The LINK™ Family Manual

Link 12/12i
Link 45/45i
LinkHub-S
LinkHub-E
LinkUSB/USBi

Table of Contents

List of Figures

List of Tables

Introduction

The LINK™ devices are advanced and intelligent interfaces to be used with Maxim Integrated 1-Wire® and iButton® components. Each uses superior digital and analog methods to accomplish more reliable operations on a wide variety of network topologies. They simplify network communications and offer some diagnostic tools as well.

The LINK™ devices 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. In addition, the firmware is adaptive and automatically makes adjustments for a variety of 1-Wire® network parameters.

The LINK™ devices were designed in cooperation with Maxim Integrated. Maxim Integrated is the originator and manufacturer of the 1-Wire® and iButton® devices as well as various accessories. Some terms used in this document may be trademarks of Maxim Integrated and are used with their permission. Application note 244 from the Maxim-Dallas website describes the improvements implemented by the LINK™ devices in greater detail.

Copyright Notices

Contact Information

iButtonLink
1221 Innovation Drive STE 117
Whitewater, WI 53190

Phone: (262) 662-4029
Fax: (262) 436-2403
Email: info@iButtonLink.com
Website URL: www.iButtonLink.com

Extended Mode Slaves

General

Extended mode slaves are slaves based on a microprocessor instead of a 1-Wire® part from Maxim-Dallas. They exist on the same bus with Maxim-Dallas 1-Wire® parts but do NOT participate in normal mode communications. When the host selects extended mode, the Maxim-Dallas parts disappear from the bus and the extended mode slave appears.

Extended mode is not available on the LinkOEM or the LinkHub.

Extended Mode vs. Normal Mode

Extended mode is a proprietary extension to the 1-Wire® protocol by iButtonLink. LINK^{TM} devices use ASCII commands to select which kind of slave is active on the bus.

Entering extended mode takes advantage of the fact that family code 0x00 is invalid. If Maxim-Dallas ever deploys a normal mode part for family code 0x00, that part can NEVER be mixed with extended mode slaves.

Extended mode is very similar to normal mode with several important differences:

- Extended mode slaves only respond to command sequences that begin with an extended mode reset (<Reset120>).

- Extended mode slaves do NOT emit a presence pulse. Instead, an extended mode slave will return a Status byte after a <Reset120>; bit 0, which is the same as the presence bit in normal mode.

When entering extended mode from normal mode, the LINK^{TM} devices issue the command sequence found below in Table 1 to exit normal mode. Table 2 shows the status byte definition for extended mode.

Command	Description
<Reset480>	Normal mode reset
0x55	Match rom
0x00	Family code (not valid for Maxim-Dallas parts)

Table 1: Table of extended mode commands

Bit	Result	Definition
Bit 0	Clear	Extended mode slave is present
Bit 1	Clear	Extended mode slave is capable of resume
Bit 2	Clear	Extended mode slave is in alarm
Bit 3	Undefined (Reserved)	Reserved
Bit 4	Undefined (Reserved)	Reserved
Bit 5	Undefined (Reserved)	Reserved
Bit 6	Undefined (Reserved)	Reserved
Bit 7	Undefined (Reserved)	Reserved

Table 2: Extended mode status byte definitions

There is no support for extended mode slaves using the 9097U emulation command set.

Serial numbers of normal mode slaves are allocated and managed by Maxim-Dallas. The serial numbers of extended mode slaves are allocated and managed by iButtonLink, LLC

DISCLAIMER:
iButtonLink will only support extended mode slaves manufactured by iButtonLink. Extended mode is a proprietary extension of the 1-Wire® protocol by iButtonLink. Agreements between iButtonLink and Maxim-Dallas to protect their intellectual property do not extend to other manufacturers.

DS2480B/9097U Emulation

The DS2480B is the basis for the Maxim DS9097U series of 1-Wire® adaptors. The LINK™ devices are programmed to closely emulate Maxim DS2480B/9097U serial port interfaces. This allows existing software to be used without modification. It also allows the LINK™ devices to be used in place of the Maxim DS2480B/9097U when increased performance or reliability is required.

The DS2480B/9097U functions that are not emulated by the LINK™ devices are described below:

Calibration
Unlike the DS2480B/9097U, LINK™ devices do not require calibration from the serial port. This is due to the LINK™ devices' crystal-controlled time bases, meaning that drift in the interface data rate and waveform timing is not an issue. The LINK™ devices will ignore the expected calibration byte for DS2480B/9097U.

Overdrive Speed
While the LINK™ devices do not support overdrive speed bus communication, this is not an issue for most applications. Higher speed mode cannot be supported because the analog components in the bus interfaces are carefully tuned for optimum long-line and short-line performance at standard speed.

Flex Timing Modes
The LINK™ devices use 1-Wire® waveform timing that is carefully and dynamically controlled by intelligent control algorithms. Changes to flex timing mode that could be done in the DS2480B/9097U are unnecessary and are not supported in the LINK™ devices. However, the LINK™ devices will emulate the flex mode registers and will appear to behave like the DS2480B/9097U for software compatibility.

EPROM Programming
The LINK™ devices do not support +12V pulses for programming EPROM 1-Wire® or iButton® devices. However, programming pulses for 5 V devices (EEPROM) are supported.

Device Version
When queried using the normal DS2480B/9097U method, the LINK™ devices report "device version 7". This allows the software to distinguish the LINK™ from a standard Maxim DS2480B/9097U interface, which returns a "3". Since single-port LINK™ devices also return "device version 7", the user can use the "   " (space) ASCII command to determine if a LinkHub is present.

Important Information for the LinkOEM, the LinkHub-E, and the LinkHub-S

LinkOEM

- LinkOEM will function with 5V TTL signals only.

LinkHub-S

- Each LinkHub-S contains a DS2401 serial number chip on the internal 1-Wire® bus. The serial number chip will show up on 1-Wire® searches unless the internal bus is deactivated.

- The LinkHub-S is a powered device providing +12 V and +5 V power on the same cable as the 1-Wire® bus. In addition, local EEPROM storage is provided for user configuration data.

- There are five 1-Wire® busses (one internal and four external) that are logically or'ed into a single composite 1-Wire® bus. Software written for the Maxim Integrated DS2480B/9097U interface will work even though the 1-Wire® devices are physically on separate busses. Each external bus on the LinkHub-S can be up to 1,000 feet long, while external busses on the Link, LinkOEM, and LinkUSB can be up to 3,000 feet long. This yields a logical 1-Wire® network of 4,000 feet with four times the network weight of a standard 1-Wire® bus.

LinkHub-E

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

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

- There are five 1-Wire® busses (one internal and four external) that are logically or'ed into a single composite 1-Wire® bus. Software written for the Maxim Integrated DS2480B/9097U interface will work even though the 1-Wire® devices are physically on separate busses. Each external bus on the LinkHub can be up to 1,000 feet long, while external busses on the Link, LinkOEM, and LinkUSB can be up to 3,000 feet long. This yields a logical 1-Wire® network of 4,000 feet with four times the network weight of a standard 1-Wire® bus.

ASCII Control Functions

General

In addition to the DS2480B/9097U emulation, The LINK^{TM} devices also provide many 1-Wire® bus functions in response to ASCII commands. This allows the LINK^{TM} devices to be used with any terminal program or serial port communications API to perform iButton®, 1-Wire® or, in some cases, extended mode bus functions without software drivers, making it easy to use. The LINK™ device 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/9097U emulation.

Case Sensitivity and Symbols

ASCII commands are case sensitive. Make sure to be aware of the case that the command requires.

Please also be aware of the difference between a single quotation mark, also known as an apostrophe, (') and an accent mark (`). In some cases, an apostrophe may prompt a command, but it will not be the same as the command an accent mark prompts.

Active Channel Mask
Command " \C " (backslash + uppercase C)

Command Compatible With: LinkHub-S | LinkHub-E

This command sets the active channel mask for certain LINK^{TM} devices. The channel mask will "mask" any number of the ports on the master so that they do not communicate with the connected slaves, allowing for communication with specific slaves. This command is followed by a two-digit hexadecimal value representing a bit mask for channels that should be active. There are five 1-Wire® busses. The bit patterns are listed in the table below.

A " 1 " makes the channel active, meaning that the master will report the slaves connected to the ports. A " 0 " makes the channel inactive, meaning that the ports will be "masked," and those ports will not be read.

Bit Pattern	Command Input (Syntax)	Channel Activated
0x37	\C37	All busses enabled
0x01	\C01	External bus 1
0x02	\C02	External bus 2
0x04	\C04	External bus 3
0x20	\C10	External bus 4
0x10	\C20	Internal bus

Table 3: Table showing the bit patterns used to activate the different channels

SIMILAR COMMANDS:

- " \$ " – Channel Reporting On/Off
- " \c " – Report Active Channel Mask

Arrival/Departure Scanning
Command " $ " (dollar sign)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkOEM | LinkUSB

This command starts the arrival/departure scanning for some LINK[TM] devices. Figure 1 below shows an arrival on a LinkUSB. Scanning is terminated by a Reset, First, Next, or a number of other operations.

NOTE: A bus scanning function is included in many LINK products to scan and report the arrival or departure of a new 1-Wire® device on a 1-Wire® bus. When turned ON using the " $ " command, the scanning system continually performs "First" and "Next" device discovery operations. It will build a table of device serial numbers, reporting up to eight (for LinkHub-S and LinkHub-E) or 10 (for Link, LinkOEM, and LinkUSB) devices. The **arrival** will be reported as a string with an exclamation point (!), a comma (,), and then the arriving device's serial number. The **departure** will be reported as a question mark (?), followed by a comma (,), and then the serial number of the device departed from the bus. This will make sure that the device is actually gone by doing multiple reads.

```
! , 8000000202214928
! , 9A00000208AC5528
```

Figure 1: Command " $ " showing the arrival of a T-Sense and a T-Probe on a LinkUSB

```
? , 8000000202214928
? , 9A00000208AC5528
```

Figure 2: Command " $ " showing the departure of a T-Sense and a T-Probe on a LinkUSB

On The LinkHub-E and LinkUSB, the current contents of the table can be reported with a " * " (asterisk) command while scanning is active.

NOTE: The scanning system will not work properly if more than eight (for LinkHub-S and LinkHub-E) or 10 (for Link, LinkOEM, and LinkUSB) 1-Wire® or iButton® devices are present on the bus. Exceeding these numbers will exhaust the Link's memory capacity.

Auxiliary Line High Impedance
Command " & " (ampersand)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkOEM | LinkUSB

For the Link, the LinkOEM, and the LinkUSB, this command sets the auxiliary line to an input and reports the current input level.

For the LinkHub-S and the LinkHub-E, this command reports the current state of the switch +5V line on the output connectors.

NOTE: The Link supports the standard RJ-11/RJ-45 type 1-Wire® bus connection, where the two center contacts of the RJ connector are the data and ground connections to the 1-Wire® bus. An auxiliary pin in the RJ connector can be added and used as an output to supply a low level (0 V) or a high level (5 V). It can also be an input, sensing a 0 - 5 V logic level. This extra I/O defaults to input (Hi-Z) mode.

NOTE: The Link USB supports the standard RJ-45 type 1-Wire® bus connection, where the center two contacts of the RJ connector are the data and ground connections to the 1-Wire® bus. An auxiliary pin in the RJ connector can be added and used. By default, this line is driven to the HIGH impedance state and left unpowered. It can be an output to supply a low level (0 V) or a high level (5 V). It can also be used as an input, sensing a 0 - 5 VDC logic level. When set to the 5 V level using the "d" command, this line can provide power for 1-Wire® devices. The current available is limited to approximately 75mA.

```
1
```

Figure 3: The LinkUSB reporting the current input level is 1

For the LinkHub-S, the LinkHub-E, and the LinkOEM, the load on the Aux line should be limited to less than 20 mA.

SIMILAR COMMANDS:
- " d " – Auxiliary Line High Power and Low Impedance
- " z " – Auxiliary Line Low Power and Impedance
- " " " – Auxiliary Output Isolation Mode

Auxiliary Line High Power & Low Impedance
Command " d " (lowercase D)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkOEM | LinkUSB

For the Link, the LinkOEM, and the LinkUSB, this command sets the auxiliary line to HIGH level (default) output.

For the LinkHub-S and the LinkHub-E, this command turns on the switched +5V line to the output connectors.

The Link supports the standard RJ-11/RJ-45 type 1-Wire® bus connection, where the two center contacts of the RJ connector are the data and ground connections to the 1-Wire® bus. An Auxiliary pin in the RJ connector can be added and used as an output to supply a low level (0 V) or a high level (5 V). It can also be an input, sensing a 0 - 5 V logic level. This extra I/O defaults to input (Hi-Z) mode.

The Link USB supports the standard RJ-45 type 1-Wire® bus connection where the center two contacts of the RJ connector are the data and ground connections to the 1-Wire® bus. An Auxiliary pin in the RJ connector can be added and used. By default, this line is driven to the HIGH impedance state and left unpowered. It can be an output to supply a low level (0 V) or a high level (5 V). It can also be used as an input, sensing a 0 - 5 V logic level. When set to the 5 V level using the "d" command, this line can provide power for 1-Wire® devices. The current available is limited to approximately 75mA.

NOTE: For the LinkHub-S, the LinkHub-E, and the LinkOEM, the load on the Aux line should be limited to less than 20 mA

SIMILAR COMMANDS:
- " & " – Auxiliary Line High Impedance
- " z " – Auxiliary Line Low Power and Impedance
- " " " – Auxiliary Output Isolation Mode
.

Auxiliary Line Low Power & Impedance
Command " z " (lowercase Z)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkOEM | LinkUSB

For The Link, The LinkOEM, and The LinkUSB, this command takes the auxiliary line to a LOW level and LOW impedance.

For The LinkHub-S and The LinkHub-E, this command turns off the switched +5 V line to the output connectors.

For the LinkHub-S, the LinkHub-E, and the LinkOEM, the load on the auxiliary line should be limited to less than 20 mA.

The Link supports the standard RJ-11/RJ-45 type 1-Wire® bus connection, where the two center contacts of the RJ connector are the data and ground connections to the 1-Wire® bus. An auxiliary pin in the RJ connector can be added and used as an output to supply a low level (0 V) or a high level (5 V). It can also be an input, sensing a 0 - 5 V logic level. This extra I/O defaults to input (Hi-Z) mode.

The Link USB supports the standard RJ-45 type 1-Wire® bus connection where the center two contacts of the RJ connector are the data and ground connections to the 1-Wire® bus. An Auxiliary pin in the RJ connector can be added and used. By default, this line is driven to the HIGH impedance state and left unpowered. It can be an output to supply a low level (0 V) or a high level (5 V). It can also be used as an input, sensing a 0 - 5 V logic level. When set to the 5 V level using the "d" command, this line can provide power for \1-Wire® devices. The current available is limited to approximately 75 mA.

SIMILAR COMMANDS:
- " d " – Auxiliary Line High Power and Low Impedance
- " & " – Auxiliary Line High Impedance
- " " " – Auxiliary Output Isolation Mode

Auxiliary Output Isolation Mode
Command " " " (quotation)

Command Compatible With: Link | LinkOEM | LinkUSB

On LINK[TM] devices, this command will cause the auxiliary output to enter isolation mode and become an image of the 1-Wire® bus transmitter drive waveform. This signal can be useful when trying to accomplish an isolated 1-Wire® connection or whenever a 2-Wire connection is preferred. The original 1-Wire® bus remains unaffected. Figure 4 shows how isolation mode runs on a 1-Wire® device.

Setting the auxiliary output level using the " z " or " d " commands will turn this feature off, as will any reset of the device.

When in the isolation mode, the LINK[TM] devices output the 1-Wire® driving waveform on the aux pin. This can be used to extend the 1-Wire® bus using other signaling protocols. It can also be used to isolate the bus using optical or galvanic isolation devices. The following diagram shows how isolation mode could be used to extend the 1-Wire® port on the LINK[TM] devices.
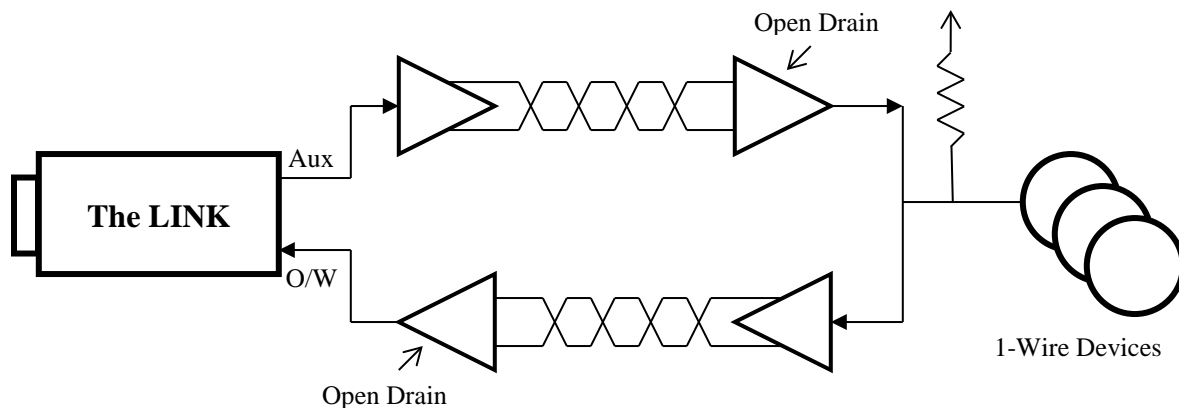
Figure 4: Isolation Mode Extending 1-Wire® Port on LINK Device

The LINK[TM] devices also have a mode where the timing of the 1-Wire® waveforms is slowed to accommodate the longer turn-around times of extended host connections like that shown above. In normal mode, network lengths of up to 3,000 feet can be used (depending on the medium). In extended mode, network lengths of between 2,000 and 6,000 feet can be achieved.

SIMILAR COMMANDS:
- " & " - Auxiliary Line High Impedance
- " d " – Auxiliary Line High Power and Low Impedance
- " z " – Auxiliary Line Low Power and Impedance

Bit Mode
Command " j " (lowercase J)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkOEM | LinkUSB

This command places the LINK device into bit mode. Only single ASCII digits of " 0 " or " 1 " value are allowed and single ASCII digits are returned.

Hitting ENTER (carriage return) will end bit mode.

SIMILAR COMMANDS
- " ~ " - Bit Mode with Strong Pull-Up
- " b " – Byte Mode
- " p " – Byte Mode with Strong Pull-Up

Bit Mode with Strong Pull-Up
Command " ~ " (tilde)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkOEM | LinkUSB

This command places the LINK^{TM} device into byte mode. Power (strong pull-up) is applied to the bus after the last bit of the first byte is issued. Only single ASCII digits of " 0 " or " 1 " value are allowed and single ASCII digits are returned.

Hitting the ENTER (carriage return) will end bit mode.

NOTE: Subsequent bytes will be performed but will not be followed by strong pull-up.

SIMILAR COMMANDS:
- " j " – Bit Mode
- " b " – Byte Mode
- " p " – Byte Mode with Strong Pull-Up
- " . " – Turn Off Dynamic Pull-Up

Byte Mode
Command " b " (lowercase B)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkOEM | LinkUSB

This command places the LINK^{TM} devices into byte mode. The next two characters entered will be taken as a hexadecimal byte value that is 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.

Hitting ENTER (carriage return) will end byte mode.

For The LinkHub-E, " x " will also end byte mode.

SIMILAR COMMANDS:
- " j " – Bit Mode
- " ~ " - Bit Mode with Strong Pull-Up
- " p " – Byte Mode with Strong Pull-Up

Byte Mode Subcommands

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkUSB

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

- " **P** " (uppercase P) – Issue 64 byte reads to the 1-Wire® bus and return results. This subcommand is used to read a memory page from devices that have a page size of 64 bytes.
  - o The Link and LinkUSB will return 128 characters followed by <CR><LF>.
  - o The LinkHub-E will return 64 pairs of hex characters.

- " **p** " (lowercase P) – Issue 66 byte reads to the 1-Wire® bus and return results. This subcommand is used to read a memory page with CRC from devices that have a page size of 64 bytes. Returns 132 characters followed by <CR><LF>.
  - o The LinkHub-E will return 66 pairs of hex characters.

- " **M** " (uppercase M) – Issue 32 byte reads to the 1-Wire® bus and return results. This subcommand is used to read a memory page from devices that have a page size of 32 bytes. Returns 64 characters followed by <CR><LF>.
  - o The LinkHub-E will return 32 pairs of hex characters.

- " **m** " (lowercase M) – Issue 34 byte reads to the 1-Wire® bus and return results. This subcommand is used to read a memory page with CRC from devices that have a page size of 32 bytes. Returns 68 characters followed by <CR><LF>.
  - o The LinkHub-E will return 34 pairs of hex characters.

- **Can be used with The LinkHub-E ONLY**: " ***** " (asterisk) – Inserts carriage return, linefeed pair of characters into the returned data stream. Provides input terminator for host program without leaving byte mode. Generates no activity on 1-Wire® bus and is not echoed.

Byte Mode with Strong Pull-Up
Command " p " (lowercase P)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkOEM | LinkUSB

This command places the LINK^{TM} devices into byte mode with strong pull-up. The next two characters entered will be taken as a hexadecimal byte value that 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.

Hitting ENTER (carriage return) will end byte mode.

For The LinkHub-E, " x " will also end byte mode.

This command differs from the " b " command in that power (strong pull-up) is applied to the bus after the last bit of the first byte is issued.

Subsequent bytes will be performed but will not be followed by strong pull-up.

SIMILAR COMMANDS:
- " j " – Bit Mode
- " ~ " - Bit Mode with Strong Pull-Up
- " b " – Byte Mode
- " p " – Byte Mode with Strong Pull-Up
- " . " – Turn Off Dynamic Pull-Up

Channel Reporting On-Off
Command " \$ " (backslash + dollar sign)

Command Compatible With: LinkHub-E

This command toggles channel reporting on or off for certain LINK^{TM} devices. 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.

SIMILAR COMMANDS:
- " \C " - Active Channel Reporting
- " \c " - Report Active Channel Mask

Display

Command "   " (space)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkOEM | LinkUSB

This command causes **The Link** to display its ID string and version number.

This command causes **The LinkHub-S** to display its firmware version number.

```
LinkHUB V1.11
```

Figure 5: LinkHub-S displaying its firmware version (1.11)

This command causes **The LinkHub-E** to display its firmware version.

```
LinkHUB-E V1.11
```

Figure 6: LinkHub-E displaying its firmware version (1.11)

This command   causes **The LinkOEM** to display its ID string and version number. An example can be found below in Figure 7.

```
LinkOEM V1.7
```

Figure 7: LinkOEM displaying its ID string (LinkOEM) and version number (V1.7)

This command causes **The LinkUSB** to display its ID string and version number. An example can be found below in Figure 8 below.

```
LinkUSB V1.7
```

Figure 8: LinkUSB displaying its ID string (LinkUSB) and version number (V1.7)

SIMILAR COMMANDS:

- " /s " – Display Serial Number

Display Serial Number
Command " \s " (backslash + lowercase S)

Command Compatible With: Link | LinkUSB

This command displays the hardware serial number of certain LINK^{TM} devices. The serial number is displayed followed by <CR><LF> and is formatted the same as all devices.

The Link and The LinkUSB family codes are 0xFE

The Link and The LinkUSB serial numbers are for registration and software validation purposes. Each device serial number is guaranteed unique. At the present time, the Link and the LinkUSB serial numbers do not participate in any searches and cannot be changed after manufacturing.

Please refer to the **Important Information** section for information on the serial numbers of the LinkHub-S and LinkHub-E.

Family First Search
Command " \f " (backslash + lowercase F)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkUSB

This command is a "Family First" search command for certain LINK^{TM} devices. It is similar to the standard " f " or "First" command, except the "Family First Search" command expects the next two hex characters to match the 1-Wire® family code included in the search. This causes the search to start with a particular family code. The standard " n " or "Next" command is used to retrieve the next 1-Wire® serial number matching the family code from the bus search.

Check the family code returned since the family search does not always end on the last device n the bus. A "?" is returned instead of the normal "+" or "-" character if there is a family search mismatch or overshoot. This is shown below in Figure 9.

```
?,  8000000202214928
?,  9A00000208AC5528
?,  88000015871E8801
```

Figure 9: A LinkHub-S showing that the search was mismatched, indicated by the " ? " in front of each string of serial numbers

The " \f " requires two hex characters as data.

S<small>IMILAR COMMAND</small>(<small>S</small>):
- " f " – First
- " n " – Next

First
Command " f " (lowercase F)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkOEM | LinkUSB

This command performs a 1-Wire® bus "first" operation for some LINK^{TM} devices. This operation searches the bus and finds the first 1-Wire® or iButton® device and displays the device serial number prefixed with a single character to indicate if there are more (" + ") or no more (" - ") parts remaining to be found. Figure 10 shows an example on the LinkUSB. When no more devices are found, the command will display the serial number of the first device found.

```
+, 8000000202214928
```

Figure 10: Command " f " displaying that there is another device found, indicated by "+"

```
+, 8000000202214928
-, 9A00000208AC5528
```

Figure 11: Command "f" showing that there is another device found after the first search, indicated by the " + ". Then, command "n" is used in ordert to bring up the next device found. After this device, there are no other devices on the bus, indicated by the " – ".

For The Link and The LinkUSB, if locator reporting is turned on, the device serial number should be followed by a comma and the serial number of the LinkLocator associated with the 1-Wire® device.

```
-,9700000208994528,FFFFFFFFFFFFFFFF
```

SIMILAR COMMAND(S):
- " n " – Next
- " \f " – Family Search First
- " $ " – Scanning Mode

Help for \ (backslash) Commands
Command " \h " (backslash + lowercase H)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkUSB

This command displays a list of commands that begin with a " \ " (backslash) for LINK™ devices. The **Link** and **LinkUSB** have their own set of expanded commands, found in Table 4. The **LinkHub-S** and **LinkHub-E** also have their own expanded commands, found Table 5.

Each line is terminated with <CR><LF>

```
\\ = Sniff mode
\h = help
\L = Locator reporting on
\l = Locator reporting off
\f = Family Search first
\O = Normal slave mode
\P = Set Page read settings*
\p = Report Page read settings
\X = Extended slave mode
\s = Report Link serial number
```

Table 4: Expanded commands for the Link and LinkUSB

```
\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
```

Table 5: Expanded commands for the LinkHub-S and LinkHub-E

Level Test
Command " l " (lowercase L)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkOEM | LinkUSB

This command will test the level of the 1-Wire® bus and report " 0 " if the bus is low or " l " if the bus is high, followed by an ENTER (carriage return) for some LINK^{TM} devices.

```
1
```

Figure 12: Command " l " returning a "1" to indicate that the level of the bus is high

<u>SIMILAR COMMANDS:</u>
- " x " – Low Level

LinkLocator Reporting Armed
Command " \L " (backslash + uppercase L)
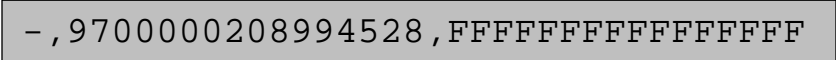
Command Compatible With: Link | LinkUSB

This command arms the reporting of LinkLocator data for the following commands:
- First (" f ")
- Family Search First (" \f ")
- Next (" n ")
- Arrival/Departure Scanning (" $ ")

The LinkLocator serial number associated with the 1-Wire® address is appended to the reporting line. If there is not a LinkLocator associated with the 1-Wire® address, a serial number of " FFFFFFFFFFFFFFFF " is reported.

```
-,9700000208994528,FFFFFFFFFFFFFFFF
```

Figure 13: Command "\L" showing that there is not a LinkLocator associated with the 1-Wire® address

LinkLocator Reporting Disarmed
Command " \l " (backslash + lowercase L)
Command Compatible With: The Link | The LinkUSB

This command can be used with the following LINK™ devices:
- The Link
- The LinkUSB

This disables LinkLocator reporting on certain LINK™ devices.

List of Available Commands
Command " h " (lowercase H)

Command Compatible With: **Link | LinkHub-S | LinkHub-E (except for character commands followed by *) | LinkOEM (except for character commands followed by ∂) | LinkUSB (except for character commands followed by §)**

This command displays the list of LINK ASCII commands that are available for the LINK^{TM} devices. These are in addition to the standard DS2480/DS9097U binary command codes that are available. Table 6 shows all single-character commands that are available on the LINK^{TM} devices.

NOTE: each line is terminated with <CR><LF>

```
f = First            \ = Expanded Command Set ∂
n = Next             ! = Thru *§∂
r = Reset            l = Level
( = Extended ∂       : = 115200 *§
b = Byt(NN+)         ^ = 57600 *
j = Byt(N+)          ` = 38400 *
p = BytW/P           , = 19200 *
~ = BitW/P           x = BusLo
d = Aux+             $ = Scan ∂
z = Aux-             * = List ∂
& = Aux?             h = Help
t = Search Type
```

Table 6: Table showing single character commands

<u>SIMILAR COMMAND(S):</u>
- " \h " – Help for commands beginning with " \ " (backslash)

Low Level
Command " x " (lowercase X)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkOEM | LinkUSB

This command will take the 1-Wire® bus to a low level for LINK™ devices. This function is used to cause a bus-wide reset by robbing power from all the bus devices for a few seconds.

Issue a reset (" r ") command or any bit or byte command to return the 1-Wire® bus to functionality.

```
1
0
```

Figure 14: The first line shows that that bus was previously at level 1. Next, command " x " is used, which shows that the bus is now at a low level.

SIMILAR COMMANDS:
- "1" – Level Test

Next

Command " n " (lowercase N)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkOEM | LinkUSB

This command performs a 1-Wire® bus "next" operation for some LINK^TM devices. This operation searches the bus and returns the next 1-Wire® or iButton® device. It then displays the device serial number prefixed with a single character to indicate if there are more (" + ") or no more (" - ") parts remaining to be found. An example of this can be found in Figure 15.

```
+, 8000000202214928
-, 9A00000208AC5528
```

Figure 15: Command " f" indicating that there are more devices on the bus, shown with the " + ". Then, command " n " is entered, displaying that there are no more devices found after the second device, indicated by "-"

If this command is used again after a " - " response is received, this command will find the first part again.

If locator reporting is turned on, the device serial number should be followed by a comma and the serial number of the LinkLocator associated with the 1-Wire® device.

If an " \f " (family search first command) is in progress, "next" may return a "?" if encountering a family code overshoot.

SIMILAR COMMAND(S):
- " f " – First
- " \f " – Family Search First
- " $ " – Scanning Mode

Pass-Thru Mode
Command " | " (vertical bar)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkOEM | LinkUSB

This command will cause the LINK^{TM} device 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 inverted to the serial port input line.

This mode can be used to bypass LINK^{TM} devices 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 pass-thru mode is a power-on reset of the LINK^{TM} device.

RAM & Register States Dump (LinkHub-E)
Command " \! " (backslash + exclamation point)

Command Compatible With: LinkHub-E

This command causes a formatted dump of the RAM contents and register states to be produced on the serial interface. A watchdog timer reset of the LINK^{TM} device occurs after the dump is complete.

The LINK^{TM} is rebooted at the completion of this command.

RAM & Register States Dump (LinkHub-S)
Command " ! " (exclamation point)

Command Compatible With: LinkHub-S

This command causes a formatted dump of the LINK^{TM} devices' RAM contents and register states to be produced on the serial interface. A watchdog timer reset of the LINK^{TM} device occurs after the dump is complete.

Read EEPROM ASCII Data
Command " \r " (backslash + lowercase R)

Command Compatible With: LinkHub-S | LinkHub-E

This command reads the ASCII data from EEPROM. Data is read from the EEPROM and returned as ASCII characters. The read is ended by a carriage return from the EEPROM data stream. A 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>
\M000100\r\r\r\r
```

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>
```

Read EEPROM Hex Data
Command " \R " (backslash + uppercase R)

Command Compatible With: LinkHub-S | LinkHub-E

This command reads the EEPROM hex data for certain LINK^{TM} devices. This command is followed by a two-digit hexadecimal number specifying the number of bytes to read. The contents of each EEPROM byte will be returned as a two-digit hex number ending with a carriage return.

<u>**EXAMPLE:**</u>

The command " \M003FF\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 ".

Relax Timing
Command " (" (left parenthesis)

Command Compatible With: Link | LinkHub-E | LinkUSB

This command relaxes the 1-Wire® timings of the above LINK^{TM} devices and helps to accommodate long bus lengths. If you are experiencing bus errors due to reflections from busses over 300 feet in length, use this timing.

Report Active Channel Mask
Command " \c " (backslash + lowercase C)

Command Compatible With: LinkHub-S | LinkHub-E

This command reports the active channel mask. The LinkHub-S and The LinkHub-E will respond with a two-character hex value reporting the current channel mask contents.

```
37
```

Figure 16: A LinkHub-S returning a two-character hex value of 37 to indicate that there is activity on the channel mask

SIMILAR COMMANDS:
- " \C " - Active Channel Mask
- " \$ " - Channel Reporting On/Off

Report EEPROM Read/Write Address
Command " \m " (backslash + lowercase M)

Command Compatible With: LinkHub-S | LinkHub-E

This command reports EEPROM read/write addresses for certain LINK™ devices. The next EEPROM read/write address will be returned followed by a carriage return.

```
\m
0001FF<CR>
```

Figure 17: Command " \m " returning "0001FF<CR> ," meaning that the next byte to be read/written is " 0001FF " hex.

Report Maximum Internal EEPROM Address
Command " \E " (backslash + uppercase E)

Command Compatible With: LinkHub-S | LinkHub-E

This command reports the maximum internal EEPROM address for certain LINK^{TM} devices. The LINK^{TM} devices will respond with a six-digit hexadecimal number that represents the address specified using read/write EEPROM commands.

The LinkHub-E and LinkHub-S include 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.

NOTE: The **32 KB** LinkHub-S and LinkHub-E have 32678 EEPROM addresses and will respond with 007FFF.

Report Scan List
Command " * " (asterisk)

Command Compatible With: LinkHub-E | LinkUSB

This command causes some LINK^{TM} devices to report a scan list. If scanning is not active when this command is entered, the response is a single-character carriage return.

A bus scanning function is included in many LINK^{TM} products to scan and report the arrival or departure of a new 1-Wire® device on a 1-Wire® bus. When turned ON using the " $ " command, the scanning system continually performs "First" and "Next" device discovery operations. It will build a table of device serial numbers, reporting up to eight (for LinkHub-S and LinkHub-E) or 10 (for Link, LinkOEM, and LinkUSB) devices. The **arrival** will be reported as a string with an exclamation point (!), a comma (,), and then the arriving device's serial number. The **departure** will be reported as a question mark (?), followed by a comma (,), and then the serial number of the device departed from the bus. This will make sure that the device is actually gone by doing multiple reads.

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

```
!, 8000000202214928
!, 9A00000208AC5528

9A00000208AC5528
8000000202214928
```

Figure 18: The first two lines show the results of arrivals being scanned with command " $ ". The next line is a line break, which was caused by command " * ". Following the line break are the results of command " * ", which are the serial numbers of the T-Probe

The scanning system will not work properly if more than eight (for LinkHub-E) or 10 (for LinkUSB) 1-Wire® or iButton® devices are present on the bus. Exceeding these numbers will extend the Link's memory capacity.

SIMILAR COMMANDS:
- " $ " – Arrival/Departure Scanning

Search Type
Command " t "   (lowercase T)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkOEM | LinkUSB

For some LINK^{TM} devices, this command will change the search type of "f" (first) and "n" (next) when followed by a two-character hexadecimal value.

Using " tF0 " will make the search normal.

Using " tEC " will make the search conditional and will only discover devices for which the search conditions are satisfied.

Set EEPROM Read/Write Address
Command " \M " (backslash + uppercase M)

Command Compatible With: LinkHub-S | LinkHub-E

This command sets the EEPROM read/write addresses for certain LINK^{TM} devices. This command is followed by a six-digit hexadecimal address specifying the next EEPROM address to be read/written.

EXAMPLE:

" \M000030 " would specify address 30 (hex) as the next read/write address

Sniffer Mode
Command " \\ " (backslash + backslash)

Command Compatible With: Link45/45i | LinkOEM**

This command causes the above LINK^{TM} devices to enter 1-Wire® sniffer mode and switch to the 57600-baud serial port rate. Sniffer mode is a way to watch the communications on a 1-Wire® bus. This is necessary to keep up with standard speed 1-Wire® bus data. The 1-Wire® sniffer function listens to the 1-Wire® bus and decodes data on the bus. The LINK^{TM} device stops serving as the bus master in this mode. Data detected on the device is converted into hexadecimal bytes and displayed. Each time a 1-Wire® bus reset is detected, a carriage return and line feed <CR><LF> are sent. This mode is used to debug 1-Wire® master programs by capturing the actual data bytes that are observed on the bus.

Overdrive speed is not supported by the sniffer mode.

**NOTE: For the LinkOEM, sniffer mode is activated through a single backslash " \ ".

In sniffer mode, only whole bytes are reported, so any partial bytes that occur prior to a reset will not be reported.

Sending a break condition will cause the device to be reset to default settings and to the default 9,600 baud serial port data rate.

NOTE: The Maxim DS1410 series of parallel-port 1-Wire® interfaces generates narrow out-of-spec pulses that may not make it through filtering, so the bus master may not work with the sniffer function. This filtering is an important part of LINK^{TM} devices' analog front end. DS9097U adaptors or other LINK^{TM} devices will work well as "sniff-able" bus masters.

For the LinkHub-S and LinkHub-E devices, this also works as an escape character for the expanded command set.

Status of Bus & Bus Reset
Command " r " (lowercase R)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkOEM | LinkUSB

This command performs a 1-Wire® bus reset and returns the status of the bus.

Status Response	Description
P	No error; at least one slave is on the bus
N	No error; no slaves are on the bus.
S	Short; the 1-Wire™ bus is shorted.

Each line is terminated with <CR><LF>

The Link and The Link USB will display the status byte as a two-character hexadecimal value following the status character ("P", "N", "S") in extended mode.

NOTE: Extended mode is not available on The LinkOEM or the LinkHub.

Turn Off Dynamic Pull-Up
Command " . " (period)

Command Compatible With: Link | LinkHub-S | LinkHub-E | LinkOEM | LinkUSB

This command will turn OFF the dynamic pull-up (DPU) driver in the 1-Wire® bus interface on some LINK™ devices.

The DPU helps extend the useable length of the 1-Wire® bus by increasing the charge current at the 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 turned off. The LINK™ devices respond with a carriage return/line feed. The DPU is turned back ON by reset of the LINK™ devices through a break or power cycle (NOT by the reset command).

Write EEPROM ASCII Data
Command " \w " (backslash + lowercase W)

Command Compatible With: LinkHub-S | LinkHub-E

This command writes ASCII data to EEPROM for certain LINK^{TM} devices. This command is followed by ASCII characters terminated by a carriage return <CR>. The carriage return is written to the EEPROM.

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

Figure 19: Command " \M000100\wThis is a test<CR> " writes 15 characters to EEPROM beginning with address " 000100 ". The next EEPROM read/write address would be " 00010F ".

Write EEPROM Hex Data
Command " \W " (backslash + uppercase W)

Command Compatible With: LinkHub-S | LinkHub-E

This command writes EEPROM hex data for some LINK^{TM} devices. This command is followed by two hex characters representing the next byte to be written. A carriage return stops the write. The carriage return is not written to the EEPROM.

<u>EXAMPLE:</u>

The command " \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.

The <CR> was not written to the EEPROM

With the LinkHub-E, the " \W " command is followed by one or more pairs of hex characters representing the byte(s) to be written.

19,200 Baud Serial Port Data Rate
Command " , " (comma)

Command Compatible With: Link | LinkHub-S | LinkOEM | LinkUSB

The host terminal will be required to switch to 19,200 baud before it can communicate with the LINK device further.

This command switches the LINK device to the 19,200 baud serial port data rate. When a break condition is detected, the device will reset to the 9,600 baud data rate. Sending the " ^ " command followed by more 9,600 baud data will often cause the device to reset and the speed to return to 9,600 baud.

NOTE: The 1-Wire® bus with relaxed timing for long lines can only process bits at a rate of about 14,000 bits per second (bps). If a user sends characters to the LINK faster than the 1-Wire® protocol rate, the device will drop characters and cause errors. 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 a value greater than 19,200 bps, the host commands must be passed to assure than the 1-Wire® bus is not overrun.

38,400 Baud Serial Port Data Rate
Command " ` " (accent)

Command Compatible With: Link | LinkHub-S | LinkOEM | LinkUSB

The host terminal will be required to switch to 38,400 baud before it can communicate with the LINK™ device further.

This command switches the Link device to the 38,400 baud serial port data rate. When a break condition is detected, the device will reset to the 9,600 baud data rate. Sending the " ^ " command followed by more 9,600 baud data will often cause the device to reset and the speed to return to 9,600 baud.

NOTE: The 1-Wire® bus with relaxed timing for long lines can only process bits at a rate of about 14,000 bits per second (bps). If a user sends characters to the LINK faster than the 1-Wire® protocol rate, the device will drop characters and cause errors. 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 a value greater than 19,200 bps, the host commands must be passed to assure than the 1-Wire® bus is not overrun.

57,600 Baud Serial Port Data Rate
Command " ^ " (caret)

Command Compatible With: Link | LinkHub-S | LinkOEM | LinkUSB

The host terminal will be required to switch to 57,600 baud before it can communicate with the LINK^{TM} device further.

This command switches the LINK^{TM} device to the 57,600 baud serial port data rate. When a break condition is detected, the device will reset to the 9,600 baud data rate. Sending the " ^ " command followed by more 9,600 baud data will often cause the device to reset and the speed to return to 9,600 baud.

NOTE: The 1-Wire® bus with relaxed timing for long lines can only process bits at a rate of about 14,000 bits per second (bps). If a user sends characters to the LINK^{TM} faster than the 1-Wire® protocol rate, the device will drop characters and cause errors. 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 a value greater than 19,200 bps, the host commands must be passed to assure than the 1-Wire® bus is not overrun.

115,200 Baud Serial Port Data Rate
Command " : " (colon)

Command Compatible With: LinkHub-S | LinkOEM

Before utilizing this command, the host terminal will be required to switch to 115,200 baud before it can communicate with the LINK^{TM} device further.

This command switches the LINK^{TM} device to the 115,200 baud serial port data rate. When a break condition is detected, the device will reset to the 9,600 baud data rate. Sending the " ^ " command followed by more 9,600 baud data will often cause the device to reset and the speed to return to 9,600 baud.

NOTE: The 1-Wire® bus with relaxed timing for long lines can only process bits at a rate of about 14,000 bits per second (bps). If a user sends characters to the LINK^{TM} faster than the 1-Wire® protocol rate, the device will drop characters and cause errors. Streaming bytes using the " b " command will fail if the baud rate is set to more than 19,200 bps because the host will overrun the 1-Wire® bus. When the baud rate is set to a value greater than 19,200 bps, the host commands must be passed to assure than the 1-Wire® bus is not overrun.

APPENDIX A: Power Issues and Requirements

The Link12, Link12i, Link45, and **Link45i** devices obtain power to operate by "robbing" it from two of the host DBN-9 serial port's available signal lines. These signals (DTR and RTS) should both be held in the high state for best performance. However, the LINK device will often operate just fine with only one of these lines high. The LINK^{TM} devices use an efficient switching regulator to convert the port pin power to useable levels with minimal losses.

Different terminal programs handle the signal lines in different ways. If both RTS and DTR are taken low for a few seconds, The LINK device will be robbed of all power and will be reset to its power-up state. If the host computer is a very low power type, such as a laptop or palm-top computer, it may not provide sufficient operation power for the LINK device. In the situation where port power may not be sufficient, it is important to make sure that the host program holds both DTR and RTS in the high state.

Many very small (palm-top) computers generate serial signals that are as low as 3 volts (which are not legal RS232 voltages). These computers will NOT work with the LINK^{TM} devices. Desktop PCs and quality laptops will provide valid RS232 signal levels and more than enough power to operate the LINK device.

The LINK^{TM} devices provide an auxiliary I/O line on the RJ connector. This line, by default, is driven to the high impedance state and left unpowered. When set to the 5V level by the " d " command, this line can be used to provide power for type 1-Wire® devices. However, the current available from this output is limited. The amount of current that the Aux line can provide depends on the amount of power available from the serial port used. As a rule, the load on the Aux line should be limited to less than 20 mA.

Conversely, the **LinkHub-S and LinkHub-E require an external power source in order to operate.** These two products need a supply that provides **500 mA of +12 V**. The DC barrel should have the specifications found below in Table 7.

OD	5.5 mm
ID	2.5 mm
Length	11.8 mm
Other	Negative to center post

Table 7: DC barrel specifications

APPENDIX B: 1-Wire® Communications Examples

The sequence for reading a DS18B20 temperature sensor via a LINK device is:
1. Issue a 1-Wire® reset using " r "

2. Enter byte mode using " b " (in pull-up mode if supported by device " p ") and address the ROM by sending " 0x55 " followed by the ROM address in reverse byte order

    a. **For the LinkHub-S and the LinkHub-E,** send " 55 " followed by the ROM address in reverse byte order
        i. Example: if the discovered ID is " E60000003DA0E128 ", you would address it as " 28E1A03D000000E6 "

3. Send the convert command " 44 " and exit byte mode <CR>

4. Wait at least 900 ms for the conversion to complete, then issue a 1-Wire® reset " r "

5. Enter byte mode using " b " and address the ROM as before

6. Send the read command of " BE "
    a. **For The LinkHub-S, The LinkHub-E, and The LinkOEM,** send the read command of " BE "

7. Send two read commands as " FFFF ". The echoed/returned data will contain the temperature reading in Intel (little-endian) order as a 16 bit signal integer (may be formatted as four hex characters)

8. Exit byte mode <CR>

EXAMPLE:
The returned value will be in 1/16 degree Celsius increments. A return value of 5701 represents 0x0157, or 343 in decimal. Dividing by 16 gives you 21.4° C (70.6° 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 of the 18B20 is 85°C (185°F). If you receive this value, it might indicate that this device has never executed a convert. It is possible to address an 18B20 and NOT have enough power available for it to execute a convert.

APPENDIX C: LED Indicator

The LinkUSB includes an LED indicator located on a corner near the USB cable attachment. The LED illuminates when there is traffic on the 1-Wire® bus.

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 side indicates the presence of 1-Wire® bus traffic. The LED will illuminate when there is 1-Wire® traffic on any bus. Figure 20 below shows the location of the ports and LED indicators on the LinkHub-E.

The LinkHub-S has two LED indicators on the right side. The LED on the left is a heartbeat indicating that the LinkHub-S is running. When idle, the LED will cycle once a second. The LED on the right indicates 1-Wire® bus traffic and will illuminate when there is 1-Wire® traffic on any bus.

APPENDIX D: Link 12/Link 12i Wiring Diagram

Pin outs viewed looking into The LINK 1-Wire® socket
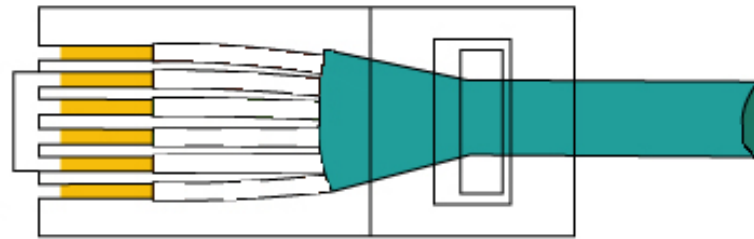
EIA/TIA-568B

Figure 20: Standard EIA/TIA 568 B RJ-45 pins

Used with thanks to Lp for public domain image: https://commons.wikimedia.org/wiki/File:RJ-45_TIA-568A_Left.png#/media/File:RJ-45_TIA-568B_Left.png

Pin	Connection	Cable Color
1	No Connection	White and Green
2	Auxiliary	White and Orange
3	No connection	Blue and White
4	1-Wire® Data	White and Blue
5	Auxiliary	Orange and White
6	1-Wire® Data Return	Green and White

APPENDIX E: Link 45/Link 45i Wiring Diagram

Pin outs viewed looking into The LINK 1-Wire® socket

Wire colors are standard for category 5 (cat 5) cables.
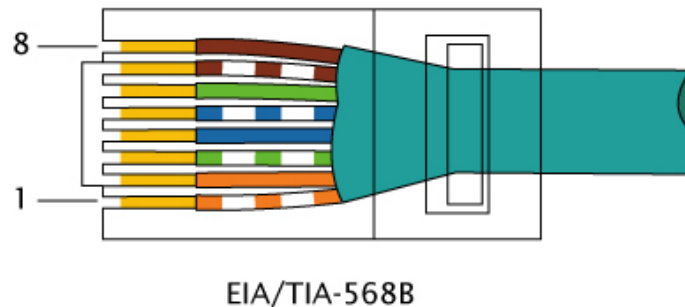
EIA/TIA-568B

Figure 21: Standard EIA/TIA 568 B RJ-45 Colors

Used with thanks to Lp for public domain image: https://upload.wikimedia.org/wikipedia/commons/6/60/RJ-45_TIA-568B_Left.png

Pin	Connection	Cable Color
1	No Connection	Orange and White
2	No Connection	Orange
3	No Connection	Green and White
4	1-Wire® Data	Blue
5	1-Wire® Return	White and Blue
6	Auxiliary	Green
7	No Connection	White and Brown
8	No Connection	Brown

APPENDIX F: LinkUSB Wiring Diagram

Pin outs viewed looking into the LinkUSB 1-Wire® socket

Wire colors are standard for cat 5 cable
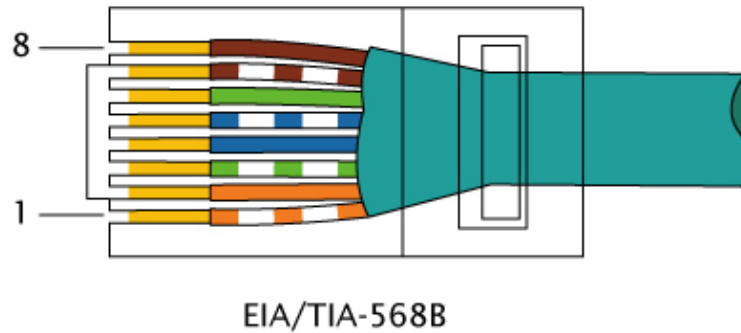
EIA/TIA-568B
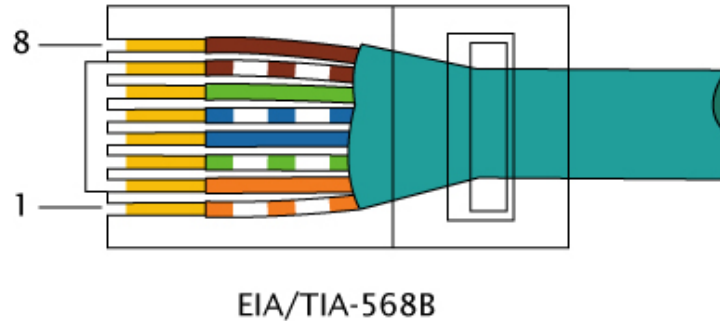
Figure 22: Standard EIA/TIA 568 B RJ-45 Colors

Used with thanks to Lp for public domain image: https://upload.wikimedia.org/wikipedia/commons/6/60/RJ-45_TIA-568B_Left.png

Pin	Connection	Cable Color
1	Ground	Orange and White
2	+5V Power Out *Limited to 50mA	Orange
3	No Connection	Green and White
4	1-Wire® Data	Blue
5	Ground	White and Blue
6	Auxiliary	Green
7	No Connection	White and Brown
8	No Connection	Brown

APPENDIX G: LinkHub-S External Wiring Diagram

Pin outs viewed looking into the LinkHub-S external 1-Wire® port.

Wire colors are standard for cat 5 cable
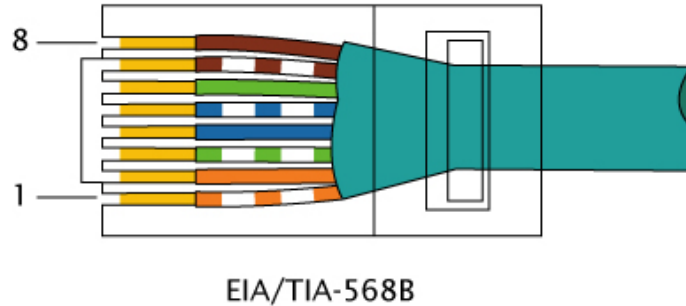
EIA/TIA-568B

Figure 23: Standard EIA/TIA 568 B RJ-45 Colors

Used with thanks to Lp for public domain image: https://upload.wikimedia.org/wikipedia/commons/6/60/RJ-45_TIA-568B_Left.png

Pin	Connection	Cable Color
1	Ground	Orange and White
2	+5V Power Out *Limited to 50mA	Orange
3	Ground	Green and White
4	1-Wire® Data	Blue
5	Ground	White and Blue
6	Auxiliary	Green
7	V+ (+12VDC)	White and Brown
8	Ground	Brown

Power Supply

The LinkHub-S uses a barrel type connector with **negative to the center post**, dimensions: OD:5.5 mm, ID:2.5 mm, Length: 11.8mm.

Nominal +12 V DC supply is required with an acceptable supply range of range of 9-15 volts DC.A regulated supply is preferred. A minimum supply current output of 250 mA is required with 500 mA recommended for additional power to 1-Wire® network devices. A 500 mA +12V DC supply with international certifications and various country AC power plugs is available from iButtonLink: http://www.ibuttonlink.com/pages/power-supplies
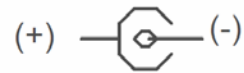
Figure 24: LinkHub-E +12 V DC Power Supply Connector

Serial

Standard Serial RJ-45 type connector.

LinkHub	RJ-45 Jack	DB9-F	EIA
Signal Name	Pin #	Pin #	Signal Name
GND	1	5	GND
232TX	2	3	TXD
232CTS	4	8	CTS
232RX	6	2	RXD
232DTR	7	2	DTR

APPENDIX H: LinkHub-S Wiring Diagram

Pin outs viewed looking into the LinkHub-S port

Wire colors are standard for cat 5 cable.

Figure 25: Standard EIA/TIA 568 B RJ-45 Colors

Used with thanks to Lp for public domain image: https://commons.wikimedia.org/wiki/File:RJ-45_TIA-568A_Left.png

Pin	Connection	Cable Color
1	Return	White and Green
2	V+ (+12V)	Green
3	Switched +5V (Aux)	White and Orange
4	Return	Blue
5	1-Wire® Data	White and Blue
6	Return	Orange
7	+5V	White and Brown
8	Return	Brown

APPENDIX I: LinkHub-E Wiring Diagram

The LinkHub-E has four 1-Wire® network ports, one Ethernet port and one power supply connector as in Figure 26.
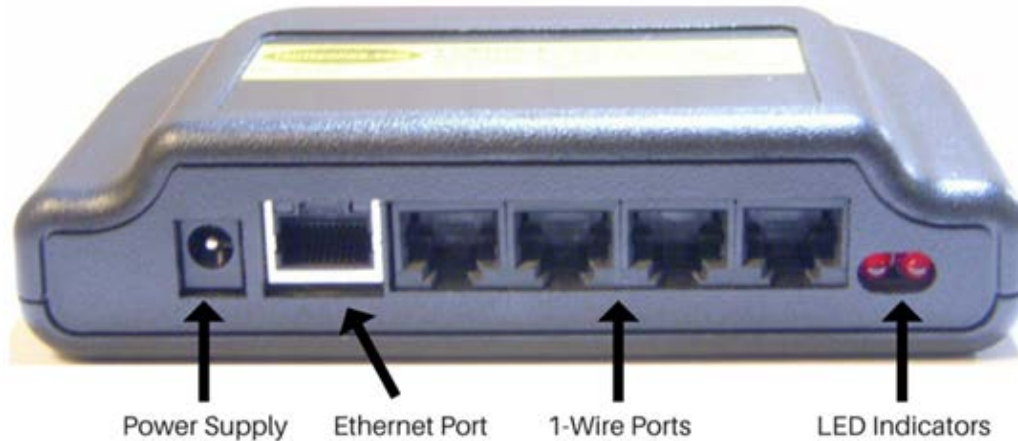
Figure 26: LinkHub-E Connections

Power Supply

The LinkHub-E uses a barrel type connector with **negative to the center post**, dimensions: OD:5.5 mm, ID:2.5 mm, Length: 11.8mm.

Figure 27: LinkHub-E +12 V DC Power Supply Connector

Nominal +12 V DC supply is required with an acceptable supply range of range of 9-15 volts DC.A regulated supply is preferred. A minimum supply current output of 250 mA is required with 500 mA recommended for additional power to 1-Wire® network devices. A 500 mA +12V DC supply with international certifications and various country AC power plugs is available from iButtonLink: http://www.ibuttonlink.com/pages/power-supplies

Ethernet

Standard 10/100 Ethernet RJ-45 type connector.

1-Wire Ports

Each of the four 1-Wire® ports on the LinkHub-E use the following pin out for all wiring.
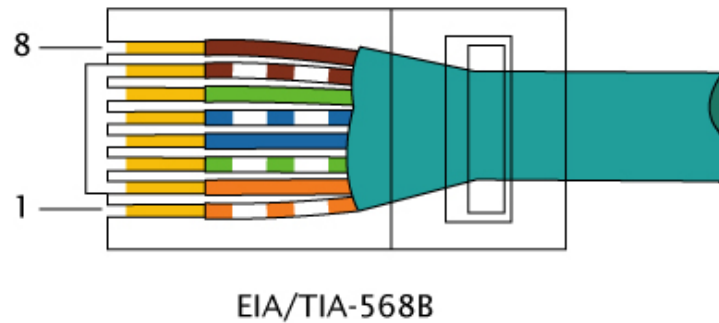
EIA/TIA-568B

Figure 28: Standard EIA/TIA 568B RJ-45 Colors

Used with thanks to Lp for public domain image: https://commons.wikimedia.org/wiki/File:RJ-45_TIA-568A_Left.png

Pin	EIA/TIA 568B Cable Color	1-Wire® Connection	Notes
1	Orange and White	Ground	
2	Orange	+5V	Unregulated and directly connected to the LinkHub power supply input.
3	Green and White	Ground	
4	Blue	1-Wire® Data	
5	White and Blue	1-Wire® Return	
6	Green	Switched 5	
7	White and Brown	+12V	400 mA maximum with standard power adapter at 25C; maximum 250 mA max per 1-Wire® port
8	Brown	+5 Return	Ground

WARNING:

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

APPENDIX J: LinkOEM Pin Outs

Form factor is 24 Pin DIP with 0.600 inch spacing. This follows JEDEC 24 pin 0.6" standards.
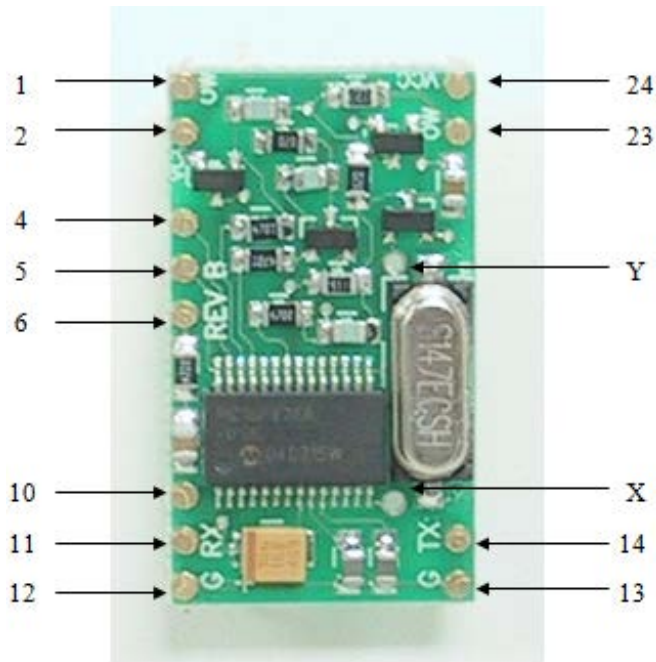
Figure 29: LinkOEM Pin Outs

1 – OW Bus (repeated on 23)

2 – Vcc +5V (repeated on 24)

4 – RB7/PGD used for programming (Pic pin 23)

5 – RB6/PGC used for programming (Pic pin 24)

6 – Master Clear used for programming and reset (Pic pin 1)

10 – Auxiliary

11 – Serial Receive Data

12 – Vss (Return)

13 – Vss (Return)

14 – Serial Transmit Data

X – I2C Clock (Pin 14 – Rc3/SCK/SCL)

Y – I2C Data (Pin 15 – RC4/SDI/SDA)

23 – OW Data (repeated on 1)

24 – Vcc +5V (repeated on 2)

APPENDIX K: FxB Protocol

FxB^{TM} protocol has greatly improved the reliability of 1-Wire® slave-based networks. The FxB^{TM} protocol has 100 times the noise immunity of 1-Wire® and 20 times the parasitic power available to the slave.

A network must consist entirely of FxB^{TM} slaves. However, each 1-Wire® slave can be used with the addition of the FxB^{TM} translator. The FxB^{TM} translator chip takes the FxB^{TM} waveforms from the main communications bus and translates them into compatible 1-Wire® waveforms. The 1-Wire® bus behind the translator can be a maximum of 4 inches long. Translators are available in either chip level (for inclusion in new packages) or packaged (for retrofitting existing slaves).

The LINK^{TM} devices automatically recognize a bus based upon FxB^{TM} protocol after the first reset is issued.

FxB^{TM} is licensed from Alicit Engineering (Chris Fox) by iButtonLink. Alicit currently has a patent pending on the FxB^{TM} protocol.

APPENDIX L: USING 1-WIRE® VIEWER WITH A LINKUSB

The LinkUSB emulates both a DS2480B and a DS9097U. Because of these emulations, the Dallas/Maxim 1-Wire® Viewer will work with the LinkUSB. Users do not need to do anything special in order for the LinkUSB to work with 1-Wire® Viewer, but 1-Wire® Viewer itself can be difficult to install and use.

Maxim provides detailed installation instructions for the 1-Wire® Viewer. These are updated periodically, and can be found at this page. However, users have found some problems and questions not completely covered by the Maxim documentation. Answers to these problems are found below.

1. When downloading the Maxim drivers and the 1-Wire® Viewer, select either the 32-bit version or the 64-bit version. The version selected must match the version of Java installed on the computer. It is possible for a 32-bit version of Java to be installed on a 64-bit version of Windows. If this is the case, 1-Wire® Viewer will not load.

   To fix this, the user should download the other (appropriate) version of 1-Wire® Viewer or download the other (appropriate) version of Java. Either will solve the issue.

2. For the LinkUSB, coost DS9097U as the adapter type. This will require a COM port number. Follow these steps to determine the COM port number:
   a. Make sure that the LinkUSB is plugged in and that Windows has found a driver. The driver should load automatically as it is included in the Windows distribution.
   b. Open "Windows Explorer"/"File Explorer."
   c. Right click on "Computer," "My Computer," or "This PC," and choose "Manage."
   d. Click on "Device Manager" in the Computer Management application.
   e. Find "Ports" in the list that appears and click the arrow next to it to expand it.
   f. Look down the list to find a "USB Serial Port" entry. The COM number will be in parenthesis.
   g. Enter the COM number in the 1-Wire® Viewer setup screen.

3. The 1-Wire® Viewer program will only use COM port numbers 0 – 15. However, it is possible to have a COM port number greater than 15. If this occurs, please follow the following steps to change the COM port number for the LinkUSB.
   a. Make sure that the LinkUSB is plugged in and that Windows has found a driver. The driver should load automatically as it is included in the Windows distribution.
   b. Open "Windows Explorer"/"File Explorer."
   c. Right click on "Computer," "My Computer," or "This PC," and choose "Manage."
   d. Click on "Device Manager" in the Computer Management application.
   e. Find "Ports" in the list that appears and click the arrow next to it to expand it.
   f. Look down the list to find a "USB Serial Port" entry.
   g. Right click on the USB Serial Port entry and choose "Properties"
   h. Click on "Port Settings" and then the "Advanced…." button.

i. In the "COM Port Number" field at te top, use the drop-down menu to select a new COM port number equal or less than 15 that is **not** marked as used.
j. Click "OK" and also click "OK" on the "Properties" window.
k. Unplug and reinsert the LinkUSB.
l. Enter the new COM number in the 1-Wire® Viewer setup screen.

Using 1-Wire® Viewer with Other Link Devices

It should be noted that 1-Wire® Viewer does not readily work with all Link devices. It does not, for example, recognize the LinkHub-S. There is a trick to get around this, but it requires also having a Link45.

Set up a Link45 and plug it into a properly configured serial port. Start 1-Wire® Viewer. 1-Wire® Viewer will recognize the Link45 and initialize. While 1-Wire® Viewer is still running, unplug the Link45 and plug a LinkHub-S into the same serial port. 1-Wire® Viewer will continue to run and detect all the sensors connected to the LinkHub-S.

Index of Commands by Device

Glossary of Terms

#

<CR>

    A single ASCII carriage return character

<LF>

    Line feed; a single ASCII line feed character that indicated the line should be scrolled

<Reset120>

    An extended mode bus reset that consists of a bus low for 120us. Extended mode slaves will respond with a status byte after a <Reset120>

<Reset480>

    A normal mode bus reset as defined by Maxim-Dallas. Consists of a bus low for 480us followed by a presence pulse issued by all normal mode slaves present on the bus (60-240us)

A

Arrival

    An arrival is reported after a new device's serial number with the correct CRC8 and a non-zero family code appears on two subsequent full-discovery passes

ASCII

    [Coding language](#)

B

Break

    A break happens when the transmitter drives the line to a zero state for longer than is supported by the protocol. This may happen because of a transmission error **or** to intentionally reset the interface. This will cause Link devices to reset.

C

Channel mask

    Set value in the LINK to make different channels or combination of channels active or inactive

CRC

    [Cyclic redundancy check](#); error correcting code to make sure communications are correct

D

Departure
>A device will be reported as a departure when it has been present on the bus, but then not found in 10 successive discovery cycles.

Drive waveform
>The shape of the voltage rise or fall on the 1-Wire® bus

DTR
>Data terminal ready

Dynamic pull-up
>Active control of the voltage increase on the 1-Wire® bus

E

EEPROM
>Electrically Erasable Programmable Read Only Memory.

H

Hexadecimal
>Numeral system with base of 16. Uses 0 – 9 to represent zero to nine and A, B, C, D, E, F to represent 10 – 15

I

I/O line
>Input output line

L

Link
>When seen after "Commands Compatible With", this refers to both the Link12/12i and Link45/45i. The Link 12/12i is no longer being produced.

M

Master
>The device that communicates with the slaves and communicates the information from the slaves to another source.

Mismatch
> The "Family First" search command does not find any devices with the same family codes

O

Or'ed
> Binary logic "or". There are five 1-Wire® busses (one internal and four external) that are logically or'ed into a single composite 1-Wire® bus. Software written for the Maxim Integrated DS2480B/9097U interface will work even though the 1-Wire® devices are physically on separate busses. Each external bus on the LinkHub-S can be up to 1,000 feet long, while external busses on the Link, LinkOEM, and LinkUSB can be up to 3,000 feet long. This yields a logical 1-Wire® network of 4,000 feet with four times the network weight of a standard 1-Wire® bus.

Overshoot
> The "Family First" search command has started back at the beginning of the list of devices already found

P

Parasitic power
> A power supply for a device taken directly from the 1-Wire® interface.

Presence pulse
> How a device shows that it is present on the bus

R

RTS
> Ready to send; the Link12/12i and Link45/45i also use this to provide power.

S

Serial port communications API
> Application programming interface; supplies libraries of software to use in an application

Slaves
> Sensors that communicate with the master

Sniffer
> Device that watches the communications; way to observe 1-Wire® traffic

Strong pull-up

    Power is applied to the bus after the last bit of the first byte is issued

T

TTL

    Transistor-transistor logic; switches and maintains logic states through circuits built from bipolar transistors

W

Watchdog timer

    Protection device that resets unit if it fails to perform its action in the specified amount of time

Revision History

Date	Revision	Initials
7/07/16	• Updated RJ-45 connection information and diagrams • Updated Table of Contents	SI
7/21/16	• Added Appendix L • Added revision history • Updated Table of Contents	SI
4/3/17	• Fixed syntax in Table 3 • Fixed bit pattern in Table 3	SI
8/21/17	• Edited version numbers and footers so they are all consistent	SI