up in the IPSETUP program. Address information will be displayed just below for either mode.

IP Mode (static or dynamic)

	DHCP Assigned Settings	Static Settings
IP Address	<input type="text"/>	192.168.1.7
Network Mask	<input type="text"/>	255.255.255.0
Gateway	<input type="text"/>	192.168.1.1
DNS Server	<input type="text"/>	68.94.156.1

Next you'll want to set the Serial Port to match IPSETUP as well. The most common configuration is data bits set to 8, Parity to None and Stop Bits to 1. For now, click the Submit New Settings button to save these changes.

Serial Data Port Settings

Baud Rate	<input type="text" value="2400"/>
Data Bits	<input type="text" value="8"/>
Data Parity	<input type="text" value="None"/>
Stop Bits	<input type="text" value="1"/>

If you are using a Parallax Device

You should now be able to run the **PINKTestV1.0.BS2** program which will test serial communication with the PINK Module.

FTP Password Protection

Setting the FTP Password will prevent anonymous FTP connections and it also password protects the configuration, modify variable and diagnostic pages. Be sure to choose a password you will remember. If you forget it you will be locked out of the PINK Module!

Configuration and FTP Password Settings

Configuration user name	<input type="text" value="username"/>	
Configuration password	<input type="password" value="*****"/>	Blank for no password
Configuration password(again)	<input type="password" value="*****"/>	

Web Interface Password Protection

Setting the Web Access Password will protect any files that contain the password "filter text". For example, if the filter text is "lock", then files containing that text, such as "locktest.htm" or "iconlock.jpg", will prompt for the correct username and password. Setting a username and password while leaving the filter field blank will protect all files.

Web access Password Settings

Web file user name	<input type="text"/>	
Web file password	<input type="password"/>	Blank for no password
Web file password(again)	<input type="password"/>	
Web file password filter	<input type="text"/>	Any web file served that has this text in the file name will be password protected (Blank for password all files)

Custom Web Pages

Custom Web Pages can be uploaded to the PINK Module via FTP. You can use a free FTP client such as WinSCP, or a web browser FTP plug-in.

Variables

The PINK Module has 100 general purpose variables and 18 dedicated system variables. Each variable can store up to 64 bytes max. The first 20 general purpose variables (0 through 19) can be written to flash memory as power-on defaults. The remaining general purpose variables (20 through 99) are RAM variables only and the data is lost when power is cycled.

The email and UDP system variables can also be written to flash memory as power-on defaults.

Web Variables

Variables 1-20 can be written to Flash memory. The remainder are RAM only.

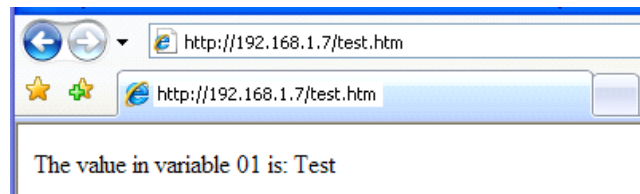
Variable name	Current Value	Description	Update RAM	Update Flash
Nb_var00=	[WELCOME!]	Web variable 00	Modify	Set Default Value
Nb_var01=	[NA]	Web variable 01	Modify	Set Default Value
Nb_var02=	[NA]	Web variable 02	Modify	Set Default Value

All variables can be accessed and modified through the Modify Variables web interface by browsing to **XXX.XXX.XXX.XXX/vv_Show.htm** (where **XXX.XXX.XXX.XXX** is the IP Address of the PINK Module). You set the default values for these variables which can be set to default (written to flash) by clicking the Set Default Value link. This page is a good resource to debug setting variables through the serial port since clicking refresh on this page will update what is currently in the variables.

Web Access to Variables

Accessing any variable within a web page is easy by using an HTML reference to the variable name exactly as it appears in the Modify Variables page. To create an example web page which displays the value of general purpose variable 01, open a text editor such as notepad and enter the following:

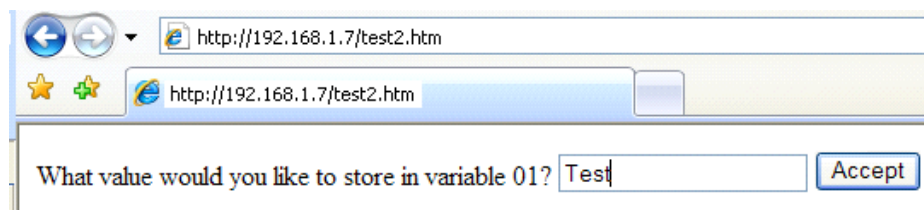
```
<html>
The value in variable 01 is: <Nb_var01>
</html>
```



Save the file as **test.htm**. Open an FTP connection as outlined previously and copy the file to the PINK Module. Browsing to **XXX.XXX.XXX.XXX/test.htm** (where **XXX.XXX.XXX.XXX** is the IP Address of the PINK Module) displays the web page you just created. Using the Modify Variables web page (in another window) you can change the value of variable 01 then refresh the original test browser page to see the results change. If you haven't set the variable it will most likely contain **NA**.

Writing to variables via web page can be accomplished with the HTML POST method. To create an example web page that changes the value of variable 01, open a text editor and enter the following:

```
<html>
<form method="post" action="test.htm"
<p>
What value would you like to store in variable 01?
<input name="Nb_var01" type="text" maxlength="64">
<input type="submit" value="Accept">
</p>
</form>
</html>
```



Save the file as **test2.htm** and copy the file to the PINK Module using the previously discussed FTP methods. Browsing to **XXX.XXX.XXX.XXX/test2.htm** (where **XXX.XXX.XXX.XXX** is the IP Address of the PINK Module) opens a form page that can be used to change the value in variable 01. Once the 'Accept' button is clicked the original **test.htm** page will be loaded to display the new value.

The *name* attribute specifies the variable that will be affected. The *maxlength* attribute can be used to limit the number of characters the form will accept. The *value* attribute allows you to change the text that appears on the submit button. The *action* attribute allows you to specify the page to load on submit.

Several HTML and JavaScript examples are included on the PINK Product web page, including those listed above.

Serial Access to Variables

To access variables through the serial port you must refer to the variable name by the last 2 letters used in the Modify Variables web page. To read the contents of a variable you would use the command format: **!NBORXX**

Note that **XX** should be replaced with the 2-letter name of the variable. For example, to read a single byte value from the general purpose variable 01 you would use the command format: **!NBOR01**

```
SEROUT TX, Baud, ["!NBOR01"]      ' Send the command
SERIN  RX, Baud, [nbvar]           ' Get the value back
```

To read the contents of the Current IP Address Status Variable (string) you would use the command format: **!NBORSI**

```
SEROUT TX, Baud, ["!NBORSI"]      ' Send the command
SERIN  RX, Baud, [STR nbvar\16\CLS] ' Input string into array
```

Note that in each of the above examples the **!NB** is followed by the number *zero*, *not* the letter O.

BASIC Stamp 2 program examples for reading variables are included on the PINK Product web page.

To write to variables you would use the command format: **!NBOWXX:DD**

Note that **XX** should be replaced with the 2-letter name of the variable and **DD** should be replaced with the data to be sent to the variable. The command must be followed with a 0 byte (null). This can be accomplished in the BASIC Stamp Editor by using the pre-defined constant **CLS**. The data will be stored as raw bytes so it doesn't matter what format it is in. The manner you send it is how it will be stored.

```
SEROUT TX, Baud, ["!NBOW01:100", CLS] ' Store ASCII string '100' in variable 01
```

In the above example ASCII values for 1, 0 and 0 are stored (string format) using 3 bytes. To store 150 as a raw binary value you would store the value directly as below (notice the value is not in quotes).

```
SEROUT TX, Baud, ["!NBOW02:", 150, CLS] ' Store value 150 in variable 02
```

...or via a variable...

```
SEROUT TX, Baud, ["!NBOW03:", nbvar, CLS] ' Store value in nbvar in variable 03
```

In this example the BASIC Stamp variable **nbvar** would contain a byte value (200 for example) which would be stored in the PINK general purpose variable 03. Had **nbvar** been a word variable, the lowbyte and highbyte would have to be stored as two bytes. BASIC Stamp 2 program examples for writing variables are included on the PINK Product web page. *Important Note* for serial communication with BASIC Stamps: The BS2, BS2e and BS2pe are not fast enough to communicate with the PINK Module serially at 9600 bps when transferring string data from the variables. It is for this reason that the default examples use 2400 bps. If you're using a BS2sx, BS2p or BS2px you can safely raise the serial port baud rate of the PINK Module to 9600 bps. Be sure to make the change in IPSETUP program (and click Set) as well as through the configuration web page (and click Submit New Settings).

BS2sx, BS2p and BS2px program examples for reading and writing variables are included on the PINK Product web page. Some of these examples make use of the BS2p Scratch Pad RAM for transferring large strings without using up variable memory. These examples also use 9600 bps.

System Variables

The PINK Module has 18 System Variables:

- **Nb_varET** is the recipient's TO email address
- **Nb_varEF** is the senders FROM email address. Note: Some ISP's require that this field match the account email address or username for authentication.
- **Nb_varES** is the email SUBJECT
- **Nb_varEC** is the email CONTENT. This variable will hold the actual email message body, which is a maximum of 64 bytes.
- **Nb_varEV** is the email SMTP Server. The Modify Variables web page has a link to Send Test Email. This allows you to test your email settings to verify they are correct.
- **Nb_varEU** is the email account USERNAME. This variable may be optional depending on your SMTP server.
- **Nb_varEP** is the email account PASSWORD. This variable may be optional depending on your SMTP server.
- **Nb_varEA** is the email AUTHENTICATION. This variable will determine whether authentication may or may not be used depending on your email provider. If the variable is blank authentication is not used. Any value/data stored here will enable authentication.
- **Nb_varST** is the PINK Status Register. This is a read-only variable. Reading this variable clears bits 1, 4 and 5.
 - **Bit0** is the Network Status bit. This bit is set if a network connection is detected and cleared otherwise.
 - **Bit1** is the Variable Update bit. This bit is set if a variable has been updated via the web page (HTTP POST method) since the last read of the Status Register.
 - **Bit2** is the Email busy status bit. This bit is set if the PINK is busy sending email and cleared when it is ready to send. This bit changes automatically but is cleared after the mail is sent or fails. This bit is *not* affected by the *Send Test Email* link in the Modify Variables web page.
 - **Bit3** is not used.
 - **Bit4** is the Email complete status bit. This bit is set when email was successfully sent. This bit will be cleared when the Status Register is read or when the next email fails.
 - **Bit5** is the UDP Message received status bit. This bit is set when a UDP Message is received. Reading the Status Register will clear this bit.
 - **Bit6** is not used.
 - **Bit7** is not used.
- **Nb_varSI** contains the Current IP Address assigned to the PINK Module.
!NB0RSI to read, !NB0WSI:<x.x.x.x> to write.
- **Nb_varSN** contains the Current Network Mask assigned to the PINK Module.
!NB0RSN to read, !NB0WSN:<x.x.x.x> to write.
- **Nb_varSG** contains the Current Default Gateway Address assigned to the PINK Module.
!NB0RSG to read, !NB0WSG:<x.x.x.x> to write.
- **Nb_varSD** contains the Current DNS Address assigned to the PINK Module.
!NB0RSD to read, !NB0WSD:<x.x.x.x> to write.

- **Nb_varSU** contains the Address of the Last Incoming UDP Message to the PINK Module. This is a read-only variable.
- **Nb_varSV** is the Last Post Variable Number. This is a read-only variable. This variable contains the number of the last general purpose register that was updated via the web (HTTP POST method). You should read this variable whenever Bit 1 of the Status Register is set. The value should be between 0 and 99. This register is set to \$FF after a read or if no variables have been updated.
- **Nb_varBI** contains the Destination IP Address for sending a UDP Message.
- **Nb_varBM** contains the UDP Message Content. Setting this prior to sending a UDP Message will send the specified data once the Send UDP Message command has been executed. If a UDP Message is received, its contents will be stored here erasing any previous data. Be sure to store your out-going message again before sending if you have received a UDP Message.
- **Nb_varBP** contains the UDP Message Port. The factory default value is 10000. This value can be changed to any valid port number. You can also change the default value.

All variables that are not Read-Only can be changed via the serial interface or via the Modify Variables web page. All of the Email variables can also be set to *default* values using the Modify Variables web page, as can the UDP Variables and the first 20 general purpose variables. These variables *cannot* be set to *default* values via the serial interface or via HTML command, only the Modify Variables web page.

Most of the System Variables contain string data and can be read as such using the same format as when reading the general purpose variables as a string. The Status Register and the Last Post Variable can, and should be read as a binary value by omitting the 'R' from the command.

```
SEROUT TX, Baud, ["!NB0ST"]      ' Send the command (no R)
SERIN  RX, Baud, [nbvar]          ' Get the raw value back (one byte)
```

```
SEROUT TX, Baud, ["!NB0SV"]      ' Send the command (no R)
SERIN  RX, Baud, [nbvar]          ' Get the raw value back (one byte)
```

If you read these registers as string data you will need to use a HEX formatter to decode the incoming value as shown below. NOTE: This method will not clear the Last Post Register.

```
SEROUT TX, Baud, ["!NB0RST"]      ' Send the command (normal read)
SERIN  RX, Baud, [HEX nbvar]      ' Get ASCII string into variable as value
```

Changing the HTTP Port Number

The default HTTP port number for the web server is 80. The port number can be read and/or modified with the following commands:

```
!NB0RHP      Read http port number.
!NB0WHP:<port> Set http port number. Must be between 1024 and 65535.
!NB0SR      Reset the device.
```

Note that a reboot of the device is required for a new port number to take effect.

Sending Email

The PINK Module can send email messages to notify you of events or for other purposes. To send an email, you can preset the email variables to *Default Values* in the Modify Variables web page if the data does not need to change. To send an email with the BASIC Stamp, all the required variables should be set and then the Send Mail command should be sent as shown in the example below. NOTE: You should substitute the actual data below with your own email address and server information.

```
SEROUT TX, Baud, ["!NB0WET:someone@somewhere.com", CLS]
SEROUT TX, Baud, ["!NB0WEF:PINKsmtp@gmail.com", CLS]
SEROUT TX, Baud, ["!NB0WES:Test Message From PINK", CLS]
SEROUT TX, Baud, ["!NB0WEC:Message Content Goes Here!", CLS]
SEROUT TX, Baud, ["!NB0WEV:smtp.gmail.com", CLS]
SEROUT TX, Baud, ["!NB0WEU:PINKsmtp@gmail.com", CLS] ' (username)
SEROUT TX, Baud, ["!NB0WEP:PINKpassword", CLS] ' (password)
SEROUT TX, Baud, ["!NB0WEA:S", CLS] ' Turn Authentication On (Optional)
SEROUT TX, Baud, ["!NB0WEW:465", CLS] ' Turn Authentication On (Optional)
SEROUT TX, Baud, ["!NB0SM"] ' Command To Send Email
```

You can then send your message by simply sending the Send Mail command shown below.

```
SEROUT TX, Baud, ["!NB0SM"] ' Command To Send Email
```

Once the Send Mail command has been sent you should immediately start monitoring bit 2 of the Status Register for it to be cleared (set to 0). Once this happens you should immediately check bit 4. If this bit is set (to 1) then the email was sent successfully. If it is cleared (to 0) then the email was not sent (failed). Bit 4 will only be valid on the read where bit 2 was cleared. An additional read will not be correct since the previous read will have cleared the Status Register. BASIC Stamp 2 program examples for sending email are included on the PINK Product web page.

Sending UDP Messages

The PINK Module can send and receive UDP (User Datagram Protocol) messages. These could be used to send and receive command information, status updates or for other purposes. A UDP Message can be sent or received but unlike email there is no way to know if it was successfully received by the target when sent. The only way to be sure your UDP Message gets to its target would be to have the target acknowledge (ACK) with a return UDP Message. This is relatively easy since the Nb_varSU register contains the IP Address of the last incoming UDP Message.

To send a UDP message you must first specify the destination IP Address, UDP Port and message content, then send the Send UDP Message command as shown below. In the example below you would replace the BI data with your target IP Address and the BM data with your own message data, which could contain values or ASCII text (up to 64 bytes).

```
SEROUT TX, Baud, ["!NB0WBI:192.168.1.6", CLS]
SEROUT TX, Baud, ["!NB0WBM:Please Send Me A Reply To Port 10000!", CLS]
SEROUT TX, Baud, ["!NB0WBP:10000", CLS]
SEROUT TX, Baud, ["!NB0SB"] ' Send UDP Message
```

These variables could be set to *Default Values* in the Modify Variables page if the data does not change. If the variables are already set, you can simply send the Send UDP Message command as shown below.

```
SEROUT TX, Baud, ["!NB0SB"] ' Send UDP Message
```

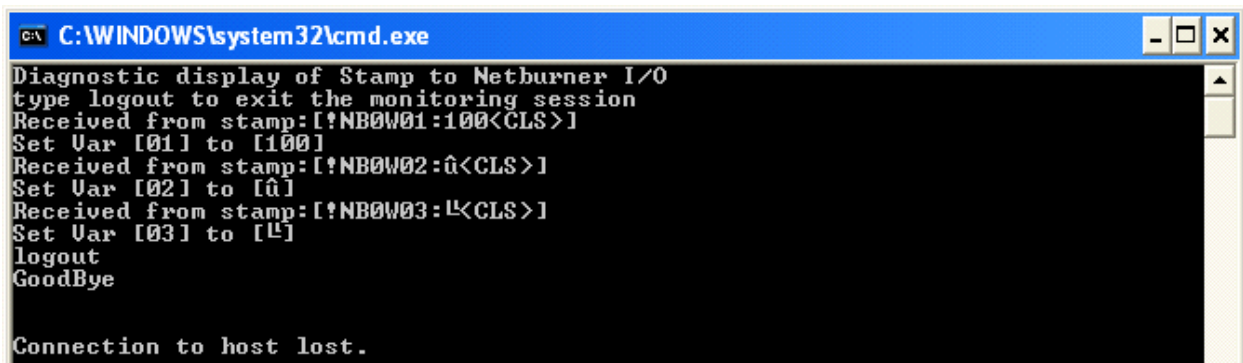
Note that incoming UDP Messages overwrite the outgoing message. Keep this in mind when sending UDP Messages and not setting the Message Content variable.

Telnet Debug Session

To monitor serial traffic between the PINK Module and the microcontroller you can establish a Telnet Session with the PINK Module. To do this using a telnet client, open a command prompt and type:

telnet XXX.XXX.XXX.XXX (where **XXX.XXX.XXX.XXX** is the IP Address of the PINK Module)

This will open a window that displays all serial communication between your microprocessor PINK Module. Below is a sample telnet debugging session when running the PINKWriteVariable.bs2 program.



```
C:\WINDOWS\system32\cmd.exe
Diagnostic display of Stamp to Netburner I/O
type logout to exit the monitoring session
Received from stamp:[!NB0W01:100<CLS>]
Set Var [01] to [100]
Received from stamp:[!NB0W02:û<CLS>]
Set Var [02] to [û]
Received from stamp:[!NB0W03:<CLS>]
Set Var [03] to []
logout
GoodBye
Connection to host lost.
```

As you can see the window shows the !NB0W01:100 command sent from the BASIC Stamp and terminated with a NULL (shown as <CLS>) as well as a message showing that Var 01 was set to 100. This information can be useful for debugging as well as other purposes. You can also see that the next two variables were set to binary values and result in strange characters being printed which represent those byte values as a printable ASCII character.

Resources and Downloads

Check out the PINK product page at www.netburner.com for example programs, example HTML/JavaScript, utility software and more.

Specifications

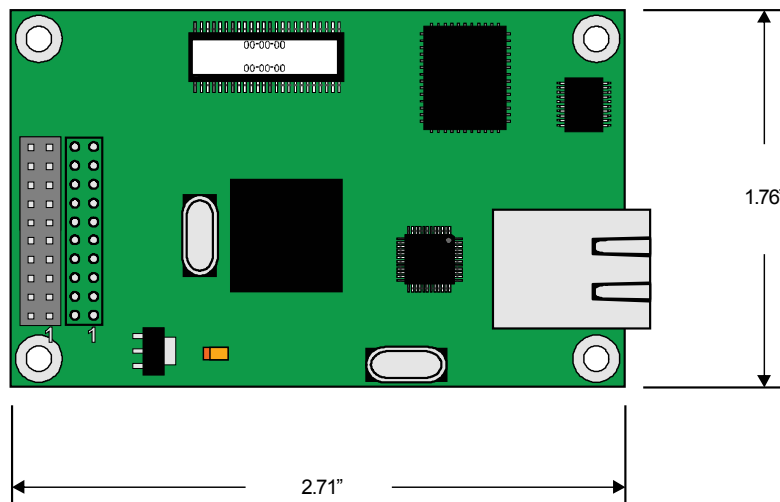
Symbol	Quantity	Minimum	Typical	Maximum	Units
Vdd	Supply Voltage [†]	-5%	3.3	+5%	V
Temp	Operating Temperature Range	-40	-	85	C

[†] Suggested Power Supply Current is 350mA\

Pin Definitions and Ratings (JP1)

Pin	Name	Function
1		
2	Vdd (+3.3V)	+3.3VDC Power
3		
4		
5	RTS0	TTL Serial Port 0 Flow Control (Serial Port)
6		
7		
8	CTS1	TTL Serial Port 1 Flow Control (Debug Port)
9	RTS1	TTL Serial Port 1 Flow Control (Debug Port)
10	TX0	TTL Serial Port 0 Transmit Data (Serial Port)
11	RX0	TTL Serial Port 0 Flow Control (Serial Port)
12	TX1	TTL Serial Port 1 Transmit Data (Debug Port)
13	RX1	TTL Serial Port 1 Receive Data (Debug Port)
14	CTS0	TTL Serial Port 0 Flow Control (Serial Port)
15		
16		
17		
18		
19		
20	GND	Ground (Vss)

Module Dimensions



Please visit the NetBurner website SB70LC product web page for the a detailed SB70LC-100IR Mechanical Diagram.

LEDs

On the RJ-45 connector LED1 is 10/100 (lit is 100) and LED2 is data activity.

Serial Interface Troubleshooting

If you are unable to communicate serially with your PINK Module, please see the list below for the most common problems and solutions to see if you can correct the issue prior to calling Tech Support.

- Be sure the Baud Rate has been set correctly on both your microprocessor and the PINK. If necessary, power the PINK Module off and then back on to verify that these settings persist by checking the web page value.
- The .BS2 programs in the examples use the BASIC Stamp 2 and will work with the BS2e and BS2pe, however if you're using a different module such as the BS2sx, BS2p or BS2px you may have the baud rate parameter in your SERIN/SEROUT set incorrectly. Always be sure you are using the correct baud value for the baud rate and BASIC Stamp model you are using. Alternately you can use the examples provided for these other BASIC Stamp models.
- Try different I/O pins. A previous improper connection may have damaged the I/O pin(s) you are using causing lack of communication. If possible verify the I/O pin using a known working serial device such as a Serial LCD or GPS Module. Lighting an LED does not necessarily mean the I/O pin works. A pin can fail in different ways.
- Be sure your TX/RX I/O pins are not backwards. The I/O pin you are sending (SEROUT) serial data on is the BASIC Stamp TX pin which would connect to the RX0 pin on the PINK Module. Conversely the pin you are receiving (SERIN) on is the BASIC Stamp RX pin and should be connected to the PINK TX0 pin. These connections are shown in the Quick Start Circuit on page 2.
- Be sure the power supply is adequate to power the PINK Module. The PINK Module requires at least a 350mA Power Supply. If you're using batteries switch to a wall supply. Batteries are not reliable for troubleshooting, especially on a high current draw device such as the PINK Module.
- Be sure the Serial Baud Rate selected is within the capability of the microcontroller you're using. For example, you should not use a baud rate of 9600 bps for the BS2, BS2e or BS2pe since the module may not be able to keep up with the serial data coming in reliably at that rate when using formatters or dealing with string data. The PINK Module's response time to a serial command is ~2.7ms for compatibility with all BASIC Stamps.
- Be sure you're using a 0 (number zero) and not an O (letter o) in the commands sent to the PINK. Some commands are terminated with a NULL or CLS and some are not. Some commands use the R or W to designate a Read or Write, while others do not. Verify the command structure if you're not getting the expected results. When in doubt run the test code provided.
- When checking the Email success bit (bit 4), remember that it must be checked during the same read cycle of the status register when the Email busy (bit 2) bit clears. This is because it is cleared during that read so subsequent reads will not be accurate. This is covered in the example program PINKSendEMail.bs2

Network Interface Troubleshooting

Due to the vast complexity and uniqueness of various computer software, hardware and networks, Technical Support cannot assist with network troubleshooting issues (See Important Information on Page 1). We can however provide some troubleshooting guidelines to help solve common network issues.

- My PINK Module cannot be seen by the IPSETUP program...
 - Be sure the PINK is in the same IP Address range as the PC you are trying to communicate from. If your PC is set to a static IP range of 192.168.X.X and the PINK Module is set to a 10.10.X.X range then you may not be able to access it. Similarly, if the PINK Module is set for DHCP and there is no DHCP server the PINK may not have a valid IP Address. Often you can still see the PINK Module using IPSETUP in these cases.
 - Disable/Turn Off any firewalls you have running. Operating systems typically have a built-in firewall. Security Utilities such as Norton Security/Anti-Virus, McAfee, Zone Alarm, etc. can all block network traffic they do not deem as safe. When in doubt disable any such firewall utilities.
 - You can use MTTTY to setup the IP address of your NetBurner device. Attach a serial cable to the SB72 Adapter Board, and from your host computer to the PINK. Make sure to power the board with the included wall wart power supply. Open up MTTTY (which can be downloaded from the product webpage), reset your NetBurner device, and type an uppercase A (i.e. A) in the MTTTY window before the (current) wait time expires to get into the NetBurner monitor program. You will see the NetBurner prompt (i.e. nb>) in the MTTTY window. At the nb> prompt type the word setup and press the Enter key on your keyboard. You are now in the MTTTY setup screen. Change option 1 to a static IP address that can be accessed by your computer network. Remember to press the S key on your keyboard to save the change, then press the X key to exit out of the setup program. If you picked a valid IP address and you should see your device with IPSetup.
- My PINK Module can be seen by my PC but not on the Internet.
 - Most likely you are using a router. By default and in its very nature a router is a firewall because it prevents the devices connected to it from being seen on the outside since you cannot directly access the local IP Address from the Internet. Most routers have a DMZ Port which will allow all traffic to your ISP Assigned IP Address to go directly through. Bear in mind that this usually allows all ports, including those used by the email and telnet system. Therefore security can be compromised using the DMZ, as telnet traffic could possibly be viewed by anyone who knows your IP Address.
 - Most routers employ 'Port Forwarding' to allow devices to be seen outside the local network. The location and options for this setting can vary from one router to the next. Usually you will have an option to specify the port to be forwarded. You will almost always want to use Port 80 (TCP) and in some cases the Port # used for UDP.
 - Many routers that support Port Forwarding require you to use Static IP Addresses on the Local Network or assign DHCP Device Names to identify the target device. The PINK Module supports a DHCP Device name if your router should require it.
- I cannot send Email from my PINK Module.
 - In order to send Email from the PINK Module it must be setup properly on the network and have access to an SMTP mail server. The SMTP server may or may not require authentication.
 - Some SMTP servers require the FROM field to contain either the username or email address associated with the server account. Try your username or email address here.
 - There are free mail server programs you can setup, however we do not have any information on these programs. It is recommended to search Google for such software.