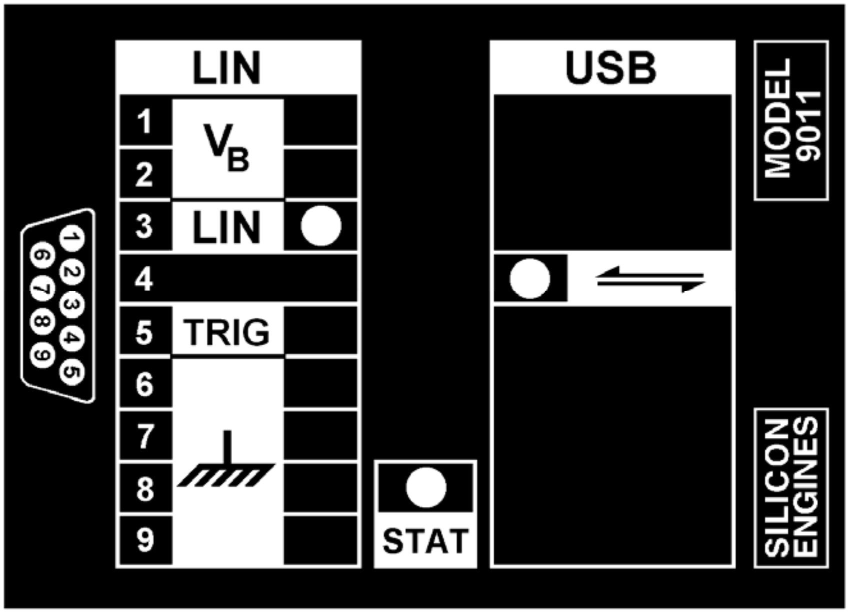
LIN/USB CONVERTER

MODEL 9011

USER'S GUIDE



MODEL 9011 REPLACES MODEL 9004

For a comparison of features, please see Part 12.

WINDOWS® COMPATIBILITY

The Model 9011 is compatible with Windows 2000, Windows XP, Windows Vista, and Windows 7

COMMENTS

We would appreciate receiving corrections and suggestions regarding this document and the product it describes. Please email to sales@siliconengines.net

SILICON ENGINES

3550 W Salt Creek Lane
Suite 105
Arlington Heights, IL 60005 USA
847-637-1180
FAX 847-637-1185
www.siliconengines.net

CONTENTS

1. INTRODUCTION

1.1. SCOPE

This document is the User's Guide for the Silicon Engines ***LIN/USB Converter, Model 9011***, a compact electronic device that allows a personal computer to connect to an automotive communications data link compatible with the Local Interconnect Network (LIN) protocol.

1.2. RELATED AUTOMOTIVE PROTOCOL CONVERTERS

In addition to the Model 9011 LIN/USB Converter, Silicon Engines also offers:

- **Model 9001**, 9141/RS-232 Converter
- **Model 9002**, 14230/RS-232 Converter
- **Model 9003,** LIN/RS-232 Converter
- **Model 9004,** LIN/USB Converter *(older version, replaced by the Model 9011)*
- **Model 9009**, 14230/USB Converter *(older version, replaced by the Model 9010)*
- **Model 9010**, 14230/USB Converter.

For details on these devices, check our web site, www.siliconengines.net.

1.3. MODEL 9011 FUNCTIONS

- **Protocol conversion:** Provides a robust and complete conversion between the automotive LIN protocol and USB (Universal Serial Bus), for connection to a personal computer.
- **Physical layer conversion:** Converts between LIN voltage levels (automotive battery) and USB voltage levels (5 volts).
- **LIN data speeds:** Programmable for operation from 1,000 bps (bits per second) to 21,000 bps over the LIN bus. This range includes the standard speeds recommended in the LIN specifications—2400, 9600, and 19,200 bps. It also covers the LIN data rate currently proposed for use by Detroit OEMs—10,416 bps. It also covers a non-standard data rate currently used in Europe—20,833 bps. The data rate can be set to any non-standard speed within the covered range to test the ability of slave mode ECUs to synchronize with master ECUs.
- **USB data speed:** Operates at USB full speed, 12 megabits per second raw data rate over the USB bus.
- **Operating modes:** Programmable to operate as a LIN bus master, as a LIN bus slave, or in PC control mode.
- **Master mode:** When programmed as the LIN bus master, generates the 13-bit synchronization break, synchronization field, and identifier field at the start of each LIN message frame. Automatically inserts parity bits per the LIN specifications, and calculates the LIN checksum. Sends LIN messages to LIN slaves. Accepts programming from a connected PC using the CLD format defined by the LIN specifications. Once programmed, generates specific master-mode messages automatically, without further PC intervention. A master node also sends acknowledgement messages with error codes if applicable to the PC for all LIN packets that it has sent. In master mode, it also relays all other messages that did not originate from itself as received messages back to the PC. Can also be programmed to keep slave modes awake by generating periodic wake-up messages. Can also be programmed to send an incrementing count in the payload of a master task message.

- **Slave mode:** When programmed as a LIN bus slave, responds to messages received from the LIN master over the LIN bus, sending preset data in the appropriate time slot. Accepts programming from a connected PC using the CLD format defined by the LIN specifications. Programmable as to which specific messages to respond to, and as to default data to send for each message. A slave node also sends acknowledgment messages with error codes if applicable to the PC for all LIN packets that it responds to. For all other traffic it acts as a listener, forwarding a copy of each LIN packet to the PC.
- **PC control mode:** When in PC control mode, decodes all messages on the LIN bus, and passes them via USB to a connected PC. Also capable of generating LIN messages as a LIN bus master under control of the PC. Can also be programmed to keep slave modes awake by generating periodic wake-up messages.
- **LIN bus loading:** Provides a LIN bus load resistor of 1 KΩ when programmed for master mode. Automatically switches to 30 KΩ when programmed for slave mode. Load resistor is programmable in PC control mode.
- **Supplied PC software:** A complete Visual Basic® 6.0 PC software program—operating under Microsoft® Windows®—is provided with the LIN/USB Converter, with source code provided. Utilities are provided for configuring the LIN/USB Converter; downloading CLD files; sending and receiving LIN bus messages; and logging LIN message traffic to a file on the PC. There are reusable header files as well, for both Visual C++ 6.0 and full source code written in Visual Basic 6.0 and Visual Basic .NET 2005, intended to be used with the LINUSB.DLL or LINUSB64.DLL (for 64-bit processing) that handles the interface to the LIN/USB Converter.

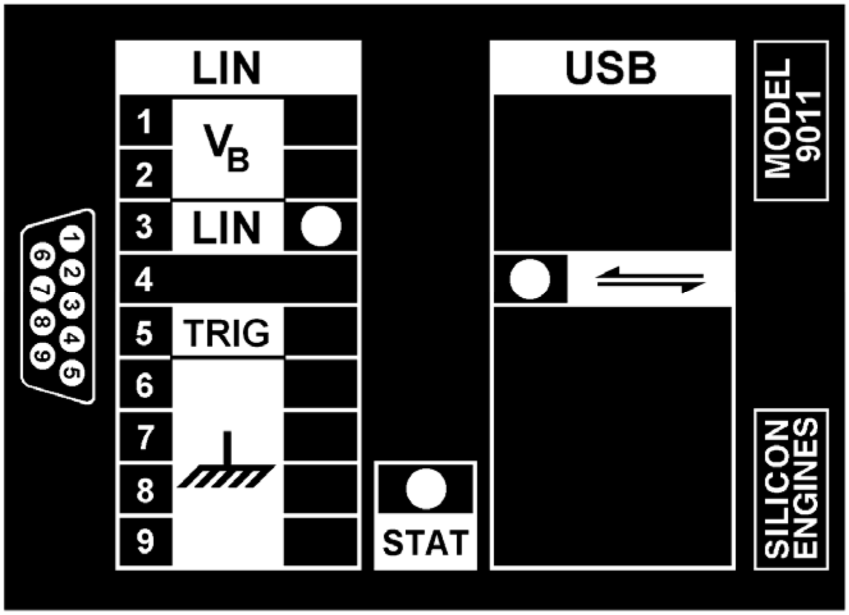
The Model 9011 is compatible with Windows 2000, Windows XP, Windows Vista, and Windows 7.

- **Multiple port capability:** The PC software handles multiple connected LIN/USB Converters simultaneously through a single user interface. A typical arrangement might be one USB master, two USB slaves, and a device operating in PC control mode. Hot-swapping of LIN/USB Converters is permitted.
- **Configuration Language Description:** A utility is provided for downloading CLD files into a LIN/USB Converter. This provides a very convenient way to configure LIN/USB Converters as master or slave nodes with automatic repeating messages. A functional LIN network can be set up for testing purposes without the need to write embedded software for each LIN node.
- **Stop on Error, Trigger on Error:** In addition, the LIN/USB Converter can be programmed to ***stop on error,*** useful in debugging LIN communications problems. It can also be programmed to assert a pulse on the TRIGGER output on an error, so that the user can quickly and easily locate and identify errors in the system, using an oscilloscope or logic analyzer.
- **Trigger on Specific Messages:** The LIN/USB Converter can be programmed to assert a pulse on the TRIGGER output when a specific message is seen on the bus. This feature works in any of the modes that the LIN/USB Converter is in.
- **Windows® DLL:** An easy-to-use DLL file is provided to allow the user to write custom LIN interfaces. Like the provided software program, the DLL supports multiple connected LIN/USB Converters. There is also a 64-bit DLL provided for 64-bit application support.

2. HARDWARE OUTLINE

2.1. ENCLOSURE

The LIN/USB Converter is housed in a compact plastic enclosure.



TOP PANEL OF ENCLOSURE
FIGURE 2.1.1

2.2. ENCLOSURE SIZE

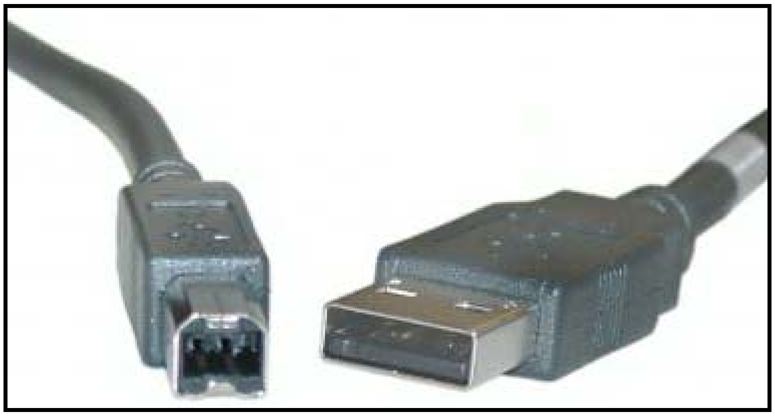
WIDTH	HEIGHT	DEPTH
4.375 IN	3.25 IN	1.5 IN
111 MM	82,6 MM	38,1 MM

ENCLOSURE DIMENSIONS
FIGURE 2.2.1.

2.3. USB CONNECTOR

The connector at the right side of the Model 9010 14230/USB Converter is a USB type B female receptacle. It is intended for connection via a USB cable to a USB port on a personal computer.

A suitable USB cable is provided with each Model 9010, measuring six feet (1,8 meters) in length.



USB CABLE FROM MODEL 9011 TO PERSONAL COMPUTER
FIGURE 2.3.1.

At one end of the cable is a USB type B male plug. The type B plug is shown at the left in the illustration above. It plugs into the USB type B receptacle on the Model 9011.

At the other end of the cable is a USB type A male plug. This type A plug is shown at the right in the illustration above. It plugs into a USB port on the PC.

2.4. LIN CONNECTOR

The connector at the left of the Model 9011 LIN/USB Converter is a type DB9M plug (9-pin male D sub-miniature).

A cable with a DB9F (female DB9) socket plugs in here. Four signals are supported: VBATT, the LIN line, a trigger output, and ground.

PIN NO.	SYMBOL	SIGNAL	DESCRIPTION
1-2	V_B	VBATT	BATTERY POWER
3	**LIN**	LIN	LIN DATA LINE
4	■	NC	*NO CONNECTION*
5	TRIG	TRIGGER	TRIGGER OUTPUT SIGNAL
6-9	⏚	GROUND	POWER AND SIGNAL RETURN

LIN CONNECTOR PIN-OUTS
FIGURE 2.4.1.

These pin-outs, as well as the locations of the pins within the 9-pin connector, are shown on the Model 9011 top panel *(Fig. 2.1.1).*

2.5. LIN CABLE TO SPECIFIC ECU

Users must construct a special cable to connect from the ECU to the Model 9011 LIN/USB Converter. Or contact Silicon Engines for assistance.

At the Model 9011 side, use a DB9F connector. Connect VBATT to pin 1, the LIN data line to pin 3, the TRIG signal to pin 5 (if it is used in your application), and ground to pin 6. The locations of the pins within the connector are shown on the top panel decal *(see Fig. 3.1.1). (See Part 3 below for more information on VBATT and GND.)*

2.6. ELECTRICAL SPECIFICATIONS

PARAMETER	MINIMUM	TYPICAL	MAXIMUM	CONDITIONS/COMMENTS
OPERATING VOLTAGE RANGE	+8 VDC	+13.8 VDC	+18 VDC	BATTERY TO MODEL 9011
LIN/USB CONVERTER LOAD RESISTOR—MASTER MODE	950 Ω	1,000 Ω	1,050 Ω	1 KΩ ±5%
LIN/USB CONVERTER LOAD RESISTOR—SLAVE MODE	28.5 KΩ	30 KΩ	31.5 KΩ	30 KΩ ±5%

LIN/USB CONVERTER ELECTRICAL SPECIFICATIONS
FIGURE 2.6.1

3. POWER REQUIREMENTS

3.1. CONNECTING BATTERY AND GROUND

When working with an automotive ECU, the BAT and GND lines are typically connected—along with the LIN bus line—to a suitable connector on the ECU. Both the LIN/USB Converter and the ECU are powered from the same source of power.

The LIN/USB Converter can also be powered by a DC power supply that connects to building AC power lines.

3.2. INPUT VOLTAGE AND CURRENT

The LIN/USB Converter contains built-in power supply circuitry that generates needed power from VBATT and GND. The LIN/USB Converter is compatible with 12-volt battery systems (8 to 18 VDC), as specified by [*LIN Physical Layer Specification*, Rev. 2.1, Sec. 6.5.4, Table 6.6, Parameter 9].

References to standards and specifications appear in this document in square brackets, such as: [LIN Protocol Specification]. Please see Part 11 for a list of references.

The unit generates +5 VDC internally for digital logic and the LED indicators.

SPECIFICATION	MIN.	TYP.	MAX	UNITS	CONDITIONS
SUPPLY VOLTAGE	8.0		18.0	VDC	CONTINUOUS OPERATION
			60	VDC	LOAD DUMP, 100 MS MAX.
			-60	VDC	REVERSE BATTERY
SUPPLY CURRENT		TBD		MA	VBATT=+12 VDC

SUPPLY POWER SPECIFICATIONS
FIGURE 3.2.1.

3.3. LOAD DUMP PROTECTION

The LIN/USB Converter contains circuitry for protection against automotive load dump transients up to the maximum levels shown above. These levels are adequate for most current vehicle designs.

However, if higher transient levels are anticipated, measures should be taken to protect the LIN/USB Converter. One method is to power the device from an AC line-powered DC power supply, rather than from the vehicle's battery.

3.4. SEPARATE BATTERY SOURCES

If the ECU and the LIN/USB Converter are powered from separate sources:

- **Grounds:** The ground of the LIN/USB Converter must be connected to the grounds of the ECU and of both power sources.

- **BAT:** The BAT voltage provided to the LIN/USB Converter should be within the ranges specified above, and within ±3 VDC of the BAT voltage provided to the ECU.

3.5. REVERSE BATTERY PROTECTION

The LIN/USB Converter is protected against inadvertent reverse battery connection. The unit will not operate properly with reversed power inputs, but will not be damaged, so long as the negative voltage is within the range specified above.

4. LAMP FUNCTIONS

4.1. USB LAMP

At the right of the top panel of the LIN/USB Converter is a bi-color LED lamp, marked **USB**.

LED PATTERN	CONDITION
STEADY GREEN	THIS DEVICE ENUMERATED BY PC USB SOFTWARE, CONVERTER OPERATING NORMALLY
GREEN BLINKING	USB TRAFFIC DETECTED
RED BLINKING	THIS DEVICE NOT YET ENUMERATED BY PC
STEADY RED	NEW SOFTWARE BEING DOWNLOADED; HARDWARE ERROR; NOT CONNECTED TO PC OR SUSPENDED
FAST STEADY FLASHING	DEVICE STACK OVERFLOW; IF EMI EVENT THEN RESET DEVICE; IF NOT EMI EVENT THEN CONTACT SILICON ENGINES

USB LAMP PATTERNS
FIGURE 4.1.1.

4.2. LIN LAMP

At the left of the top panel of the LIN/USB Converter is a bi-color LED lamp, marked **LIN**.

LED PATTERN	CONDITION
STEADY GREEN	CONVERTER OPERATING NORMALLY, LIN BUS IDLE
GREEN BLINKING	LIN TRAFFIC DETECTED
STEADY RED	NEW SOFTWARE BEING DOWNLOADED; HARDWARE ERROR

LIN LAMP PATTERNS
FIGURE 4.2.1.

5. LIN/USB MESSAGE CENTER—INSTALLATION

5.1. FUNCTIONALITY

The **LIN USB Message Center** software utility is provided with the LIN/USB Converter. When installed in a personal computer running Windows®, this program allows you to send messages through the LIN/USB Converter to a connected ECU.

The Model 9011 is compatible with Windows 2000, Windows XP, Windows Vista, and Windows 7.

5.2. INSTALLATION

The **LIN USB Message Center** software is provided on a CD that comes with the product. To install the software, insert the CD into your computer's CD drive. Run the program InstallSEMessageCenterAnd9004And9011Drivers[xxxx].EXE (where [xxxx] is the latest version) in the root folder of the CD. This program will install the software on your computer. If you want to install just the drivers and not the Message Center then run the program InstallSE9004And9011Drivers[xxxx].EXE (where [xxxx] is the latest version) in the root folder of the CD.

Please install the Message Center software first before plugging in the LIN/USB Converter for the first time.

5.3. INSTALLATION OF USB DEVICE DRIVERS

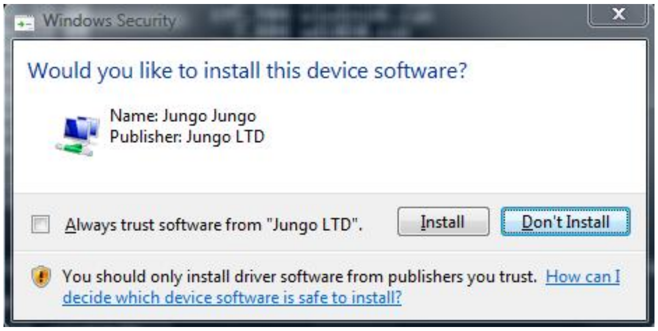
The USB device drivers are installed by either of the two executables in the root folder of the CD. The CD that comes with the product also contains in the Drivers folder the files necessary for USB initialization. These files are also present in the directory `C:\Program Files\Silicon Engines\SELINUSB`. Here are the steps necessary to install the USB device drivers:

Message Center: Install the LIN/USB Message Center software into your PC.

USB: Connect the USB cable from the Model 9011 to your PC.

5.4. WINDOWS 7 AND VISTA INSTALLATION SCREENS

If you are running Windows 7 or Vista, you should see a screen as shown below.



Windows Security

Would you like to install this device software?

Name: Jungo Jungo
Publisher: Jungo LTD

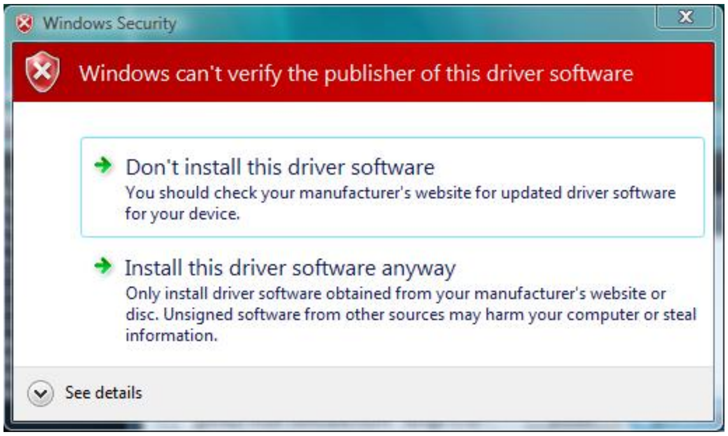
□ Always trust software from "Jungo LTD".

You should only install driver software from publishers you trust. How can I decide which device software is safe to install?

DEVICE SOFTWARE INSTALL SCREEN
FIGURE 5.4.1

Note: Jungo is a supplier of PC-side USB software utilities which Silicon Engines uses for the LIN/USB Message Center.

Go ahead and click the INSTALL button. You should then see the following screen:



Windows Security

Windows can't verify the publisher of this driver software

➔ Don't install this driver software

You should check your manufacturer's website for updated driver software for your device.

➔ Install this driver software anyway

Only install driver software obtained from your manufacturer's website or disc. Unsigned software from other sources may harm your computer or steal information.

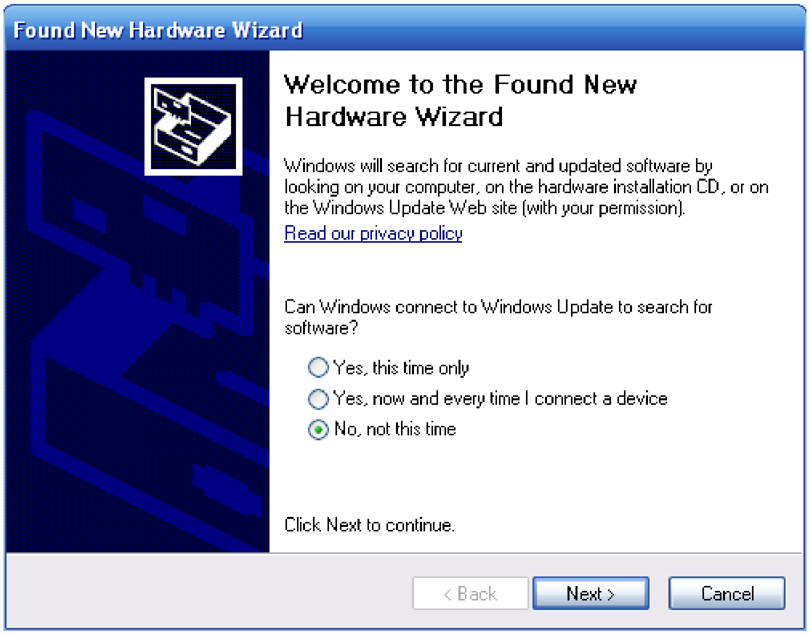
WINDOWS PUBLISHER CHECK
FIGURE 5.4.2

Please click on INSTALL THIS DRIVER SOFTWARE ANYWAY.

The driver software for Windows 7 or Windows Vista should now be installed.

5.5. WINDOWS XP AND 2000 INSTALLATION SCREENS

If you are running Windows XP or 2000 you should see a screen as shown below.



Found New Hardware Wizard

Welcome to the Found New Hardware Wizard

Windows will search for current and updated software by looking on your computer, on the hardware installation CD, or on the Windows Update Web site (with your permission).

Read our privacy policy

Can Windows connect to Windows Update to search for software?

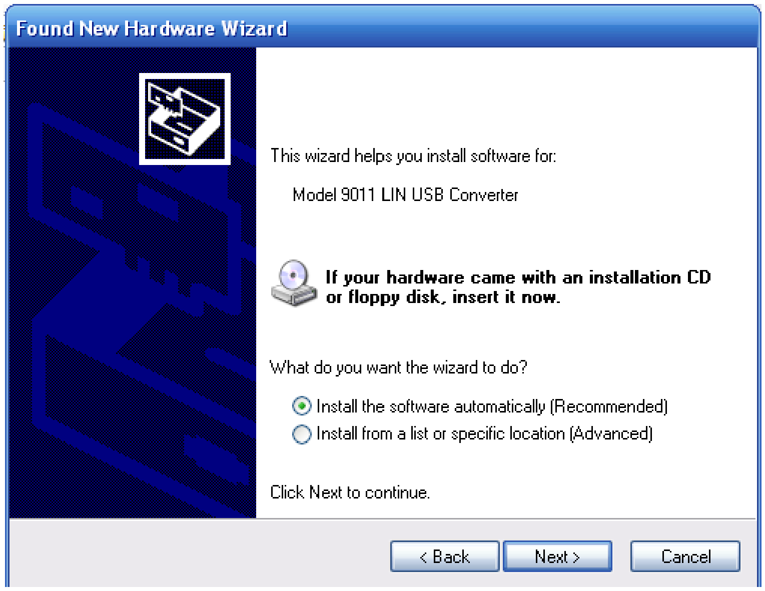
○ Yes, this time only
○ Yes, now and every time I connect a device
● No, not this time

Click Next to continue.

WINDOWS XP AND 2000 INSTALLATION SCREEN
FIGURE 5.5.1

Go ahead and click the NO, NOT THIS TIME button.

You should then see the following screen:



Found New Hardware Wizard

This wizard helps you install software for:

Model 9011 LIN USB Converter

If your hardware came with an installation CD or floppy disk, insert it now.

What do you want the wizard to do?

● Install the software automatically (Recommended)

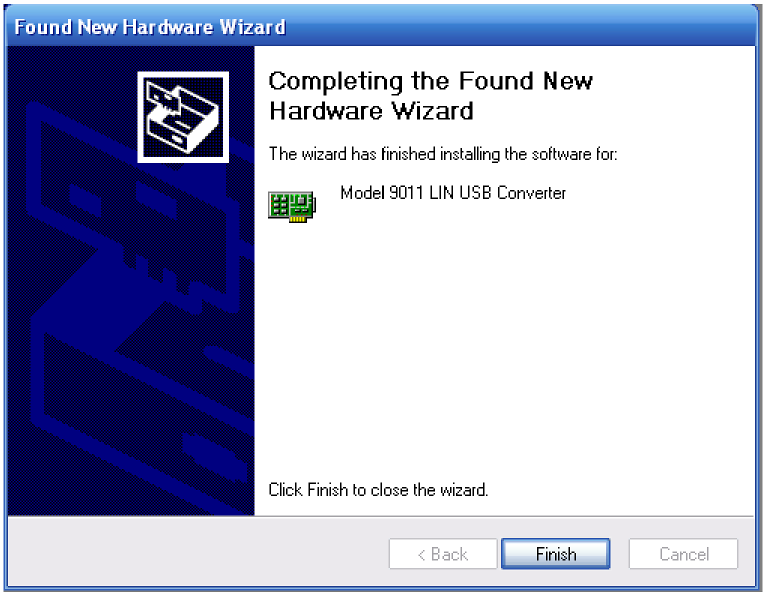
○ Install from a list or specific location (Advanced)

Click Next to continue.

WINDOWS XP/2000 INSTALL SCREEN #2
FIGURE 5.5.2

Please click on INSTALL THIS SOFTWARE AUTOMATICALLY.

You should next see the following screen:



Found New Hardware Wizard

Completing the Found New Hardware Wizard

The wizard has finished installing the software for:

Model 9011 LIN USB Converter

Click Finish to close the wizard.

WINDOWS XP/2000 INSTALL SCREEN #3
FIGURE 5.5.3

Please click the FINISH button. The software should now be installed on your computer.

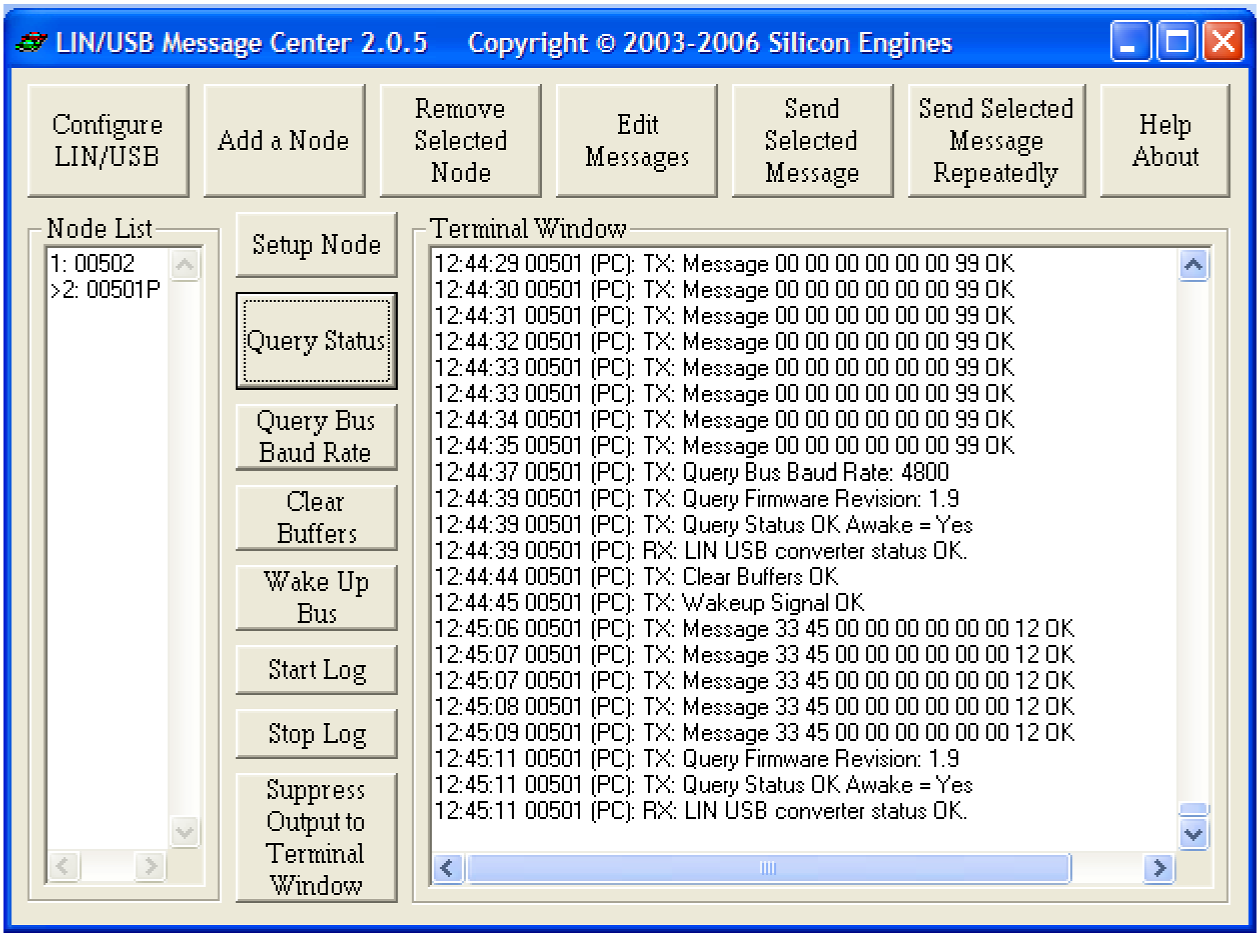
6. LIN USB MESSAGE CENTER

6.1. PROVIDED PC SOFTWARE

The LIN USB Message Center is a PC-based program for controlling one or more LIN/USB Converters. This program is provided at no additional cost with the device. Full source code for the LIN USB Message Center is also provided so that users can build their own applications with the least amount of learning curve.

6.2. MAIN MESSAGE SCREEN

The Main Message Screen is the top-level control center for the Message Center program.



```
12:44:29 00501 (PC): TX: Message 00 00 00 00 00 00 99 OK
12:44:30 00501 (PC): TX: Message 00 00 00 00 00 00 99 OK
12:44:31 00501 (PC): TX: Message 00 00 00 00 00 00 99 OK
12:44:32 00501 (PC): TX: Message 00 00 00 00 00 00 99 OK
12:44:33 00501 (PC): TX: Message 00 00 00 00 00 00 99 OK
12:44:33 00501 (PC): TX: Message 00 00 00 00 00 00 99 OK
12:44:34 00501 (PC): TX: Message 00 00 00 00 00 00 99 OK
12:44:35 00501 (PC): TX: Message 00 00 00 00 00 00 99 OK
12:44:37 00501 (PC): TX: Query Bus Baud Rate: 4800
12:44:39 00501 (PC): TX: Query Firmware Revision: 1.9
12:44:39 00501 (PC): TX: Query Status OK Awake = Yes
12:44:39 00501 (PC): RX: LIN USB converter status OK.
12:44:44 00501 (PC): TX: Clear Buffers OK
12:44:45 00501 (PC): TX: Wakeup Signal OK
12:45:06 00501 (PC): TX: Message 33 45 00 00 00 00 00 12 OK
12:45:07 00501 (PC): TX: Message 33 45 00 00 00 00 00 12 OK
12:45:07 00501 (PC): TX: Message 33 45 00 00 00 00 00 12 OK
12:45:08 00501 (PC): TX: Message 33 45 00 00 00 00 00 12 OK
12:45:09 00501 (PC): TX: Message 33 45 00 00 00 00 00 12 OK
12:45:11 00501 (PC): TX: Query Firmware Revision: 1.9
12:45:11 00501 (PC): TX: Query Status OK Awake = Yes
12:45:11 00501 (PC): RX: LIN USB converter status OK.
```

LIN/USB MESSAGE CENTER MAIN MESSAGE SCREEN
FIGURE 6.2.1

- **Terminal Window:** An active scrolling window showing all messages to and from LIN/USB Converters. Each line contains a time stamp, node number, transmit/receive indication, and message description.
- **Configure LIN Box:** Sets up attached LIN/USB Converters.
- **Node List:** The ***Node List*** window shows the currently active LIN nodes. The alphabetic suffix after the node number shows whether a LIN/USB Converter node has been programmed as a master node (M), as a slave node (S), or in PC control mode (P).

- **Node Control:** The ***Add a Node*** and ***Setup Node*** buttons allow you to add and identify additional LIN/USB Converters. Use ***Add a Node*** to add a node that does not appear in the Node List to the Node list to be available for Selecting. To select a node, simply place the mouse cursor over the Node's serial number in the Node List and click the mouse button. Use ***Setup Node*** to assign a node to a particular function, such as PC-controlled, or master or slave node.
- **Other LIN nodes:** Nodes on the LIN bus that are not LIN/USB Converters—LIN-compatible automotive ECUs—will not appear in the ***Node List***. However their message traffic will appear in the ***Terminal Window***.
- **Message Control:** The ***Edit Messages***, ***Send Selected Message***, and ***Send Selected Message(s) Repeatedly*** buttons are used in PC Control Mode to define LIN messages, and to send them, once or repeatedly.
- **Query Status:** Used to cause a connected LIN/USB Converter to report its current firmware version and operating status, including any existing error messages.
- **Query Baud Rate:** Use this button to determine the currently selected LIN/USB Converter's bus baud rate. Use this if there is any doubt about what baud rate the LIN/USB Converter is currently operating at. The typical cause for confusion is when a node is setup using a CLD file, which contains a set baud rate command that may or may not match the baud rate in the Configuration menu. To revert from a baud rate from a CLD file to the one on the Configuration menu, simply click on the Configure button after the node has been setup from the CLD file.
- **Bus wake-up:** Used to send a message to wake up any LIN nodes that may have entered low-power sleep mode. Note that if wake-ups are enabled, wake-up pulses are sent before each message is sent on a sleeping bus automatically.
- **Buffer control:** The ***Start Log*** button causes all bus activity to be logged to a designated PC file. The ***Stop Log*** button stops data logging. ***Suppress Output to Terminal Window*** will freeze the messages in the terminal window, while allowing other LIN activities to continue in the background.

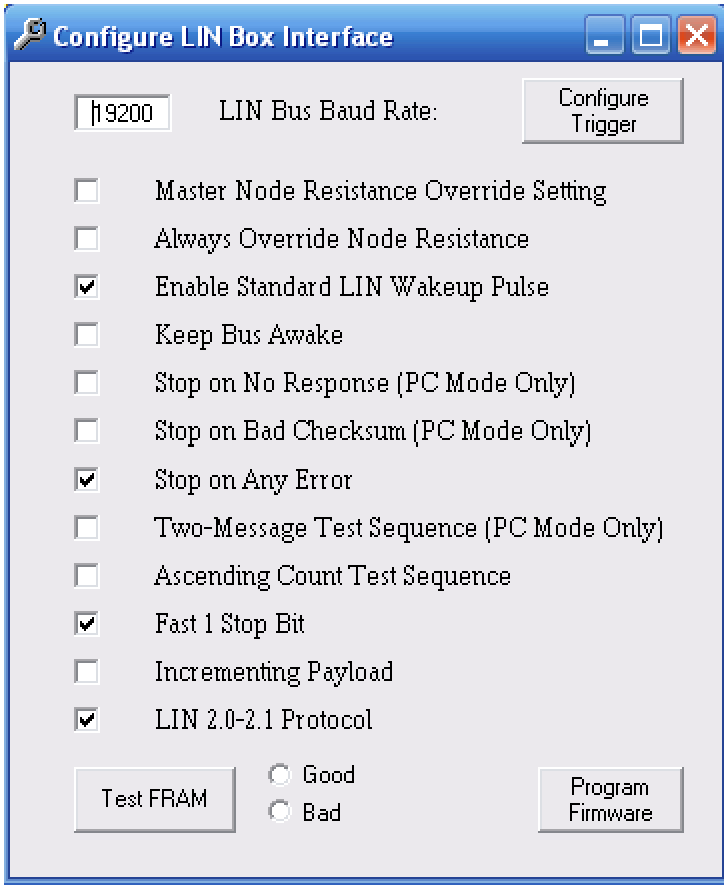
6.3. MASTER MODE, SLAVE MODE, PC CONTROL MODE

Each LIN/USB Converter defaults to PC control mode when shipped from the factory. When in PC control mode, it sends and receives messages under control of the personal computer. Thereafter, the LIN/USB Converter boots up into the last mode that it was programmed in.

The device can also be programmed to operate as the LIN bus master, or as a LIN bus slave. This programming is performed by loading a ***Configuration Language Description*** (CLD) file into the device. Also known as a ***LIN Description File*** (LDF), the format and contents are defined in the LIN protocol specifications. The CLD file covers one or more connected LIN/USB Converters. It specifies whether each node is a master or a slave.

6.4. CONFIGURATION SCREEN

This screen allows you to configure the LIN interface.



19200 LIN Bus Baud Rate:

☐ Master Node Resistance Override Setting
☐ Always Override Node Resistance
☑ Enable Standard LIN Wakeup Pulse
☐ Keep Bus Awake
☐ Stop on No Response (PC Mode Only)
☐ Stop on Bad Checksum (PC Mode Only)
☑ Stop on Any Error
☐ Two-Message Test Sequence (PC Mode Only)
☐ Ascending Count Test Sequence
☑ Fast 1 Stop Bit
☐ Incrementing Payload
☑ LIN 2.0-2.1 Protocol

○ Good
○ Bad

CONFIGURATION SCREEN
FIGURE 6.4.1

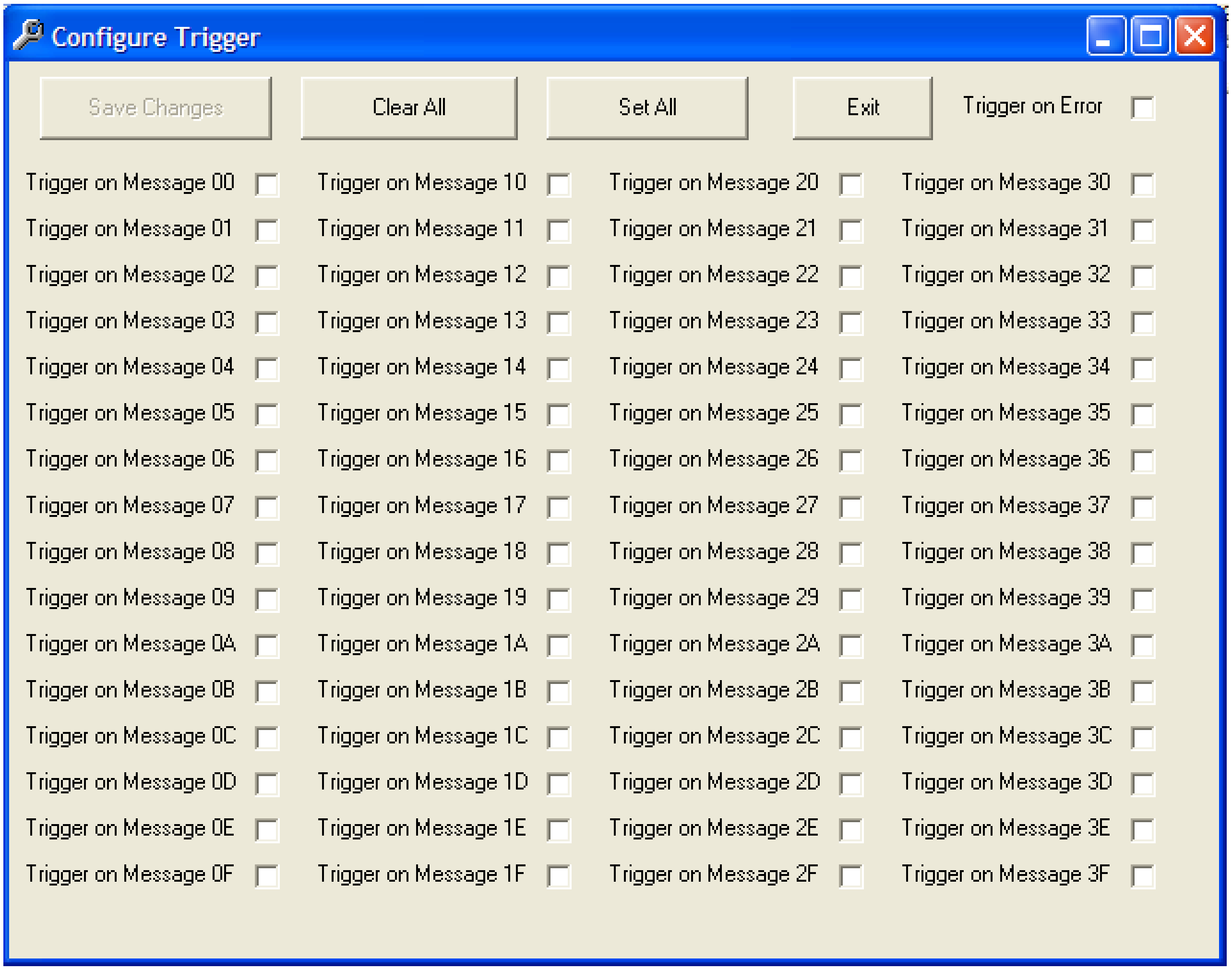
- **LIN Bus Baud Rate:** Used to program the baud rate, in bits per second, for the LIN bus. Valid entries are from 1,000 to 21,000 bps.
- **Master Mode Load Resistor Override Setting:** Used to select whether the master mode LIN bus load resistor—1 KΩ—is activated. Note that this also depends on the next option, ***Always Override Node Resistance***. If that box is unchecked, then the load resistor is added on an intelligent basis (always using the master load resistor if in Master mode emulation or if the converter is in PC-controlled mode and sending a Master Task message). If that box is checked, then the ***Master Node Resistance Override Setting*** is used. If not checked, then the slave mode resistor is switched in—30 KΩ. The resistor selection is automatic when the LIN/USB Converter is set up to be in master mode or slave node. Generally just one node on the LIN bus—acting as the LIN bus master—should use the 1 KΩ value, while all others should use the 30 KΩ value.
- **Fast 1 Stop Bit:** This setting allows the LIN/USB device to be configured in the fastest possible transmission mode (if checked). Each data byte sent over the LIN bus will have just one STOP bit per data word. If unchecked, the LIN/USB device will send two stop bits after each byte to allow extra time for slower ECUs.

  Note: It is not recommended to use Fast 1 Stop Bit mode in a system that utilizes the older Model 9004 LIN/USB Converters on the LIN system bus. That device has lower USB data processing capabilities that might result in data errors when operating in this mode.

- **Incrementing Payload:** This setting only applies when the LIN/USB device is setup as a Master Node from a CLD file. When this is the case, and this option is checked, the first two bytes of each master task message sent in the master delivery schedule will increment with each successive message. If this option is unchecked, each master task message will contain the static data from the CLD file. The incrementing payload option can be useful for testing an ECU's ability to respond to each and every message from a master.
- **Stop on Any Error:** This setting is stored in non-volatile storage on the LIN/USB device as well as the mode of operation the LIN/USB device is in. If the LIN/USB device is in Master or Slave mode, the device will automatically stop emulating the Master or Slave node if any error is detected (when the Stop on Any Error box is checked). This includes Slave Not Responding errors but does not include receive checksum mismatch errors since that is handled by decode logic on the PC. Note that continuous PC transmissions can be set to stop on checksum error since continuous PC transmissions are handled by the PC.
- **Test FRAM:** Press this button to test the internal data memory within the LIN/USB Converter. After approximately two minutes, either the Bad or Good radio button will become checked, indicating the result of the test. If the unit tests bad, contact Silicon Engines to arrange for servicing.

  Note: This button is provided for backward compatibility with the Model 9004 LIN/USB Converter. The FRAM is a serial non-volatile memory device within the Model 9004 that stores LIN message traffic. The FRAM ferromagnetic memory has a finite data life, and this button allows the user to test the memory. The FRAM in a Model 9004 will not test bad unless it has undergone extremely heavy use for many years. The Model 9011 uses serial SRAM with an unlimited data life.
- **Miscellaneous message controls:** Buttons provide a variety of message control options, including wake-up messages, control of repeated messages, error handling, and test modes.

- **Configuring Triggers:** Click on the ***Configure Trigger*** button to access the trigger setup window. The trigger setup window looks like this:



Trigger on Error ☐

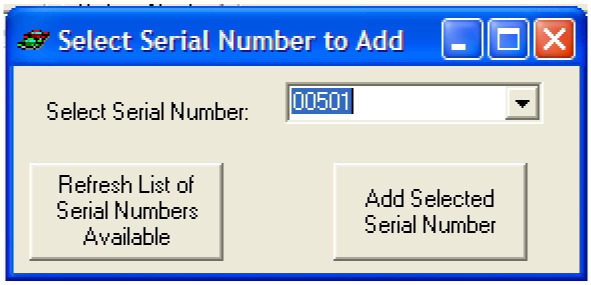
Trigger on Message 00 ☐	Trigger on Message 10 ☐	Trigger on Message 20 ☐	Trigger on Message 30 ☐
Trigger on Message 01 ☐	Trigger on Message 11 ☐	Trigger on Message 21 ☐	Trigger on Message 31 ☐
Trigger on Message 02 ☐	Trigger on Message 12 ☐	Trigger on Message 22 ☐	Trigger on Message 32 ☐
Trigger on Message 03 ☐	Trigger on Message 13 ☐	Trigger on Message 23 ☐	Trigger on Message 33 ☐
Trigger on Message 04 ☐	Trigger on Message 14 ☐	Trigger on Message 24 ☐	Trigger on Message 34 ☐
Trigger on Message 05 ☐	Trigger on Message 15 ☐	Trigger on Message 25 ☐	Trigger on Message 35 ☐
Trigger on Message 06 ☐	Trigger on Message 16 ☐	Trigger on Message 26 ☐	Trigger on Message 36 ☐
Trigger on Message 07 ☐	Trigger on Message 17 ☐	Trigger on Message 27 ☐	Trigger on Message 37 ☐
Trigger on Message 08 ☐	Trigger on Message 18 ☐	Trigger on Message 28 ☐	Trigger on Message 38 ☐
Trigger on Message 09 ☐	Trigger on Message 19 ☐	Trigger on Message 29 ☐	Trigger on Message 39 ☐
Trigger on Message 0A ☐	Trigger on Message 1A ☐	Trigger on Message 2A ☐	Trigger on Message 3A ☐
Trigger on Message 0B ☐	Trigger on Message 1B ☐	Trigger on Message 2B ☐	Trigger on Message 3B ☐
Trigger on Message 0C ☐	Trigger on Message 1C ☐	Trigger on Message 2C ☐	Trigger on Message 3C ☐
Trigger on Message 0D ☐	Trigger on Message 1D ☐	Trigger on Message 2D ☐	Trigger on Message 3D ☐
Trigger on Message 0E ☐	Trigger on Message 1E ☐	Trigger on Message 2E ☐	Trigger on Message 3E ☐
Trigger on Message 0F ☐	Trigger on Message 1F ☐	Trigger on Message 2F ☐	Trigger on Message 3F ☐

CONFIGURE TRIGGER SCREEN
FIGURE 6.4.2

Select any triggers you would like to make occur by the selected LIN/USB converter and then click on ***Save Changes*** to make the triggers permanently remembered by the LIN/ USB converter. Then, when the specified event occurs, a pulse will appear on the trigger output line that is available on the LIN connector. *(See Part 7 for the electrical specifications for the TRIGGER output signal.)*

6.5. ADDING LIN NODES

It is necessary to specify the LIN nodes connected to the LIN bus. Use the ***Add a Node*** button on the main message screen to bring up the following screen:


Select Serial Number to Add
Refresh List of Serial Numbers Available
Add Selected Serial Number

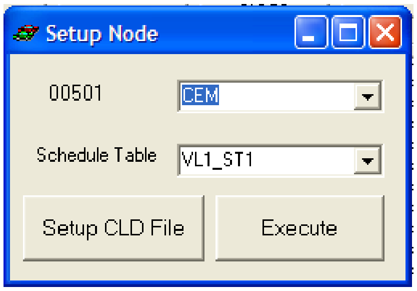
Select Serial Number: 00501

ADDING LIN NODES
FIGURE 6.5.1

To add a new serial number, click on the ***Add Selected Serial Number*** button. If another device has been connected since the screen was invoked, click on ***Refresh List of Serial Numbers Available*** to see if the new device is now present in the list. To see all the nodes that are identified as connected (or already appear in the node list) click on the drop-down box arrow to the right of the selection box and click on the node that you want to add.

6.6. LOADING CLD FILES

To load the CLD file features into a LIN/USB Converter, return to the Main Message Center screen, and click on the ***Setup Node*** button. The following screen appears:

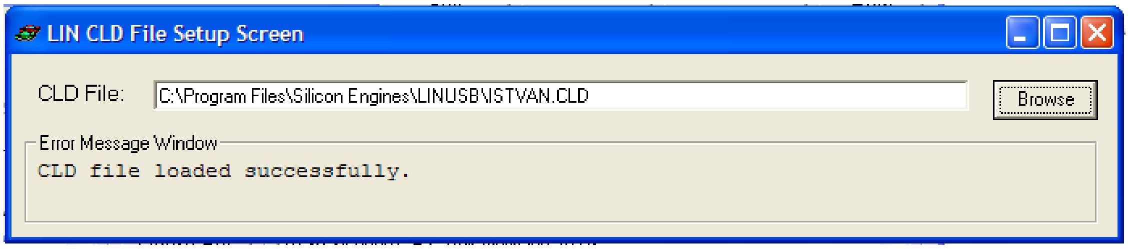


00501 CEM

Schedule Table VL1_ST1

SETUP NODE SCREEN
FIGURE 6.6.1

Next press the ***Setup CLD File*** button to load a CLD file. The following screen appears:



CLD File: C:\Program Files\Silicon Engines\LINUSB\ISTVAN.CLD

Error Message Window

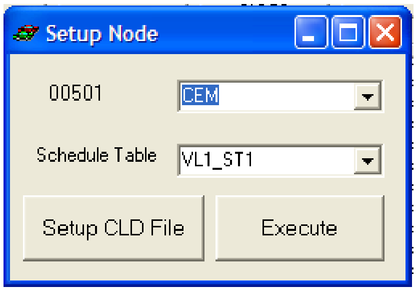
```
CLD file loaded successfully.
```

CLD FILE SETUP SCREEN
FIGURE 6.6.2

If you do not have a CLD or LDF file already, then it is recommended that you load the file provided with the LIN USB Message Center (ISTVAN.CLD). That file is located in the \Program Files\Silicon Engines\LINUSB directory. Browse to that file and then confirm that it says that it has loaded successfully. The LIN USB Message Center should be able to recognize any CLD or LDF file that is compatible with the 1.3, 2.0, or 2.1 LIN specifications.

6.7. EXECUTING MASTER NODE SCHEDULES

Returning to the ***Setup Node*** screen:


Setup Node

00501: CEM

Schedule Table: VL1_ST1

SETUP NODE SCREEN
FIGURE 6.7.1

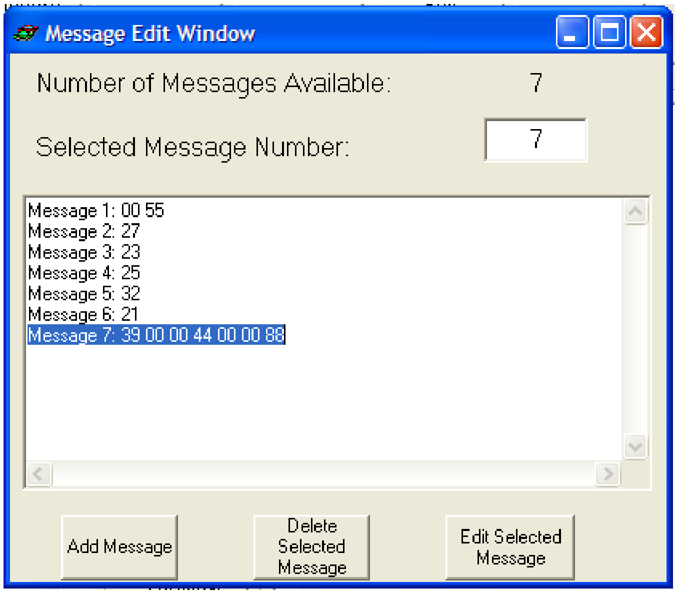
When the top window shows the name of a master node, the ***Schedule Table*** window becomes active. It provides a list of the CLD schedules currently available. Click on the arrow at the right of this window to select the specific CLD schedule that you wish to activate.

Once you have selected the appropriate schedule, click on the ***Execute*** button. The master mode schedule will be executed.

If you would like this LIN/USB module to emulate a slave node, then choose a slave node name from the top drop-down box. You don't need to select anything in the second drop-down box for slave nodes.

6.8. EDITING LIN MESSAGES

To view and edit messages within the LIN Message Center, press the ***Edit Messages*** button. The following screen will pop up:



Message Edit Window

Number of Messages Available: 7

Selected Message Number: 7

Message 1: 00 55
Message 2: 27
Message 3: 23
Message 4: 25
Message 5: 32
Message 6: 21
Message 7: 39 00 00 44 00 00 88

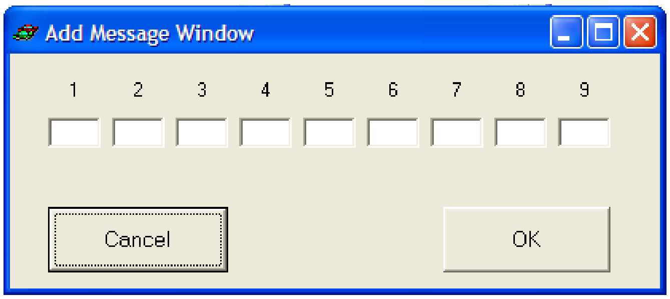
MESSAGE EDIT WINDOW
FIGURE 6.8.1.

To select a message, click on the message you would like to send next or to remove. That message will become highlighted (surrounded by gray). Now you can send the message by clicking on ***Send Selected Message*** on the main screen.

To remove a message, click on the message you want to remove, and press ***Delete Selected Message***.

6.9. ADDING A NEW LIN MESSAGE

To add a message, click on ***Add Message*** and the following screen will pop up:



Add Message Window

ADD MESSAGE WINDOW
FIGURE 6.9.1.

In this screen you can enter a message that the LIN Message Center will be capable of sending later. Data entry is in hex.

If you enter just one hex number in box 1, then this will be a master mode message expecting a response that will be supplied by a slave device on the LIN bus.

If you fill more than one box, then this will be a full master/slave task message.

Note: the LIN checksum is computed automatically and should not be included in the data entered here.

Note: the first byte must have bits 6 and 7 cleared. In other words, the first byte must be less than or equal to 3F hex. The two high-order bits (parity bits) will be filled in automatically.

When you are done entering the message, press the ***OK*** button and the message will appear in the ***Message Edit Window*** on the main message screen.

6.10. SENDING A LIN MESSAGE

To send the most recently selected message in the ***Message Edit Window***, press ***Send Selected Message*** on the main screen.

6.11. SENDING A LIN MESSAGE REPEATEDLY

You can send a message repeatedly by clicking on ***Send Selected Message Repeatedly***. This will cause the selected message to be sent to the selected nodes over and over.

When you click on ***Two-Message Test Sequence***, two messages will be repeated over and over, subject to the stop-on-error rules specified on the Configuration Screen.

The Configuration Screen can be set up so that repeating messages will stop if a bad checksum is encountered; if no response is received; or when any error is encountered. These options are useful for testing the LIN capabilities of an ECU on the LIN bus.

7. TRIGGER OUTPUT SIGNAL

7.1. TRIG OUTPUT, PIN 5 ON LIN CONNECTOR

The LIN/USB Converter can be configured to output a pulse on its TRIGGER output when a specified LIN message event occurs, or in the event of a specified error. This signal appears on pin 5 of the LIN connector.

7.2. TRIGGER SIGNAL DESCRIPTION

The idle state of the TRIGGER signal is 0 volts.

The TRIGGER signal goes high to a level of approximately 4.6 to 5.0 volts when a programmed event occurs. It remains high for approximately 1.5 milliseconds, then returns to the 0-volt idle condition.

A connected oscilloscope or logic analyzer should be set up to trigger on the rising edge of the TRIGGER signal.

7.3. TRIGGER SIGNAL LOADING

The TRIGGER signal has a 10 KΩ resistor to ground within the LIN/USB Converter. Pulses are generated by a current-limited high-side transistor powered by an internal +5V supply within the unit. Maximum recommended load current is 1 milliamp. The high-side driver will go into current limit at approximately 2 milliamps.

8. BACKGROUND ON THE LIN BUS

8.1. LIN BUS CONSORTIUM

A consortium including Audi, BMW, DaimlerChrysler, Volkswagen, and Volvo (car manufacturers), as well as Freescale Semiconductor (formerly Motorola Semiconductor) and Volcano Communications Technologies (automotive electronics), developed the LIN Protocol Specification.

For complete specifications, point your browser to http://www.lin-subbus.org.

8.2. RELATION TO CAN, J1850

LIN, ***Local Interconnect Network***, is a low-cost automotive multiplexing concept. Automotive electronic control modules (ECUs) that implement LIN communicate with each other over a one-wire data bus. In this respect, LIN is similar to other automotive multiplexing protocols—notably CAN and J1850. However LIN has been designed to realize lower ECU costs.

CAN (Controller Area Network) is typically implemented as a two-wire bus, running at 500 kbps (SAE J2284). A single-wire, 33.3 kbps version (SAE J2411) is in use by GM. CAN embodies a collision detection and arbitration mechanism that allows each node to start sending data whenever the bus is idle, and to resolve collisions without loss of data. CAN is becoming increasingly dominant for medium- and high-speed automotive multiplexing. However CAN adds several dollars to the cost of an automotive ECU—including the cost of the CAN serial communications controller within the microcontroller; a CAN physical layer transceiver; and extra ROM and RAM to handle the higher layers of the protocol.

The SAE J1850 protocol is conceptually similar to CAN, in that it implements collision detection and arbitration. However data encoding, bus speeds, and physical layer details differ from CAN. J1850-VPW (General Motors, Chrysler) involves a single-wire bus operating at an average data rate of 10,400 bps, using variable pulse-width data coding. J1850-PWM (Ford) implements a two-wire bus operating at 41,600 bps, using pulse-width modulation. The costs of implementing J1850 are similar to the costs for CAN, and the American car manufacturers are tending to switch to CAN for new vehicle designs.

There has been a growing perception of the need for a lower-cost network for many of the ECUs in a vehicle—for example, body control modules controlling doors, memory seats, and similar functions.

8.3. OUTLINE SPECIFICATIONS

LIN is a single-wire bus that can readily be implemented by very low-cost microcontrollers. The hardware required for a slave-mode LIN ECU is a standard SCI (serial communications interface) or UART (universal asynchronous receiver/transmitter). ECUs with low application processing requirements can even handle LIN without an SCI or UART. Data speed is 1,000 to 20,000 bps. The physical layer transceiver is very simple. There is a master node in a LIN network, keeping the protocol simple. Relatively little software overhead is required. In a typical vehicle architecture, the body control ECUs interconnect using LIN (rather than CAN or J1850). A master LIN module serves as the LIN bus master, and provides a bridge to the CAN network for interchange of information with higher-speed ECUs.

With respect to the physical layer, LIN is similar to ISO-9141, an automotive diagnostic standard published by the International Standards Organization in 1989. LIN is also similar to ISO-14230, an updated and extended diagnostic standard published in 1999, implementing ***Keyword Protocol 2000.*** However these diagnostic protocols are active only when a tester device is connected to the ECU, at the factory or in a service depot. LIN is a protocol for exchanging information between modules when the vehicle is moving down the road.

8.4. BUS TOPOLOGY

The LIN bus line is a bi-directional, half-duplex, serial input/output line for exchange of information between automotive ECUs (electronic control units).

Up to 16 modules may be connected on the LIN bus line. Recommended maximum wire length is 40 meters (131 feet) [*LIN Physical Layer Specification*, Rev. 2.1, Sec. 6.5.5, Table 6.11].

The LIN bus terminal on each ECU is connected to like terminals on other ECUs within the vehicle.

One LIN module acts as the master node. The other ECUs act as LIN slaves. Each LIN slave communicates over the LIN bus when it receives its unique ID code from the LIN master. There is no collision arbitration (unlike CAN or J1850).

8.5. PHYSICAL LAYER

Within the LIN bus master, the LIN bus is terminated to VBATT through a pull-up resistor with a value of 1 KΩ. Within each LIN slave, the LIN bus is terminated to VBATT by a pull-up resistor with a value of 30 KΩ. A diode is placed in series with each pull-up resistor, within each ECU, to prevent the LIN bus from providing a sneak path to power the ECU if it has lost its connection to VBATT [*LIN Physical Layer Specification*, Rev. 2.1, Sec. 6.5.1].

8.6. SERIAL DATA FORMAT

For the most part, the LIN bus uses asynchronous serial data format compatible with the SCI (serial communications interface) or UART (universal asynchronous receiver/transmitter) peripherals within popular microcontrollers. The data format is 8N1. Bits are sent in 10-bit words consisting of a START bit, eight data bits (least significant bit first), no parity bit, and one STOP bit [*LIN Physical Layer Specification,* Rev. 2.1, Sec. 6.4.3].

The data format is similar to RS-232, except that the serial data signals on the LIN bus are non-inverted, unipolar (VBATT and GROUND), while the serial data signals on the RS-232 TXD and RXD lines are inverted, bipolar (± 10 to 15 V).

BIT DESCRIPTION	LOGIC LEVEL	LIN VOLTAGE	RS-232 VOLTAGE
IDLE LINE	1	VBATT	-10 V
START BIT	0	GROUND	+10 V
DATA BITS	0	GROUND	+10 V
	1	VBATT	-10 V
STOP BIT	1	VBATT	-10 V

SERIAL DATA LEVELS FOR LIN, RS-232
FIGURE 8.6.1

8.7. LIN CIRCUITRY WITHIN EACH ECU

A typical LIN ECU includes a LIN transceiver integrated circuit. The transmitter output stage of the LIN transceiver is an open-drain MOSFET (or open-collector bipolar transistor) that is normally off. The pull-up resistors to VBATT within the various LIN ECUs cause the LIN bus line to rise to the level of VBATT when the LIN bus is idle. An ECU selectively activates this transmitter transistor when transmitting on the LIN bus line.

The receiver input stage within the LIN transceiver consists of a voltage comparator that slices at approximately 50% of VBATT to distinguish between high and low voltage levels on the LIN bus.

A number of semiconductor manufacturers offer LIN transceivers compatible with the above specifications. A list of LIN transceivers appears in the *REFERENCES* section of this document *(Part 11).*

The LIN transmitter and receiver typically connect to an SCI (serial communications interface) or UART (universal asynchronous receiver-transmitter) within the ECU's microcontroller.

8.8. SERIAL DATA SPEED

Within a given vehicle, the LIN bus operates at a data rate from 1,000 to 21,000 bits per second (bps) [*LIN Physical Layer Specification*, Rev. 2.1, Sec. 6.3.] Three recommended LIN data rates were prescribed by an earlier version of the LIN specifications [*LIN Protocol Specification*, Rev. 1.3, *Bit rate*, p. 9].

SPEED	DATA RATE
SLOW	2400 BPS
MEDIUM	9600 BPS
FAST	19,200 BPS

LIN 1.3 RECOMMENDED STANDARD LIN BUS SPEEDS
FIGURE 8.8.1

The LIN/USB Converter can be programmed for any data rate from 1,000 to 21,000 bps. It should match the data rate selected for the target vehicle.

8.9. LIN BUS LINE CAPACITIVE LOADING

Bus line capacitance can be an important issue in LIN networks. The number of nodes, the cable capacitance, and the capacitive bus loads from the master and slave nodes all must be considered [*LIN Physical Layer Specification,* Revision 2.1, Sec. 6.5.5, Table 6.11].

PARAMETER	MINIMUM	TYPICAL	MAXIMUM	CONDITIONS/COMMENTS
TOTAL CAPACITANCE OF BUS	1	4	10	NANOFARADS
CAPACITANCE OF MASTER NODE		220		PICOFARADS
CAPACITANCE OF SLAVE NODES		220	250	PICOFARADS
LINE CAPACITANCE		100	150	PICOFARADS PER METER
LENGTH OF BUS LINE			40	METERS
NUMBER OF NODES			16	UNITS

LIN BUS LINE CAPACITIVE LOADING
FIGURE 8.9.1

8.10. HEADER FIELD

The LIN protocol requires a header field at the start of each message frame. The master LIN node always generates the header field. It consists of the following phases:

1. **Synchronization break:** A low-voltage break signal on the LIN bus lasting 13 nominal bit times at the effective data speed. This break symbol departs from the conventional asynchronous SCI/UART frame format.

2. **Synchronization break delimiter:** An idle (high-voltage) period on the LIN bus lasting one nominal bit time on the LIN bus.

3. **Synch field:** A single byte consisting of the hex value 55H, sent in standard 8N1 asynchronous data format, with one start bit, eight data bits, and a stop bit. This byte is sent at the LIN baud rate appropriate for the target ECUs. It is intended for use by slave ECUs that may not have accurate crystal-based clocks, allowing them to synchronize their responses with the LIN data rate established by the LIN bus master.

4. **Identifier field:** A single byte in 8N1 format that defines the content and length of the LIN message.

BIT NO.	BIT ID.	FUNCTION	COMMENTS
0	ID0	MESSAGE IDENTIFIER	LSB, SENT FIRST
1	ID1		
2	ID2		
3	ID3		
4	ID4		
5	ID5		
6	P0	PARITY BITS	
7	P1		MSB, SENT LAST

IDENTIFIER FIELD IN LIN MESSAGE HEADER
FIGURE 8.10.1

The least significant six bits, ID0 through ID5, specify a particular LIN message.

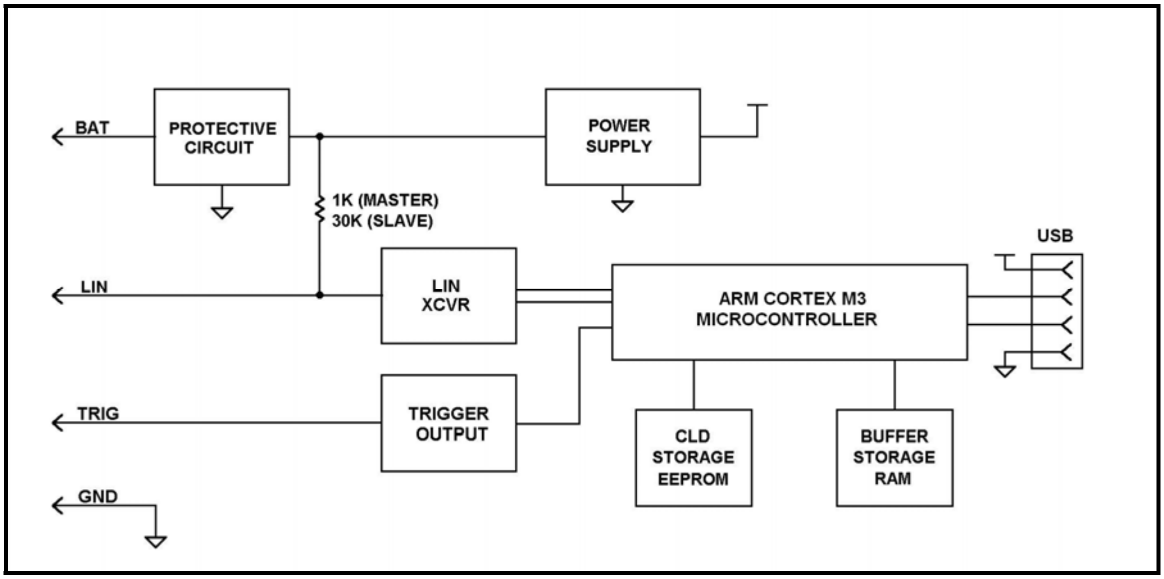
Bits P0 and P1 are parity check bits on the first six bits, ID0 through ID5. Each parity bit operates on four of the six bits ID0-ID5.

PARITY BIT	FORMULA	COMMENTS
P0	$\mathrm{ID1} \otimes \mathrm{ID1} \otimes \mathrm{ID2} \otimes \mathrm{ID4}$	EVEN PARITY
P1	$\overline{\mathrm{ID1} \otimes \mathrm{ID3} \otimes \mathrm{ID4} \otimes \mathrm{ID5}}$	ODD PARITY

PARITY CHECK BITS IN IDENTIFIER FIELD
FIGURE 8.10.2

9. MODEL 9011 HARDWARE

9.1. BLOCK DIAGRAM



MODEL 9011 BLOCK DIAGRAM
FIGURE 9.1.1.

9.2. HARDWARE DESCRIPTION

- An STMicroelectronics STM32F103 microcontroller, a member of the 32-bit ARM Cortex M3 family running at 72 MHz, handles all the processing within the Model 9011.
- A 32K by 8 high-speed serial RAM, interfaced to the ARM Cortex M3, buffers incoming LIN traffic in order to prevent data loss when USB activity is temporarily suspended due to PC activity.
- 8K by 8 serial EEPROM to provide non-volatile storage of CLD parameters for the LIN interface.
- LIN bus transceiver with slew control per LIN specifications.
- LIN data speed programmable at any integer baud rate from 1000 bps to 21,000 bps.
- USB protective diode array to protect against static damage to the MCU.
- In-system programmable firmware. The microcontroller is initially programmed at the factory using a JTAG connector. Thereafter the ARM Cortex M3 can be reprogrammed over the USB port using special programming software (such as the Message Center).
- Power supply providing needed internal voltages from VBATT, including automotive-grade features such as reverse battery protection and load dump protection.

10. EMBEDDED SOFTWARE OUTLINE

10.1. LIN FEATURES

- **LIN Engine:** Controlled by a 72 MHz microcontroller.
- **Master and slave:** Handles both LIN master mode and LIN slave mode transactions.
- **Baud rate:** LIN data speed programmable to any integer baud rate from 1,000 to 21,000 bps.

- **Synchronization:** Generates the master-mode synchronization symbol.
- **Checksums:** Automatically handles LIN checksums on the transmit side.
- **Parity ID Fields:** Automatically handles parity ID field (two upper bits of command ID) on both receive and transmit sides. Generates an error code if the parity ID field is incorrect.
- **Frame length:** Automatically detects LIN messages of any length.
- **CLD file interpreter:** Runs as a CLD node when loaded by the provided Windows® software.
- **Trigger output:** Can be set to trigger on any specific command identifier or on an error.
- **EEPROM handler:** Stores configuration data—including CLD files—in nonvolatile EEPROM memory.
- **Frame errors:** Detects errors in the LIN frame—between 9 and 11 data bits, an illegal value.
- **Sync errors:** Detects improper synchronization fields.
- **Parity errors:** Detects bad parity in the command ID frame.
- **Checksum errors:** Detects bad LIN checksums. This is done by the PC software external to the LIN/USB module.
- **No response errors:** Detects failure of a slave to respond.
- **Transmission errors:** Detects shorted LIN bus conditions—bus is high when trying to transmit a low level, or bus is low when trying to transmit a high level.
- **Sleep errors:** Reports bus traffic attempts when the LIN bus is asleep.

10.2. USB FEATURES

- **USB full-speed:** Provides a USB full-speed interface, 12 megabits/second raw data rate.
- **USB handler:** Controlled by an MCU with built-in USB interface.
- **Data buffering:** Buffers incoming LIN data in high-speed serial RAM to avoid data loss when the PC USB port is temporarily unavailable.

10.3. FIRMWARE UPGRADES

The firmware inside both microcontrollers is fully reprogrammable using the LIN/USB Message Center and the firmware files from the CD. Please view the README file installed with the program for details on the software upgrades available. The latest README file and software updates are also available on the Silicon Engines website.

11. REFERENCES

11.1. LATEST VERSIONS

The documents shown below were the latest revisions known when this document was last revised. They are generally available from the issuing organization's website. Please be sure to check with the issuing organization for updates and revisions.

11.2. LIN SPECIFICATIONS

1. *LIN Specification Package*, Revision 2.1, pages 1-27, November 24, 2006. Introduction to the LIN specification package. LIN Consortium.

2. *LIN Protocol Specification,* Revision 2.1, November 24, 2006, pages 23-53. Specifications for LIN data link layer. LIN Consortium.

3. *LIN Transport Layer Specifications,* Revision 2.1, November 24, 2006, pages 24-62. Specifications for ISO 15675-2-compatible diagnostic messages. LIN Consortium.

4. *LIN Node Configuration and Identification Specification,* Revision 2.1, November 24, 2006, pages 63-76. Specifies the manner in which a slave node is configured and identified. LIN Consortium.

5. *LIN Diagnostic Specification,* Revision 2.1, November 24, 2006, pages 77-105. Specifies communications between a diagnostic tester and slave nodes. LIN Consortium.

6. *LIN Physical Layer Specification,* Revision 2.1, November 24, 2006, pages 106-123. Specifications for LIN bit timing and voltage levels. LIN Consortium.

7. *LIN Application Program Interface Specification,* Revision 2.1, November 24, 2006, pages 124-160. Specifies a recommended application programmer's interface between the LIN software and ECU application software. LIN Consortium.

8. *LIN Node Capability Language Specification,* Revision 2.1, November 24, 2006, pages 161-171. Specifies a means to characterize an off-the-shelf LIN slave node in machine-readable syntax as a step towards plug-and-play operation.

9. *LIN Configuration Language Specification,* Revision 2.1, November 24, 2006, pages 172-191. Specifies a means to characterize a LIN network, facilitating simulation in the absence of one or more nodes.

11.3. LIN APPLICATION NOTES

1. *Software LIN Slave,* Application Note AVR308, May 2002, 12 pages. Atmel Corporation.

2. *LIN Bus and its Potential for Use in Distributed Multiplex Applications,* SAE Technical Paper 2001-02-0072, 2001, 10 pages. Delphi Automotive Systems; Society of Automotive Engineers.

3. *LIN Protocol Specification Implementation with PICmicro® MCUs,* Application Note 729, DS00729A, 2000, 35 pages. Microchip Technology Inc.

4. *Local Interconnect (LIN) Demonstration,* Application Note 2103, 2000, 68 pages. Motorola Semiconductor Products.

5. *Philips Microcontrollers in LIN Applications,* Application Note 10115, Feb. 15, 2002, 6 pages. Philips Semiconductors.

6. *LIN (Local Interconnect Network) Solutions,* Application Note 1278, Rev. 1.1, April 2002, 44 pages. STMicroelectronics.

11.4. LIN TRANSCEIVERS

1. Freescale Semiconductor, MC33399.
2. Infineon Technologies, TLE6258.
3. Melexis Microelectronic Integrated Systems, TH8080.
4. ON Semiconductor, NCV7310.
5. Philips Semiconductors, TJA1020.
6. STMicroelectronics, L9638.

11.5. WEBSITES

1. Atmel Corporation, www.atmel.com
2. Delphi Automotive Systems, www.delphi.com
3. Freescale Semiconductor, www.freescale.com
4. Infineon Technologies, www.infineon.com
5. LIN Consortium, www.lin-subbus.de
6. Melexis Microelectronic Integrated Systems, www.melexis.com
7. Microchip Technology Inc., www.microchip.com
8. ON Semiconductor, www.onsemi.com
9. Philips Semiconductors, www.semiconductors.philips.com
10. Society of Automotive Engineers, www.sae.org
11. Silicon Engines, www.siliconengines-ltd.com
12. STMicroelectronics, www.st.com

12. MODEL 9011 VERSUS MODEL 9004

12.1. MODEL 9011 REPLACES MODEL 9004

Silicon Engines has introduced the Model 9011 LIN/USB Converter as an upgrade to our earlier Model 9004 LIN/USB Converter.

The Model 9004 is being phased out and will not be offered for sale once supplies have run out.

The Model 9011 implements all of the key features of the Model 9004.

12.2. ENCLOSURE AND CONNECTORS

The Model 9011 comes in a larger enclosure than the Model 9004. The Model 9011 enclosure offers additional space for the internal circuitry and facilitates use of the higher-speed ARM Cortex M3 microcontroller.

In addition, the Model 9011 employs a commonly-available DB9 connector for its LIN interface, replacing a 4-pin Molex connector used on the Model 9004.

12.3. COMPARISON CHART

SPECIFICATION	MODEL 9011	MODEL 9004	MODEL 9011 IMPROVEMENTS
ENCLOSURE	4.375 X 3.25 X 1.5 IN.	4.0 X 1.65 X 0.95 IN.	SPACE FOR ENHANCED CIRCUITRY
LIN CONNECTOR	DB9	4-PIN MOLEX	MORE STANDARD CONNECTOR
PROCESSOR	72 MHZ ARM CORTEX M3	TWO 8-BIT PROCESSORS	ADDITIONAL PROCESSING POWER
USB SPEED	FULL-SPEED, 12 MEGABITS PER SECOND	LOW-SPEED, 1.5 MEGABITS PER SECOND	FASTER USB PERFORMANCE
WINDOWS SUPPORT	WINDOWS 2000, XP, VISTA, 7	WINDOWS 2000, XP ONLY	COMPATIBLE WITH NEARLY ALL EXISTING WINDOWS COMPUTERS

MODEL 9011 VERSUS MODEL 9004

FIGURE 12.3.1.

13. REVISION HISTORY

13.1. REVISION A

Initial release.

☑