



# **Secure Serial-to-Ethernet Factory Application User's Manual**

## **Supported Platforms:**

CB34EX  
NANO54415  
PK70EX-232  
PK70EX-485  
PK70EX-MMS  
SB70LC  
SB700EX  
SB800EX

# Table of Contents

<i>Table of Contents</i> .....	2
<b>1. Overview</b> .....	<b>4</b>
<b>2. Initial Configuration</b> .....	<b>4</b>
<b>2.1 Hardware Configuration</b> .....	<b>4</b>
2.1.1 CB34EX Device Platform .....	4
2.1.2 NANO54415 Device Platform.....	5
2.1.3 SB70LC Device Platform .....	5
2.1.4 SB700EX Device Platform.....	6
2.1.4.1 SB700EX Jumper Layout and Configuration .....	6
2.1.4.2 SB700EX DB9 Connector Pinouts .....	7
2.1.5 SB800EX Device Platform.....	7
2.1.6 NBPk70EX with Quad-UART Blade Board (NBPkBU-XXXCR).....	8
2.1.6.1 DB37-to-Quad-DB9 Serial Cable Pin-out for NBPkBU-100/232CR and -MMSCR Blade Boards	8
2.1.6.2 DB37-to-Quad-DB9 Serial Cable Pin-out for NBPkBU-485CR Blade Board .....	9
2.1.6.3 Termination Jumpers on the NBPkBU-485CR and -MMSCR Blade Boards .....	10
<b>2.2 Network Configuration</b> .....	<b>12</b>
<b>2.3 Operational Configuration</b> .....	<b>13</b>
<b>3. Web Page Configuration</b> .....	<b>13</b>
<b>3.1 Network Settings</b> .....	<b>14</b>
<b>3.2 UDP Configuration</b> .....	<b>15</b>
<b>3.3 TCP Configuration</b> .....	<b>16</b>
3.3.1 TCP Server Mode .....	17
3.3.2 TCP Client Mode.....	17
3.3.3 TCP Packetization Options, valid for TCP Sever or Client Modes .....	18
3.3.4 TCP Advanced Serial Network Settings.....	19
<b>4. SSH Configuration</b> .....	<b>20</b>
<b>4.1 SSH Key Configuration</b> .....	<b>22</b>
<b>4.2 SSH Advanced Serial Settings</b> .....	<b>23</b>
<b>5. Serial Settings</b> .....	<b>23</b>
<b>6. I2C Settings (SB70LC ONLY)</b> .....	<b>24</b>
<b>6.1 Command Protocol</b> .....	<b>25</b>
<b>7. Password Settings</b> .....	<b>26</b>
<b>8. Hypertext Transfer Protocol over SSL (HTTPS) Configuration</b> .....	<b>27</b>
<b>9. AT Commands</b> .....	<b>28</b>
<b>9.1 Enter and Exit Command Mode</b> .....	<b>28</b>
<b>9.2 Command Syntax</b> .....	<b>29</b>
<b>9.3 System/Network Configuration</b> .....	<b>29</b>
<b>9.4 Serial Port Configuration</b> .....	<b>30</b>
<b>9.5 Passwords</b> .....	<b>31</b>
9.5.1 Administrator.....	31

9.5.2	SSH.....	31
<b>9.6</b>	<b>AT Command Examples.....</b>	<b>32</b>
9.6.1	Changing the System IP Address.....	32
9.6.2	Configuring the Serial Server Listen Port.....	32
9.6.3	Configure Outgoing Network Client Connection .....	32
9.6.4	Read Analog to Digital Input Channel 1.....	33
9.6.5	Change Pin to GPIO and Set Output to High .....	33
<b>10.</b>	<b>LEDs .....</b>	<b>33</b>
<b>11.</b>	<b>RS-232 NULL Modem Wiring.....</b>	<b>34</b>
<b>12.</b>	<b>Power Connector .....</b>	<b>34</b>
<b>13.</b>	<b>Web Browsers and Proxy Servers .....</b>	<b>34</b>
<b>14.</b>	<b>Testing with a Telnet Connection .....</b>	<b>34</b>
<b>15.</b>	<b>Revision History .....</b>	<b>38</b>

## 1. Overview

The NetBurner device can function in one of the following ways:

1. The device is pre-programmed at the factory to provide serial-to-Ethernet functionality using UDP, TCP, or SSH. For the SB70LC, the device is also pre-programmed to provide Ethernet-to-I2C communication using TCP. Use the IPSetup utility to configure the device for dynamic or static IP addressing, and then connect to the device's web server page using your web browser to configure its serial and Ethernet settings (the IPSetup utility is available for download from the NetBurner web site under "Products > NetBurner Software, Protocols & Tools Suite > Deployment Tools"). Depending on the platform type, two to five UART serial ports are available in this mode of operation. Please see the hardware appendix of this manual for more information on hardware configuration and supported modes of each platform. The rest of this user's manual describes how to use your device with the secure serial-to-Ethernet factory application.
2. The NetBurner device is a very fast 32-bit single-board network computer that is customizable to suit any task. With the purchase of the NetBurner Network Development Kit (NNDK-X-KIT where "X" is the name of the platform type), you have the ability to write custom network applications, create dynamic web pages, define custom baud rates, utilize any available general-purpose digital I/O pins, and utilize peripheral features such as I<sup>2</sup>C and the SPI bus. Peripheral availability will vary between different platforms.

## 2. Initial Configuration

Three areas of configuration are required before you can start using your NetBurner device:

1. Hardware jumper configuration
2. Network configuration through the IPSetup utility
3. Operational configuration through the secure factory application's configuration web page

### 2.1 Hardware Configuration

Some device platform types may have one or more physical serial connection ports that will require jumper configuration to a desired serial mode before changing the serial modes through the serial configuration web page of the device.

#### 2.1.1 CB34EX Device Platform

For a complete explanation of the various hardware jumper configurations for supported serial modes on the DB9 and TSTRIP5 connection interfaces, please refer to the CB34EX hardware

manual, which is accessible on the CB34EX platform web page under "Documentation and Downloads" of the NetBurner web site.

## 2.1.2 NANO54415 Device Platform

The NANO54415 supports up to five UART serial ports for data tunneling. Each serial port supports the following mode settings through the factory application:

- Ports 0, 1, and 2 – RS-232, DEBUG, RS-485 Half Duplex, and RS-485 Full Duplex
- Ports 3 and 4 – RS-232 and DEBUG

A maximum of one serial port can be set to "DEBUG" mode, which uses the RS-232 protocol. The remainder of this guide will describe how to use the NANO54415 with the secure data tunneling factory application.

Note that the serial ports are limited to the following mode settings when mounted on the NANO carrier board (NANO-ADPT-100CR) due to hardware design restrictions:

- Ports 0 and 1 – RS-232 and DEBUG
- Port 2 – RS-485 half duplex and RS-485 full duplex
- Port 3 – TX and RX TTL signals only
- Port 4 – TX and RX TTL signals only (not usable on NANO-ADPT-100CR v1.1 and earlier)

## 2.1.3 SB70LC Device Platform

The SB70LC has two asynchronous UART type serial ports, referred to as "Port 0" and "Port 1". The SB70LC provides two TTL level UART ports that include TX, RX, RTS and CTS. If you need RS-232 or RS-485 electrical levels, then you must add the appropriate level shifters to the TTL level signals. "Port 0" can be configured for either RS-232 or RS-485 operation (the factory default is RS-232), while "Port 1" is always used as RS-232.

The "Debug Serial Port" ("Debug Port" for short), is used to display status messages from the application. If your application needs only a single serial port, we recommend setting the other serial port to "DEBUG".

If your application requires a single RS-232 connection, then we recommend using "Port 1" as the data port and "Port 0" as the debug port. If your application requires a single RS-485 connection, then we recommend configuring Port 1 as the debug port.

For proper I2C bus operation, 4.7k Ohm pull-up resistors must be installed on both the I2C clock (SCL) and data (SDA) signals.

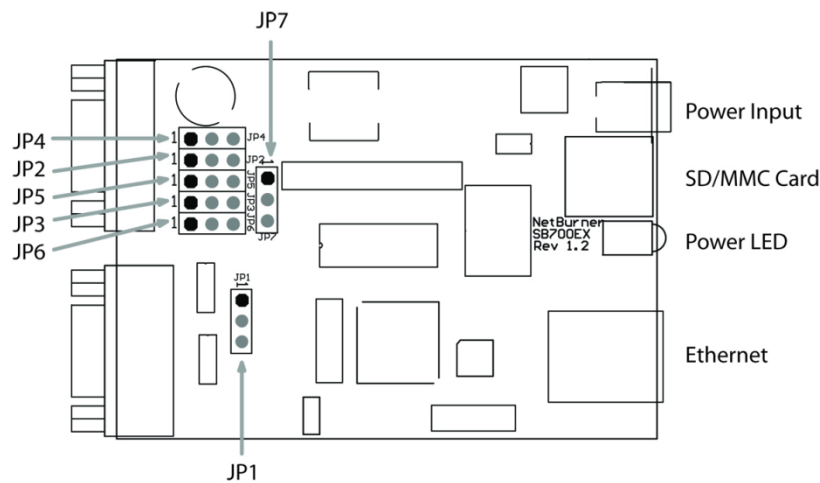
## 2.1.4 SB700EX Device Platform

The SB700EX has two asynchronous UART type serial ports, referred to as "Port 0" and "Port 1". The ports can be configured as two RS-232 ports, or one RS-232 port and one RS-485 port (half or full duplex). "Port 0" can be configured for either RS-232 or RS-485 operation (the factory default is RS-232), while "Port 1" is always used as RS-232.

The "Debug Serial Port" ("Debug Port" for short), is used to display status messages from the application. If your application needs only a single serial port, we recommend setting the other serial port to "DEBUG".

If your application requires a single RS-232 connection, we recommend using "Port 1" as the data port and "Port 0" as the debug port. If your application requires a single RS-485 connection, we recommend configuring "Port 1" as the debug port.

### 2.1.4.1 SB700EX Jumper Layout and Configuration



Mode	JP1	JP2	JP3	JP4	JP5	JP6
RS-232	[ 1 - 2 ]	[ 1 - 2 ]	[ 1 - 2 ]	[ 1 - 2 ]	[ 1 - 2 ]	---
RS-422/485	[ 2 - 3 ]	[ 2 - 3 ]	[ 2 - 3 ]	[ 2 - 3 ]	[ 2 - 3 ]	---
Enable RS-485 HD Echo	---	---	---	---	---	[ 2 - 3 ]
Disable RS-485 HD Echo	---	---	---	---	---	[ 1 - 2 ]

- JP6[2-3] should be used when using RS-485 full-duplex mode.
- JP7 selects the slew-rate limit and has three configurations:
  - [1-2] = 500 kbps communication rate
  - [2-3] = 16 Mbps communication rate
  - [No Jumper] = 250 kbps communication rate (default)

## 2.1.4.2 SB700EX DB9 Connector Pinouts

**NetBurner SB700EX Port 0-1 Signal Descriptions**

Pin	Port 0		Port 1
	RS-232 <sup>1</sup>	RS-485 <sup>1</sup>	RS-232
1	CD	-	CD
2	RX	HD- / FD TX-	RX
3	TX	HD+ / FD TX+	TX
4	DTR	-	DTR
5	GND	GND	GND
6	DSR	FD RX-	DSR
7	RTS	FD RX+	RTS
8	CTS	-	CTS
9	RI	-	RI

**Note:**

1. Port 0 can be configured as either RS-232 or RS-485

## 2.1.5 SB800EX Device Platform

The SB800EX has two asynchronous UART type serial ports, referred to as "Port 0" and "Port 1". The ports can be configured as RS-232 (default) or RS-485/422 half or full duplex. The SB800EX features jumper-less configuration, so the serial port mode is set by software configuration through the web page or serial AT command set.

The "Debug Serial Port" ("Debug Port" for short), is used to display status messages from the application. If your application needs only a single serial port, we recommend setting the other serial port to "DEBUG".

For detailed information on the connector signals please refer to the data sheet on the SB800EX product page at [www.netburner.com](http://www.netburner.com).

## 2.1.6 NBPK70EX with Quad-UART Blade Board (NBPKBU-XXXCR)

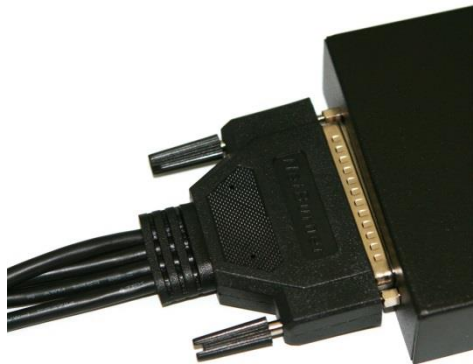
The NBPK70EX with the 4-port serial-to-Ethernet option can network-enable up to 4 RS-232/485 serial devices, depending on the type of quad-UART blade board used (or 5 if you include the debug console port which can operate in RS-232 mode only).



### NBPK70EX (Front)

The device has 6 connectors on the front panel:

- One 10/100 Ethernet port
- Two 7-24 V DC power connectors
- One SD Card reader slot
- One debug serial console port (RS-232 via UART 0)



### NBPK70EX (Rear & Side)

The device has one DB37 receptacle port on the rear panel and two LEDs on the side panel:

- NetBurner DB37-to-4xDB9 UART cable adapter for serial ports 1 to 4
- Two dual-color LEDs (not shown) on side near Ethernet jack

### 2.1.6.1 DB37-to-Quad-DB9 Serial Cable Pin-out for NBPKBU-100/232CR and -MMSCR Blade Boards

The following table lists the functions of the pins on each side of the DB37-to-quad-DB9 serial cable when the cable is connected to the NBPKBU-100/232CR or NBPKBU-MMSCR blade boards. Note that the quad RS-485 function column only applies when the NBPKBU-MMSCR blade board is used.

DB37	DB9-1	DB9-2	DB9-3	DB9-4	Quad RS-232 Function	Quad RS-485 Function
1				1	Raw Port 4 Carrier Detect	
2				2	Raw Port 4 Receive	Raw Port 4 Rx+ (FD)
3				3	Raw Port 4 Transmit	Raw Port 4 Tx- (FD/HD)
4				4	Raw Port 4 DTR	
5				5	GND	GND
6			9		Raw Port 3 Ring Indicator	
7			8		Raw Port 3 CTS	Raw Port 3 Rx- (FD)
8			7		Raw Port 3 RTS	Raw Port 3 Tx+ (FD/HD)



9			6		Raw Port 3 DSR	
10		1			Raw Port 2 Carrier Detect	
11		2			Raw Port 2 Receive	Raw Port 2 Rx+ (FD)
12		3			Raw Port 2 Transmit	Raw Port 2 Tx- (FD/HD)
13		4			Raw Port 2 DTR	
14		5			GND	GND
15	9				Raw Port 1 Ring Indicator	
16	8				Raw Port 1 CTS	Raw Port 1 Rx- (FD)
17	7				Raw Port 1 RTS	Raw Port 1 Tx+ (FD/HD)
18	6				Raw Port 1 DSR	
19	NO CONNECTION					
20				6	Raw Port 4 DSR	
21				7	Raw Port 4 RTS	Raw Port 4 Tx+ (FD/HD)
22				8	Raw Port 4 CTS	Raw Port 4 Rx- (FD)
23				9	Raw Port 4 Ring Indicator	
24			5		GND	GND
25			4		Raw Port 3 DTR	
26			3		Raw Port 3 Transmit	Raw Port 3 Tx- (FD/HD)
27			2		Raw Port 3 Receive	Raw Port 3 Rx+ (FD)
28			1		Raw Port 3 Carrier Detect	
29		6			Raw Port 2 DSR	
30		7			Raw Port 2 RTS	Raw Port 2 Tx+ (FD/HD)
31		8			Raw Port 2 CTS	Raw Port 2 Rx- (FD)
32		9			Raw Port 2 Ring Indicator	
33	5				GND	GND
34	4				Raw Port 1 DTR	
35	3				Raw Port 1 Transmit	Raw Port 1 Tx- (FD/HD)
36	2				Raw Port 1 Receive	Raw Port 1 Rx+ (FD)
37	1				Raw Port 1 Carrier Detect	

### 2.1.6.2 DB37-to-Quad-DB9 Serial Cable Pin-out for NBPKBU-485CR Blade Board

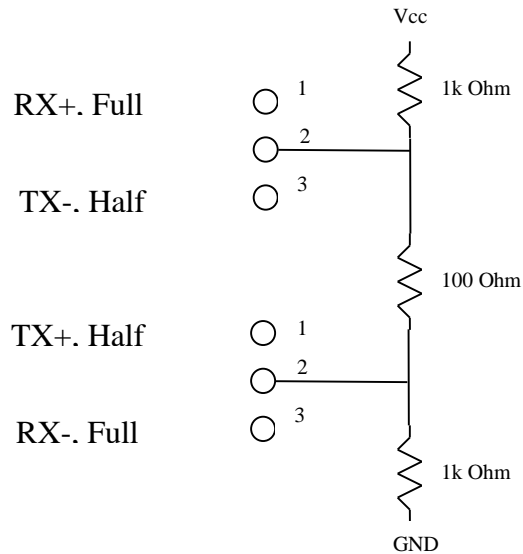
The following table lists the functions of the pins on each side of the DB37-to-quad-DB9 serial cable when the cable is connected to the NBPKBU-485CR blade board.

DB-37	DB9-1	DB9-2	DB9-3	DB9-4	Quad UART RS-485 Function
1				1	
2				2	Port 4 Tx- (FD/HD)
3				3	Port 4 Tx+ (FD/HD)
4				4	
5				5	GND
6			9		
7			8		
8			7		Port 3 Rx+ (FD)
9			6		Port 3 Rx- (FD)

10		1			
11		2			Port 2 Tx- (FD/HD)
12		3			Port 2 Tx+ (FD/HD)
13		4			
14		5			GND
15	9				
16	8				
17	7				Port 1 Rx+ (FD)
18	6				Port 1 Rx- (FD)
19	No Connection				
20				6	Port 4 Rx- (FD)
21				7	Port 4 Rx+ (FD)
22				8	
23				9	
24			5		GND
25			4		
26			3		Port 3 Tx+ (FD/HD)
27			2		Port 3 Tx- (FD/HD)
28			1		
29		6			Port 2 Rx- (FD)
30		7			Port 2 Rx+ (FD)
31		8			
32		9			
33	5				GND
34	4				
35	3				Port 1 Tx+ (FD/HD)
36	2				Port 1 Tx- (FD/HD)
37	1				

### 2.1.6.3 Termination Jumpers on the NBPKBU-485CR and –MMSCR Blade Boards

Each of the four serial ports on the blade boards that support RS-485 mode have a pair of 3-pin headers for optional resistor termination that can be added by installing jumpers. The schematic representation of the terminating resistors is shown below:



As shown above, there are two 3-pin headers for each serial port that can be jumped to short pins 1-2 or 2-3, depending on the duplex mode. To add termination in full duplex mode, add one 2-pin jumper to pins 1-2 of the RX+/TX- header, and a second 2-pin jumper to pins 2-3 of the TX+/RX- header. To add termination in half duplex mode, add one 2-pin jumper to pins 2-3 of the RX+/TX- header, and a second 2-pin jumper to pins 1-2 of the TX+/RX- header.

<b>NBPKBU-485CR Jumper Chart</b>		
<b>Port</b>	<b>RX+/TX- Header</b>	<b>TX+/RX- Header</b>
1	JP1	JP2
2	JP3	JP4
3	JP5	JP6
4	JP7	JP8

<b>NBPKBU-MMSCR Jumper Chart</b>		
<b>Port</b>	<b>RX+/TX- Header</b>	<b>TX+/RX- Header</b>
1	JP6	JP5
2	JP3	JP4
3	JP1	JP2
4	JP8	JP7

## 2.2 Network Configuration

1. Run the IPSetup utility, which is located in your NetBurner program group from the Windows Start menu. To view the advanced settings, click on the "Advanced" button. If you do not have the NNDK tools, you can download the standalone IPSetup utility from the NetBurner web site under the following sub-menu:

Products > NetBurner Software, Protocols & Tools Suite > Deployment Tools

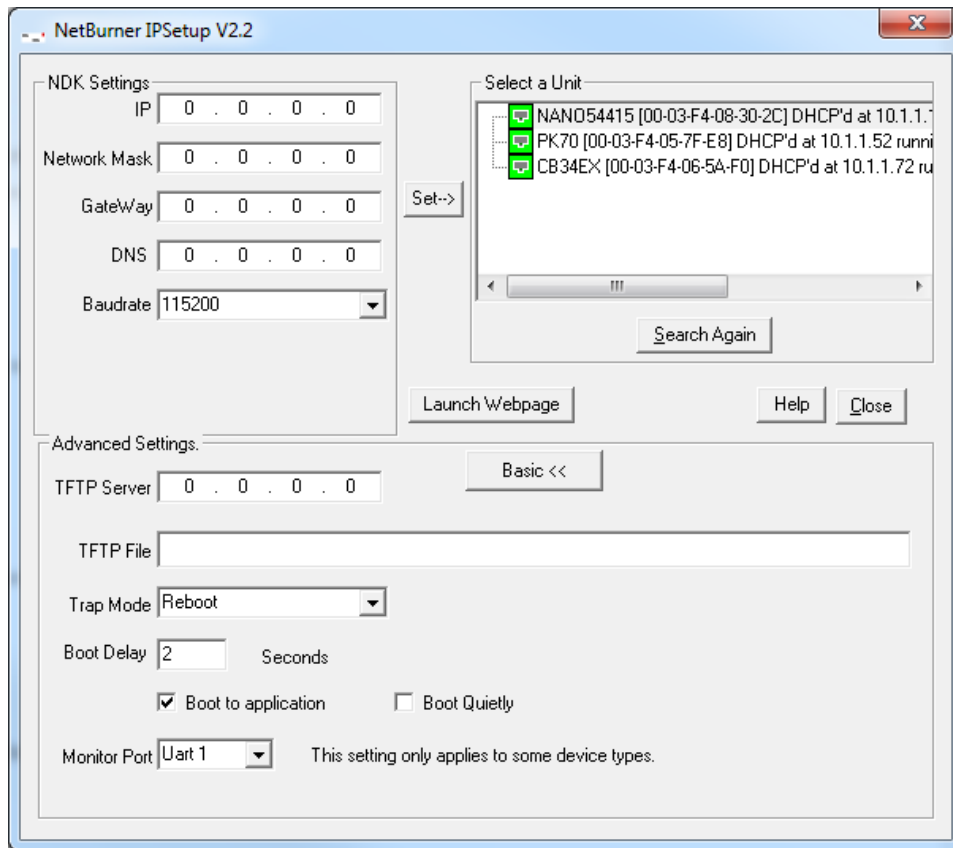
2. Locate your device in the "Select a Unit" list box by matching its MAC address. The MAC address label is located on your device. If your device does not appear in the list box, then verify that the onboard power LED, network speed and activity LEDs on the Ethernet jack are illuminated, and click the "Search Again" button. If you are still unable to see your device, remove power, verify that all your cables are properly connected, reapply power, and click the "Search Again" button. Note that the IPSetup utility uses a UDP broadcast protocol, and therefore will not operate through a router.
3. If your network supports a DHCP server, the dynamically assigned device IP address will also appear in the "Select a Unit" pane. If your network does not support a DHCP server, then you will need to configure the static device IP and mask address fields appropriate for your network. Static addresses usually reside in an address space set aside for private networks. Some common values are:

Class A: 10.0.0.0 to 10.255.255.255

Class B: 172.16.0.0 to 172.31.255.255

Class C: 192.168.0.0 to 192.168.255.255

4. After you have entered all of your values, click the "Set→" button in the center of the IPSetup utility dialog box to apply the new parameters to the selected target device in the "Select a Unit" list box.



IPSetup Window with Advanced Settings Expanded

## 2.3 Operational Configuration

Once the network parameters are set, you can use the web server interface to modify the settings of your device. To access the web page, click on the "Launch Webpage" button in the IPSetup utility, or you can open your web browser and enter the device IP address in the address bar of the browser window (for example: "http://10.1.1.72").

## 3. Web Page Configuration

Once you have configured the IP address of your device, you can connect to the device's web server to configure your serial and network settings.

### 3.1 Network Settings



**Network**

Protocol TCP/SSL ▾ (Changing will terminate all existing connections)

Device Name (for DHCP)

NetBIOS Name

Version

	Static Settings	DHCP Assigned Values	Address Mode
Device IP Address	<input type="text" value="0.0.0"/>	(10.1.1.127)	Dynamic IP (DHCP) ▾
Device Subnet Mask	<input type="text" value="0.0.0"/>	(255.255.252.0)	
Device Gateway	<input type="text" value="0.0.0"/>	(10.1.1.1)	
DNS Server	<input type="text" value="0.0.0"/>	(10.1.1.3)	
NTP Server	<input type="text" value="pool.ntp.org"/>	(current:) (63.245.214.135)	Valid NTP time
system Time:	NTP: AUG 25 2014 day: 236 (MON) 21:34:47 UTC (When page was loaded)		

Network configuration page (SB70LC platform shown)

Protocol	Selects the network protocol: SSH, TCP or UDP. Selecting a mode will enable the navigation links that apply to that protocol, and disable navigation links that do not apply. For example, if you select SSH, then the TCP and UDP configuration screens are disabled.
Device Name	This specifies the device name reported to your DHCP server, and used as the NETBIOS name. Note that NETBIOS names are limited to 14 characters and no spaces.
Version	Software version number
Address Mode	Select between dynamic and static IP address settings
Device IP Address Device Subnet Mask Device Gateway DNS Server	If "Address Mode" is set to DHCP, then your DHCP server will provide these values and they will be displayed in the "DHCP Assigned Values" column. If you wish to specify these values yourself, set the address mode to "Static IP" and type the values in the appropriate fields. Note that if you do not specify a gateway and/or DNS server you will not be able to communicate outside your local network.
NTP Server	This specifies the network time protocol (NTP) server name or address that is used to set the system time.
System Time	This displays the date and time (UTC) when the web page was last loaded. This information updates with every web page refresh.

## 3.2 UDP Configuration



[Network](#) | [UDP](#) | [TCP](#) | [SSH](#) | [Serial](#) | [Password](#) | [HTTPS](#) | [CA Certs](#) | [Advanced](#) | [Help](#)

UDP		
Settings	Port 0	Port 1
Incoming port:	<input type="text" value="0"/>	<input type="text" value="24"/>
Outgoing port:	<input type="text" value="1000"/>	<input type="text" value="1000"/>
Send output to this address:	<input type="text" value="(Enter IP Address)"/>	<input type="text" value="(Enter IP Address)"/>
Learn outbound address from last incoming packet	<input type="checkbox"/>	<input type="checkbox"/>
Number of characters to accumulate before sending UDP packet:	<input type="text" value="32"/>	<input type="text" value="32"/>
Number msec to wait for accumulated characters: 0 waits forever.	<input type="text" value="100"/>	<input type="text" value="100"/>
Send UDP frame when this character is received: (Enter NA to disable)	<input type="text" value="NA"/>	<input type="text" value="NA"/>
<input type="button" value="Submit New Settings"/>		

Device Name: CB34EXSX-5AF0 | Version: 02.03.0000

[UDP configuration page \(CB34EX platform shown\)](#)

This is only available if the UDP protocol is selected from the "Network" configuration web page. The column headings "Port 0" and "Port 1" refer to serial UARTs 0 and 1, respectively (the NANO54415 is the exception to this rule, where ports 3 and 4 are associated with UARTs 8 and 9).

Incoming Port	The UDP port to listen on for incoming network data that will be sent out the associated serial port
Outgoing Port	The UDP port used for outbound data received from the serial side
Send output to this address	The destination IP address for outgoing serial data
Learn outbound address from last incoming packet	Send outbound serial to the IP address from the last received UDP packet. Useful for clients that may have changing IP addresses.
Number of characters to accumulate before sending UDP packet	Maximum number of characters to accumulate from received UDP packets before sending them out the serial port. This setting will be overridden if the accumulation delay time setting is used and the delay time expires.
Number msec to wait for accumulated characters (0 waits forever)	Specified maximum wait time for received UDP character accumulation setting.
Send UDP frame when this character is received: (Enter NA to disable)	If a character is specified, a UDP frame will be sent upon receipt of that character from the serial port.

### 3.3 TCP Configuration



TCP			
	Port 0	Port 1	I2C Port
<b>Listen for incoming network connections</b>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Listening network port:	<input type="text" value="23"/>	<input type="text" value="24"/>	<input type="text" value="26"/>
Timeout and disconnect after this many seconds of inactivity.	<input type="text" value="60"/>	<input type="text" value="60"/>	<input type="text" value="60"/>
Allow new connection if the existing connection has been idle for this many seconds.	<input type="text" value="30"/>	<input type="text" value="30"/>	<input type="text" value="30"/>
<b>When to begin making outgoing tcp connections:</b>	<input type="text" value="Never"/>	<input type="text" value="Never"/>	
Connect on network port:	<input type="text" value="1000"/>	<input type="text" value="1000"/>	
Connect to this address:	<input type="text" value="(Enter IP Address)"/>	<input type="text" value="(Enter IP Address)"/>	
Alternate address:	<input type="text" value="(Enter IP Address)"/>	<input type="text" value="(Enter IP Address)"/>	
Timeout and disconnect after this many seconds of inactivity.	<input type="text" value="60"/>	<input type="text" value="60"/>	
Retry failed outgoing connections after this many seconds.	<input type="text" value="360"/>	<input type="text" value="360"/>	
Check and maintain valid connection at intervals in seconds.	<input type="text" value="0"/>	<input type="text" value="0"/>	
<b>Use custom packetization logic (below)</b>	<input type="checkbox"/>	<input type="checkbox"/>	
Number of characters to accumulate before sending TCP packet:	<input type="text" value="32"/>	<input type="text" value="32"/>	
Number of msec to wait for accumulated characters: 0 waits forever.	<input type="text" value="100"/>	<input type="text" value="100"/>	
Flush TCP frame when this character is received (Enter NA to disable):	<input type="text" value="NA"/>	<input type="text" value="NA"/>	
USE SSL rather than TCP for connections:	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Always Save Serial Chars regardless of connection status:	<input type="checkbox"/>	<input type="checkbox"/>	
<i>Network Settings on Serial Port - <a href="#">Advanced Serial Settings</a></i>			
<input type="button" value="Submit New Settings"/>			

TCP configuration page (SB70LC platform shown)

This is only available if the TCP protocol is selected in the "Network" configuration page. The column headings "Port 0" and "Port 1" refer to the serial ports and associated UARTs 0 and 1 (the NANO54415 is the exception to this rule, where ports 3 and 4 are associated with UARTs 8 and 9). In the case that the SB70LC is used, there will be an additional column for I2C options.



In TCP protocol mode you can configure the device to be a TCP server, which listens for incoming connections, or a TCP Client, which makes outgoing connections. In TCP Client mode, you can choose to make an outgoing connection on device power-up, or you can choose to only connect when serial data is available to send. You cannot be a TCP Client and TCP Server at the same time.

### 3.3.1 TCP Server Mode

Listen for incoming network connections	Select checkbox to enable the port to listen for incoming TCP connection requests. Checking this box will override the TCP Client mode.
Listening network port	Port number to listen on. The default port is 23 (telnet). The listen port numbers for Port0 and Port1 must be different.
Timeout and disconnect after this many seconds of inactivity.	Terminate TCP connection if no incoming network data or outgoing serial data has occurred. This is useful because there is no way to detect if a client has crashed or abnormally terminated unless unacknowledged data exists and times out. A value of 0 disables this feature.
Allow new connection if the existing connection has been idle for this many seconds.	Similar to the disconnect timeout, but does not disconnect a connection until a new connection is requested. A value of 0 disables this feature.

### 3.3.2 TCP Client Mode

When to begin making outgoing serial connections	Selects between connect on power-up and connect on serial data available.
Connect on network port	Specifies destination TCP port number
Connect to this address	Specifies destination IP address
Timeout and disconnect after this many seconds of inactivity.	Terminate TCP connection if no incoming network data or outgoing serial data has occurred. This is useful because there is no way to detect if a client has crashed or abnormally terminated unless unacknowledged data exists and times out. A value of 0 disables this feature.
Retry failed outgoing connections after this many seconds.	Number of seconds to wait before retrying an outgoing connection.
Check and maintain valid connection at intervals in seconds	Wait a given number of seconds since last received packet before sending a keep-alive packet to the remote device to verify connection. It will then wait at least two seconds for a return acknowledgement. If the remote device does not respond, then the existing connection is closed and retries until the remote device is available. A value of zero disables this feature.

### 3.3.3 TCP Packetization Options, valid for TCP Sever or Client Modes

Use custom packetization logic	Enables/disables custom packetization settings
Number of characters to accumulate before sending TCP packet	Maximum number of characters to accumulate from received TCP connection before sending them out the serial port. This setting will be overridden if the accumulation delay time setting is used and the delay time expires.
Number msec to wait for accumulated characters (0 waits forever)	Specified maximum wait time for received TCP character accumulation setting.
Flush TCP frame when this character is received (Enter NA to disable)	Send all accumulated serial data upon receipt of this character from the serial port.
Use SSL rather than TCP for connections	Checking this box enables the SSL protocol for the port connection used; unchecking the box reverts back to regular TCP connections.
Always save serial chars regardless of connection status	Checking this box allows received serial chars to be stored, whether the connection is established or not; unchecking the box discards any incoming serial characters.
DTR will reflect current connection status (this option is available only for the SB700EX platform)	When checked, the DTR output signal will toggle high or low depending on whether a serial-Ethernet connection is established for the associated port. If there is an active connection, the DTR signal will go high; a closed connection will cause DTR to go low.
Network Settings on Serial Port - Advanced Serial Settings	Link to advanced network settings that enable serial messages to be sent upon specific network events.

### 3.3.4 TCP Advanced Serial Network Settings



#### Advanced Serial

**Port 0: Serial Data Notification Settings**

Send serial message when TCP connection is established

Send serial message when TCP connection is lost  
 Message to send:

[Message Formatting Codes](#)

Send serial break when incoming TCP connection is established  
 Break interval (in tenths of a second):

Send serial break when incoming character is received (2-digit hex, i.e. "02"):

**Port 1: Serial Data Notification Settings**

Send serial message when TCP connection is established

Send serial message when TCP connection is lost  
 Message to send:

[Message Formatting Codes](#)

Send serial break when incoming TCP connection is established  
 Break interval (in tenths of a second):

Send serial break when incoming character is received (2-digit hex, i.e. "02"):



Message Formatting Codes	
Character	Data Item
%%	Prints "%"
%r	Line feed (ASCII 10)
%n	Carriage return (ASCII 13)
%x	Any hex value is %X20 (ASCII space)

## 4. SSH Configuration



[Network](#) | [UDP](#) | [TCP](#) | [SSH](#) | [Serial](#) | [Password](#) | [HTTPS](#) | [CA Certs](#) | [Advanced](#) | [Help](#)

SSH		
	Port 0	Port 1
Listen for incoming network connections	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Listening network port:	<input type="text" value="0"/>	<input type="text" value="22"/>
Timeout and disconnect after this many seconds of inactivity.	<input type="text" value="360"/>	<input type="text" value="360"/>
Allow new connection if the existing connection has been idle for this many seconds.	<input type="text" value="180"/>	<input type="text" value="180"/>
Use custom packetization logic (below)	<input type="checkbox"/>	<input type="checkbox"/>
Number of characters to accumulate before sending TCP packet:	<input type="text" value="32"/>	<input type="text" value="32"/>
Number msec to wait for accumulated characters: 0 waits forever.	<input type="text" value="100"/>	<input type="text" value="100"/>
Flush TCP frame when this character is received (Enter NA to disable):	<input type="text" value="NA"/>	<input type="text" value="NA"/>
<p><i>SSH Keys</i> <a href="#">SSH Keys</a></p> <p><i>Network Settings on Serial Port - <a href="#">Advanced Serial Settings</a></i></p>		
<input type="button" value="Submit New Settings"/>		

Device Name: CB34EXSX-5AF0 | Version: 02.03.0000

SSH configuration page (CB34EX platform shown)

Only available if the SSH Protocol is selected in Network Configuration. The column headings Port 0 and Port 1 refer to the serial ports and associated UARTs 0 and 1 (the NANO54415 is the exception to this rule, where ports 3 and 4 are associated with UARTs 8 and 9).

Listen for incoming network connections	Select checkbox to enable the port to listen for incoming SSH connection requests.
Listening network port	Port number to listen on. The default SSH port is 22. The listen port numbers for Port0 and Port1 must be different.
Timeout and disconnect after this many seconds of inactivity.	Terminate SSH connection if no incoming network data or outgoing serial data has occurred. This is useful because there is no way to detect if a client has crashed or abnormally terminated unless unacknowledged data exists and times out. A value of 0 disables this feature.
Allow new connection if the existing connection has been idle for this many seconds.	Similar to the disconnect timeout, but does not disconnect a connection until a new connection is requested. A value of 0 disables this feature.

Use custom packetization logic	Enables/disables custom packetization settings
Number of characters to accumulate before sending TCP packet	Maximum number of characters to accumulate from received SSH connection before sending them out the serial port. This setting will be overridden if the accumulation delay time setting is used and the delay time expires.
Number msec to wait for accumulated characters (0 waits forever)	Specified maximum wait time for received SSH character accumulation setting.
Flush TCP frame when this character is received (Enter NA to disable)	Send all accumulated serial data upon receipt of this character from the serial port.
SSH Keys	<a href="#">Link to the SSH key management page.</a>
Network Settings on Serial Port - Advanced Serial Settings	<a href="#">Link to advanced network settings that enable serial messages to be sent upon specific network events.</a>

## 4.1 SSH Key Configuration



[Network](#) | [UDP](#) | [TCP](#) | [SSH](#) | [Serial](#) | [Password](#) | [HTTPS](#) | [CA Certs](#) | [Advanced](#) | [Help](#)

SSH Keys		
RSA Public/Private Key Pair	Default	<a href="#">Display Public Key</a>
DSA Public/Private Key Pair	Default	<a href="#">Display Public Key</a>
RSA or DSA Key File to Install	<input type="button" value="Browse..."/> No file selected.	
		<input type="button" value="Install Key"/>

Device Name: CB34EXSX-5AF0 | Version: 02.03.0000

**SSH Keys - Key size must be at least 512 and no more than 4096 and in openSSH(openSSL) format.**

[SSH key configuration page \(CB34EX platform shown\)](#)

Only available if the SSH Protocol is selected in Network Configuration. RSA and DSA are algorithms for public-key cryptography and involve a public key and a private key. The public key can be known to everyone and is used for encrypting messages. Messages encrypted with the public key can only be decrypted using the private key. For detailed information on the NetBurner SSH implementation, please refer to the NetBurner SSH product datasheet at [www.netburner.com](http://www.netburner.com).

This configuration screen is used to display the current key configuration and provide the ability to upload your own key. Each private key also contains the public key, so uploading your own private key will also upload the public key. You can use the Display Public Key link to copy the public key and provide it for encryption purposes.

There are 3 possible key configurations. The first two are present so the system is functional if you have not uploaded your own user created keys.

1. The NetBurner factory key, designated by the name “NetBurner”. This key is built into the SSH library and will be active if no other keys are present.
2. The SSH application key, designated by the description “Default”. This key is compiled into the SSH application and will be used if no user created keys have been uploaded.
3. A key you have created and uploaded.

**IMPORTANT: When creating your own keys the key size must be at least 512 bits and no more than 4096 bits and in openSSH(openSSL) format.**

RSA Public/Private Key Pair	Displays current RSA key in use.
DSA Public/Private Key Pair	Displays current DSA key in use.
RSA or DSA Key File to Install	Use this field to select a user created key file to upload. Once selected, click on “Install Key” to upload.

## 4.2 SSH Advanced Serial Settings

These are the same as with TCP. Please refer to the TCP Advanced Serial Settings section.

## 5. Serial Settings



[Network](#) | [UDP](#) | [TCP](#) | [SSH](#) | [Serial](#) | [Password](#) | [HTTPS](#) | [CA Certs](#) | [Advanced](#) | [Help](#)

Serial	Port 0	Port 1	Port 2	Port 3	Port 4
Data Port Settings (If both are DEBUG, defaults to Port 0):	DEBUG	RS-232	RS-485 Half Duplex	RS-232	RS-232
Data Baud Rate:	115200	115200	115200	115200	115200
Custom Baud Rate:	0	0	0	0	0
Data Bits:	8	8	8	8	8
Data Parity:	None	None	None	None	None
Stop Bits:	1	1	1	1	1
Flow Control:	None	None	None	None	None
AT Commands:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="button" value="Submit New Settings"/>					

Device Name: NANO54415SX-4F14NANO54415SX-4F1 | Version: 02.03.0000

[Serial configuration page \(NANO54415 platform shown\)](#)

Depending on the device, anywhere from two to five serial ports are provided. Each port may vary in what serial modes are supported. If applicable, the hardware jumpers must also be set to the correct mode.

Data Port Settings	Each port may support one or more of the following modes: RS-232, DEBUG, RS-485 Half Duplex, and RS-485 Full Duplex
Data Baud Rate	Set a baud rate from a predefined list
Custom Baud Rate	Set any custom value to be the baud rate. "Custom" must be selected in "Data Baud Rate" for this field to be a valid. Entering '0' will default to 115.2 kbps.
Data bits	Number of data bits – valid values are 5, 6, 7, and 8.
Data parity	Parity mode – valid values are odd, even, and no parity.
Stop bits	Number of stop bits – valid values are 1 and 2.
Flow control	Set to "None" for no flow control. If using RS-232, valid selections are none, XON/XOFF software flow control, or RTS/CTS hardware flow control.

## 6. I2C Settings (SB70LC ONLY)

The SB70LC has the additional feature of Ethernet-to-I2C support. An additional I2C page link will appear in the navigation bar. The functionality is different from the UART based serial to Ethernet data pump behavior; when I2C is enabled the interface is a command/response protocol as described in the command protocol section below.

**NetBurner**  
Networking in 1 day!

[Network](#) | [UDP](#) | [TCP](#) | [SSH](#) | [Serial](#) | [I2C](#) | [Password](#) | [HTTPS](#) | [CA Certs](#) | [Advanced](#) | [Help](#)

### I2C

Data Baud Rate: 96000 ▼

Master Address: 0x5b

Save Address in Non-Volatile Memory:

Connected I2C Devices (Scan Performed on page reload): 0x20, 0x5b

I2C configuration page (SB70LC platform shown)

Data Baud Rate	Set the I2C baud rate.
Master Address	Set the 7-bit address of this master device. The input must be a two-digit hex value between 0x08 and 0x77.
Save Address in Non-Volatile Memory	Save the address in Non-Volatile memory so that the address persists after powering down the module.
Connected I2C Devices	Displays the addresses of the devices detected on the I2C bus. The scan is performed when the webpage is loaded.



## 6.1 Command Protocol

All Ethernet-to-I2C commands begin with a '#' symbol, and terminate with a carriage return and line feed: 0x0D 0x0A. All data fields are two-digit hex values.

Command	Description	Returns
#MENU or #HELP or ?	Print the main menu	<Main menu>
#WB<address><data byte>	Write a byte. Example: #WB39AC writes 0xAC to the address 0x39	<byte written> OK
#WW<address> <write length><data>	Write a buffer. Example: #WW390203AC writes 0x03AC to the address 0x39	<bytes written> OK
#RB<address>	Read a byte. Example: #RB39 reads a byte from the address 0x39	<byte read> OK
#RR<address><read length>	Read a buffer. Example: #RR390A reads 10 bytes from the address 0x39	<bytes read> OK
#WR<address><data byte>	Write a byte then read a byte. Example: #WR39AC writes 0xAC to the address 0x39 and reads a byte	<byte written>, <byte read> OK
#WA<address><data byte> <read length>	Write a byte then read a buffer. Example: #WA39AC0A writes 0xAC to the address 0x39 then reads 10 bytes	<byte written>, <bytes read> OK
#SE<new address>	Set a new I2C address. Example: #SE08 sets the address of the module to 0x08	<new address> OK
#SV<new address>	Save a new I2C address. Example: #SV0A sets the address of the module to 0x0A and saves the new address in Non-Volatile memory	<new address> OK
#ST	Print the I2C bus status.	"<I2C bus status>" OK
#RE	Reset the I2C bus.	OK
#SC	Scan the I2C bus for active devices.	<address of device>, <address of device>, ... OK

## 7. Password Settings



**Password**

Administrator

User Name:

Password:  (Leave blank for no password)

Repeat Password:

Allow AT Access

SSH User

User Name:

Password:  (Leave blank for no password)

Repeat Password:

Allow AT Access

**Device Name: CB34EXSX-5AF0 | Version: 02.03.0000**

[Password configuration page \(CB34EX platform shown\)](#)

Administrator Password	<p>This password applies to:</p> <ul style="list-style-type: none"> <li>Web page access</li> <li>Autoupdate capability for software updates</li> <li>Configuration changes with IPSetup</li> <li>SSH client log-in</li> </ul>
SSH Password	<p>Password for SSH clients with no administrative privileges to log in to the web server. Because a SSH log-in is possible with either an Admin or SSH password, an Admin password is required to enable this feature.</p>

## 8. Hypertext Transfer Protocol over SSL (HTTPS) Configuration



[Network](#) | [UDP](#) | [TCP](#) | [SSH](#) | [Serial](#) | [Password](#) | [HTTPS](#) | [CA Certs](#) | [Advanced](#) | [Help](#)

### HTTPS

SSL Public Key Certificate	Default	
RSA Public/Private Key Pair	Default	<a href="#">Display Public Key</a>
Certificate File to Install	<input type="button" value="Browse..."/>	No file selected.
Key File to Install	<input type="button" value="Browse..."/>	No file selected.

Device Name: CB34EXSX-5AF0 | Version: 02.03.0000

HTTPS - Hypertext Transfer Protocol over Secure Shell Layer (HTTPS) secure web site settings.

Key size must be at least 128 and no more than 1024 and in openssl(openSSH) format.

[HTTPS configuration page \(CB34EX platform shown\)](#)

HTTPS refers to HTTP over a secure SSL connection. To use HTTPS you can simply enter a URL of “https://xxxxx” instead of “http://xxxxx”. Your NetBurner device can simultaneously run both HTTP and HTTPS access to the web server.

HTTPS requires both a Certificate and a RSA public/private Key. The key must match the certificate and if you are creating your own self signed certificate or purchasing a certificate from a certificate authority, both the certificate and key must be uploaded at the same time. If you are creating your own certificate, called a self-signed certificate, anyone accessing the secure web page will first receive a warning message from the web browser asking to confirm whether or not the certificate should be accepted. For detailed information on the NetBurner SSL implementation, please refer to the NetBurner SSL product datasheet at [www.netburner.com](http://www.netburner.com).

There are 3 possible key/certificate configurations. The first two are present so the system is functional if you have not uploaded your own user created certificate and key.

1. The NetBurner certificate/key built into the SSL library, designated by the name “NetBurner”, which will be used if no other keys are present.
2. The application certificate/key, designated by the description “Default”. This SSL certificate and key are compiled into the application and will be used if no user created certificate and key have been uploaded.
3. A certificate and key you have uploaded.

**IMPORTANT: When creating your own key the key size must be at least 128 bits and no more than 1024 bits and in openSSL(openSSH) format.**

SSL Public Key Certificate	Displays current certificate in use.
RSA Public/Private Key Pair	Displays the current keys in use.
Certificate file to install	Selects the certificate file to upload (must also upload the key).
Key file to install	Selects the RSA public/private key pair to upload (must also upload certificate).

## 9. AT Commands

As an alternative to the web page configuration, you can configure the SB700EX over a serial port using an AT command format. The configuration sequence is:

1. Enter command mode
2. Send configuration change commands
3. Save changes and exit

The AT command operation is different for configuration commands versus GPIO pin programming commands. Please refer to “GPIO Server and Analog to Digital Readback” later in this section for additional details.

### 9.1 Enter and Exit Command Mode

AT configuration commands can only be processed when the SB700EX is in “AT Command Mode”.

To enter AT command mode:

1. Pause for 1 second (send no data).
2. Send “+++”.
3. Wait for 1 second.
4. The device will respond with “\r\nOK”.

To exit command mode:

1. Wait 30 seconds for timeout, any changes will be lost.
2. Send “AT&X” or “ATO” to exit and discard any changes.
3. Send “AT&P” to exit and save new settings.
- 4.

How does this work on SB700EX

Send “AT&F” to reset all settings to factory default (does not save).

Returns:

- 1, Returning to active connection
- 2, Returning to listening connection
- 3, Returning to active UDP mode
- 0, Returning to idle

Exiting command mode with any changes terminate all existing TCP connections.

## 9.2 Command Syntax

The format of an AT command is:

```
AT#<command>=<parameter>,<parameter>, ... . <cr>
```

The format of an AT query is:

```
AT#<command>?<cr>
```

Syntax rules:

- All white space outside quotations is ignored
- All commands and queries are terminated by a carriage return <cr> (decimal value 13), if a line feed <lf> follows a <cr>, it will be ignored.

## 9.3 System/Network Configuration

Example: AT#SYSIP=10.1.1.100<cr>

```
#SYS
```

```
IP=<ip address>
```

```
MK=<mask ip>
```

```
GW=<gateway ip>
```

```
DN=<dns ip>
```

```
DH=1 // To enable DHCP, set IP to 0.0.0.0.
```

```
NB=<device name>
```

```
UN=<string>
```

```
PW=<string>
```

### **SB700EX Only**

```
GS=<enable/disable gpio server> // 0 = disable, 1 = enable
```

```
GP=<gpio server listen port>
```

### **SB70LC and SB700EX Only (SB700EX can be set per port)**

```
MD=<T|U|S> Mode T for TCP, U for UDP and S for SSH
```

```
#CUR //Get current active values
```

```
IP?
```

```
MK?
```

```

GW?
DN?
ST? // Return port status for current port
S0? // Return port status for port 0
S1? // Return port status for port 1

```

Return Values:

- Connected to IP xx.xx.xx.xx
- Listening on port xx
- UDP mode with learned send-to IP Address : xx.xx.xx.xx
- UDP mode send to Address : xx.xx.xx.xx
- Idle

## 9.4 Serial Port Configuration

Example: AT#SER0LN=0<cr>

```

#SERn                // n = serial port number, 0, 1
SM=<R|D|H|F >      // R=RS232, D=debug, H=Half duplex RS485, F=Full duplex RS485
LN=< 0|1 >          // 1 = listen for connection 0= don't listen
BR=<baud rate >
DB=< data bits >    // 7, 8
PR=< N|O|E >        // none, odd, even
ST=< stop bits >    // 1, 2
FL=< N|S|H >        // none, software or hardware

SP=<listen network port number>
SD=<disconnect timeout in seconds>
SO=<override timeout in seconds>

CM=<N|P|R>          // outgoing connection mode: never (listen mode only),
                    // power-up or upon received serial data
CI=<dest ip name/addr>
CP=<destination port number>
CD=<disconnect timeout in seconds>
CR=<retry timeout in seconds>
KA=<check every number of seconds to maintain TCP connection>

```

### CB34EX and SB700EX Only

```
NP=< T|U|L >      // network protocol T = tcp, U = udp, L = udp learn
```

### CB34EX, NANO54415, SB70LC and SB700EX Only

```
UL=<0|1>          //UDP Learn Mode 0 or 1
```

## 9.5 Passwords

Available on CB34EX, NANO54415, SB700EX and SB70LC.

### 9.5.1 Administrator

Example: AT#SYSAUN=BobAdmin<cr>

```
#SYSA
UN=<name>           // set user name
PW=<password>       // set password
UN?                 // query user name
PW?                 // query password
```

### 9.5.2 SSH

Example: AT#SYSSUN=BobSsh<cr>

```
#SYSS
UN=<name>           // set user name
PW=<password>       // set password
UN?                 // query user name
PW?                 // query password
```

## 9.6 AT Command Examples

The following examples display the full transcript of serial communication including commands sent and replies from a SB700EX. Commands are marked in red, responses are marked in blue.

### 9.6.1 Changing the System IP Address

This example first queries the current IP address, and then changes the current IP to 10.1.1.79. Once the IP address change request is made, the settings are saved, and the device reboots.

```
+++  
OK>AT#CURIP?10.1.1.99  
OK>AT#SYSIP=10.1.1.79  
OK>AT&PIP Address Changed Rebooting  
  
Waiting 2sec(s) to start 'A' to abort
```

### 9.6.2 Configuring the Serial Server Listen Port

This example first queries the current TCP Serial Server listening port number, and then changes the current port number to 30. Once the port number change request is made, the settings are saved. No system restart was required.

```
+++  
OK>AT#SER0SP?23  
OK>AT#SER0SP=30  
OK>AT&P2,Returning to listening connection
```

### 9.6.3 Configure Outgoing Network Client Connection

This example changes the client connection's IP and port. Once the changes are made, the settings are saved. No system restart was required.

```
+++  
OK>AT#SER0CI=10.1.1.78  
OK>AT#SER0CP=30  
OK>AT&P2,Returning to listening connection
```



## 9.6.4 Read Analog to Digital Input Channel 1

This example reads the A2D pin 8. The value is immediately returned.

```
+++  
OK>AT#P08?2248
```

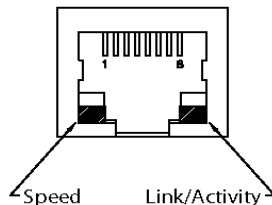
## 9.6.5 Change Pin to GPIO and Set Output to High

This example checks the current pin mode, and finds it in a2d channel 1. The mode is then updated to GPIO, and set to high.

```
+++  
OK>AT#P08M?AD1  
OK>AT#P08M=GPIO  
OK>AT#P08M?GPIO In=1  
OK>AT#P08=1  
OK>
```

## 10. LEDs

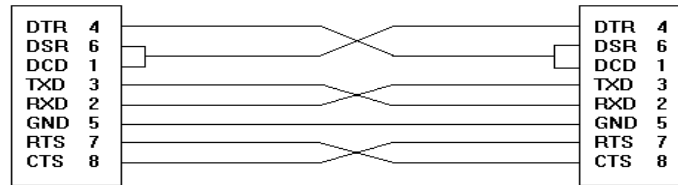
- Power LED: Illuminated while power is applied.
- LED1 on RJ-45: Ethernet speed – 10 (off) or 100 (on)
- LED2 on RJ-45: Link and data activity



Pin	Signal	Pin	Signal
1	TX+	2	TX-
3	RX+	4	N/C
5	N/C	6	RX-
7	N/C	8	N/C

## 11. RS-232 NULL Modem Wiring

The following table and diagram shows how to create a null modem cable/adaptor for RS-232 connections. A null modem cable is required if you are connecting a device such as the CB34EX or SB700EX to a standard serial port on a PC (if available).



## 12. Power Connector

Some of the devices may support multiple connection methods in receiving power. For those that support the standard 2.1-mm P5 input jack and/or 2-pin quick-disconnect terminal block, the specifications for these interfaces are shown below:

1. A standard 2.1-mm P5 input jack. The center pin has a positive ('+') polarity and the outer shell has a negative ('-') polarity.
2. A 2-pin quick-disconnect terminal block. Pin 1 (the pin that is closer to the 2.1-mm P5 input jack) has a negative polarity ('-') and pin 2 (the pin that is closer to the console serial port and MMC/SD Card slot) has a positive polarity ('+').

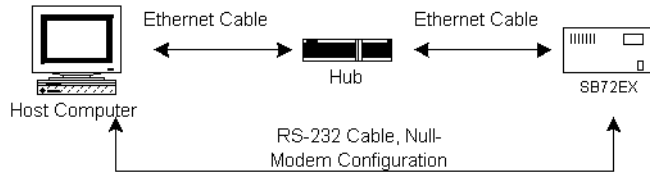
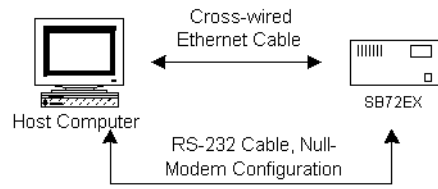
Devices that use the P5 input jack or the terminal block have a voltage input range of 7 to 24 Volts DC.

## 13. Web Browsers and Proxy Servers

If you are working on a corporate LAN that uses a proxy server for Internet web browsing, you will need to exclude the IP Address of your device in your web browser's proxy server settings/preferences. Otherwise, an attempt to connect to a web page on the LAN will fail because the proxy server will attempt to route the request outside the LAN. For most web browsers, this can be accomplished in the advanced settings for the proxy server configuration. Set the Network Mask for your host computer's network adapter and your device to 255.255.255.0.

## 14. Testing with a Telnet Connection

One quick way to test the functionality of your Serial-to-Ethernet connection is with the Telnet program and an RS-232 Serial terminal program (e.g. MTTY). To run this test, configure your system as one of the two examples shown below (null modem not required for all device platform types):

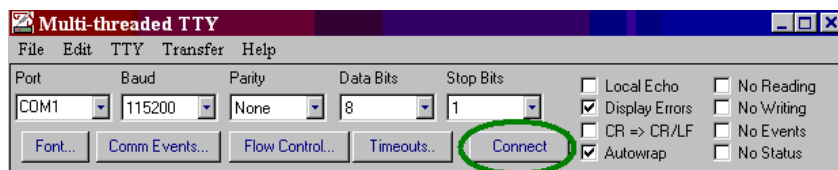


The objective of this example is to use a single host computer running telnet and a serial terminal program to send data in either direction. Therefore, if you type text in the telnet window, it should appear in the serial terminal window and visa versa. For a serial terminal, you can use MTTY (available as a download at [www.netburner.com](http://www.netburner.com)) or HyperTerminal.

In the following example, an IP Address of 10.1.1.79 will be used for the SB700EX. Note: Replace this number with the specific IP Address you assigned (if using a Static IP Address) during configuration.

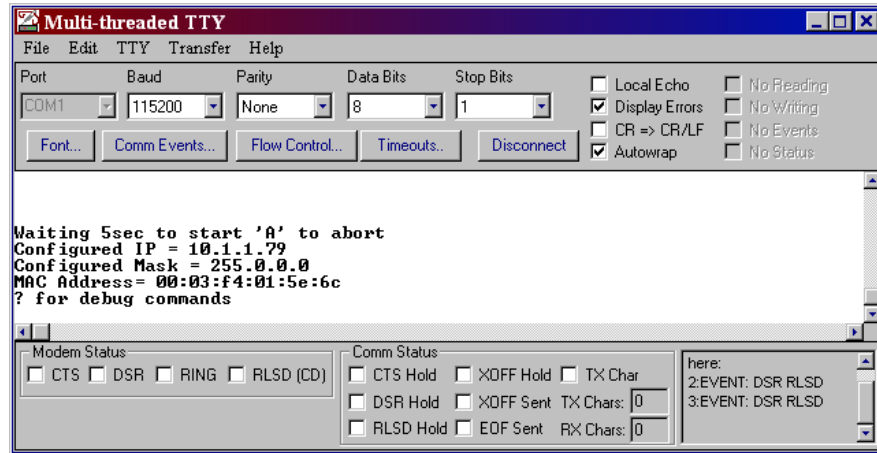
## Procedure

1. Connect your hardware in one of the above configurations.
2. Open a command prompt window on your host computer.
3. Verify everything is connected correctly by executing the command ping 10.1.1.79 and pressing the Enter key on your keyboard. You will see a valid ping response. Remember to substitute your IP Address for our example IP Address (i.e.10.1.1.79).
4. Run either HyperTerminal or MTTY. Set the baud rate to the value you assigned to the SB700EX during configuration. To use MTTY (with the factory default settings):
  - Connect your NULL modem cable from Port 0 on your SB700EX to your host computer's serial port. Remember, a standard serial cable will not work.
  - Run Mtty.exe (by double clicking its icon), available as a download from the SB700EX product page at [www.netburner.com](http://www.netburner.com). When the MTTY window appears use the factory default setting shown below:
    - The Port setting is the communication port that you connected the NULL modem cable to on your host computer.
    - The host computer and the attached SB700EX must agree on a speed or baud rate to use for the serial connection.
    - Parity checks whether the data has been lost or written over when transmitted between your host computer and your SB700EX.
    - Data Bits are the number of bits in a transmitted data package.
    - The Stop bit follows the data and parity bits in serial communication. It indicates the end of transmission.
  - Click the MTTY **Connect** button.

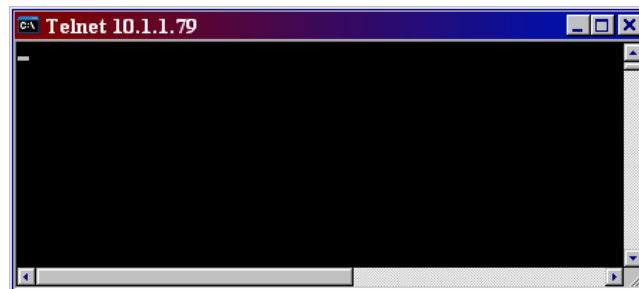


- Remove and reapply power to your SB700EX.

- The SB700EX factory application will boot.



5. Run Telnet by typing: telnet 10.1.1.79 24 after the prompt, and press the Enter key. Note: This assumes a port number of 24. You must replace this port number with the listening port number that you assigned in the “Device Connection Settings (TCP mode)” section. Remember to substitute your IP Address for our example IP Address (i.e.10.1.1.79).



6. At this point, anything you type in the Telnet window should appear in the serial terminal window and vice versa.

## 15. Revision History

<b>Revision</b>	<b>Date</b>	<b>Comments</b>
1.0	2014-09-02	Initial release as a universal platform manual for devices running the secure factory application by default
1.1	2014-09-11	Updated documentation to include hardware sections for usage of the NBPKBU-XXXCR quad-UART blade boards with the NBPK70EX-100IR
1.2	2015-03-18	Updated to include TCP outgoing connection parameter on power-up of checking and attempting to maintain a valid connection with a remote device even after it has returned from being disconnected or offline
1.3	2015-11-23	Updated to include Ethernet-to-I2C features.
1.4	2016-10-14	Added SB800EX