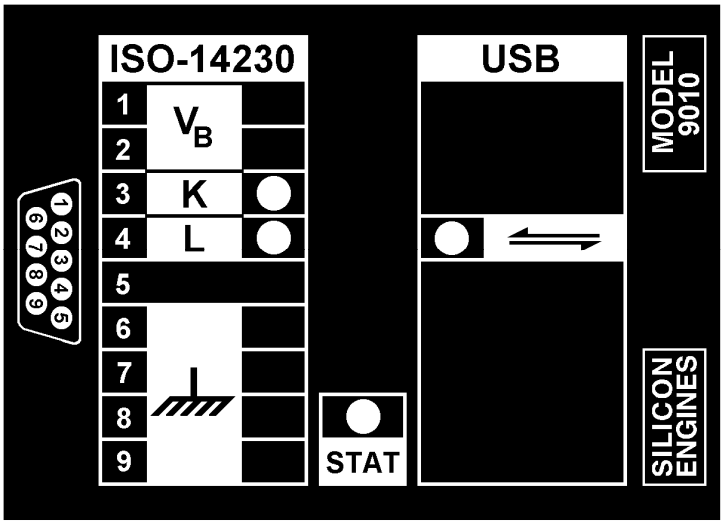
SILICON ENGINES

14230/USB CONVERTER

MODEL 9010

USER'S GUIDE



MODEL 9010 REPLACES MODEL 9009

For a comparison of features, please see Part 22.

WINDOWS® COMPATIBILITY

The Model 9010 is compatible with Windows 2000, Windows XP, Windows Vista, and Windows 7

1. TITLE PAGE

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

2. CE CONFORMANCE

DECLARATION OF CONFORMITY
MODEL 9010 14230/USB CONVERTER

Conformance to Directives and Standards:

- Emissions, 2004/108/EC EMC Directive, EN55022:2006, A1:2007
- Emissions, 2004/104/EC Auto Directive, Narrow-Band, Broad-Band
- Information Technology Equipment Immunity, EN55024:1998, A1:2001, A2:2003, using IEC 61000-4-2, 3, 4, 6, 8

Equipment Type/Environment: Automotive diagnostic equipment, light industrial environment.
Beginning Serial No.: Conformity certificate applies to Serial No. 1 and above.
Year of Manufacture: Products manufactured on or after September 1, 2011.

CE

I, the undersigned, hereby declare that the equipment specified above conforms to the above Directives.

Manufacturer:

Full name: Kerry Berland
Position: President
Place: Arlington Heights, Illinois, USA
Date: August 1, 2011

3. CONTENTS

4. OVERVIEW

4.1. INTRODUCTION

This document is the User's Guide for the **14230/USB Converter**, **Model 9010**, a compact electronic device that allows a personal computer with a USB interface to connect to an automotive diagnostic data link compatible with ISO-14230.

USB (Universal Serial Bus) is a versatile serial data interface that is found on all newer personal computers, tending to replace the traditional RS-232 and parallel port interfaces supported on earlier PCs.

ISO-14230 is a worldwide standard issued by the International Standards Organization. ISO-14230 defines the physical layer interface, as well as a standard protocol for sending and receiving diagnostic information, known as **Keyword Protocol 2000.**

The ISO-14230 standard is an updated and expanded version of an earlier automotive diagnostic standard, ISO-9141.

4.2. REPLACEMENT FOR MODEL 9009

The Model 9010 14230/USB Converter is intended to replace an earlier version of the 14230/USB Converter, the Model 9009. *Please see Part 22 for a comparison of features.*

4.3. RELATED PRODUCTS

In addition to the Model 9010 14230/USB Converter, Silicon Engines also offers:

- **Model 9001, 9141 Converter.** Converts from RS-232 to ISO-9141.
- **Model 9002, 14230/RS-232 Converter.** Converts from RS-232 to ISO-14230.
- **Model 9003, LIN/RS-232 Converter.** Converts from RS-232 to LIN, another automotive serial data link standard.
- **Model 9011, LIN/USB Converter.** Converts from USB to LIN.

For details on these products, please contact Silicon Engines.

4.4. APPLICATIONS

- **Development:** Facilitates development of an automotive ECU (electronic control unit) that supports an ISO-14230 diagnostic line, by allowing a personal computer to act as the diagnostic analyzer during software development.
- **Production:** Allows the ISO-14230 data link to serve as a port for testing the ECU, and for downloading programs, parameters, serial number, calibration data, etc.
- **Service:** Allows a personal computer to act as a diagnostic analyzer.

4.5. FUNCTIONS

- **Interface conversion:** Converts data between the ISO-14230 automotive interface, and the USB interface, for connection of an automotive ECU to a personal computer.
- **Speed conversion:** Interfaces to the PC as a full-speed USB device, communicating at a raw data rate of 12 megabits per second. Interfaces to the ECU at the ISO-14230 data rate. The ISO-14230 data rate is typically 10,400 bits per second, but any integer data rate between 1,000 bps and 115,200 bps can be selected for ISO-14230-compatible communications with the ECU.

- **Initialization:** The Model 9010 14230/USB Converter handles both 5-baud initialization, and fast initialization. Unburdens the PC from having to handle initialization timing signals.
- **Duplexing:** Converts the half-duplex ISO-14230 line, to full-duplex USB messages. Avoids the need for the PC to deal with echoed characters when it transmits packets to the ECU.
- **Reliable keep awake functionality:** The Model 9010 14230/USB Converter can be set up to transmit a TESTER PRESENT message every two seconds without continuous interaction with the PC. This offloads the PC software, and ensures that the ECU will not go to sleep even if the PC experiences severe latencies or slowdowns.
- **Flexible bus voltages:** Compatible with both 12-volt and 24-volt automotive battery systems. Also compatible with industrial systems using a 5V single-wire communications bus.
- **Listener mode:** For engineering and testing, can be put into Listener Mode to monitor bus traffic.
- **Signal indicators:** Provides three two-color LEDs to show the states of significant signal lines.

4.6. SOFTWARE SUPPORT

- **ISO-14230 USB Message Center:** Silicon Engines provides a PC software utility, the ***ISO-14230 USB Message Center,*** for use with the Model 9010. Compatible with Windows® 7, Vista®, XP, and 2000, 32-bit and 64-bit versions, this utility allows you to set-up the 14230/USB Converter, send and receive messages, and monitor ISO-14230 bus traffic. There is no extra charge for this utility.
- **Message Center source code:** We provide users with the source code for the ISO-14230 USB Message Center, so that you can modify it for custom applications, for example, end-of-line ECU tester. The source code is provided in two versions, Visual Basic 6 and Visual Basic.NET 2005.
- **Driver code:** We also provide driver files that should be installed on your PC in order to access the Model 9010 over USB.

5. ISO STANDARDS

5.1. AUTOMOTIVE PROTOCOLS

The ISO-14230 Converter is intended for use with electronic control units (ECUs) which implement [ISO-14230], published by the International Standards Organization in 1999. This standard is based on the earlier standard [ISO-9141], published by the ISO in 1989.

References to standards and specifications appear in this document in square brackets, such as: [ISO-14230-1]. Please see Part 23 for a list of references.

5.2. ISO-9141

The 1989 standard [ISO 9141] defined the physical layer of the interface between the ECU and the tester, as well as a 5-baud initialization sequence. Automotive manufacturers were free to specify the high-speed data rate, after initialization. Commonly seen high-speed data rates were 480, 600, and 10,400 bits per second.

[ISO-9141] left details on the higher layers of the interface—the structure of the data interchange between the diagnostic tester and the ECU—up to each automotive manufacturer.

Subject to legislative initiatives by the California Air Resources Board (CARB), every automotive vehicle sold in the USA starting in model year 1996 was required to provide an on-board diagnostic (OBD) connector, allowing a diagnostic tester to monitor key engine control emission parameters. Vehicle manufacturers could choose to implement OBD using [ISO-9141-2], a practice followed by Chrysler, as well as most European and Japanese vehicles sold in the USA. Alternately OBD could be implemented using [J1850], a more complex automotive protocol that supports not only diagnostics, but also in-vehicle communications. GM and Ford implemented OBD using two distinct versions of [J1850].

5.3. ISO-14230

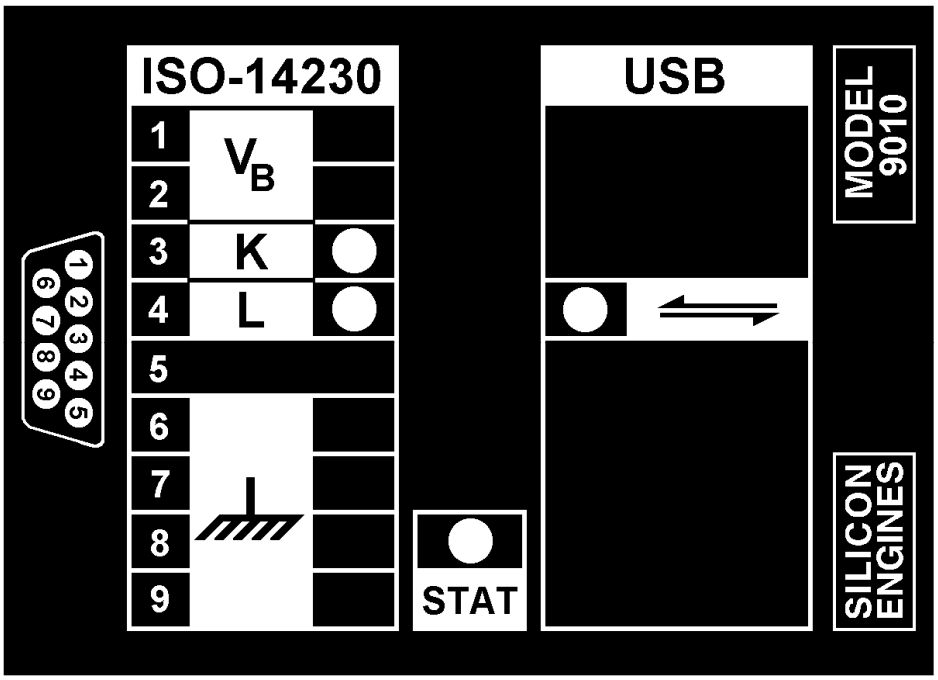
The 1999 standard [ISO-14230], **Keyword Protocol 2000**, is based on [ISO-9141], but adds many additional details:

- **Keywords:** Specifies details of an initialization protocol whereby the diagnostic tester and the ECU exchange information on how they will communicate, using coded keywords.
- **Fast initialization:** Provides a fast initialization option, in addition to the 5 baud initialization option provided in [ISO-9141].
- **L-line:** Standardizes the use of the L-line for optional use during ECU initialization.
- **High-speed protocol**: Defines many details of the high-speed data exchange phase after initialization has taken place.

6. ENCLOSURE

6.1. TOP PANEL

The Model 9010 14230/USB Converter is housed in a black plastic enclosure. Three bicolor LED indicator lamps, and one green LED, appear through clear plastic windows on the top panel. The legends on the top panel identify the functions of these lamps, and identify the signals on the ISO-14230 connector.



MODEL 9010 TOP PANEL
FIGURE 6.1.1.

6.2. 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 6.2.1.

7. CONNECTORS

7.1. ISO-14230 CONNECTOR ON CONVERTER

The connector at the left of the 14230 Converter is a type DB9M plug (9-pin male D sub-miniature).

A DB9F (female DB9) socket plugs in here. Four signals are supported: VBATT, the ISO-14230 K-line, the ISO-14230 L-line, and ground.

PIN NO.	SYMBOL	SIGNAL	DESCRIPTION
1-2	V_B	VBATT	BATTERY POWER
3	**K**	K	ISO-14230 K-LINE
4	**L**	L	ISO-14230 L-LINE
5		NC	*NO CONNECTION*
6-9	⏚	GROUND	POWER AND SIGNAL RETURN

ISO-14230 CONNECTOR PIN-OUTS
FIGURE 7.1.1.

These pin-outs, as well as the locations of the pins within the 9-pin connector, are shown on the 14230 Converter top panel legend *(Fig. 6.1.1).*

7.2. ISO-14230 CABLE TO SPECIFIC ECU

Users who intend to apply the 14230 Converter in engineering development work, or production testing—with a specific ECU—must construct a special cable to connect from the ECU to the 14230 Converter. Or contact Silicon Engines for assistance.

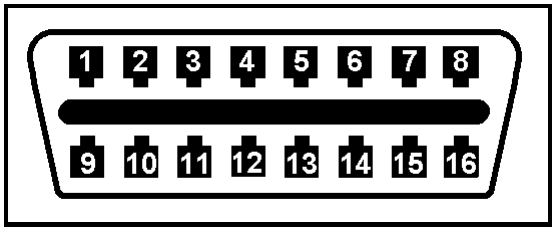
At the ISO-14230 side, use a DB9F connector. Connect VBATT to pin 1, the K-line to pin 3, the L-line to pin 4 (if it is used in your application), and ground to pin 6. The locations of the pins within the connector are shown on the top panel decal *(see Fig. 6.1.1). (See Part 8 below for more information on VBATT and GND.)*

7.3. 14230-OBD CABLE

Users who intend to use the 14230 Converter in-vehicle can purchase an optional cable from Silicon Engines. Specify the **14230-OBD Cable**. The standard cable length is 10 meters (32.8 feet).

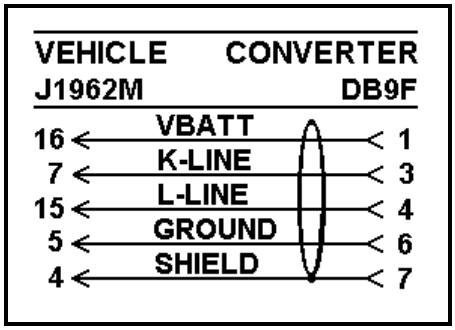
This cable is compatible with the On-Board Diagnostic connector in automotive vehicles. The OBD connector is typically located within one meter (~3 feet) of the steering wheel, in a location that does not require tools for access.

At the 14230 Converter side, the 14230-OBD cable provides a DB9F connector, as described above. On the other end is a 16-pin connector designed to plug into the OBD port, as defined in [J1962].



IN-VEHICLE ON-BOARD DIAGNOSTIC CONNECTOR
FIGURE 7.3.1.

At the vehicle side, the Silicon Engines 14230-OBD cable provides a 16-pin male J1962 connector, designed to plug into the OBD port. The 14230-OBD cable is wired as follows:



CABLE FROM VEHICLE OBD PORT TO 14230 CONVERTER
FIGURE 7.3.2.

7.4. 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 9010 TO PERSONAL COMPUTER
FIGURE 7.4.1.

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 9010.

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.

8. POWER REQUIREMENTS

8.1. 14230 POWER, USB POWER

The 14230 interface section of the Converter typically derives its power from the automotive battery.

Usually the USB section of the Converter also derives its power from the automotive battery. However when no automotive battery power is available, the Model 9010 gets its power from the USB port on the personal computer—in a manner similar to many familiar PC peripheral devices, such as a mouse or memory stick.

8.2. CONNECTING VBATT AND GROUND

When the 14230 Converter is plugged into a vehicle OBD (on-board diagnostic) port, using the 14230-0BD Cable described above, the 14230 circuitry within the device derives power from the vehicle's battery.

When working with an ECU at the component level, the VBATT and GROUND lines are typically connected—along with the K-line and (optionally) the L-line—to a suitable connector on the ECU. Both the 14230 Converter and the ECU are powered from the same source of power. The 14230 Converter is compatible with both 12-volt and 24-volt battery systems.

The 14230 Converter can also be powered by a DC power supply that connects to building AC power lines.

8.3. 14230 INPUT VOLTAGE AND CURRENT

The 14230 Converter contains built-in power supply circuitry that generates needed 14230-side power from VBATT and GND. The 14230 Converter is compatible with 12-volt battery systems (8 to 16 VDC), as well as 24-volt systems (16 to 32 VDC).

The unit generates +5 VDC and +3.3 VDC internally for its digital logic and the 14230 LED indicators.

SPECIFICATION	MIN.	TYP.	MAX	UNITS	CONDITIONS
SUPPLY VOLTAGE	8.0		32.0	VDC	CONTINUOUS OPERATION
	8.0	13.8	16.0	VDC	12 VOLT BATTERY RANGE
	16.0	27.6	32.0	VDC	24 VOLT BATTERY RANGE
			60	VDC	LOAD DUMP, 100 MS MAX.
			-60	VDC	REVERSE BATTERY
SUPPLY CURRENT		TBD		MA	VBATT=+13.8 VDC, K-LINE IDLE
		TBD		MA	VBATT=+27.6 VDC, K-LINE IDLE

SUPPLY POWER SPECIFICATIONS
FIGURE 8.3.1.

8.4. LOAD DUMP PROTECTION

The 14230 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 14230 Converter. One method is to power the device from an AC line-powered DC power supply, rather than from the vehicle's battery.

8.5. SEPARATE VBATT SOURCES

If the ECU and the 14230 Converter are powered from separate sources:

- **Grounds:** The ground of the 14230 Converter must be connected to the grounds of the ECU and of both power sources.
- **VBATT:** The VBATT voltage provided to the 14230 Converter should be within the ranges specified above, and within ±3 VDC of the VBATT voltage provided to the ECU.

8.6. REVERSE BATTERY PROTECTION

The 14230 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.

9. ISO-14230 K-LINE

9.1. BUS TOPOLOGY

The K-line is a bi-directional, half-duplex, serial input/output line for exchange of information between an ECU (electronic control unit) and a diagnostic tester.

In a production vehicle, multiple modules may be connected over the same K-line. The K-line terminal on each ECU is connected to like terminals on other ECUs within the vehicle.

Each ECU communicates over the K-line only when it receives its unique address code from the diagnostic tester. Typically only one ECU is active at once.

9.2. SERIAL DATA FORMAT

Once initialization has taken place, the 14230 Converter sends and receives data at high speed—typically 10,400 bps—using an asynchronous serial data format. 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.

The 14230 data format is similar to RS-232, except that the serial data signals on the ISO-14230 K-line are non-inverted, unipolar (VBATT and GROUND), while the serial data signals on RS-232 TXD and RXD lines are inverted, bipolar (± 10 to 15 V).

BIT DESCRIPTION	LOGIC LEVEL	ISO-14230 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 ISO-14230, RS-232
FIGURE 9.2.1

9.3. 14230 SERIAL DATA SPEED

The typical ISO-14230 high-speed data rate after initialization is 10,400 bps. The Model 9010 allows the 14230 data rate to be set at any integer value from 1,000 to 115,200 bits per second.

The standard data rate of 10.4 kbps, which is specified in ISO-14230 and related standards, is not a standard for personal computers—so one of the benefits of the Model 9010 is to communicate at baud rates that would be difficult for a PC. The embedded microcontroller within the Model 9010 controls the 14230 baud rate.

Though these data rates are not supported in [ISO-14230], the Model 9010 can also be programmed to communicate over the K-line at data rates up to 115.2 kbps. This capability is intended for use in ECU programming and testing.

9.4. K-LINE LOGIC WITHIN ECU

Within a typical ISO-14230 ECU, the K-line transmitter circuit includes an open-collector bipolar transistor (or open-drain MOSFET) that is normally off. A pull-up resistor to VBATT within the diagnostic tester causes the K-line to rise to the level of VBATT when the diagnostic tester is attached and the K-line is idle. The ECU activates this transistor when transmitting on the K-line.

Within the ECU, the K-line receiver is typically a voltage comparator, set to slice at 50% of VBATT.

These transmitter and receiver functions are typically embodied in an ISO-9141 transceiver, such as the Freescale (formerly Motorola) MC33199, ST Microelectronics L9613, or Vishay Siliconix Si9243A.

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

9.5. HALF-DUPLEX SINGLE-WIRE INTERFACE

The ISO-14230 K-line is a single-wire interface. Data flow is half-duplex, in one direction only. Either the tester is talking, or the ECU. Typically the diagnostic tester acts as the bus master, sending a command to the ECU, which then sends a response.

A direct result of the half-duplex nature of the K-line is data echo. For example, whenever the ECU sends a logic 0 to the tester on the K-line, the K-line goes low. The ECU's K-line receiver detects this condition. This means that every byte that the ECU sends to the tester, using its UART's TXD pin, is simultaneously echoed back to the ECU on its UART's RXD pin.

9.6. K-LINE LOGIC WITHIN 14230 CONVERTER

The 14230 Converter, acting as the diagnostic tester in the system, provides a pull-up resistor from the K-line to VBATT. The specified value of this pull-up resistor is 510 Ω for 12-volt battery systems, and 1 KΩ for 24-volt battery systems [ISO-14230-1 §6.2.2].

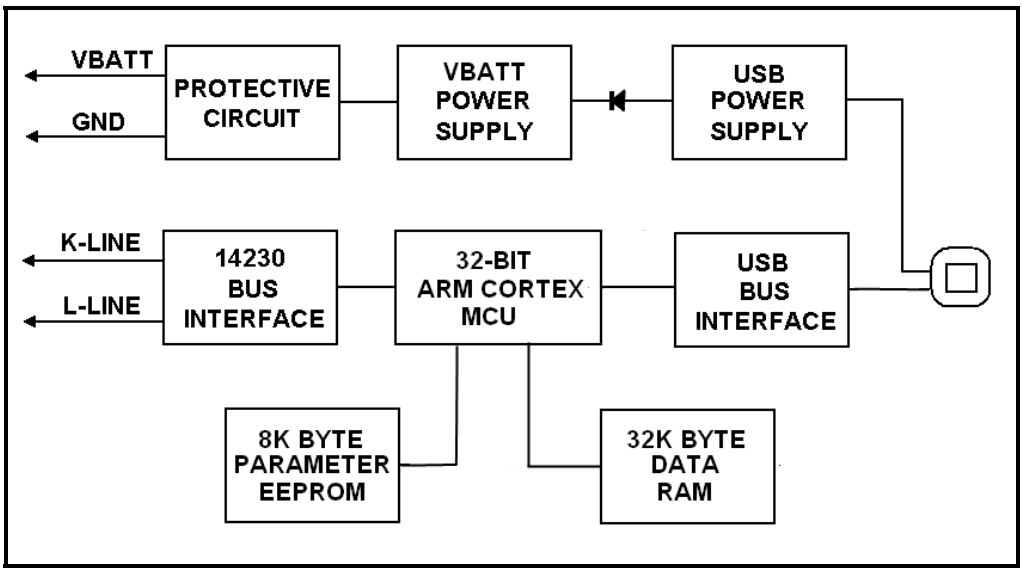
A simplified block diagram for the Model 9010 14230 Converter appears below.

The microcontroller (MCU) within the Model 9010 controls the 14230 baud rate.

The MCU also converts half-duplex K-line data to full-duplex. Specifically, when the PC sends a data byte over its USB port, to the 14230 Converter, the MCU forwards it onto the K-line. Because the K-line is half-duplex, the K-line receiver in the Converter will see this transmitted data. The 14230 MCU filters out echoed bytes and to help simplify PC software.

The 14230 Converter contains protective circuitry to assure that the maximum voltage on the K-line does not exceed 40 VDC, even in the presence of higher-voltage transients on the VBATT line [ISO-14230-1 §6.2.5].

The transceiver IC contained in the 14230 Converter contains slew-rate limiting circuitry to reduce radio frequency interference. The transceiver IC also contains current-limiting circuitry to protect its output transistor against a short to VBATT. The maximum current is below the 100-mA limit specified in the standards [ISO-14230-1 §6.2.2].



MODEL 9010 BLOCK DIAGRAM
FIGURE 9.6.1.

9.7. K-LINE ELECTRICAL SPECIFICATIONS

PARAMETER	MINIMUM	TYPICAL	MAXIMUM	CONDITIONS/COMMENTS
OPERATING VOLTAGE RANGE, 5V INDUSTRIAL MODE	+4.5 VDC	+5 VDC	+8 VDC	5 VOLT MODE
OPERATING VOLTAGE RANGE, 12 VOLT AUTOMOTIVE MODE	+8 VDC	+13.8 VDC	+16 VDC	12 VOLT MODE
OPERATING VOLTAGE RANGE, 24 VOLT TRUCK/BUS MODE	+16 VDC	+27.6 VDC	+32 VDC	24 VOLT MODE
DIAGNOSTIC TESTER LOAD RESISTOR, 5V VOLT MODE	228 Ω	240 Ω	252 Ω	240 Ω ±5%
DIAGNOSTIC TESTER LOAD RESISTOR, 12 VOLT MODE	484 Ω	510 Ω	536 Ω	510 Ω ±5%
DIAGNOSTIC TESTER LOAD RESISTOR, 24 VOLT MODE	950 Ω	1,000 Ω	1,050 Ω	1 KΩ ±5%
DIAGNOSTIC TESTER LOAD RESISTOR, NO-PULL-UP MODE	10K Ω			
MAXIMUM VOLTAGE ON K-LINE		34 VDC		VOLTAGE CLAMP ON K-LINE OUTPUT
MAXIMUM SINK CURRENT, OUTPUT LOW		60 mA		CURRENT-LIMITED TRANSCEIVER
RISE TIME/FALL TIME		0.4	2.0	µSEC, 9V ≤ VBATT ≤ 16V

K-LINE ELECTRICAL SPECIFICATIONS
FIGURE 9.7.1

9.8. SELECTING 12-VOLT VERSUS 24-VOLT MODE

The 14230 Converter contains K-line and L-line load resistors for both 5V/12-volt and 24-volt operating modes. If auto-detect voltage mode was previously selected, a voltage comparator measures the level of VBATT. If VBATT is above 16 VDC, then the load resistors are set to 1 KΩ. If VBATT is below 16 VDC, then the load resistors are switched to 510 Ω.

PARAMETER	MINIMUM	TYPICAL	MAXIMUM	CONDITIONS/ COMMENTS
VBATT THRESHOLD VOLTAGE FOR SETTING LOAD RESISTORS	15.2 VDC	16.0 VDC	16.8 VDC	16.0 VDC ± 5%

VOLTAGE THRESHOLD FOR AUTOMATIC LOAD RESISTOR SELECTION
FIGURE 9.8.1

Typically a 12-volt ECU operates at 14.4 volts and below, while a 24-volt system typically runs at 20 volts and above. Within these ranges, the 14230 Converter will automatically select the correct load resistor.

However in case the ECU must be operated close to the 16-volt threshold, the 14230 Converter provides the possibility of load resistor configuration through software. *See Section 13.1 for details.*

10. ISO-9141 L-LINE

10.1. ACTIVE DURING INITIALIZATION

In addition to the K-line, the 14230 Converter also supports the L-line. This signal is used only during initialization, and its use by ECUs is optional [ISO-14230-2 §3].

The L-line may be used by an ECU to assist in waking up from low-power sleep mode when the diagnostic tester is plugged in. The L-line may also be connected to a general-purpose port pin on the ECU's microcontroller, to handle the initialization sequence—whether 5-baud initialization, or fast initialization—in place of the ECU's UART/SCI port.

When the 14230 Converter is in the first portion of the initialization sequence, in **Initialization Mode,** the L-line outputs exactly the same signals that are present on the K-line. *(For details, see Part 15.*)

When the 14230 Converter is in **Run Mode**, the L-line is idle. The L-line output transistor within the 14230 Converter is off, and the load resistor within the 14230 Converter pulls the L-line up to VBATT.

10.2. L-LINE ELECTRICAL SPECIFICATIONS

The electrical specifications for the L-line are the same as for the K-line in the preceding section. The value of the L-line load resistor, 510 Ω or 1 KΩ, is controlled by the same logic as for the K-line.

10.3. L-LINE STATUS

Unlike the K-line, the L-line is not bi-directional. The ECU does not send data to the diagnostic tester over the L-line. However, the 14230 Converter contains circuitry for monitoring the state of the L-line, and displaying its condition on a bicolor LED.

11. LAMP FUNCTIONS AND SIGNAL FLOW

11.1. STATUS LAMP

In the center of the top panel of the 14230/USB Converter is a green LED lamp, marked **STAT**.

LED PATTERN	SIGNAL LINE CONDITION
STEADY GREEN	CONVERTER OPERATING NORMALLY
BLINKING	DATA SUCCESSFULLY RECEIVED OR TRANSMITTED ON THE K-LINE
OFF	CONVERTER NOT OPERATING PROPERLY

STATUS LAMP PATTERNS
FIGURE 11.1.1.

If the STAT lamp is off, please contact Silicon Engines for assistance.

11.2. TWO-COLOR LAMPS

Three two-color indicators on the top panel of the 14230/USB Converter show the status of the ISO-14230 and USB signal lines. Each lamp glows either green or red whenever the device is powered up.

11.3. K AND L LINE LAMPS

The lamps for the K-line and the L-line indicate voltage levels on these communications lines.

LED COLOR	SIGNAL LINE CONDITION
GREEN	HIGH VOLTAGE LEVEL ON SIGNAL AT CONNECTOR
RED	LOW VOLTAGE LEVEL ON SIGNAL AT CONNECTOR

LED COLOR CODES
FIGURE 11.3.1.

These voltage levels and lamp colors correspond to the following signal conditions:

CONN. PIN	SIGNAL NAME	DATA DIR.	LAMP COLOR	VOLTAGE LEVEL	SIGNAL FUNCTION
3	K	ECU↔PC	GREEN	VBATT	IDLE LINE; LOGIC 1(MARK); STOP BIT
			RED	0 V	START BIT; LOGIC 0 (SPACE)
4	L	ECU←PC	GREEN	VBATT	IDLE, INITIALIZATION COMPLETE
			RED	0 V	LOGIC 0 LEVELS DURING INITIALIZATION

ISO-14230 K AND L LINE LAMPS
FIGURE 11.3.2.

11.4. USB LAMP

The indicator lamp for the USB line is a single two-color LED.

LAMP COLOR	SIGNAL FUNCTION
GREEN	IDLE USB LINE
BLINKING	SUCCESSFUL COMMUNICATIONS
RED	USB SUSPENDED OR NOT CONNECTED

USB LAMP CODES
FIGURE 11.4.1.

11.5. LAMP PATTERNS, SYSTEM IDLE

ISO-14230 LAMPS				USB LAMP	
LAMP	COLOR	STATE	LEVEL	COLOR	STATE
K	GREEN	IDLE	VBATT	GREEN	IDLE
L	GREEN	IDLE	VBATT		

LAMP PATTERNS, ISO-14230 SYSTEM IDLE
FIGURE 11.5.1.

The above lamp pattern should appear when the system is idle. The 14230/USB Converter is connected to both the PC and the ECU, and the power is on. The 14230 software in the PC is *not* running, so the USB port from the PC is idle. The K-line lamp, the L-line lamp, and the USB lamp are all green.

11.6. K OR L LAMP STEADY RED

If either the K lamp or the L lamp is red when the system is idle, check the ECU and the cables for a short to ground.

The K lamp should be green most of the time in any mode. When the system is communicating, the K lamp will be green while the ISO-14230 line is idle waiting for a new command, and during logic 1 data bits, stop bits, and inter-frame idle time. The K lamp should go red only for start bits and logic 0 data bits. A steady red K lamp indicates a problem on the K-line.

The L lamp should also be green most of the time. It should go red only during ISO-14230 initialization. A steady red L lamp indicates a problem on the L-line.

If the K or L lamp is red, the problem could also be that the wrong Bus Voltage was selected. When in doubt, select auto-select bus voltage.

11.7. LAMP PATTERNS, ISO-14230 INITIALIZATION

ISO-14230 LAMPS				USB LAMP		
LAMP	COLOR	STATE	LEVEL	COLOR	STATE	LEVEL
K	GREEN WITH RED FLASHES	1/0	VBATT/ GND	BLINKS	1/0	INIT SEQUENCE STARTED
L	GREEN WITH RED FLASHES	1/0	VBATT/ GND			

LAMP PATTERNS, INIT MODE
FIGURE 11.7.1.

During ISO-14230 initialization, the USB lamp will blink, indicating that the PC has commanded **Initialization Mode**.

While the initialization sequence is under way, the K and L lamps will be green with red flashes.

When initialization is complete, a return code will be relayed to the PC over the USB interface.

11.8. LAMP PATTERNS, RUN MODE

ISO-14230 LAMPS				USB LAMP		
LAMP	COLOR	STATE	LEVEL	COLOR	STATE	LEVEL
K	GREEN WITH RED FLASHES	1/0	VBATT/ GND	BLINKING	1/0	DATA
L	GREEN	1	VBATT			

LAMP PATTERNS, RUN MODE
FIGURE 11.8.1.

During run mode, the K lamp will be green with short red flashes, corresponding to START and 0 bits in the ISO-14230 data stream. The L line is idle in Run Mode, so the L lamp should be steady green.

In Run Mode, the USB lamp will be blinking, corresponding to the data being sent Thto and from the PC.

12. SOFTWARE CONTROL

12.1. COMMANDS VIA USB INTERFACE

The Model 9010 14230/USB Converter is under the control of a personal computer, connecting via the USB interface.

12.2. ISO-14230 PROTOCOL

The 14230 MCU within the Model 9010 manages the details of the ISO-14230 protocol, including the ISO-14230 initialization sequence, and communications at 10,400 baud (or other programmed baud rate).

See Parts 15-16 below for details on ISO-14230 protocol functions.

12.3. ISO-14230 USB MESSAGE CENTER

Silicon Engines provides a program called the ***ISO-14230 USB Message Center*** for use with the Model 9010 14230/USB Converter. It operates on a personal computer running Windows®. This utility provides the means to set up the Model 9010, and to communicate with an ECU with an ISO-14230 interface. There is no extra charge for this software.

See Parts 17-20 below for details on the ISO-14230 USB Message Center.

12.4. MODEL 9010 AS 14230 BUS HOST

The most common use model for the Model 9010 is to act as the 14230 bus host. It connects over USB to a personal computer running the ***ISO-14230 USB Message Center***, or custom software written for a specific system. In this case, the Model 9010 should be configured for the operating voltage of the system—5V, 12V, or 24V. This causes the Model 9010, acting as bus master, to apply a bus load resistor matching the ISO-14230 bus specifications for the operating voltage in use. *(See Section 13.1 on setting the operating voltage.)*

Typically the client modules on the 14230 bus will require initialization sequences to start a communication session. *(See Sections 13.2-13.4 for an introduction to Model 9010 bus initialization commands.)*

12.5. MODEL 9010 IN LISTENER MODE

A useful operating mode for the Model 9010 is Listener Mode. The Model 9010 connects via USB to a PC running the USB Message Center, or custom software. Some other device in the system—such as a diagnostic analyzer—acts as the 14230 bus host. One or more other modules—such as automotive electronic control units—act as 14230 bus clients. In this mode, the diagnostic analyzer typically provides the bus load resistor, so the Model 9010 should typically be set for no pull-up resistor. *(See Section 18.7 on setting the Model 9010 for no pull-up resistor.)*

Actually the Model 9010 is always in Listener Mode. Provided that the Model 9010 has been programmed for the baud rate that corresponds to the 14230 bus, it will always show bus traffic on the ISO-14230 USB Message Center screen—whether messages are generated by a 14230 bus client ECU; by a separate 14230 bus host (such as a diagnostic analyzer); or by the Model 9010 itself.

However in a system where there is a separate 14230 bus master (such as a diagnostic analyzer), we provide the option to turn off the Model 9010 load resistor, to avoid the possibility that duplicated load resistors could interfere with proper bus waveforms.

12.6. CUSTOM CONTROL SOFTWARE

Many users need customized PC software—for example, to run end-of-line test programs on a specific ECU.

To support development of custom control software, Silicon Engines makes available the source code for our ISO-14230 USB Message Center. We also provide the ***14230USB.DLL (x32) and 14230USB64.DLL*** (x64) libraries that contains functions that can be called by C, C++, or Visual Basic® code. For the remainder of this specification, it will be assumed the user is using the 14230USB.DLL for x32 deployment, but if the user desires x64 (64-bit) deployment, they should substitute references to 14230USB.DLL with 14230USB64.DLL.

13. ISO-14230 CONTROL COMMANDS

13.1. SET OPERATING VOLTAGE

This USB command sets the operating voltage of the 14230/USB Converter for 12-volt mode, 24-volt mode, 5V mode, no pull-up mode, or auto-detect mode. *(See Section 9.8 on auto-detect mode.)*

The operating voltage setting is stored in non-volatile memory within the Model 9010, so that at power up, it will come up in the mode last programmed over the USB link. Auto-detect mode is the default mode if this function has not been programmed otherwise.

All the supported voltage mode options can be set from the ISO-14230 USB Message Center.

Custom PC software should make use of the function `Iso14230UsbSetBusCharacteristics` in driver library `14230USB.DLL`. This function will return a code indicating success or failure and the reason for failure. *See the Programmer's Reference Manual, 14230/USB Converter, PRM9009-9010, for a description of the voltage mode routines in 14230USB.DLL.*

13.2. INITIALIZATION ALTERNATIVES

To communicate with most ECUs that implement ISO-14230, the diagnostic tester must complete an initialization sequence. The 14230/USB Converter supports four initialization alternatives:

5-Baud Initialization: Low-speed initialization at 5 baud, as originally provided by [ISO 9141].

CARB Initialization: 5-baud initialization consistent with the specifications for on-board diagnostics (OBD) according to the California Air Resources Board specifications [ISO-9141-2, ISO-14230-4].

Fast initialization: High-speed initialization, as more recently defined by [ISO-14230-2].

When one of the above three options is selected, the 14230/USB Converter handles all the details of initialization in response to commands received from the PC over the USB interface.

13.3. FAST INIT COMMAND

This command performs fast initialization of the attached ECU. *For details on initialization, see Part 15.*

Fast initialization can be commanded from the ISO-14230 USB Message Center.

Custom PC software should make use of the function `Iso14230UsbFastInit` in driver library `14230USB.DLL`. This function will return a code indicating success or failure and the reason for failure. *See the Programmer's Reference Manual, 14230/USB Converter, PRM9009-9010, for a description of the initialization routines in 14230USB.DLL.*

Note: Many ECUs require periodic TESTER PRESENT messages following initialization to stay in communications mode. *See Section 13.5.*

13.4. SLOW INIT COMMAND

This command is used to perform slow initialization of the attached ECU. Slow initialization applies both to ECUs that use the legacy 5-baud initialization prescribed by [ISO-9141], as well as to ECUs meeting the CARB requirements. *For details on initialization, see Part 15.*

Slow initalization can be commanded from the ISO-14230 USB Message Center.

Custom PC software should make use of the function `Iso14230UsbExtendedSlowInit` in driver library `14230USB.DLL`. This function will return a code indicating success or failure and the reason for failure. *See the Programmer's Reference Manual, 14230/USB Converter, PRM9009-9010, for a description of the initialization routines in 14230USB.DLL. (Note: the routine Iso14230UsbSlowInit is also provided for compatibility with software that utilizes both the 9009 and the 9010 interchangeably. The only difference between the two is that the extended command includes an "alternate" argument for alternate slow init).*

Note: Many ECUs require periodic TESTER PRESENT messages following initialization to stay in communications mode. *See Section 13.5.*

13.5. TESTER PRESENT MESSAGE COMMANDS

These commands set up TESTER PRESENT messages that will be sent periodically by the 14230/USB Converter. Many ECUs require periodic TESTER PRESENT messages following initialization to stay in communications mode.

When TESTER PRESENT messages are activated, the 14230/USB Converter automatically sends these messages every two seconds to the connected ECU. These messages are automatic—once set up they do not need to be generated by the PC.

The 14230/USB Converter supports either the standard TESTER PRESENT message, or a user-defined message. Unless a custom message is defined, the TESTER PRESENT message will use the parameters defined in the FAST INIT or SLOW INIT commands, including the functional argument and the source and target addresses.

In the event that the connected ECU requires a non-standard TESTER PRESENT message, the 14230/USB Converter can generate a user-defined custom message that is up to 31 bytes long.

TESTER PRESENT messages can be controlled from the ISO-14230 USB Message Center.

Custom PC software should make use of the function `Iso14230UsbSetContinualWakeup` in driver library `14230USB.DLL` to control whether TESTER PRESENT messages are sent. The function `Iso14230UsbSetCustomTesterPresentMessage` is used to define a custom message. *See the Programmer's Reference Manual, 14230/USB Converter, PRM9009-9010, for a description of TESTER PRESENT routines in 14230USB.DLL.*

13.6. REQUEST FOR KEYWORD OR SYSTEM ID

This command requests the 14230/USB Converter to return the keyword that it received from the ECU during the most recent initialization sequence. *For details on keywords and initialization, see Part 15.*

A request for the most recent keyword can be generated from the ISO-14230 USB Message Center.

Custom PC software should make use of the function `Iso14230UsbGetMostRecentKeyword` in driver library `14230USB.DLL`. This function will return a code indicating success or failure and the reason for failure. *See the Programmer's Reference Manual, 14230/USB Converter, PRM9009-9010, for a description of keyword request routine in 14230USB.DLL.*

For alternate slow initialization, the application software will also have to know the system ID. To retrieve the most recent system ID, you utilize the routine Iso14230UsbGetMostRecentSystemID in driver library 14230USB.DLL. You can also press the Request Keyword button in the ISO-14230 USB Message Center and if configured for alternate slow initialization, the most recent system ID will also be reported.

13.7. SET ISO-14230 BAUD RATE

This command sets the data rate that will be used by the 14230/USB Converter to communicate with the ECU, after initialization is complete. Valid settings are from 1,000 to 115,200 bps. The standard data rate is 10,400 bps, as specified by [ISO-14230-2, §5.2.4.2.3.1].

The ISO-14230 baud rate setting is stored in non-volatile memory within the Model 9010, so that at power up, it will come up set for the baud rate last programmed over the USB link.

The baud rate can be programmed from the ISO-14230 USB Message Center.

Custom PC software should make use of the function `Iso14230UsbSetHighSpeedBaudRate` in driver library `14230USB.DLL`. This function will return a code indicating success or failure and the reason for failure. *See the Programmer's Reference Manual, 14230/USB Converter, PRM9009-9010, for a description of the high speed baud rate routines in 14230USB.DLL*

(Note: the routine `Iso14230UsbSetBaudRate` *is also provided for applications that use the Model 9009 and Model 9010 interchangeably; in that case, the baud rate specified cannot be higher than 10,417. The Model 9009 cannot operate at baud rates higher than 10,417.)*

13.8. SET HIGH SPEED DUPLEX MODE

This command only applies if the baud rate is higher than 10,417. In that case, the Model 9010 is using an actual UART for the data transmission and is not checking every bit assertion. In that case, for guaranteed delivery applications, the high speed duplex mode should be turned on and then every transmitted byte will be echoed back if it transmitted properly. The application would then be responsible for making sure that the echoed back data matched the transmitted data. For baud rates below 10,417, this is not required because if the bit did not assert properly on the bus, an error code is automatically generated.

The ISO-14230 high speed duplex mode setting is stored in non-volatile memory within the Model 9010, so that at power up, it will come up with the setting last programmed over the USB link.

This setting can also be programmed from the ISO-14230 USB Message Center.

Custom PC software should make use of the function `Iso14230UsbSetHighSpeedDuplex` in driver library `14230USB.DLL`. This function will return a code indicating success or failure and the reason for failure. *See the Programmer's Reference Manual, 14230/USB Converter, PRM9009-9010, for a description of the high speed duplex routine in 14230USB.DLL.*

13.9. SET INTER-BYTE DELAY (P4)

This command sets the inter-byte delay—the time period between bytes that are transmitted by the 14230/USB Converter to communicate with the ECU. Valid settings are from 0 to 51 milliseconds. The inter-byte delay is also known as the P4 time period, and its standard value is 5 milliseconds, as specified by [ISO-14230-2, §4.5.1].

The inter-byte delay setting is stored in non-volatile memory within the Model 9010, so that at power up, it will come up set for the delay interval last programmed over the USB link.

The inter-byte delay can be programmed from the ISO-14230 USB Message Center.

Custom PC software should make use of the function `Iso14230UsbSetInterByteDelay` in driver library `14230USB.DLL`. This function will return a code indicating success or failure and the reason for failure. *See the Programmer's Reference Manual, 14230/USB Converter, PRM9009-9010, for a description of the inter-byte delay routines in 14230USB.DLL.*

13.10. SEND MESSAGE

This command requests the 14230/USB Converter to send a specified message to the ECU over the K-line. This message is transmitted using the interbyte spacing specified in the interbyte spacing command.

A message can be entered and sent to the connected ECU using the ISO-14230 USB Message Center.

Custom PC software should make use of the function `Iso14230UsbWriteDataMsg` in driver library `14230USB.DLL`. This function will return a code indicating success or failure and the reason for failure. *See the Programmer's Reference Manual, 14230/USB Converter, PRM9009-9010, for a description of the status request routine in 14230USB.DLL.*

13.11. REQUEST FOR STATUS CODE

This command requests the 14230/USB Converter to return its current status.

A request for the current status of the 14230/USB Converter can be generated from the ISO-14230 USB Message Center.

Custom PC software should make use of the function `Iso14230UsbReadStatus` in driver library `14230USB.DLL`. This function will return a code indicating success or failure and the reason for failure. *See the Programmer's Reference Manual, 14230/USB Converter, PRM9009-9010, for a description of the status request routine in 14230USB.DLL.*

13.12. CLEAR MCU BUFFERS

This command requests the 14230/USB Converter to clear its internal transmit and receive buffers as well as any lingering error codes.

A request to clear the MCU buffers within the 14230/USB Converter can be generated from the ISO-14230 USB Message Center.

Custom PC software should make use of the function `Iso14230UsbClearBuffers` in driver library `14230USB.DLL`. This function will return a code indicating success or failure and the reason for failure. *See the Programmer's Reference Manual, 14230/USB Converter, PRM9009-9010, for a description of the clear MCU buffers routine in 14230USB.DLL.*

13.13. REQUEST 14230/USB CONVERTER REVISION LEVEL

This command requests the 14230/USB Converter to return a value indicating its revision level. The returned value has two parts. The first is the revision number for the firmware loaded into the USB MCU. The second is the revision number for the 14230 MCU firmware.

A request to determine the revision level can be generated from the ISO-14230 USB Message Center.

Custom PC software should make use of the function `Iso14230UsbReadFirmwareRevision` in driver library `14230USB.DLL`. This function will return a code indicating success or failure and the reason for failure. *See the Programmer's Reference Manual, 14230/USB Converter, PRM9009-9010, for a description of the revision level routine in 14230USB.DLL.*

14. CONVERTER RESPONSE PACKETS

14.1. STATUS CODE RESPONSE PACKET

The 14230/USB Converter will return a STATUS CODE response packet in response to any of the following commands from the PC:

- FAST INIT *(Section 13.3)*
- SLOW INIT *(Section 13.4)*
- SET UP CUSTOM TESTER PRESENT MESSAGE *(Section 13.5)*
- SET ISO-14230 BAUD RATE *(Section 13.7)*
- SET ISO-14230 HIGH SPEED DUPLEX MODE *(Section 13.8)*
- SEND MESSAGE *(Section 13.10)*
- REQUEST FOR STATUS *(Section13.11)*
- CLEAR MCU BUFFERS *(Section 13.12)*

The status code will be returned to the user when making the function call to the `14230USB.DLL`. The table below shows the status codes.

HEX VALUE	COMMAND MEANING	LIKELY DIAGNOSIS/ COMMENTS
00H	CONVERTER FAILED TO BRING K-LINE AND L-LINE LOW	K-LINE AND L-LINE SHORTED TO VBATT
01H	CONVERTER FAILED TO BRING K-LINE LOW	K-LINE SHORTED TO VBATT
02H	CONVERTER FAILED TO BRING L-LINE LOW	L-LINE SHORTED TO VBATT
03H	K-LINE AND L-LINE FAILED TO RETURN HIGH	BUS CONTENTION, SHORT TO GROUND
04H	K-LINE FAILED TO RETURN HIGH	BUS CONTENTION, SHORT TO GROUND
05H	L-LINE FAILED TO RETURN HIGH	BUS CONTENTION, SHORT TO GROUND
09H	NO RESPONSE RECEIVED TO INITIALIZATION	BAD ECU ADDRESS, WIRING ERROR, ECU SOFTWARE FAULT
0AH	IMPROPER SYNC BYTE RECEIVED	5-BAUD INITIALIZATION ERROR
0BH	IMPROPER INVERSE ADDRESS RECEIVED	5-BAUD INITIALIZATION ERROR
0CH	IMPROPER START COMMUNICATIONS POSITIVE RESPONSE	FAST INITIALIZATION ERROR
0DH	BAD START COMMUNICATIONS POSITIVE RESPONSE CHECKSUM	FAST INITIALIZATION ERROR
0EH	BUS BUSY	THERE IS ALREADY A LOT OF TRAFFIC ON THE BUS WHICH DOES NOT PERMIT THE TRANSMISSION REQUESTED
20H	RECEIVED INVALID CHECKSUM FROM PC	COMMUNICATIONS NOISE, PC SOFTWARE ERROR
21H	RECEIVED INVALID COMMAND FROM PC	COMMUNICATIONS NOISE, PC SOFTWARE ERROR
30H	RECEIVE BUFFER OVERFLOW (FROM PC-USB)	FLOW CONTROL FAILURE
31H	TRANSMIT BUFFER OVERFLOW (TO PC-USB)	MORE TRAFFIC ON K-LINE THAN SYSTEM CAN KEEP UP WITH
4FH	NO ERROR PRESENT	SYSTEM OPERATING PROPERLY
50H	K-LINE BUS TRANSMIT BUFFER OVERFLOW (TO ECU)	FLOW CONTROL FAILURE
51H	K-LINE BUS RECEIVE BUFFER OVERFLOW (FROM ECU)	FLOW CONTROL FAILURE
E0H	CONVERTER MCU ROM CHECKSUM ERROR	HARDWARE FAULT, CONTACT SILICON ENGINES
E1H	EEPROM LOCKOUT ERROR	HARDWARE FAULT, CONTACT SILICON ENGINES
F7H	PC XMIT BUFFER OVERFLOW	SOFTWARE IS TRYING TO TRANSMIT TOO MANY MESSAGES/COMMANDS BEFORE USB HANDLER CAN SEND THEM
F9H	PC RECV BUFFER OVERFLOW	SOFTWARE IS NOT SPENDING ENOUGH TIME REMOVING DATA FROM 14230USB.DLL RECEIVE BUFFERS
FCH	OUT OF MEMORY	THE PC HAS RUN OUT OF MEMORY

14230/USB CONVERTER RESPONSE CODES
FIGURE 14.1.1.

14.2. KEYWORD RESPONSE PACKET

The 14230/USB Converter will return a KEYWORD response packet in response to the following command from the PC:

- REQUEST FOR KEYWORD *(Section 13.6)*

This response packet includes the two-byte ISO-14230 keyword received by the 14230/USB Converter, from the ECU, during the most recent initialization sequence. The keyword is formatted as two hexadecimal bytes.

If the 14230/USB Converter has not yet initialized the attached ECU, the keyword returned will be 0000H.

When initialization has occurred, the most significant keyword byte will be 8FH, and the least significant keyword byte will be as specified by [ISO-14230-2, §5.2.4.1, Table 8].

14.3. SYSTEM ID RESPONSE PACKET

The 14230/USB Converter will return a SYSTEM ID response packet in response to the following command from the PC:

- REQUEST FOR SYSTEM ID *(Section 13.6)*

This response packet includes the two-byte ISO-14230 system ID received by the 14230/USB Converter, from the ECU, during the most recent alternate slow initialization sequence. The system ID is formatted as two hexadecimal bytes.

If the 14230/USB Converter has not yet initialized the attached ECU with alternate slow initialization, the system ID returned will be 0000H.

14.4. REVISION LEVEL RESPONSE PACKET

The 14230/USB Converter will return a REVISION LEVEL response packet in response to the following command from the PC:

- REQUEST 14230 CONVERTER REVISION LEVEL *(Section 13.13)*

The response packet includes two one-byte hex values that show the 14230/USB Converter revision levels. The first one-byte hex value is the major revision level of the MCU firmware. The second one-byte hex value is the minor revision level of the MCU firmware.

14.5. RECEIVE DATA PACKETS

The 14230/USB Converter will package receive data into packets and then send them to the PC in receive data packets. All valid data at the proper baud rate on the K-line that is received by the 14230/USB Converter is packaged into packets one to four bytes long.

All data received by the 14230/USB Converter can be monitored using the ISO-14230 USB Message Center.

Custom PC software should make use of the driver library `14230USB.DLL`. *See the Programmer's Reference Manual, 14230/USB Converter, PRM9009-9010, for details on how to receive these data packets.*

A status code is returned with each receive data packet to indicate whether any errors occurred during or before the receipt of the data in the packet.

Note: Echoes from TESTER PRESENT messages (which are generated by the 14230/USB Converter) will be sent back to the PC. These packets are from one to 31 bytes in length.

TESTER PRESENT Messages can be monitored using the ISO-14230 USB Message Center.

15. INITIALIZATION DETAILS

15.1. HANDLED BY MCU

The MCU within the 14230 Converter takes care of the details of ECU initialization, in response to the RS-232 commands described above.

This section gives details on the functions performed by the 14230 Converter during initialization.

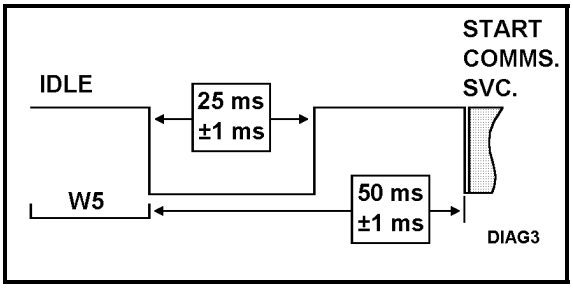
15.2. FAST INITIALIZATION

The 14230 Converter supports Fast Initialization as defined by [ISO-14230-2 §5.2.4.2.3.1]. This method offers much faster initialization than the 5-baud sequence defined by [ISO-9141].

The Fast Initialization sequence involves the following steps:

1. **Wake-Up Pattern:** The 14230 Converter sends a Wake-Up Pattern to the ECU. The K-line must first be idle (logic 1, VBATT level) for at least the [ISO-14230-2] W5 period, 300 ms minimum. The Wake-Up Pattern starts with a 25 ms (±1 ms) low signal (logic 0) on both the K-line and the L-line. The K-line and the L-line then return to a high level, ending after a total of 50 ms (±1 ms). After the Wake-Up Pattern, the L-line is idle.

2. **Start Communications Service:** The tester next sends a Start Communications Service request frame over the K-line to the ECU at the ISO-14230 high-speed data rate, typically 10,400 bps. 10,400 bps is the industry-standard high-speed data rate after fast initialization [ISO-14230-2 §5.2.4.2.3.1], but the 14230 Converter also supports non-standard baud rates *(see Section 9.6).* The Start Communications Service frame includes the physical or functional address so that only the intended ECU(s) are activated for this session. *(For details on the Start Communications Service frame, see Section 11.3.)* The falling edge of the start bit of the first byte of this frame marks the end of the Wake-Up Pattern time period.



FAST INITIALIZATION
FIGURE 15.2.1

3. **Start Communications Service Positive Response:** The ECU sends this acknowledgment packet back to the tester at the high-speed ISO-14230 data rate, typically 10,400 bps. *(For details on the Start Communications Positive Response frame, see Section 15.4.)*

The MCU within the 14230 Converter takes care of the two 25-millisecond time periods. It also sends the Start Communications Service frame at 10,400 bps (typical), and receives the Start Communications Service Positive Response frame.

For compatibility with [ISO-14230], the ECU should be programmed to recognize the 25 ms low level on the K-line or L-line. Typically it monitors the L-line directly, bypassing its serial communications interface. Upon detection of the 25 ms (± 1 ms) low level, the ECU detects that a Fast Initialization sequence is in process, and waits to receive the Start Communications request at 10,400 bps (typical) from the tester over the K-line.

15.3. START COMMUNICATIONS REQUEST

FRAME SECTION	SERVICE → / BYTE ↓	START COMMS. REQUEST	COMMENTS
HEADER	BYTE 1	81H	FORMAT BYTE, PHYSICAL ADDRESSING
		C1H	FORMAT BYTE, FUNCTIONAL ADDRESSING
		80H	FORMAT BYTE, ALTERNATE PHYSICAL ADDRESSING
		C0H	FORMAT BYTE, ALTERNATE FUNCTIONAL ADDRESING
	BYTE 2	*	ECU ADDRESS
	BYTE 3	*	TESTER ADDRESS
	BYTE 4 (ALTERNATE MODE ONLY)	01H	ALTERNATE HEADER LENGTH BYTE
DATA	BYTE 1	81H	CODE FOR START COMMUNICATIONS REQUEST
CHECKSUM	BYTE 1	Σ	CHECKSUM

START COMMUNICATIONS REQUEST
FIGURE 15.3.1

The14230 Converter sends a Start Communications Request over the K-line at the ISO-14230 high-speed data rate (typically 10,400 bps) as part of the Fast Initialization sequence. This packet has three sections—header, data, and checksum—consistent with the format of all high-speed data messages as defined by [ISO-14230-3].

The header section of the Start Communications Request contains three bytes. The first byte is a format byte with a value of 81H, C1H, 80H, or C0H specifying whether physical or functional addressing will be used in the header section of this frame. The addresses of the ECU and of the tester appear next within the header section. Note that 80H and C0H are used in the “alternate” method of initialization, which is used by Kia and Hyundai, for example. To invoke the alternate method of initialization, specify a lower case “p” or “f” in the fast initialization command.

The data section of the Start Communications Request contains one byte, 81H, which is the Service Identifier Byte assigned for a Start Communications Request [ISO-14230-2 §5.2.4.3.2].

The checksum is calculated on the previous bytes in the packet.

15.4. START COMMUNICATIONS POSITIVE RESPONSE

The ECU should reply to the Start Communications Request with a Start Communications Positive Response. Depending on the specific Keyword 2000 variation implemented by the ECU, this packet can take on any of the following four formats.

FRAME SECTION	SERVICE → / BYTE ↓	START COMMS. POS. RESPONSE	COMMENTS
HEADER	BYTE 1	1X YYYYYY BINARY	FORMAT BYTE [YYYYYY ≠ 0]
	BYTE 2	*	TESTER ADDRESS
	BYTE 3	*	ECU ADDRESS
DATA	BYTE 1	C1H	CODE FOR START COMMUNICATIONS RESPONSE
	BYTE 2	8F	HIGH-ORDER KEYWORD
	BYTE 3	*	LOW-ORDER KEYWORD
CHECKSUM	BYTE 1	Σ	CHECKSUM

START COMMUNICATIONS POSITIVE RESPONSE, FORMAT #1
FIGURE 15.4.1

FRAME SECTION	SERVICE → / BYTE ↓	START COMMS. POS. RESPONSE	COMMENTS
HEADER	BYTE 1	1X 00000 BINARY	FORMAT BYTE
	BYTE 2	*	TESTER ADDRESS
	BYTE 3	*	ECU ADDRESS
	BYTE 4	*	LENGTH BYTE
DATA	BYTE 1	C1H	CODE FOR START COMMUNICATIONS RESPONSE
	BYTE 2	8F	HIGH-ORDER KEYWORD
	BYTE 3	*	LOW-ORDER KEYWORD
CHECKSUM	BYTE 1	Σ	CHECKSUM

START COMMUNICATIONS POSITIVE RESPONSE, FORMAT #2
FIGURE 15.4.2

FRAME SECTION	SERVICE → / BYTE ↓	START COMMS. POS. RESPONSE	COMMENTS
HEADER	BYTE 1	0X YYYYYY BINARY	FORMAT BYTE [YYYYYY ≠ 0]
DATA	BYTE 1	C1H	CODE FOR START COMMUNICATIONS RESPONSE
	BYTE 2	8F	HIGH-ORDER KEYWORD
	BYTE 3	*	LOW-ORDER KEYWORD
CHECKSUM	BYTE 1	Σ	CHECKSUM

START COMMUNICATIONS POSITIVE RESPONSE, FORMAT #3
FIGURE 15.4.3

FRAME SECTION	SERVICE → / BYTE ↓	START COMMS. POS. RESPONSE	COMMENTS
HEADER	BYTE 1	0X 00000 BINARY	FORMAT BYTE
	BYTE 2	*	LENGTH BYTE
DATA	BYTE 1	C1H	CODE FOR START COMMUNICATIONS RESPONSE
	BYTE 2	8F	HIGH-ORDER KEYWORD
	BYTE 3	*	LOW-ORDER KEYWORD
CHECKSUM	BYTE 1	Σ	CHECKSUM

START COMMUNICATIONS POSITIVE RESPONSE, FORMAT #4
FIGURE 15.4.4

15.5. COMPLETION OF FAST INITIALIZATION

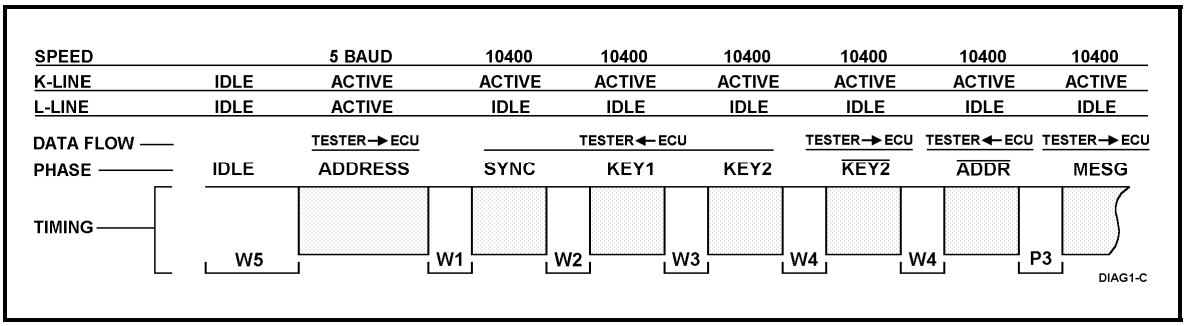
The MCU within the 14230 Converter handles the entire Fast Initialization sequence—including generation of the Wake-Up Pattern, sending the Start Communications Service packet to the ECU at 10,400 bps (typical), and receiving the Start Communications Positive Response packet at 10,400 bps (typical) from the ECU.

If initialization was successful, the 14230 Converter sends the PC a status code response packet over the USB interface that includes the code 4FH, no error *(Section 14.1)*.

If initialization was unsuccessful, the 14230 Converter sends the PC a status code response packet that includes an error code indicating the problem encountered *(Section 14.1)*.

15.6. 5 BAUD INITIALIZATION SEQUENCE

The 5-baud initialization protocol is illustrated in the following drawing. Each step is explained below, and then the time periods W1-W5 and P3 are defined.



5-BAUD INITIALIZATION SEQUENCE
FIGURE 15.6.1

1. **Idle state:** When a diagnostic tester is connected to the ISO 14230 K-line, the pull-up resistor to VBATT within the tester assures that the K-line is at an idle level of at least 80% of VBATT whenever communications are not actively occurring. The L-line is at the same level.

2. **5-baud address:** To start diagnostic communications using the 5-baud sequence, the diagnostic tester sends an initializing address code over both the K-line and the L-line at 5 baud (5 bits per second). The ECU is programmed to recognize a specific address code, and that address code must appear in the 5-baud start-up message from the diagnostic tester before the ECU will respond.

The 5-baud address is transmitted using NRZ (non-return-to-zero) coding, i.e., standard asynchronous start-stop UART or teletypewriter format. A high level (VBATT level) represents a logic 1, and a low level (0 volts nominal) represents a logic 0. An idle line is at a marking level (logic 1, VBATT level). A start bit (logic 0) starts each character.

The vehicle manufacturer determines the 5-baud address.

If the diagnostic tester implements ***physical addressing***, the 5-baud address is transmitted using 7O1 serial data format (seven data bits, odd parity bit, and one stop bit) [ISO-14230-2, §A.1]. The least significant address bit is transmitted first. The stop bit is a logic 1 level.

If the diagnostic tester implements ***functional addressing,*** the 5-baud address is transmitted using 8N1 serial data format (eight data bits, no parity bit, and one stop bit) [ISO-14230-2, §A.2]. The LSB is transmitted first, and the stop bit is a logic 1.

3. **L-line:** After the tester sends the 5-baud address on the K-line and L-line, the L-line goes idle (VBATT level). Its role is complete, and the L-line stays idle until the next time that the tester enters initialization mode.

4. **Baud rate synchronization pattern:** Upon receipt of its address at 5 baud, the ECU sends back a baud rate synchronization byte at the data speed it will use during the upcoming diagnostic session. This byte is 55 hex, sent at the high-speed rate that will be used during the diagnostic session. Coding is NRZ, 8N1. This sync character appears on the line as 0101010101 binary (start and stop bits underlined), providing an alternating bit pattern that an automotive diagnostic tester can use to determine the bit rate.

5. **14230 data rate:** The 14230 Converter communicates at a fixed 14230 data rate of 10,400 bps, unless programmed for a different speed *(see Section 13.7)*.

6. **Protocol keyword:** After sending the 55 hex sync byte, the ECU sends a two-byte keyword at the high-speed data rate (typically 10,400 bps). This two-byte keyword defines the data format to be used. We will use an example value of 2005 decimal. The keyword defines certain protocol characteristics that will govern high-speed communications after initialization, as defined in [ISO-14230-2 § 5.2.4.1]. Valid keywords fall into the range from 2000 to 2031 (decimal).

   The keyword is sent as two keybytes, each using 7O1 format (one start bit, seven data bits, an odd parity bit, and one stop bit). Keybyte 1, transmitted first, contains the seven low-order bits, and keybyte 2 contains the seven high-order bits. Within each byte, the least significant bit is transmitted first. As an example, the keyword 2005 decimal is transmitted as follows:

2005 DECIMAL															
MSB							LSB	MSB							LSB
PAR	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	PAR	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}
1	0	0	0	1	1	1	1	1	1	0	1	0	1	0	1
8F HEX								D5 HEX							
LAST ← ← ← ← ← TIME ORDER OF TRANSMISSION ← ← ← ← ← ***FIRST***															

5-BAUD KEYWORD DATA FORMAT
FIGURE 15.6.2

As shown above, 2005 decimal (07D5 hex), transmitted in 7O1 serial data format, is equivalent to 8FD5 hex in 8N1 format.

7. **Inverted keybyte acknowledgment:** Upon receipt of the two key bytes, the tester returns keybyte 2, logically inverted, in 8N1 format, at 10,400 bps (typical). Inverted keybyte 2 is 70 hex.

8F							
1	0	0	0	1	1	1	1
0	1	1	1	0	0	0	0
70							

INVERTED KEYBYTE ACKNOWLEDGMENT
FIGURE 15.6.3

8. **Inverted address code:** To complete the initialization handshake, the ECU sends its address code, logically inverted, in 8N1 format, at 10,400 bps (typical). The specific value of the inverted address code will of course depend on the address code selected by the manufacturer. The following shows inverted address codes for two example ECU addresses of A7 and B0.

EXAMPLE ADDRESS CODE #1								EXAMPLE ADDRESS CODE #2							
A7								B0							
1	0	1	0	0	1	1	1	1	0	1	1	0	0	0	0
0	1	0	1	1	0	0	0	0	1	0	0	1	1	1	1
58								4F							

INVERTED ADDRESS CODE EXAMPLES
FIGURE 15.6.4

15.7. 5 BAUD INITIALIZATION TIMING

			MILLISECONDS			
TIME	**DESCRIPTION**	**XMTR**	**MIN**	**MAX**	**NOM**	**TIMEOUT**
W5	IDLE AT LOGIC 1 BEFORE ADDRESS CODE SENT	TSTR	300	--	300	--
W1	FROM ADDRESS TO SYNC	ECU	60	300	65	550
W2	FROM SYNC TO KEYBYTE 1	ECU	5	20	10	110
W3	FROM KEYBYTE 1 TO KEYBYTE 2	ECU	0	20	10	110
W4	FROM KEYBYTE 2 TO INVERTED KEYBYTE 2	TSTR	25	50	30	110
	FROM INVERTED KEYBYTE 2 TO INVERTED ADDRESS	ECU	25	50	30	110
P3	FROM INVERTED ADDRESS TO NEXT PACKET	TSTR	55	5,000	1,000	5,000

5-BAUD INITIALIZATION TIMING
FIGURE 15.7.1

In the above chart, the *MIN* and *MAX* columns show specifications from [ISO-14230-2, §5.2.4.2.2.1] and [ISO-14230-2, §4.5.1]

The *XMTR* column shows which side is next to transmit, the tester or the ECU. The side that is transmitting is responsible to observe the time period.

The *NOM* column shows the nominal time period that the transmitting device uses.

The *TIMEOUT* column shows the time limit used by the receiving device, usually larger than the specified maximum, after which the receiving device stops waiting.

15.8. COMPLETION OF 5-BAUD INITIALIZATION

The MCU within the 14230 Converter handles the entire 5-baud initialization sequence as described above.

If initialization was successful, the 14230 Converter sends the PC a status code response packet that includes the code 4FH, no error *(Section 14.1)*.

If initialization was unsuccessful, the 14230 Converter sends the PC a status code response packet that includes an error code indicating the problem encountered *(Section 14.1)*.

15.9. ALTERNATE METHOD OF SLOW INITIALIZATION

The command `Iso14230UsbExtendedSlowInit` provides an alternate argument. If the alternate argument is set to 1, this specifies the alternate method of slow initialization. This method is used, for example, with certain Fiat models. In this method, every byte sent from the converter box is echoed back by the ECU before continuing. This means that ordinary message control cannot be performed with the Silicon Engines 14230 USB Message Center because ordinary message control does not wait after every byte sent for an echo. Therefore, in order to use the Model 9010 14230 USB Converter with this type of ECU, special custom software will have to be written by the implementer. The initialization is handled by this command, but special software will have to be written for handling actual message traffic.

This command also includes the existence of the "System ID" – a 16 bit number that is conveyed by the ECU during the initialization process. The System ID of the ECU can be queried, after successful initialization using this method, using the `Iso14230UsbGetMostRecentSystemID` function.

The format of the data sent back and forth is very similar to Figure 15.6.1, except that after Key 1 and Key 2 are conveyed, the two bytes of the System ID are also conveyed.

16. ISO-14230 DATA PROTOCOL

16.1. OPERATION IN RUN MODE

The PC software is responsible for the details of Keyword Protocol 2000, as defined in [ISO-14230-3], once initialization has been completed.

16.2. ERRORS IN RUN MODE

The 14230/USB Converter checks the K-line to make sure that, bit by bit, the data on the K-line matches the data commanded to be sent for baud rates of 10,417 or less. If an error is detected—for example, a K-line short to VBATT or ground—then the 14230/USB Converter will report this error with a status code attached to any Receive Data Packets, or in response to a Status Request. The status code is an error code specifying the problem *(Section 14.1)*.

The 14230/USB Converter rejects serial data words with framing errors—for example, the stop bit was not seen at the normal logic 1 level.

The 14230/USB Converter does not check for errors at the protocol layer—that is, Keyword Protocol 2000 data or timing errors.

When running at a baud rate greater than 10,417, the bit assertions are not checked and error checking becomes the responsibility of the application. The application can turn on full duplex mode for baud rates greater than 10,417 which will provide an echo of all bytes transmitted. The echo will only be correct if there were no errors on the bus. So the application should check the echoed data for guaranteed delivery applications for baud rates greater than 10,417.

16.3. RUN-MODE TIME-OUT FUNCTIONS

If there is no K-line traffic for over five seconds, then the ECU may time out and stop responding to commands.

The PC can keep the 14230/USB Converter in Run Mode by periodically sending TESTER PRESENT packets [ISO-14230-3 §6.4] whenever the data link is otherwise idle. *See Section 13.5 on TESTER PRESENT commands.*

17. MESSAGE CENTER—INSTALLATION

17.1. FUNCTIONALITY

The **ISO-14230 USB Message Center** software utility is provided with the 14230/USB Converter. When installed in a personal computer running Windows®, this program allows you to send messages through the 14230/USB Converter to a connected ECU. This program is compatible with Windows® 2000, XP, Vista, and 7.

17.2. INSTALLATION

The **ISO-14230 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 InstallSEMessageCenterAnd9009And9010Drivers[xxxx].EXE (where [xxxx] is the latest version) in the root folder of the CD. This program will install the software on your computer.

Please install the Message Center software first before plugging in the 14230/USB Converter for the first time.

17.3. INSTALLATION OF USB DEVICE DRIVERS

The USB device drivers are installed by any of the three 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\SE14230USB`. Here are the steps necessary to install the USB device drivers:

Message Center: Install the ISO-14230 USB Message Center software into your PC (see Sec 17.2).

Module power: Power up the 14230/USB Converter. The lamps should come on.

USB: Connect the USB cable from the Model 9010 to your PC.

17.4. WINDOWS 7 AND VISTA INSTALLATION SCREENS

If you are running Windows 7 or Vista, you should see a screen as shown below. *Note: Jungo is a supplier of PC-side USB software utilities.*

DEVICE SOFTWARE INSTALL SCREEN
FIGURE 17.4.1

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

WINDOWS PUBLISHER CHECK
FIGURE 17.4.2

Please click on INSTALL THIS DRIVER SOFTWARE ANYWAY.

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

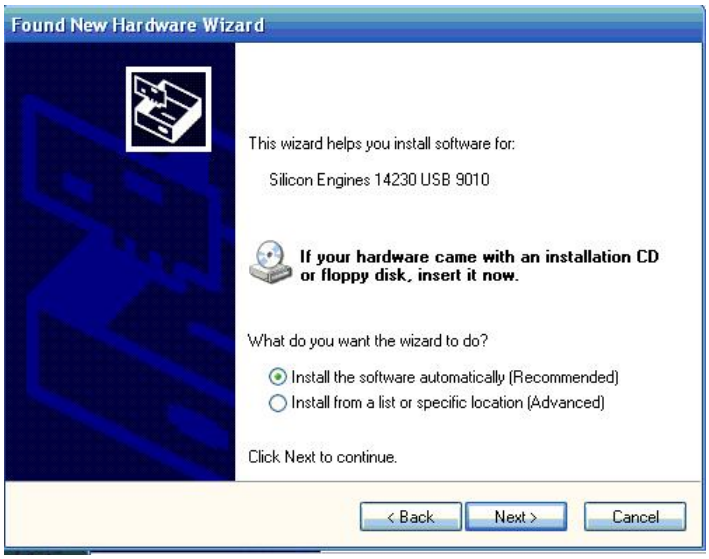
17.5. WINDOWS XP AND 2000 INSTALLATION SCREENS

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

WINDOWS XP AND 2000 INSTALLATION SCREEN
FIGURE 17.5.1

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

You should then see the following screen:



WINDOWS XP/2000 INSTALL SCREEN #2
FIGURE 17.5.2

Please click on INSTALL THIS SOFTWARE AUTOMATICALLY.

You should next see the following screen:

WINDOWS XP/2000 INSTALL SCREEN #3
FIGURE 17.5.3

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

18. MESSAGE CENTER—BASIC OPERATIONS

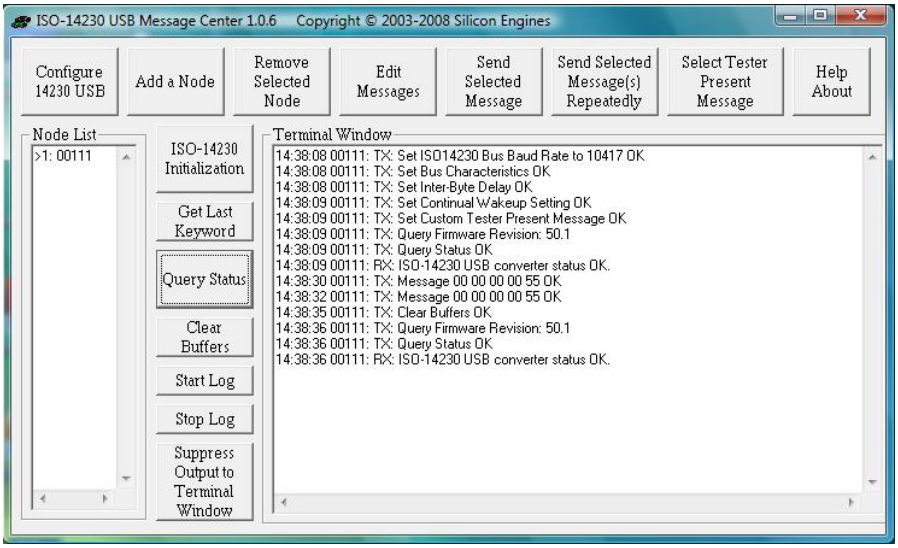
In this chapter, Part 18, we will discuss how to set up the system for basic functions.

In Part 19 we will discuss sending and receiving messages.

In Part 20, we will discuss advanced options.

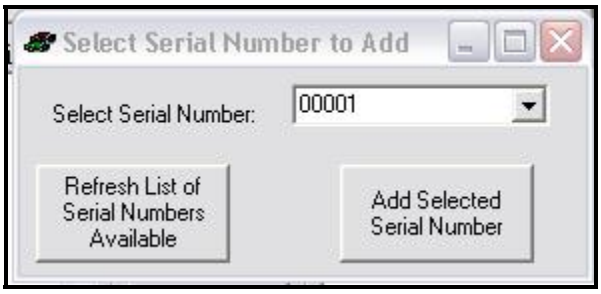
18.1. MAIN MESSAGE CENTER SCREEN

The **Main Message Center** screen appears below. It consists of 15 function buttons, a **Node List,** and a **Terminal Window**. Brief time-stamped messages appear in the **Terminal Window** to show messages flowing between the PC, the 14230/USB Converter, and the connected ECU. The Node List shows all connected nodes. In the picture below, no nodes have been connected yet.



MAIN MESSAGE CENTER SCREEN
FIGURE 18.1.1

To connect a new node, select the **Add a Node** button. You should see a diagram like the one below:

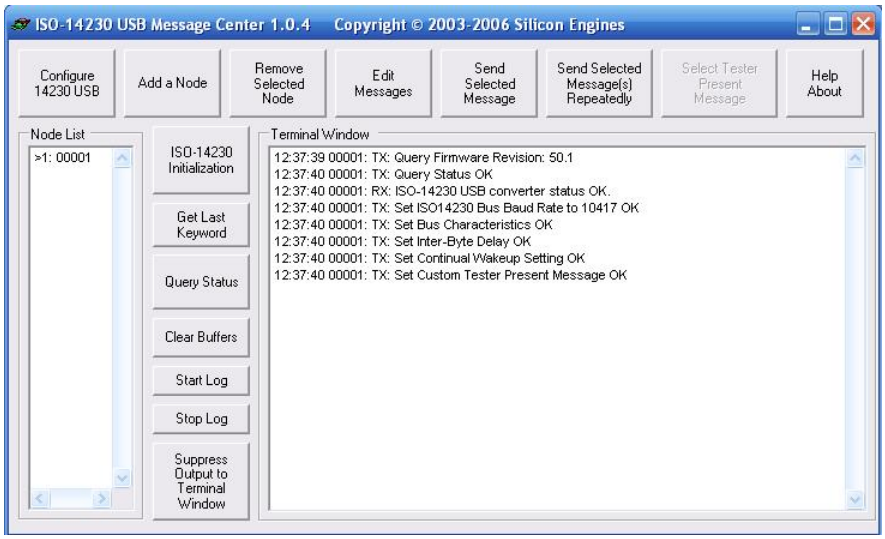


ADD A NODE SCREEN
FIGURE 18.1.2

On this screen, all recognized 14230/USB Converter devices will be listed, by serial number, in the drop-down box at the top of the window. You can select which serial numbered Converter to add, using the drop-down arrow button.

If the status of the connected 14230/USB Converters has changed—for example, a new Converter has been plugged in—then you can press the **Refresh List** button to update the list of recognized 14230/USB Converters in the drop-down list.

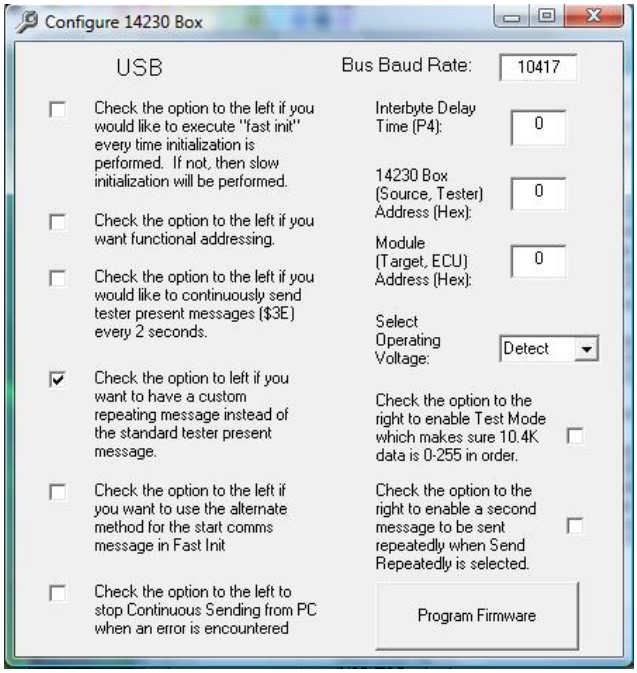
When the 14230/USB Converter that you want to add appears in the drop-down window, press **Add Selected Serial Number**. You should then see the result as in the picture below:



NODE LIST SHOWS ONE CONNECTED CONVERTER
FIGURE 18.1.3

18.2. CONFIGURE SYSTEM

The first step in using the **ISO-14230 USB Message Center** software is set up the PC software and the 14230/USB Converter. Click on the **Configure System** button. The following screen should appear.



CONFIGURATION SCREEN
FIGURE 18.2.1.

18.3. SELECTING THE INITIALIZATION MODE

The 14230/USB Converter handles initialization of the connected ECU automatically. But you must specify the initialization mode used by the ECU to which you are connecting—5 baud, or fast initialization. *(For technical details, see Part 15 above.)*

To select fast initialization, use the PC mouse to point to the **FAST INIT** box in the **Configuration Screen**. Click on the mouse to turn the check mark on and off. When this box is checked, the 14230/USB Converter will use fast initialization. Otherwise it will use 5-baud initialization.

You can also check the box for functional addressing versus physical addressing. Another box sets the alternate method of initialization. The selected options should match the options that the target ECU will recognize.

Don't forget to set the source and target addresses as well. These should match the parameters that the ECU expects for proper communications.

18.4. SPECIFYING THE ISO-14230 DATA RATE

Normally the 14230/USB Converter will communicate with the ECU at the standard data rate of 10,400 bps, once initialization has been completed [ISO-14230, §5.2.4.2.3.1]. However the 14230/USB Converter also allows you to specify non-standard data rates. Enter the desired baud rate in the box provided. Valid entries are from 1,000 to 115,200 bps.

18.5. SPECIFYING THE INTER-BYTE DELAY

Normally the 14230/USB Converter will communicate with the ECU with 5 millisecond delays between bytes. However the 14230/USB Converter also allows you to specify other inter-byte delays. Enter the desired delay time, in milliseconds, in the box provided. Valid entries are from 0 to 51 (milliseconds). *For details, see Section 13.9.*

18.6. SPECIFYING THE ADDRESS MODE

The ISO-14230 protocol offers the option of routing packets by ECU address (physical) or by message contents (functional). The default mode is physical addressing. To send data using functional addressing, check the option box for functional addressing.

When the 14230/USB Converter has been configured for 5-baud initialization and for physical addressing, the 14230/USB Converter will transmit the ECU address using 7O1 data format, during the 5-baud initialization sequence.

When the 14230/USB Converter has been configured for 5-baud initialization and for functional addressing, the 14230/USB Converter will transmit the ECU address using 8N1 data format, during the 5-baud initialization sequence. *(For details, see Section 15.6.)*

18.7. SPECIFYING THE OPERATING VOLTAGE

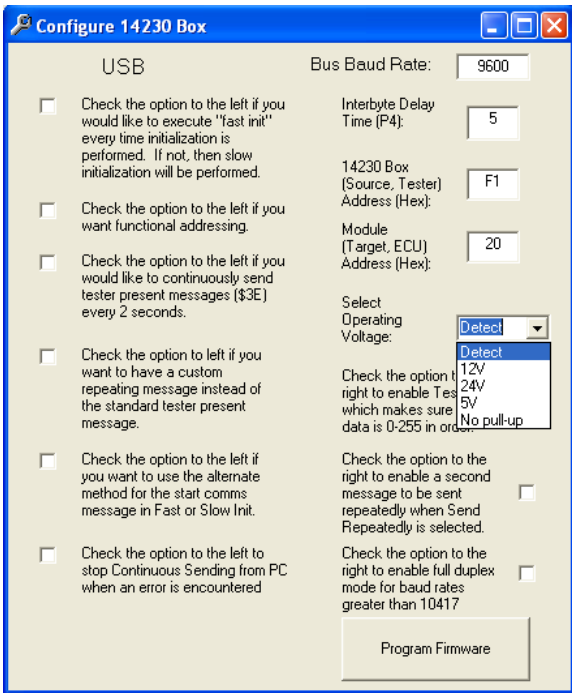
The 14230/USB Converter has been designed to operate with automotive vehicles using 12-volt batteries, as well as truck and off-the-road vehicles using 24-volt supplies.

The Model 9010 also has two additional operating modes: ***5V mode***, useful for industrial one-wire 5V serial data buses; and ***no pull-up mode***, in which the Model 9010 does not apply any pull-up resistor at all on the bus.

The Model 9010 has internal switches that change its internal K-line and L-line load resistors to match the selected mode. *(See Section 13.1 for details.)*

In the box marked **Select Operating Voltage**, select one of five options: **Auto Detect** (the 14230/USB Converter detects the voltage and switches automatically), **12V**, **24V, 5V,** or **no pull-up**.

The **No pull-up** option is recommended for **Listener Mode** *(see Section 12.5)*.



PULL-DOWN MENU FOR SELECTING OPERATING VOLTAGE
FIGURE 18.7.1.

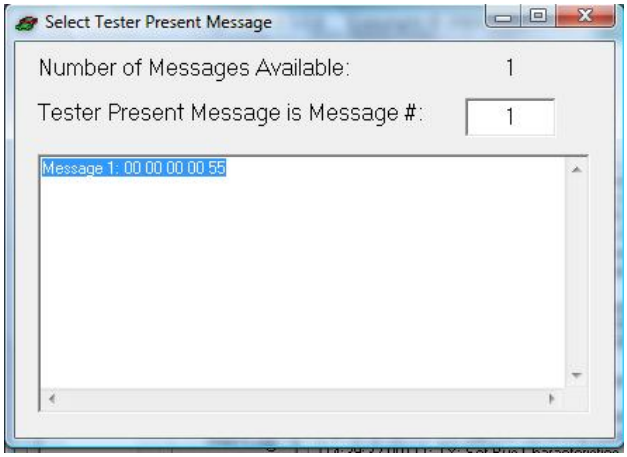
18.8. CONTINUOUS MESSAGE TRANSMISSION OPTIONS

To send a periodic transmission of a TESTER PRESENT message, then click **Send TESTER PRESENT messages every 2 seconds.** The default message is the standard $3E TESTER PRESENT message.

When you select **Send Message Continuously,** the 14230/USB Converter sends the message that has been selected at a periodic rate.

If you check the box on the Configuration screen that says **enable a second message to be sent repeatedly**, the message number just before the selected message number will also be sent at a periodic rate, at the same repetition rate as the selected message. This allows two messages to be sent periodically.

If the message being transmitted is not the one you need, then you need to click on **custom repeating message**. Then you can click on the button **Select Tester Present Message** from the main screen, which will give you a screen like the one below:



SELECT TESTER PRESENT MESSAGE
FIGURE 18.8.1.

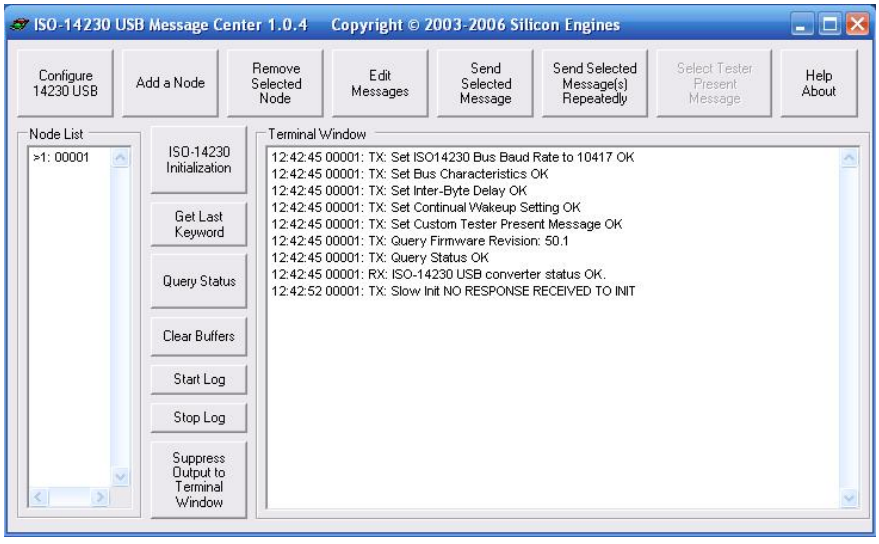
The message you select on this screen will be sent by the Message Center software to the 14230/USB Converter. The Converter will use that message as its TESTER PRESENT message every two seconds, as long as that feature is checked in the Configuration screen.

You can also set the Message Center to send a steadily increasing sequence of values to test a transmission link. Check the box **Enable Test Mode**. Note that you can also set up a 14230/USB Converter to RECEIVE a steadily increasing sequence of values in order, and report any mismatches, by also checking the same box.

18.9. SEND INITIALIZATION SEQUENCE

Once you have set up the system, you can click on the **Send Initialization Sequence** button on the **Main Message Center** screen. You should see the message go out on the **Terminal Window**, and after that, a message indicating that the ECU has responded.

If the ECU does not respond, you will see an error message explaining what type of error occurred. An example is shown below:



SCREEN SHOWING INITIALIZATION FAILURE
FIGURE 18.9.1.

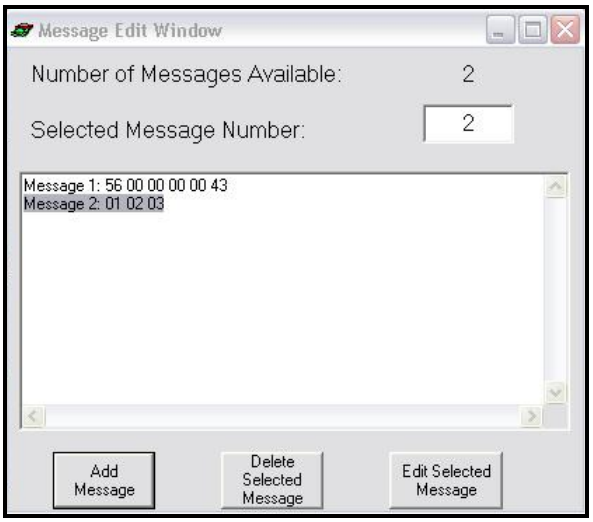
18.10. TROUBLESHOOTING INITIALIZATION PROBLEMS

- **FAST INIT/5-baud:** If the ECU does not respond using fast initialization, try 5-baud (slow) initialization.
- **Addressing mode:** Makes sure that the addressing mode—physical or functional—matches what the ECU expects for the Start Communications sequence.
- **Alternate method:** Try checking the **alternate method for fast or slow initialization** box and see if that works. Certain models of ECU use an alternate method of initialization.
- **ECU and Tester address:** Verify that you have the correct ECU address. Also make sure you have defined a tester address that the ECU recognizes as a valid tester address.
- **Wake-up from sleep:** Sometimes ECUs are designed to go to sleep after a short period, and wake up only when an operator control is actuated. Try pushing buttons on the ECU to make sure it is awake.
- **L line:** Some ECUs require connecting the L line to a wakeup line on the ECU.
- **Wrong Bus Voltage Setting:** Make sure that the operating bus voltage selection is correct for this ECU and this bus voltage. If in doubt, select auto-detect for operating bus voltage.

19. MESSAGE CENTER—MESSAGE CONTROL

19.1. GENERATING A CUSTOM MESSAGE

To generate a custom message to send to the ECU, bring up the **Main Message Center** screen. Click on the **Select/Edit Messages** button. The **Message Edit Window** will appear.

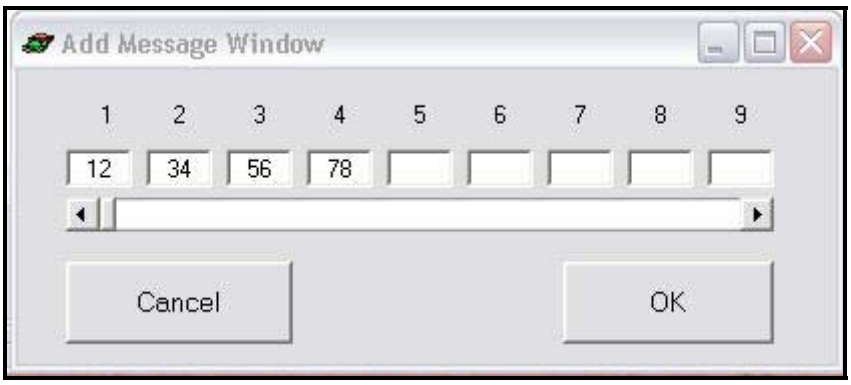


MESSAGE EDIT WINDOW
FIGURE 19.1.1.

To select a message, position the mouse cursor over the message you wish to select, and push the mouse button.

19.2. ADD MESSAGE WINDOW

To add a new message, click on the **Add Message** button in the **Message Edit Window** *(see previous section)*. The **Add Message Window** will appear. Enter a hex value in each of the byte boxes to make up a message.



ADD MESSAGE WINDOW
FIGURE 19.2.1.

19.3. SEND SELECTED MESSAGE

Once you have entered a custom message, you can send that message by clicking on the **Send Selected Message** button on the **Main Message Center** screen. You should see the message go out on the **Terminal Window**.

19.4. CLEAR TERMINAL WINDOW

You can erase the old messages appearing in the **Terminal Window** by pressing the **Clear Terminal Window** button on the **Main Message Center** screen. This makes room to view new messages.

20. MESSAGE CENTER—ADVANCED OPTIONS

20.1. REQUEST STATUS

In the **Main Message Center** screen**,** check the box **Request Status** to cause the PC to send a request to the 14230/USB Converter as to the current status of the system. You should see a response on the **Terminal Window**. *(For a list of response codes and error messages, and the likely diagnosis of underlying faults, see Section 14.1.)*

20.2. REQUEST LAST KEYWORD/SYSTEM ID

In the **Main Message Center** screen**,** check the box **Request Last Keyword** to cause the PC to send a request to the 14230/USB Converter to return the most recent keyword that it has received from the connected ECU. You should see a response in the **Terminal Window**. This can be used in debugging ECU communications. *(See Section 14.1.)* If alternate method of slow initialization is selected, this button will also request the last System ID as well (*see Section 15.9*).

20.3. CLEAR BUFFERS IN 14230/USB CONVERTER

In the **Main Message Center** screen**,** check the box **Clear ISO-14230 Box Buffers** to cause the PC to send a request to the 14230/USB Converter to clear its internal data storage buffers. This may be necessary in the event of an error in communications with a connected ECU.

20.4. TESTER PRESENT MESSAGES

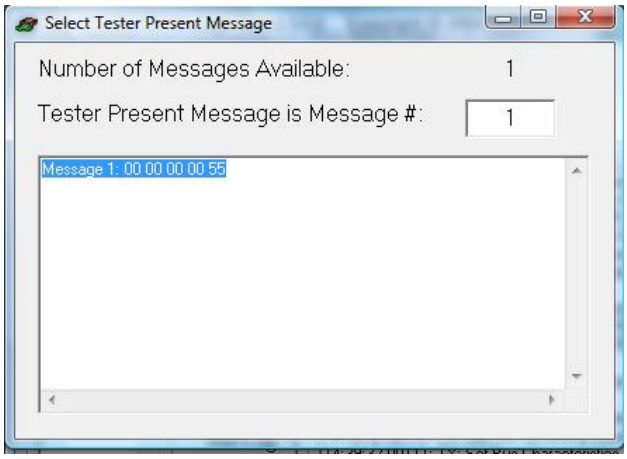
In the **Configuration Screen,** check the box for **TESTER PRESENT Messages** to send TESTER PRESENT messages (hex 3E) every two seconds to the ECU. These messages are typically generated by a diagnostic tester to keep the addressed ECU awake. Many ECUs automatically disconnect and enter low-power sleep mode after a time-out period if they do not receive diagnostic messages.

Note that if 3E is not a valid TESTER PRESENT message for your system, then you can define and run a custom message. *See the next section below.*

20.5. SENDING A CUSTOM TESTER PRESENT MESSAGE

In the **Configuration Screen,** check the box for **Custom TESTER PRESENT Message** to send your own message (in place of the default **TESTER PRESENT message**) periodically to the ECU. Use the **Message Edit Window** to control custom messages, and the **Add Message Window** to add new messages, as previously described.

Next you need to specify which message will be repeated. Click on the **Select TESTER PRESENT Message** button on the **Main Message Center** screen. This brings up the following screen:



SELECTING THE REPEATED MESSAGE
FIGURE 20.5.1.

Specify the message you want to be repeated by positioning the mouse cursor over the message you want, and clicking the mouse button. Alternately enter the message number in the box provided.

20.6. GENERATING AN ASCENDING TEST SEQUENCE

Check the box for **Test Mode** in the **Configuration Screen** to enter a mode in which the PC and 14230/USB Converter check for a test sequence from the connected ECU. Intended for use in communications testing, this sequence expects the ECU to generate messages in sequential ascending order: 00, 01, 02, …., FE, FF, 00, 01, 02, etc. The sequence can start at any value, but the bytes that follow must be in order.

If a byte is missed, the program will stop, and an error message will appear in the Terminal Window.

20.7. GENERATING A LOG FILE

There are two boxes on the Main Message Center screen that allow you to generate a log file on your PC, in order to record the messages that appear in the Terminal Window.

To start a log file, click on the **Start Log** button. You will be prompted to specify the name of the log file on your PC. Messages will be stored in human-readable ASCII text format, with new line characters after each line.

To stop the log file, click on the **Stop Log** button.

21. SOFTWARE UPGRADES

21.1. AVAILABILITY

Silicon Engines updates the firmware for the Model 9010 embedded microcontroller, as well as the PC-side software, from time to time. These upgrades implement bug fixes, and add new features requested by customers.

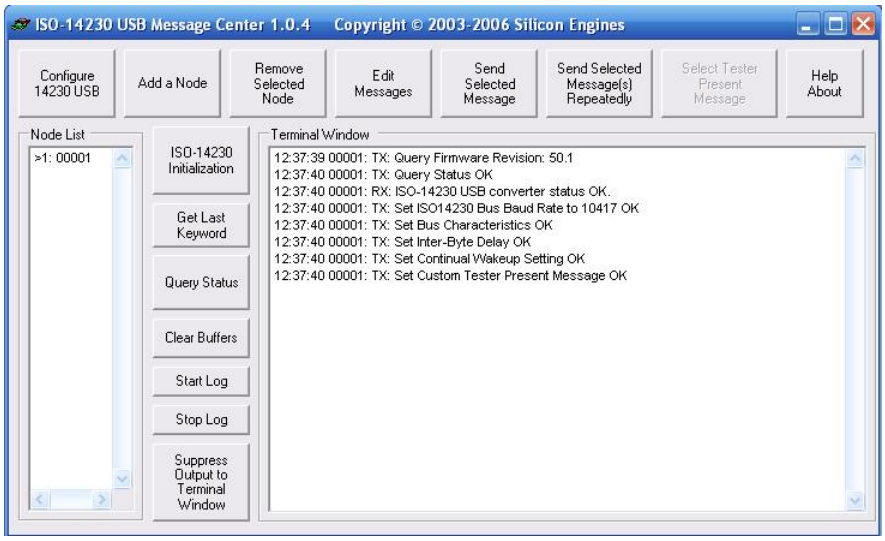
To check whether there is a new version of the firmware for your device, please go to:

www.siliconengines.net

Navigate to the PRODUCTS section; find the Model 9010 Converter; and find the link to the available software releases.

21.2. FIRMWARE VERSIONS

In order to determine which firmware version is running on your Model 9010, use the QUERY STATUS button in the Main Message Center screen.



QUERY STATUS BUTTON SHOWS FIRMWARE REVISION
FIGURE 21.2.1

In the example screen above, the first line in the Terminal Window indicates Firmware Revision 50.1.

The Model 9010 software version consists of two parts: major and minor. For example, version 50.1 indicates major version 50, minor version 1.

In normal operation, the major software version will be 50. Contact Silicon Engines to determine the latest minor version level.

If you see a major version of between 40 and 49, this indicates that the Model 9010 has lost its main program. Version 40-49 indicates that only the program bootloader is currently installed. You will need to upgrade the firmware, using the S19 file in the UPGRADE folder on the CD provided with the Model 9010. Or use a later S19 file from a newer CD downloaded from the Silicon Engines website.

21.3. UPGRADING

The firmware upgrade functionality for the embedded microcontroller within the Model 9010 is available through the Configuration Menu in the Message Center.

22. MODEL 9010 VERSUS MODEL 9009

22.1. MODEL 9010 REPLACES MODEL 9009

Silicon Engines has introduced the Model 9010 14230/USB Converter as an upgrade to our earlier Model 9009 14230/USB Converter.

The Model 9010 implements all of the key features of the Model 9009.

The Model 9009 is being phased out and will not be offered for sale once supplies have run out.

22.2. COMPARISON CHART

SPECIFICATION	MODEL 9010	MODEL 9009	MODEL 9010 IMPROVEMENTS
INTERNAL PROCESSOR	72 MHZ ARM CORTEX M3	TWO 8-BIT PROCESSORS	ADDITIONAL PROCESSING POWER
14230 K-LINE BUS SPEED	UP TO 115,200 BPS	UP TO 10,400 BPS	CAN BE USED FOR FIRMWARE DOWNLOADS AND TESTING
14230 K-LINE BUS VOLTAGE	5V, 12V, 24V	12V, 24V	COMPATIBILITY WITH INDUSTRIAL APPLICATIONS USING SINGLE-WIRE 5-VOLT BUS
USB SPEED	FULL-SPEED, 12 MEGABITS PER SECOND	LOW-SPEED, 1.5 MEGABITS PER SECOND	FASTER USB PERFORMANCE
WINDOWS SUPPORT	WINDOWS 7, VISTA, XP, 2000	WINDOWS XP, 2000	COMPATIBLE WITH NEARLY ALL EXISTING WINDOWS COMPUTERS

MODEL 9010 VERSUS MODEL 9009
FIGURE 22.2.1.

23. REFERENCES

23.1. LATEST VERSIONS

The documents shown below are the latest revisions when this document was last revised. Please be sure to check with the issuing standards organization for updates and revisions.

23.2. ISO STANDARDS

To obtain copies of International Standards Organization documents in the USA, contact www.ansi.org. To obtain copies outside the USA, contact www.iso.org. Standards can be ordered directly from the ISO, or from an affiliated standards organization in each country (such as ANSI in the USA).

ISO-9141, Road Vehicles—Diagnostic Systems—Requirements for Interchange of Digital Information, 1989-10-1.

ISO-9141-2, *Road Vehicles—Diagnostic Systems—Part 2: CARB Requirements for Interchange of Digital Information,* 1994-02-01.

ISO-9141-2, *Road Vehicles—Diagnostic Systems—Part 2: CARB Requirements for Interchange of Digital Information, Amendment 1,* 1996-12-01.

ISO-9141-3, *Road Vehicles—Diagnostic Systems—Part 3: Verification of the Communication Between Vehicle and OBD II Scan Tool,* 1998-12-15.

ISO-14230-1, *Road Vehicles—Diagnostic Systems—Keyword Protocol 2000—Part 1: Physical Layer*, 1999-03-15.

ISO-14230-2, *Road Vehicles—Diagnostic Systems—Keyword Protocol 2000—Part 2: Data Link Layer*, 1999-03-15.

ISO-14230-3, *Road Vehicles—Diagnostic Systems—Keyword Protocol 2000—Part 3: Application Layer*, 1999-03-15.

ISO-14230-4, *Road Vehicles—Diagnostic Systems—Keyword Protocol 2000—Part 4: Requirements for Emission-Related Systems*, 2000-06-01.

23.3. SAE STANDARDS

To obtain copies of SAE standards, contact www.sae.org. The first publication shown is a paperback book containing the separate standards listed below, plus many additional related standards, current as of its publication date.

1. SAE HS-3000, SAE On-Board Diagnostics for Light and Medium Duty Vehicles Standards Manual, 2003 edition.

2. J1962, *Diagnostic Connector*, December 2001.

3. J1850, *Class B Data Communications Network Interface*, May 2001.

24. REVISION HISTORY

24.1. REVISION E

1. Section 4.5: Added Listener Mode to the feature list.
2. Section 9.7: Added specifications for bus load resistor in 5V mode. Added minimum bus load resistor specification in no pull-up mode.
3. Section 12.4: Added section describing typical use of Model 9010 as 14230 bus host.
4. Section 12.5: Added section describing use of Model 9010 in Listener Mode.
5. Section 18.7: Added language clarifying that the no pull-up option is recommended for Listener Mode. Added picture of pull-down menu for selecting the operating voltage.

24.2. REVISION D

Figure 22.2.1: Corrected data in comparison chart between Model 9010 and Model 9009.

24.3. REVISION C

First general release.

24.4. REVISIONS B, A

Obsolete documents applying to Model 9010 beta prototypes.

☑