

ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Visit our website - *Click HERE*

The LinkHub-E™
Users Guide

An Advanced, Intelligent, Multiport Network Interface for the iButton™ and 1-Wire™ Bus



03/31/2010

Software Version 1.1i and above

Copyright © 2008, 2009, 2010 by iButtonLink LLC
iButton and 1-Wire are trademarks of Dallas Semiconductor Corp, Dallas, Texas, USA.

Introduction:

The LinkHub-E™ is an advanced, intelligent, multiport network interface for use with Dallas Semiconductor 1-Wire™ and iButton™ components. It uses superior digital and analog methods to accomplish reliable operation on a wide variety of network topologies. It also offers many functions that simplify network communications and some diagnostic tools as well.

There are five (5) 1-Wire busses (1 internal and 4 external) which are logically or-ed into a single composite 1-Wire bus. Software written for the Dallas Semiconductor DS2480B interface will work even though the 1-Wire devices are physically on separate busses. Each external bus can be up to 1000 feet long. This yields a logical 1-Wire network of 4000 feet with 4 times the network weight of a standard 1-Wire bus.

The LinkHub-E provides +12V and +5V power on the 1-Wire bus cables.

The LinkHub-E is designed to operate reliably on long and short 1-Wire busses. This is accomplished by a special analog interface design that uses matched impedances and slew rate controls, as well as smart cable pre-charge. The firmware (programming) is adaptive and automatically adjusts many 1-Wire network parameters.

The LinkHub-E is based on the Link, designed in cooperation with Dallas Semiconductor Corp., Dallas, Texas, the originator of 1-Wire and iButton devices and various accessories. Some terms used in this document may be trademarks of Dallas Semiconductor Corporation, and are used with their permission.

DS2480B Emulation:

The LinkHub-E is programmed to closely emulate a Dallas DS2480B serial port to 1-Wire line driver so that existing software can use it without modification. (The DS2480B is the basis for the Dallas DS9097 series of 1-Wire adapters.) This allows The LinkHub-E to be used in place of the Dallas DS2480B in most cases where increased performance or reliability is required. Some DS2480B functions are <u>not</u> emulated by the LinkHub-E, as described below.

Calibration:

Unlike the DS2480B, the LinkHub-E does not require calibration from the serial port because it has a crystal-controlled time base. This means that drift in the interface data rate and waveform timing is not an issue. The LinkHub-E expects a calibration byte for DS2480B software compatibility but it is ignored.

Copyright © 2008, 2009, 2010 by iButtonLink LLC
iButton and 1-Wire are trademarks of Dallas Semiconductor Corp, Dallas, Texas, USA.

Overdrive Speed:

The LinkHub-E does not support Overdrive speed 1-Wire bus communication. The analog components in the bus interface are carefully tuned for optimum long- and short-line performance at standard speed and the higher speed mode cannot also be supported. However, for the vast majority of applications, this is not an issue.

Flex Timing Modes:

The LinkHub-E 1-Wire waveform timing is carefully and dynamically controlled by adaptive control algorithms. Flex mode timing changes supported by the DS2480B are unnecessary and are not supported in the LinkHub-E. However, the LinkHub-E emulates the Flex Mode registers and will appear to operate like a DS2480B for software compatibility.

EPROM Programming:

The LinkHub-E does not support +12V pulses for programming EPROM 1-Wire or iButton devices. Programming pulses for 5V devices (EEPROM) are supported.

Device Version:

The LinkHub-E reports device version 7 when queried using the normal DS2480B method. This allows the software to distinguish the LinkHub-E from a standard Dallas DS2480B interface, which at the time of this writing returns a 3. Since the Link (single port) also returns a device version 7, the user can use the " " (space) ASCII command to determine if a LinkHub-E is present.

Embedded EEPROM Memory:

The LinkHub-E includes 32K Bytes of EEPROM memory for user configuration data. This EEPROM is not part of the 1-Wire bus structure but provides local storage for general purpose use.

LinkHub-E Serial Number:

Each LinkHub-E contains a DS2401 Silicon Serial Number chip on the internal 1-Wire bus, providing an absolutely unique identity. The serial number will appear on 1-Wire searches unless the internal bus is deactivated. It may be conveniently used to identify the particular LinkHub-E or to verify that the 1-wire bus reading functions are working correctly.

Copyright © 2008, 2009, 2010 by iButtonLink LLC
iButton and 1-Wire are trademarks of Dallas Semiconductor Corp, Dallas, Texas, USA.

LED indicators:

The LinkHub-E has two LED indicators on the right side. The LED on the left is a heartbeat indicating that the LinkHub-E processor is running. While the processor is idle, the LED will illuminate at one-second intervals. The LED on the right indicates the presence of 1-Wire bus traffic. The LED will illuminate when there is 1-Wire traffic on any bus.

Connecting to the LinkHub-E:

After the LinkHub-E has an IP Address assigned either manually or via DHCP (see Appendix C), use one of two methods to issue commands and receive responses.

1. Establish a telnet session to it on TCP port 10001.
2. Disable Telnet Mode (see Appendix C) and use a virtual COM port like the Lantronix Com Port Redirector to define a COM port using the LinkHub-E's IP address and port 10001. That COM port can be used with programs like HyperTerm to send ASCII commands, or by programs like DigiTemp and the Dallas OneWireViewer to communicate with the LinkHub-E like a DS9097.

ASCII Commands:

In addition to DS2480B emulation, the LinkHub-E also provides many 1-Wire bus functions in response to ASCII commands. This allows the LinkHub-E to be used with any terminal program or telnet socket API to perform iButton or 1-Wire bus functions without software drivers. The LinkHub-E functions that can be performed have been coded for normal ASCII keys so that all the basic 1-Wire functions can be performed manually on the keyboard of a terminal program, or using programming languages that are limited to ASCII serial port I/O. As a rule, these special commands consist of every command code that is not used in the DS2480B emulation.

The LinkHub-E ASCII commands are letters and symbols as follows:

Key **" " (space)** – Displays the LinkHub-E firmware version.

Key **h** – Displays a list of LinkHub-E ASCII commands that are available in addition to standard DS2480 binary command codes:

```
f=First         \=Expanded Command Set
n=Next          !=Thru
r=Reset         l=Level
(=Extend
b=Byt(NN+)
j=Bit(N+)
p=BytW/P
~=BitW/P        x=BusLo
d=Aux+          $=Scan
z=Aux-          *=List
&=Aux?          h=Help
```

Copyright © 2008, 2009, 2010 by iButtonLink LLC
iButton and 1-Wire are trademarks of Dallas Semiconductor Corp, Dallas, Texas, USA.

```
t=Search Type
```

Note that the text `"!=Thru"` is incorrect and will be removed in a future LinkHub-E firmware release.

Key **r** – Performs a 1-Wire bus reset and returns the status of the bus, either a "P" representing the presence of one or more devices on the bus, an "N" representing no devices on the bus, or an "S" representing a shorted bus.

Key **b** – Places the LinkHub-E into byte mode. The next two characters entered will be taken as a hexadecimal byte value, which is then issued onto the bus. The response byte is then displayed in hexadecimal. Subsequent pairs of hexadecimal characters will also generate bytes, allowing for streaming of bytes without intervening commands. An "x" character or hitting ENTER (Carriage Return) will end Byte mode.

The following subcommands are recognized in byte mode instead of a 2 character hex pair:

P... Issue 64 read byte commands to the OW bus and return 64 pairs of hex characters. This subcommand is used to read a memory page from devices that have a page size of 64 bytes.

p ... Issue 66 read byte commands to the OW bus and return 66 pairs of hex characters. This subcommand is used to read a memory page WITH CRC from devices that have a page size of 64 bytes.

M ... Issue 32 read byte commands to the OW bus and return 32 pairs of hex characters. This subcommand is used to read a memory page from devices that have a page size of 32 bytes.

m ... Issue 34 read byte commands to the OW bus and return 34 pairs of hex characters. This subcommand is used to read a memory page WITH CRC from devices that have a page size of 32 bytes.

***** … Inserts a carriage return, linefeed pair of characters into the returned data stream. Generates no activity on any 1-Wire bus. The "*" is not echoed. Provides an input terminator for a host program without leaving byte mode.

Key **p** – Performs the same type of operation as the "**b**" key above, except that power (strong pull-up) is applied to the bus after the last bit of the first byte is issued. Subsequent bytes generated will not be followed by strong pull-up. Hitting ENTER (Carriage Return) will end the Byte mode.

Copyright © 2008, 2009, 2010 by iButtonLink LLC
iButton and 1-Wire are trademarks of Dallas Semiconductor Corp, Dallas, Texas, USA.

Key **j** – Performs the same type of operation as the "**b**" key above, except with single bits. Single ASCII digits of 0 or 1 value only are allowed, and single ASCII digits are returned. Hitting ENTER (Carriage Return) will end the Bit mode.

Key **~ (tilde)** – Performs the same type of operation as the "**j**" key above except that power (strong pull-up) is applied to the bus after the first bit is issued. (Subsequent bits generated will not be followed by strong pull-up.)

Key **f** – Performs a 1-Wire bus "first" operation. This operation searches the bus and finds the first 1-Wire or iButton device and displays the device serial number prefixed with a "+" or "-" character to indicate if there are more ("+") or no more ("-") parts remaining to be found.

Key **n** – Performs a 1-Wire bus "next" operation. This operation searches the bus and returns the next 1-Wire or iButton device, and displays the device serial number prefixed with a "+" or "-" character to indicate if there are more ("+") or no more ("-") parts remaining to be found. (If used again after a "-" response is received, this function finds the first part again.) Note that if a Family Search (the **\f** command) is in progress, next may return a "?" if encountering a family code overshoot.

Key **t** – When followed by a two-character hexadecimal value, will change the search type to this function command value. Using "tF0" will make the search normal. Using "tEC" will make the search conditional and will discover only devices for which the search conditions are satisfied. (See the data sheets for each individual Dallas iButton or 1-Wire device for specific search type command codes.)

Key **l (lower case L)** – This will test the composite 1-Wire bus level and report a "0" if any bus is low or a "1" if all busses are high, followed by a carriage return.

Key **x** – Forces all 1-Wire busses to a low level. Cancelled by a reset ("r") command or any bit or byte command to restore the 1-Wire busses to operation. This function is used to cause a bus-wide reset by robbing power from all devices on the busses for a few seconds.

Key **z** – Turn off the switched +5V line to the LinkHub-E output connectors.

Key **d** – Turn on the switched +5V line to the LinkHub-E output connectors.

Key **&** – Report the current intended state of the switched +5V line on the LinkHub-E output connectors.

Key **$** – Start arrival/departure scanning (See *Scanning* below.)

Key ***** – Report scan list. If scanning is NOT active the response is a single carriage return character. (See *Scanning* below.)

Copyright © 2008, 2009, 2010 by iButtonLink LLC
iButton and 1-Wire are trademarks of Dallas Semiconductor Corp, Dallas, Texas, USA.

Key **** (**backslash**) – Escape character for a LinkHub-E expanded command set command. The \ command by itself does nothing. It must be followed by an expanded command character. This differs from a standard Link which uses this key for sniffer mode that is not supported by the LinkHub-E.

Key **\f** (**backslash f**) – Family search first. Similar to the standard first command, the family search first expects the next two hex characters to specify the 1-Wire family code to be included in the search. The standard "n" (next) command is used to retrieve the next 1-Wire serial number from the bus search. Since the family search may not terminate on the last device on the bus, care should be taken to check the family code returned for each serial number. In the case of a family search overshoot, a "?" is returned instead of the normal "+" or "-". See the First ("f") command.

Key **\h** (**backslash h**) – Display help for the expanded command set:

```
LinkHub Expanded Commands

\h = help
\$ = Toggle Channel Mask Display
\C = Set Active Channel Mask
\c = Report Active Channel Mask
\E = Report max EEPROM address
\M = Set EEPROM next address
\m = Report EEPROM next address
\R = Read EEPROM HEX data
\r = Read EEPROM ASCII data
\W = Write EEPROM HEX data
\w = Write EEPROM ASCII data
\f = Family Search first
\! = Crash and Restart LinkHub
```

Key **\C** (**backslash uppercase C**) – Set the active Channel Mask. This command is follow by a two digit hex value representing a bitmask for channels which should be active. There are five 1-Wire busses. The bit patterns used to form the mask are:

```
0x37 ...  All busses enabled
0x01 ...  External bus 1
0x02 ...  External bus 2
0x04 ...  External bus 3
0x10 ...  Internal bus
0x20 ...  External bus 4
```

A one (1) makes the channel active. A zero (0) makes the channel inactive. Example: to enable only the internal bus and bus 2, enter \C11.

Key **\c** (**backslash c**) – Report the active Channel Mask. The LinkHub-E will respond with a two digit hex value reporting the current Channel Mask.

Copyright © 2008, 2009, 2010 by iButtonLink LLC
iButton and 1-Wire are trademarks of Dallas Semiconductor Corp, Dallas, Texas, USA.

Key **\E (backslash uppercase E)** – Report maximum internal EEPROM address. The LinkHub-E will respond with a 6 digit hex number which represent the address which can be specified with Read/Write EEPROM commands. The 32KB LinkHub-E has 32768 EEPROM addresses. A 256KB LinkHub-E has 262136 addresses. A 512KB LinkHub-E has 524272 addresses. A 32KB LinkHub-E will respond with 007FFF.

Key **\M (backslash uppercase M)** – Set EEPROM Read/Write Address. The \M command is followed by a 6 digit Hex Address specifying the next EEPROM address to be read/written. Example: \M000030 would specify address 30 (hex) as the next read/write address.

Key **\m (backslash lowercase m)** – Report EEPROM Read/Write Address. The next EEPROM read/write address will be returned followed by a carriage return. Example: \m might return 0001FF<CR> meaning the next byte to be read/written is 0001FF hex.

Key **\R** (**backslash uppercase R**) – Read hex data from EEPROM. The \R command is followed by a 2 digit hex number specifying the number of bytes to read. The contents of each EEPROM byte will be returned as a two digit hex number terminated with a carriage return.

Example: **\M0003FF\R04** will return the contents of EEPROM addresses 0003FF through 000402 followed by a carriage return. The next EEPROM read/write address would be 000403 after executing the example.

Key **\r (backslash lowercase r) –** Read ASCII data from EEPROM. Data is read from the EEPROM and returned as ASCII characters. The read is terminated by a carriage return from the EEPROM data stream. The carriage return is also sent to the user.

Example: After entering some ASCII data into the EEPROM with:

```
\M000100\wThis is a test<CR>
\wThis is line two<CR>
\WThis is line three<CR>
\wThis is the last line<CR>
```

The command **\M000100\r\r\r\r** would return:

```
This is a test<CR>
This is line two<CR>
This is line three<CR>
This is the last line<CR>
```

Key **\W** (**backslash uppercase W**) – Write hex data to EEPROM. The \W command is followed by one or more pairs of hex characters representing the byte(s) to be written. A carriage return terminates the write. The carriage return is NOT written into the EEPROM.

Copyright © 2008, 2009, 2010 by iButtonLink LLC
iButton and 1-Wire are trademarks of Dallas Semiconductor Corp, Dallas, Texas, USA.

Example: **\M0000FF\W416243<CR>** will put the ASCII string "AbC" into bytes 0000FF through 0000102 of the EEPROM memory. The next EEPROM read/write address would be 0000103 after the execution of the example. Note that the <CR> was NOT written to the EEPROM.

Key **\w (backslash lowercase w) –** Write ASCII data to EEPROM. The \w command is followed by ASCII characters terminated by a carriage return <CR>. The carriage return IS written into the EEPROM. Example:

\M000100\wThis is a test<CR> writes 15 characters to EEPROM beginning at address 000100. The next EEPROM read/write address would be 00010F after the execution of the example.

Key **\$ (backslash dollar sign)** – Toggles channel reporting on or off. When channel reporting is on, reported 1-Wire device serial numbers are followed by a comma and the number of their 1-Wire bus. The internal bus is reported as bus number five (5).

Key **\!** (**backslash exclamation point**) Cause a formatted dump of the LinkHub-E RAM contents and register states to be produced on the serial interface. A watchdog timer reset of the LinkHub-E occurs after the dump is complete.

Key **|** (**vertical bar**) – This will cause the LinkHub-E to enter a pass-thru mode. In this mode, all activity on the serial port output line is passed through (inverted) to the 1-Wire bus, and all activity on the 1-Wire bus is passed-through (inverted) to the serial port input line. This mode can be used to bypass the LinkHub-E and allow the serial port direct access to the 1-Wire bus. Because the device is no longer able to interpret serial data in this mode, the only way to get out of the pass-thru mode is by a power-on-reset of the LinkHub-E. Since this can be extremely difficult to do to a remotely located LinkHub-E, this command will be removed in a future LinkHub-E firmware release.

Note 1: The 1-Wire bus with relaxed timing suitable for long lines can only process bits at a rate of about 14,000 per second. Streaming bytes using the (b) command will fail if the baud rate is set to more than 19,200 because the host will overrun the 1-Wire bus. When the baud rate is set to any value greater than 19,200 the host commands must be paced to assure that 1-Wire bus overrun does not occur. Pacing means waiting for the character response before sending another character. Even though the LinkHub-E has a bigger typeahead buffer (64 characters) than the original Link, pacing will ensure that the LinkHub-E does not overrun the 1-Wire bus.

Key . (**period**) – This turns OFF the dynamic pull-up (DPU) driver in the 1-Wire bus interface. The DPU helps extend the useable length of the 1-Wire bus by increasing the charge current at appropriate times in the 1-Wire waveform. In the rare event that the action of the DPU causes a problem on shorter networks, this command allows it to be

Copyright © 2008, 2009, 2010 by iButtonLink LLC
iButton and 1-Wire are trademarks of Dallas Semiconductor Corp, Dallas, Texas, USA.

turned off. The LINK responds with a carriage return, line feed. The DPU is turned back ON by any reset of the LinkHub-E.

Key **" ("** (**left parenthesis**) – Relaxes the 1-Wire bus timings to accommodate long bus lengths. Invoke this timing if you are experiencing bus errors due to reflections from busses over 300 feet in length.

Power Issues:

The LINK provides an Auxiliary I/O line on the RJ connector. The LinkHub-E provides switched +5V on the same pin.

Scanning:

It is often useful to scan the 1-Wire bus and report the ***arrival*** of a new 1-Wire device or the ***departure*** of a device from the bus. The LinkHub-E includes a bus scanning function for this purpose. When turned ON using the '$' character, the scanning system continually performs First and Next device discovery operations and builds a table of up to eight (8) device serial numbers. As a new device appears on the 1-Wire bus, and after the same device serial number (with correct CRC8 and a non-zero family code) has been observed on two subsequent full discovery passes, an Arrival is reported as a string with a "!" (exclamation-point) character, a comma, and the arriving device serial number.
When a device has been present on the bus and then is not found in ten (10) successive complete discovery cycles, it is reported as a Departure with a "?" (question mark) followed by a comma and the serial number of the device that departed from the bus. (This de-bounces the departure.)
While scanning is active the current contents of the table can be reported with the "*" command.

Scanning is terminated by a Reset, First, Next or any number of other operations.

The scanning system will not work properly if more than 8 iButton or 1-Wire devices are present on the bus. This is due to the limited memory available in the LinkHub-E.

Questions about the LinkHub-E should be directed to info@iButtonLink.com.

Copyright © 2008, 2009, 2010 by iButtonLink LLC
iButton and 1-Wire are trademarks of Dallas Semiconductor Corp, Dallas, Texas, USA.

Appendix A – 1-Wire Communications Examples

The sequence for reading a DS18B20 temp sensor via the LinkHub-E™ is straightforward:

1. Issue a 1-Wire reset (`r`).
2. Enter byte mode in pull-up mode (`p`) and address the ROM by sending `55` followed by the ROM address in reverse byte order (that is, if the discovered id is `E60000003DA0E128` you would address it as `28E1A03D000000E6`).
3. Send the convert command `44`.
4. Exit byte mode (CR).
5. Wait at least 900ms for the conversion to complete.
6. Issue a 1-Wire reset (`r`).
7. Enter byte mode (`b`) and address the ROM as before.
8. Send the read command `BE`.
9. Send two read commands as `FFFF`. The echoed data will contain the temperature reading in Intel (little-endian) order as a 16-bit signed integer (as 4 hex characters)
10. Exit byte mode (CR).

The returned value will be a signed value in 1/16 deg. C increments. A return value of `5701` represents 0x0157, or 343 in decimal. Dividing by 16 yields 21.4 degrees C, or 70.6 degrees F.

Mind the sign bit, or values below freezing will appear to be unusually warm by several thousand degrees.

Debugging hint:

The power up default reading from the DS18B20 before it has done a conversion is 85 degrees C (185 degrees F). Look carefully if you receive this value. This indicates that a convert has never been executed by this device since power up. It is possible to address a device and NOT have enough power available for it to execute a convert.

Copyright © 2008, 2009, 2010 by iButtonLink LLC
iButton and 1-Wire are trademarks of Dallas Semiconductor Corp, Dallas, Texas, USA.

Appendix B – Wiring Diagrams

Pinouts

LinkHub-E External 1-Wire Ports

(viewed looking into port)



Pin 1 Ground
Pin 2 V+ (+12V)
Pin 3 Switched 5(Aux)
Pin 4 Ground
Pin 5 1-Wire Data
Pin 6 Ground
Pin 7 +5V
Pin 8 Ground

Warning – There is no current limiting provided on the +12V pin. Shorting this to ground will severely damage the LinkHub-E.

Copyright © 2008, 2009, 2010 by iButtonLink LLC
iButton and 1-Wire are trademarks of Dallas Semiconductor Corp, Dallas, Texas, USA.

Appendix C – Network Interface Notes

The LinkHub-E Ethernet interface is implemented with a standard Lantronix XPort. More information and the Lantronix Device Installer may be downloaded from iButtonLink at http://www.ibuttonlink.com/manuals.aspx or from Lantronix at http://www.lantronix.com.

The LinkHub-E's only differences from the Lantronix XPort factory defaults are:

1. Channel 1 is set to baud rate 115200
   This is required to match the serial port of the LinkHub-E's microprocessor
   No Parity, 1 Stop bit are also required, but are Lantronix factory defaults.

2. The Disconnect Mode is set to 40 hex.
   This enables Telnet Mode (or Telnet Com Port Cntrl), which disables local character echoing unless the telnet client program asks for it. If a virtual COM port will be used you must change Disconnect Mode to 0.

3. The Inactivity Timeout is set to ten seconds.

4. a. The IP Address is usually set to Autoconfigure, and the LinkHub-E will obtain an IP address from a DHCP server if one is available on the network, or use AutoIP to generate an address in the range of 169.254.x.x with a subnet mask of 255.255.0.0.

   b. Some units have a predefined fixed IP address. These units have a "default IP:" label on the bottom specifying the address. You can use this address to connect to the LinkHub-E for further configuration. Once the IP address has been changed please remove the label.

XPort configuration can be done with the Lantronix Device Installer, which includes web and telnet configuration options. It can also be done with a standard web browser connection to port 80, or a telnet connection to port 9999.

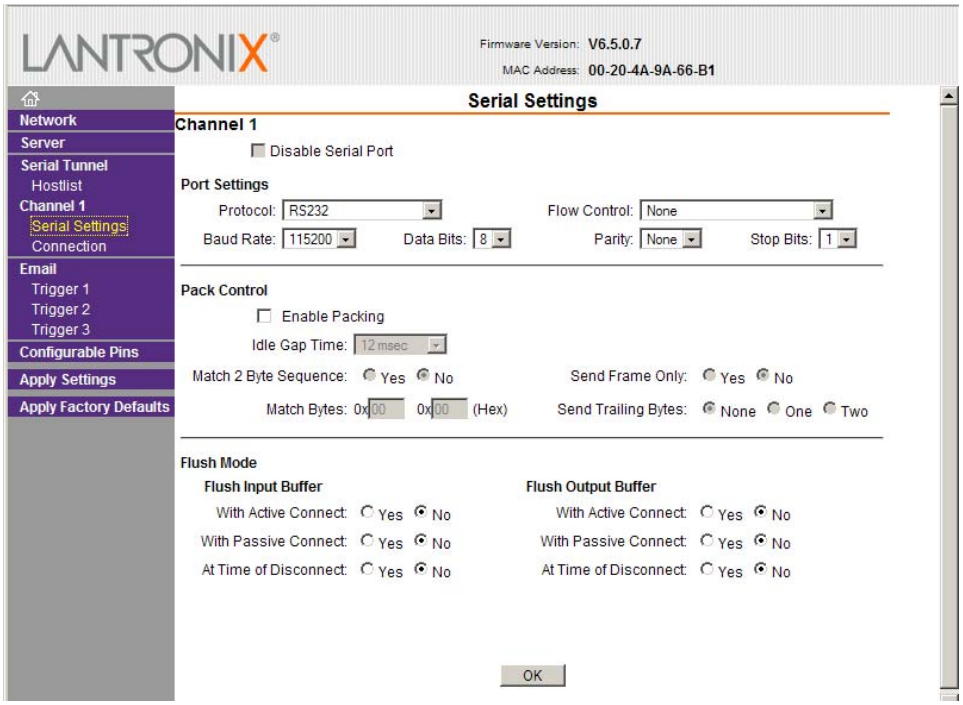
To use web configuration highlight the LinkHub-E's XPort in Device Installer and select the Web Configuration tab, or with a standard web browser connect to the LinkHub-E's IP address on port 80. The username and password are blank by default, so click the OK button in the authentication popup window.

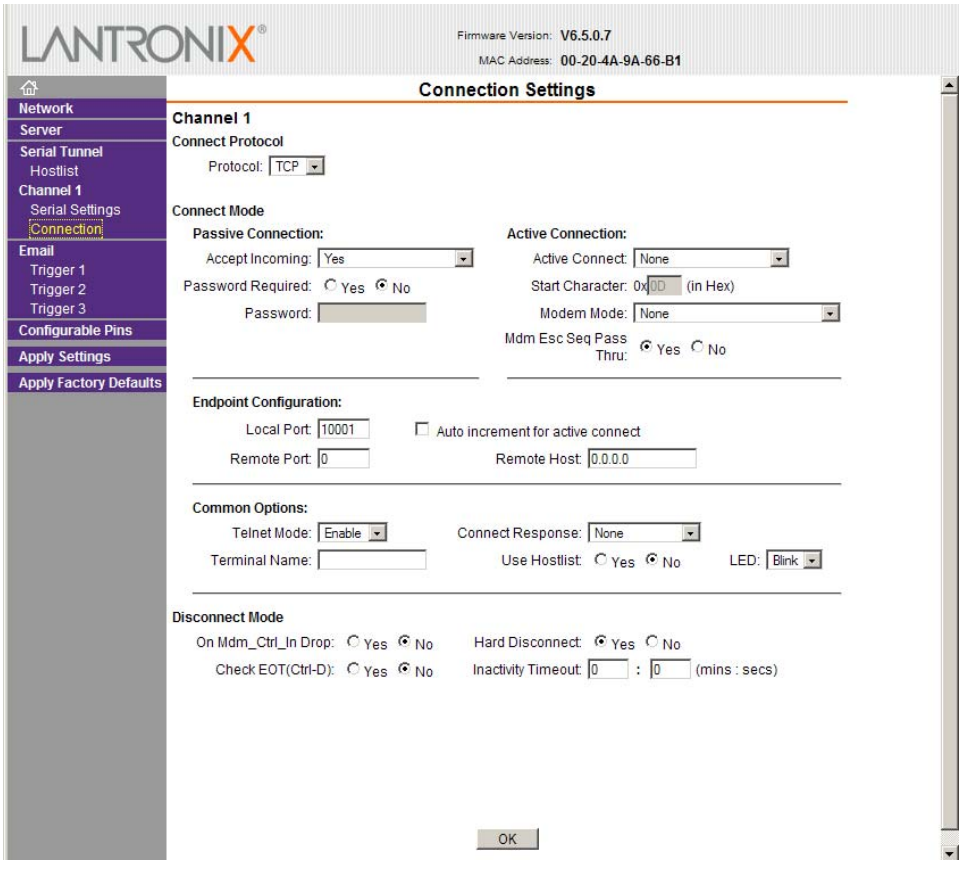
In Channel 1 Serial Settings select the 115200 Baud Rate and click OK.

In Channel 1 Connection under Common Options select Telnet Mode (or Telnet Com Port Cntrl) Enable. If a virtual COM port will be used select Telnet Mode Disable. Under Disconnect Mode set Inactivity Timeout to 0:10 and click OK.

Copyright © 2008, 2009, 2010 by iButtonLink LLC
iButton and 1-Wire are trademarks of Dallas Semiconductor Corp, Dallas, Texas, USA.

Screen shots of Web Configuration page settings





Copyright © 2008, 2009, 2010 by iButtonLink LLC
iButton and 1-Wire are trademarks of Dallas Semiconductor Corp, Dallas, Texas, USA.

To use telnet configuration establish a telnet session to the LinkHub-E on port 9999.

Press Enter for setup mode.
Under Change Setup enter 1 for Channel 1
At Baudrate (xxxx)? enter 115200
Hit return to accept the current values until DisConnMode appears
At DisConnMode (00) ? enter 40
Hit return to accept the current values until DisConnTime appears
At DisConTime (00:00) ? enter 0, hit return, enter 10 and hit return again
Hit Return to accept the current values until Change Setup appears again, then enter 9 for Save and Exit

Telnet configuration example:

```
telnet 192.168.0.2 9999
MAC address 00204A9A66B1
Software version V6.5.0.7 (070919) XPTEXE

Press Enter for Setup Mode


*** basic parameters
Hardware: Ethernet TPI
IP addr 192.168.0.2, gateway 192.168.0.1,netmask 255.255.255.0

*** Security
SNMP is              enabled
SNMP Community Name: public
Telnet Setup is      enabled
TFTP Download is     enabled
Port 77FEh is        enabled
Web Server is        enabled
Web Setup is         enabled
ECHO is              disabled
Enhanced Password is disabled
Port 77F0h is        enabled

*** Channel 1
Baudrate 115200, I/F Mode 4C, Flow 00
Port 10001
Connect Mode : C0
Send '+++' in Modem Mode enabled
Show IP addr after 'RING' enabled
Auto increment source port disabled
Remote IP Adr: --- none ---, Port 00000
Disconn Mode : 40 (Telnet Com Port Cntrl Enabled) for telnet links, or
               00 if a virtual COM port will be used
               Disconn Time:  00:00
Flush   Mode : 00
Terminal name:

*** Expert
```

Copyright © 2008, 2009, 2010 by iButtonLink LLC
iButton and 1-Wire are trademarks of Dallas Semiconductor Corp, Dallas, Texas, USA.

```
TCP Keepalive    : 45s
ARP cache timeout: 600s
CPU performance: Regular
Monitor Mode @ bootup : enabled
RS485 tx enable  : active low
HTTP Port Number : 80
SMTP Port Number : 25
MTU Size: 1400
Alternate MAC: disabled
Ethernet connection type: auto-negotiate

*** E-mail
Mail server: 0.0.0.0
Unit       :
Domain     :
Recipient 1:
Recipient 2:

- Trigger 1
Serial trigger input: disabled
  Channel: 1
  Match: 00,00
Trigger input1: X
Trigger input2: X
Trigger input3: X
Message :
Priority: L
Min.  notification interval: 1 s
Re-notification interval  : 0 s

- Trigger 2
Serial trigger input: disabled
  Channel: 1
  Match: 00,00
Trigger input1: X
Trigger input2: X
Trigger input3: X
Message :
Priority: L
Min.  notification interval: 1 s
Re-notification interval  : 0 s

- Trigger 3
Serial trigger input: disabled
  Channel: 1
  Match: 00,00
Trigger input1: X
Trigger input2: X
Trigger input3: X
Message :
Priority: L
Min.  notification interval: 1 s
Re-notification interval  : 0 s
```

Copyright © 2008, 2009, 2010 by iButtonLink LLC
iButton and 1-Wire are trademarks of Dallas Semiconductor Corp, Dallas, Texas, USA.

```
Change Setup:
  0 Server
  1 Channel 1
  3 E-mail
  5 Expert
  6 Security
  7 Defaults
  8 Exit without save
  9 Save and exit            Your choice ? 1

Baudrate (9600) ? 115200
I/F Mode (4C) ?
Flow (00) ?
Port No (10001) ?
ConnectMode (C0) ?
Send '+++' in Modem Mode  (Y) ?
Show IP addr after 'RING'  (Y) ?
Auto increment source port  (N) ?
Remote IP Address : (000) .(000) .(000) .(000)
Remote Port  (0) ?
DisConnMode (00) ? 40 (Telnet Com Port Cntrl Enabled) if using telnet
FlushMode   (00) ?
DisConnTime (00:10) ?:
SendChar 1  (00) ?
SendChar 2  (00) ?
Terminal name () ?


Change Setup:
  0 Server
  1 Channel 1
  3 E-mail
  5 Expert
  6 Security
  7 Defaults
  8 Exit without save
  9 Save and exit            Your choice ? 9

Parameters stored ...
```

The XPort restarts to activate the changes, disconnecting the telnet connection.

Copyright © 2008, 2009, 2010 by iButtonLink LLC
iButton and 1-Wire are trademarks of Dallas Semiconductor Corp, Dallas, Texas, USA.

Documentation Change Log

3-18-2010	Added Connection Timeout information
3-31-2010	Added Fixed default IP information

Copyright © 2008, 2009, 2010 by iButtonLink LLC
iButton and 1-Wire are trademarks of Dallas Semiconductor Corp, Dallas, Texas, USA.

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP