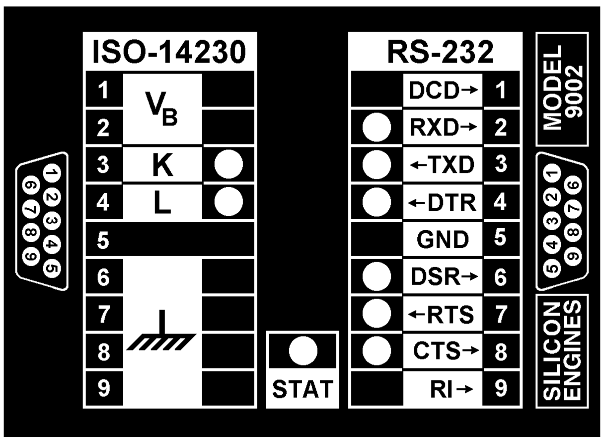
14230/RS-232 CONVERTER

MODEL 9002

USER'S GUIDE



COMMENTS

We would appreciate receiving corrections and suggestions regarding this document and the product it describes. Please email to 14230@siliconengines.net

SILICON ENGINES

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

CONTENTS

1. OVERVIEW

1.1. INTRODUCTION

This document is the User's Guide for the **14230/RS-232 Converter**, **Model 9002,** a compact electronic device that allows a personal computer to connect to an automotive diagnostic data link compatible with ISO-14230. 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. In addition to the Model 9002 14230 Converter, Silicon Engines offers a **9141 Converter, Model 9001**. For details on the Model 9001, check our web site, www.siliconengines.net.

1.2. 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.

1.3. FUNCTIONS

- **Level conversion:** Converts signals between ISO-14230 levels and RS-232 (CCITT V.24) levels, for connection to a personal computer.
- **Vehicle battery power:** Compatible with both 12-volt and 24-volt battery systems.
- **Speed conversion:** Interfaces to the ECU at the ISO-14230 data rate. The ISO-14230 data rate is typically 10,400 bits per second, but any data rate between 1,000 bps and 10,417 bps can be selected. Interfaces to the PC at 19,200 bits per second, avoiding the need for the PC to handle non-standard baud rates.
- **Initialization:** The Model 9002 14230 Converter handles both 5-baud initialization, and fast initialization. Avoids the need to use special PC communications libraries to handle 5-baud data, or to use precision PC timing routines to handle fast initialization timing.
- **Duplexing:** Converts the half-duplex ISO-14230 line, to standard full-duplex RS-232 signals. Avoids the need for the PC to deal with echoed characters.
- **Signal indicators:** Provides eight two-color LEDs to show the states of all significant signal lines.

2. ISO STANDARDS

2.1. 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 16 for a list of references.

2.2. 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].

2.3. ISO-14230

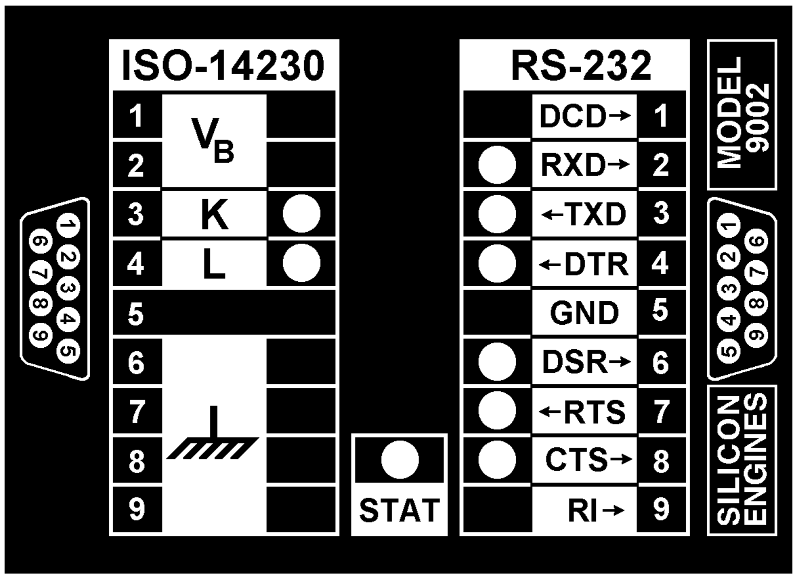
The 1999 standard [ISO-14230], **Keyword Protocol 2000**, is based on [ISO-9141], but adds many additional details:

- **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.

3. ENCLOSURE

3.1. TOP PANEL

The 14230 Converter is housed in a black plastic enclosure. Eight 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 and RS-232 connectors.



MODEL 9002 TOP PANEL
FIGURE 3.1.1.

3.2. 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 3.2.1.

4. CONNECTORS

4.1. 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 4.1.1.

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. 3.1.1)*.

4.2. ISO-14230 CABLE TO SPECIFIC ECU

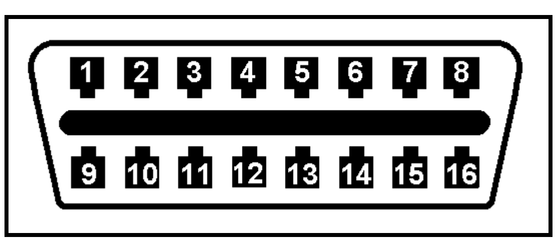
Users who intend to apply the 14230 Converter in engineering development work, or production testing—with a specific ECU—must construct a special cable to connect from the ECU to the 14230 Converter. Or contact Silicon Engines for assistance.

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. 3.1.1). (See Part 5 below for more information on VBATT and GND.)*

4.3. 14230-OBD CABLE

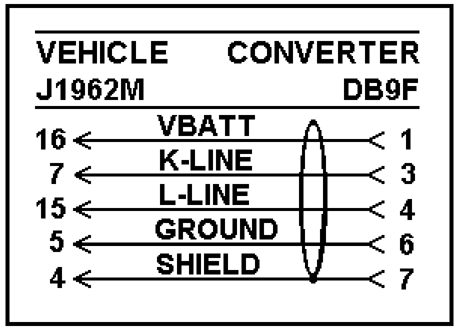
Users who intend to use the 14230 Converter in-vehicle can purchase an optional cable from Silicon Engines. Specify the **14230-OBD Cable**. The standard cable length is 10 meters (32.8 feet).

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 a standard automotive OBD (on-board diagnostic) port, as defined in [J1962]. This diagnostic connector is typically located within one meter (~3 feet) of the steering wheel, in a location that does not require tools for access.



IN-VEHICLE ON-BOARD DIAGNOSTIC CONNECTOR
FIGURE 4.3.1.

At the vehicle side, the 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:



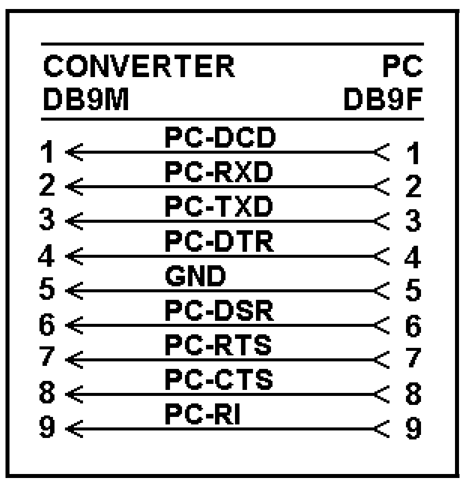
CABLE FROM VEHICLE OBD PORT TO 14230 CONVERTER
FIGURE 4.3.2.

4.4. RS-232 CONNECTOR

The connector at the right of the 14230 Converter is a DB9F type, designed for connection to the COM1, COM2, or COM3 port of a PC. Signal wires are shown on the top panel artwork *(Fig. 3.1.1)*.

Connector polarities and signal names have been chosen for compatibility with the RS-23.2 (CCITT V.24/V.28) serial port of a recent model IBM®-compatible PC. PC-AT and later computer models typically provide a DB9M connector at their serial ports.

Typically the 14230 Converter connects to the PC over a DB9 extender cable about 6 feet (2 M) long. A DB9M connector plugs into the 14230 Converter at one end of the cable, and a DB9F connector plugs into the PC at the other end. This cable provides straight-through wiring—pin 1 connects to pin 1, pin 2 to pin 2, etc. A cable of this description is provided with each 14230 Converter.



STRAIGHT-THROUGH RS-232 CABLE
FIGURE 4.4.1.

Notice: recommended practice is to connect the RI line on the 14230 Converter to the RI line on the PC's RS-232 serial port. To prevent interference with the functioning of the 14230 Converter, do not connect its RI signal in any other manner.

4.5. 25-PIN SERIAL PORT

If your computer provides a 25-pin DB25F RS-232 serial port connector, it will be necessary to install an adapter with a DB25M plug on one side, and a DB9M plug on the other, to adapt the computer to the 14230 Converter. This type of adapter is commonly sold to adapt serial mouse devices to 25-pin ports, and should be readily available from the closest computer store—or contact Silicon Engines for assistance.

5. POWER REQUIREMENTS

5.1. 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, it 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.

5.2. INPUT VOLTAGE AND CURRENT

The 14230 Converter contains built-in power supply circuitry that generates needed 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 internally for digital logic and the LED indicators, as well as ±10 VDC for the RS-232 interface.

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		65		MA	VBATT=+13.8 VDC, K-LINE IDLE
		42		MA	VBATT=+27.6 VDC, K-LINE IDLE

SUPPLY POWER SPECIFICATIONS
FIGURE 5.2.1.

5.3. 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.

5.4. 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.

5.5. 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.

6. ISO-14230 K-LINE

6.1. 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.

6.2. 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 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 the 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 6.2.1

The typical ISO-14230 high-speed data rate is 10,400 bps, not a standard speed for RS-232. A microcontroller within the 14230 Converter performs data rate conversion, communicating with the PC over the RS-232 interface at 19,200 bps, and at up to 10,400 bps over the ISO-14230 K-line.

6.3. K-LINE LOGIC WITHIN ECU

Within a typical ISO-14230 ECU, the K-line transmitter is an open-collector transistor 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 open-collector 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.

The transmitter and receiver functions may also be embodied in an ISO-9141 transceiver, such as the Motorola MC33199, ST Microelectronics L9637, 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.

6.4. 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.

6.5. 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].

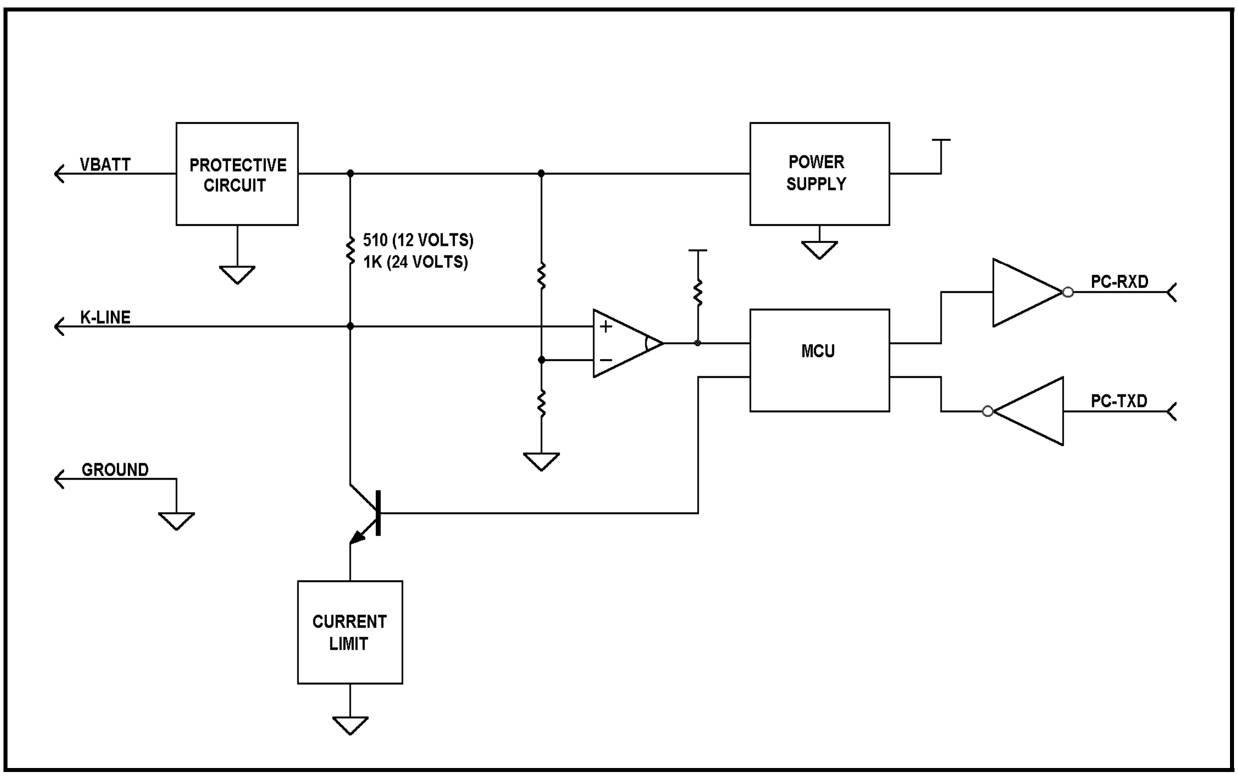
A simplified block diagram for the K-line logic within the 14230 Converter appears below.

The microcontroller (MCU) does baud-rate conversion. It sends and receives data over the RS-232 TXD and RXD lines at 19,200 bps, and sends and receives data over the ISO-14230 K-line at the ISO-14230 high-speed data rate, typically 10,400 bps.

The MCU also converts half-duplex K-line data to full-duplex RS-232. Specifically, when the PC sends a data byte over the PC-TXD line, to the 14230 Converter, the MCU forwards it onto the K-line. Because the K-line is half-duplex, the K-line comparator will see this transmitted data. The MCU filters out echoed bytes 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 14230 Converter 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].



K-LINE LOGIC WITHIN 14230 CONVERTER
FIGURE 6.5.1.

6.6. K-LINE ELECTRICAL SPECIFICATIONS

PARAMETER	MINIMUM	TYPICAL	MAXIMUM	CONDITIONS/COMMENTS
OPERATING VOLTAGE RANGE, 12 VOLT MODE	+8 VDC	+13.8 VDC	+16 VDC	12 VOLT MODE
OPERATING VOLTAGE RANGE, 24 VOLT MODE	+16 VDC	+27.6 VDC	+32 VDC	24 VOLT MODE
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%
MAXIMUM VOLTAGE ON K-LINE		34 VDC		VOLTAGE CLAMP ON K-LINE OUTPUT
MAXIMUM SINK CURRENT, OUTPUT LOW		58 mA		CURRENT-LIMITED OUTPUT TRANSISTOR

K-LINE ELECTRICAL SPECIFICATIONS
FIGURE 6.6.1

6.7. SELECTING 12-VOLT VERSUS 24-VOLT MODE

The 14230 Converter contains K-line and L-line load resistors for both 12-volt and 24-volt operating modes. At power up, 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 6.7.1

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 resistors.

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 Sec. 9.5 for details.*

7. ISO-9141 L-LINE

7.1. 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 11.)*

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.

7.2. 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.

7.3. 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.

8. LAMP FUNCTIONS AND SIGNAL FLOW

8.1. STATUS LAMP

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

LED PATTERN	SIGNAL LINE CONDITION
STEADY GREEN	CONVERTER OPERATING NORMALLY
BLINKING	CONVERTER MCU ERROR
OFF	CONVERTER NOT OPERATING PROPERLY, OR IN PROGRAM MODE

STATUS LAMP PATTERNS
FIGURE 8.1.1.

At power-up, the MCU within the 14230 Converter does a quick self-diagnostic test. A steady STAT lamp indicates that the MCU has passed the test. If the lamp is blinking or off, please contact Silicon Engines for assistance.

8.2. TWO-COLOR LAMPS

Eight two-color indicators on the top panel of the 14230 Converter show the status of key RS-232 and ISO-14230 signal lines. Each lamp glows either green or red whenever the device is powered up.

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 8.2.1.

8.3. ISO-14230 K AND L LINES

Two LED indicators show the voltage levels on the ISO-14230 K and L lines.

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 8.3.1.

8.4. RS-232 SIGNAL LINES

Six LED indicators show the voltage levels on key RS-232 signal lines. The names of the RS-232 signals on the 14230 Converter top panel are from the point of view of the connected personal computer.

CONN. PIN	SIGNAL NAME	DATA DIR.	LAMP COLOR	TYP. LEVEL	SIGNAL FUNCTION
1	DCD	CONV→PC	*(NONE)*	+10 V	DATA CARRIER DETECT, HIGH WHEN POWER IS ON
2	RXD	CONV→PC	GREEN	+10 V	START BIT; LOGIC 0 (SPACE)
			RED	-10 V	IDLE LINE; LOGIC 1 (MARK); STOP BIT
3	TXD	CONV←PC	GREEN	+10 V	START BIT; LOGIC 0 (SPACE)
			RED	-10 V	IDLE LINE; LOGIC 1 (MARK); STOP BIT
4	DTR	CONV←PC	GREEN	+10 V	ON=RUN MODE
			RED	-10 V	OFF=INITIALIZATION MODE
5	GND	---	*(NONE)*	0 V	SIGNAL COMMON
6	DSR	CONV→PC	GREEN	+10 V	14230 COMMUNICATIONS READY
			RED	-10 V	CONVERTER REQUESTS TO BE INITIALIZED, OR CONVERTER HAS DETECTED ERROR
7	RTS	CONV←PC	GREEN	+10 V	REQUEST TO SEND, FLOW CONTROL, PC READY TO RECEIVE FROM CONVERTER
			RED	-10 V	PC NOT READY TO RECEIVE FROM CONVERTER
8	CTS	CONV→PC	RED	+10 V	DATA SET READY, FLOW CONTROL, CONVERTER READY TO RECEIVE FROM PC
			GREEN	-10 V	CONVERTER NOT READY TO RECEIVE FROM PC
9	RI		(NONE)	0 V	RING INDICATOR, NOT USED FOR ISO-14230 COMMUNICATIONS , CONNECT ONLY TO PC-RI

RS-232 SIGNALS
FIGURE 8.4.1.

8.5. RXD: SERIAL DATA FROM CONVERTER TO PC

The RXD line leads to the RECEIVE DATA input of the PC's RS-232 serial port. Voltage levels are ±10 volts nominal. Data speed is fixed at 19,200 bps. Data format is 8N1—one start bit, eight data bits, no parity bit, and one stop bit.

8.6. TXD: SERIAL DATA FROM PC TO CONVERTER

The TXD line originates from the TRANSMIT DATA output of the PC's RS-232 serial port. Voltage levels are typically ±10 volts nominal. Data speed is fixed at 19,200 bps. Data format is 8N1—one start bit, eight data bits, no parity bit, and one stop bit.

8.7. DTR: PC INITIALIZATION REQUEST

The PC uses its DTR (Data Terminal Ready) line as a handshake signal to the 14230 Converter to control whether the Converter is in **Initialization Mode** or **Run Mode**.

When the PC sets DTR low, it is requesting the 14230 Converter to enter **Initialization Mode**. The Converter should respond by setting DSR low. When both DTR and DSR are low, the Converter is in Initialization Mode. *See Part 9 below for details on initialization commands.*

When both DTR and DSR are high, the 14230 Converter is in **Run Mode**. The Converter treats 19,200 bps data from the TXD line as data to be transmitted to the ECU at the ISO-14230 high-speed data rate, typically10,400 bps. Responses received from the ECU at 10,400 bps (typical) over the K-line are sent to the PC over the RXD line at 19,200 bps.

8.8. DSR: CONVERTER INITIALIZATION REQUEST

The 14230 Converter uses the DSR (DATA SET READY) line as a handshake signal to the PC to control whether the Converter is in **Initialization Mode** or **Run Mode**.

The Converter sets DSR low when the system is first powered up, and the system has not yet been initialized. It may also set DSR low when a protocol error indicates the need for configuration. The PC should respond by setting DTR low. When both DTR and DSR are low, the Converter is in Initialization Mode. *See Part 9 below for details on initialization commands.*

When both DTR and DSR are high, the 14230 Converter is in **Run Mode**.

8.9. RTS: PC FLOW CONTROL

The PC can optionally use the RTS (REQUEST TO SEND) line for flow control. The PC sets RTS high to indicate that it is prepared to receive RS-232 data from the 14230 Converter. The PC sets RTS low to cause the 14230 Converter to pause in sending data to the PC.

The PC serial port should be configured for hardware flow control. RTS should be set ON except when desired to pause incoming data from the 14230 Converter.

8.10. CTS: CONVERTER FLOW CONTROL

The 14230 Converter uses the CTS (REQUEST TO SEND) line for flow control. The 14230 Converter sets CTS high to indicate that it is prepared to receive RS-232 data from the PC. The 14230 Converter sets CTS low to cause the PC to pause in sending data to the 14230 Converter.

The PC serial port should be configured for hardware flow control. The 14230 Converter will set CTS ON except when desired to pause incoming data from the PC.

8.11. RI: FACTORY TEST ONLY

The 14230 Converter does not use the RI (RING INDICATOR) line during normal operations. This line is an input to the PC, typically used by an external modem to signal that there is an incoming call. The RI line is used by Silicon Engines for factory test purposes.

Notice: recommended practice is to connect the RI line on the 14230 Converter to the RI line on the PC's RS-232 serial port. To prevent interference with the functioning of the 14230 Converter, do not connect its RI signal in any other manner.

8.12. LAMP PATTERNS, SYSTEM IDLE

ISO-14230 LAMPS				RS-232 LAMPS			
LAMP	**COLOR**	**STATE**	**LEVEL**	**LAMP**	**COLOR**	**STATE**	**LEVEL**
K	GREEN	IDLE	VBATT	RXD	RED	IDLE	-10 VDC
L	GREEN	IDLE	VBATT	TXD	RED	IDLE	-10 TO -15 VDC
				DTR	RED	INIT MODE	-10 TO -15 VDC
				DSR	RED	INIT MODE	-10 VDC
				RTS	RED	NOT READY	-10 TO -15 VDC
				CTS	RED	NOT READY	-10 VDC

LAMP PATTERNS, ISO-14230 SYSTEM IDLE
FIGURE 8.12.1.

The above lamp pattern should appear when the system is idle. The 14230 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 RS-232 signal lines from the PC are all at idle levels. The RS-232 lamps are all red, while the ISO-14230 K lamp is green.

8.13. 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.

8.14. LAMP PATTERNS, ISO-14230 INITIALIZATION

ISO-14230 LAMPS				RS-232 LAMPS			
LAMP	COLOR	STATE	LEVEL	LAMP	COLOR	STATE	LEVEL
K	GREEN WITH RED FLASHES	1/0	VBATT/ GND	RXD	RED WITH GREEN FLASHES	1/0	INIT SEQUENCE
L	GREEN WITH RED FLASHES	1/0	VBATT/ GND	TXD	RED WITH GREEN FLASHES	1/0	INIT SEQUENCE
				DTR	RED	INIT	-10 TO -15 VDC
				DSR	RED	INIT	-10 VDC
				RTS	GREEN	READY	+10 TO +15 VDC
				CTS	GREEN	READY	+10 VDC

LAMP PATTERNS, INIT MODE
FIGURE 8.14.1.

During ISO-14230 initialization, the DTR lamp will be RED, indicating that the PC has commanded **Initialization Mode**. The DSR lamp will also be RED, indicating that the 14230 Converter is in initialization mode.

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

When initialization is complete, the DTR and DSR lamps will switch to green.

8.15. LAMP PATTERNS, RUN MODE

ISO-14230 LAMPS				RS-232 LAMPS			
LAMP	COLOR	STATE	LEVEL	LAMP	COLOR	STATE	LEVEL
K	GREEN WITH RED FLASHES	1/0	VBATT/ GND	RXD	RED WITH GREEN FLASHES	1/0	HI-SPEED DATA
L	GREEN	1/0	VBATT	TXD	RED WITH GREEN FLASHES	1/0	HI-SPEED DATA
				DTR	GREEN	RUN	+10 TO +15 VDC
				DSR	GREEN	RUN	+10 VDC
				RTS	GREEN	READY	+10 TO +15 VDC
				CTS	GREEN	READY	+10 VDC

LAMP PATTERNS, RUN MODE
FIGURE 8.15.1.

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

9. INITIALIZATION MODE

9.1. INITIALIZATION ALTERNATIVES

To communicate with an ECU, the diagnostic tester must normally complete an initialization sequence. The 14230 Converter supports three initialization alternatives:

1. **5-Baud Initialization:** Low-speed initialization at 5 baud, as originally provided by [ISO 9141].
2. **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].
3. **Fast Initialization:** High-speed initialization, as newly defined by [ISO-14230-2].

The 14230 Converter handles all the details of initialization in response to commands received from the PC over the RS-232 interface.

The 14230 Converter can also be connected in parallel with an external diagnostic tester that performs the initialization sequence:

4. **Listener mode:** An external diagnostic tester does the initialization. *See Sec. 9.10 below.*

9.2. ENTERING INITIALIZATION MODE

To enter Initialization Mode, the PC turns off its DTR line. The RS-232 DTR line switches to a –10 to –15 VDC level. The DTR lamp on the 14230 Converter top panel will be red.

The 14230 Converter responds by turning off its DSR line (if it was not already off). The RS-232 DSR line goes to a –10 volt level. The DSR lamp on the 14230 Converter top panel will be red.

When both DTR and DSR are low, the 14230 Converter is in Initialization Mode. The PC sends commands to the 14230 Converter over the TXD line at 19,200 bps, 8N1 format. The 14230 Converter responds over the RXD line at 19,200 bps, 8N1 format.

9.3. INITIALIZATION COMMAND STRUCTURE

When in Initialization Mode, the PC sends commands in packets. The 14230 Converter responds to each command with a response packet.

Each command packet starts with a fixed five-byte header, as follows:

BYTE NO.	HEX VALUE	ASCII CHAR.
1	31H	1
2	34H	4
3	32H	2
4	33H	3
5	30H	0

INITIALIZATION COMMAND HEADER
FIGURE 9.3.1.

After the five-byte header, one or more command bytes are sent, with the specific values determined by the specific commands, as described further below.

9.4. CHECKSUM BYTES

At the end of each packet are two checksum bytes. The first checksum byte is calculated as the sum of the preceding bytes in the command packet, including all the bytes in the packet, including the five-byte header, but excluding the two checksum bytes themselves. The addition is performed modulo-256—in other words, any overflow beyond 255 decimal is discarded.

The second checksum byte is the bitwise inversion of the first checksum byte. The two checksum bytes added together will equal FF hex.

In the command charts below, the symbol **Σ** indicates the checksum byte, and the symbol **/Σ** indicates the inverted checksum byte.

9.5. SET OPERATING VOLTAGE

This command sets the operating voltage of the 14230 Converter for 12-volt or 24-volt mode. After the five-byte header, the sixth byte is the letter "V." The seventh byte selects between automatic, 12-volt, and 24-volt modes. Automatic mode is the default mode at power-up *(see Sec. 6.7 above)*.

BYTE NO.	HEX VALUE	ASCII CHAR.	COMMAND VALUE
6	56H	V	VOLTAGE MODE COMMAND
7	00H		SET TO AUTOMATIC MODE
	0CH		SET TO 12-VOLT MODE
	18H		SET TO 24-VOLT MODE
8	Σ		CHECKSUM BYTE
9	/Σ		INVERTED CHECKSUM BYTE

VOLTAGE MODE COMMAND
FIGURE 9.5.1.

9.6. SET ISO-14230 HIGH-SPEED BAUD RATE

This command is supported by 14230 Converter firmware Rev. 6 or later, and PC software revision 1.4 or later. Please contact Silicon Engines if you need assistance in updating older versions.

This command sets the high-speed data rate that will be used by the 14230 Converter to communicate with the ECU, after initialization is complete. Valid ranges are 1,000 to 10,417 bps. The standard data rate is 10,400 bps, as specified by [ISO-14230-2, §5.2.4.2.3.1]. The 14230 Converter powers up set to 10,400 bps, and it will use the standard data rate unless the PC sends it this command with another valid setting, for compatibility with ECUs that run at non-standard baud rates.

After the five-byte header, the sixth byte is the letter "B." The seventh and eighth bytes specify the baud rate (bits per second).

BYTE NO.	HEX VALUE	ASCII CHAR.	COMMAND VALUE
6	43H	B	BAUD RATE COMMAND
7	*		BAUD RATE, MS BYTE
8	*		BAUD RATE, LS BYTE
9	Σ		CHECKSUM BYTE
10	/Σ		INVERTED CHECKSUM BYTE
* SET TO MATCH HIGH-SPEED BAUD RATE REQUIRED			

BAUD RATE COMMAND
FIGURE 9.6.1.

9.7. SET INTER-BYTE DELAY (P4)

This command is supported by 14230 Converter firmware Rev. 9 or later, and PC software revision 1.5 or later. Please contact Silicon Engines if you need assistance in updating older versions.

This command sets the inter-byte delay—the time period between bytes that are transmitted by the 14230 Converter to communicate with the ECU. Valid ranges are 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 14230 Converter powers up set to 5 milliseconds, and it will use this value for the inter-byte delay unless the PC sends it this command with another valid setting. The purpose of the P4 time period is to accommodate slower ECUs with microcontrollers that may need extra time to respond to commands from the diagnostic tester.

After the five-byte header, the sixth byte is the letter "I." The seventh byte specifies the inter-byte delay, in milliseconds. Values greater than 51 will cause the command to be ignored.

BYTE NO.	HEX VALUE	ASCII CHAR.	COMMAND VALUE
6	49H	I	INTER-BYTE DELAY COMMAND
7	*		DELAY VALUE
8	Σ		CHECKSUM BYTE
9	/Σ		INVERTED CHECKSUM BYTE
* SET TO MATCH DELAY TIME REQUIRED			

INTER-BYTE DELAY COMMAND
FIGURE 9.7.1.

9.8. FAST INIT COMMAND

This command performs fast initialization of the attached ECU.

BYTE NO.	HEX VALUE	ASCII CHAR.	COMMAND VALUE
6	46H	F	FAST INIT COMMAND
7	50H	P	PHYSICAL ADDRESSING
	46H	F	FUNCTIONAL ADDRESSING
	70H	p (LOWER CASE)	PHYSICAL ADDRESSING, ALTERNATE METHOD
	66H	f (LOWER CASE)	FUNCTIONAL ADDRESSING ALTERNATE METHOD
8	*		SOURCE ADDRESS (PC, TESTER)
9	*		DESTINATION ADDRESS (ECU)
10	Σ		CHECKSUM BYTE
11	/Σ		INVERTED CHECKSUM BYTE
* SET TO MATCH VALUES SET BY ECU MANUFACTURER			

FAST INIT COMMAND
FIGURE 9.8.1.

For details on the fast initialization sequence, see Part 11.

If initialization is successful, the 14230 Converter will send a response packet to the PC with a response code of 4F hex, NO ERROR PRESENT *(see Sec. 10.2)*. The 14230 Converter will then raise DSR to a high level to indicate that initialization is complete. The PC should raise DTR to confirm the shift to Run Mode. If the PC does not raise DTR within 5.0 seconds, the 14230 Converter will deassert DSR and re-enter Initialization Mode.

9.9. 5-BAUD AND CARB INIT COMMANDS

These commands perform 5-baud or CARB initialization of the attached ECU.

BYTE NO.	HEX VALUE	ASCII CHAR.	COMMAND VALUE
6	35H	5	ISO-9141 5-BAUD INIT
	43H	C	CARB INITIALIZATION
7	50H	P	PHYSICAL ADDRESSING
	46H	F	FUNCTIONAL ADDRESSING
	70H	p (LOWER CASE)	PHYSICAL ADDRESSING, ALTERNATE METHOD
	66H	f (LOWER CASE)	FUNCTIONAL ADDRESSING ALTERNATE METHOD
8	*		SOURCE ADDRESS (PC, TESTER)
9	*		DESTINATION ADDRESS (ECU)
10	Σ		CHECKSUM BYTE
11	/Σ		INVERTED CHECKSUM BYTE
* SET TO MATCH VALUES SET BY ECU MANUFACTURER			

5-BAUD, CARB INIT COMMANDS
FIGURE 9.9.1.

For details on the 5-baud initialization sequence, see Part 11.

If initialization is successful, the 14230 Converter will send a response packet to the PC with a response code of 4F hex, NO ERROR PRESENT *(see Sec. 10.2)*. The 14230 Converter will then raise DSR to a high level to indicate that initialization is complete. The PC should raise DTR to confirm the shift to Run Mode. If the PC does not raise DTR within 5.0 seconds, the 14230 Converter will deassert DSR and re-enter Initialization Mode.

9.10. GO TO LISTENER MODE

This command is supported by 14230 Converter firmware Rev. 10 or later, and PC software revision 1.6 or later. Please contact Silicon Engines if you need assistance in updating older versions.

This command requests the 14230 Converter to enter Run Mode without performing any initialization sequence. This mode of operation is typically used in conjunction with an external diagnostic tester, which performs the initialization. After the PC sends this command, and receives the acknowledgment from the 14230 Converter, the PC waits for the DSR line to go high, indicating the shift to Run Mode. ***Note:*** If the 14230 Converter has been set to Run Mode during the time when an external diagnostic tester performs the initialization sequence, it may indicate a transient error condition, because it is expecting high-speed bus traffic. However the 14230 Converter will not interfere with the bus traffic.

BYTE NO.	HEX VALUE	ASCII CHAR.	COMMAND VALUE
6	47H	G	GO TO LISTENER MODE
7	Σ		CHECKSUM BYTE
8	$/\Sigma$		INVERTED CHECKSUM BYTE

GO TO LISTENER MODE COMMAND
FIGURE 9.10.1.

9.11. REQUEST FOR KEYWORD

This command requests the 14230 Converter to return the keyword that it received from the ECU during the most recent initialization sequence. *(See Part 11 on keywords.)*

BYTE NO.	HEX VALUE	ASCII CHAR.	COMMAND VALUE
6	4BH	K	REQUEST KEYWORD
7	45H		CHECKSUM BYTE
8	6AH		INVERTED CHECKSUM BYTE

KEYWORD REQUEST COMMAND
FIGURE 9.11.1.

9.12. REQUEST FOR STATUS CODE

This command requests the 14230 Converter to return a code reporting its current status.

BYTE NO.	HEX VALUE	ASCII CHAR.	COMMAND VALUE
6	3FH	?	RETURN STATUS CODE
7	39H		CHECKSUM BYTE
8	C6H		INVERTED CHECKSUM BYTE

STATUS CODE REQUEST COMMAND
FIGURE 9.12.1.

9.13. CLEAR MCU BUFFERS

This command requests the 14230 Converter to clear its internal transmit and receive buffers. This command is necessary to recover from a transmit or receive buffer overflow within the 14230 Converter, as indicated by 30H or 31H status codes *(see Sec. 10.2)*. These errors can be avoided when the PC implements hardware flow control using the CTS and RTS lines *(see Secs. 8.9 & 8.10)*.

BYTE NO.	HEX VALUE	ASCII CHAR.	COMMAND VALUE
6	58H	X	CLEAR MCU BUFFERS
7	52H		CHECKSUM BYTE
8	ADH		INVERTED CHECKSUM BYTE

CLEAR CONVERTER BUFFERS
FIGURE 9.13.1.

9.14. SEND MESSAGE WHILE IN INIT MODE

This command requests the 14230 Converter to send a specified message to the ECU, at 10,400 bps, even though the system is in Initialization Mode. This is useful for example to send a STOP COMMUNICATIONS message in the case that the ECU is in Run Mode, but the 14230 Converter is in Initialization Mode. It can also be used for software development purposes.

Byte number 7 in this command specifies the number of bytes in the message, N. The maximum value of N is 8.

BYTE NO.	HEX VALUE	ASCII CHAR.	COMMAND VALUE
6	4D	M	SEND MESSAGE IN INIT MODE
7	*		LENGTH OF MESSAGE (N)
8 TO (8+N-1)	*		MESSAGE DATA, (N) BYTES
(8+N)	Σ		CHECKSUM BYTE
(8+N+1)	/Σ		INVERTED CHECKSUM BYTE

SEND MESSAGE WHILE IN INIT MODE
FIGURE 9.14.1.

9.15. REQUEST 14230 CONVERTER REVISION LEVEL

This command requests the 14230 Converter to return a value indicating its revision level.

BYTE NO.	HEX VALUE	ASCII CHAR.	COMMAND VALUE
6	52	R	REQUEST REVISION LEVEL
7	4C		CHECKSUM BYTE
8	B3		INVERTED CHECKSUM BYTE

REQUEST CONVERTER REVISION LEVEL
FIGURE 9.15.1.

9.16. REQUEST FOR SYSTEM ID

This command requests the 14230 Converter to return the system ID that it received from the ECU during the most recent alternate slow initialization sequence. *(See Section 11.9 on alternate slow initialization.)*

BYTE NO.	HEX VALUE	ASCII CHAR.	COMMAND VALUE
6	5AH	Z	REQUEST SYSTEM ID
7	54H		CHECKSUM BYTE
8	BBH		INVERTED CHECKSUM BYTE

SYSTEM ID REQUEST COMMAND
FIGURE 9.16.1

10. 14230 CONVERTER RESPONSE PACKETS

10.1. RESPONSE PACKET FORMAT

When in Initialization Mode, the 14230 Converter sends response packets to the PC, over the RS-232 interface, at 19,200 bps, only in response to commands initiated by the PC.

Each response packet consists of eight or nine bytes. It starts with a fixed five-byte header, as follows:

BYTE NO.	HEX VALUE	ASCII CHAR.
1	52H	R
2	45H	E
3	53H	S
4	50H	P
5	3AH	:

RESPONSE PACKET HEADER
FIGURE 10.1.1.

After the five-byte header, one or two data bytes are sent, as described further below.

The response packet is terminated by two checksum bytes, calculated in the same way as for command packets, as described above.

10.2. STATUS CODE RESPONSE PACKET

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

- SET ISO-14230 HIGH-SPEED BAUD RATE *(Sec. 9.6)*
- FAST INIT *(Sec. 9.8)*
- 5-BAUD AND CARB INIT *(Sec. 9.9)*
- GO TO LISTENER MODE *(Sec. 9.10)*
- REQUEST FOR STATUS *(Sec.9.12)*
- CLEAR MCU BUFFERS *(Sec. 9.13)*
- SEND MESSAGE WHILE IN INIT MODE *(Sec. 9.14)*

The sixth byte within this packet indicates whether or not the 14230 Converter has currently detected an error, according to the chart below.

The seventh and eighth bytes within this packet are the checksum bytes.

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
04H	K-LINE FAILED TO RETURN HIGH	BUS CONTENTION
05H	L-LINE FAILED TO RETURN HIGH	BUS CONTENTION
06H	K-LINE AND L-LINE LOW FOR 1 SECOND	K-LINE AND L-LINE SHORTED TO GROUND
07H	K-LINE LOW FOR 1 SECOND	K-LINE SHORTED TO GROUND
08 H	L-LINE LOW FOR 1 SECOND	L-LINE SHORTED 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
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-TXD)	FLOW CONTROL FAILURE
31H	TRANSMIT BUFFER OVERFLOW (TO PC-RXD)	FLOW CONTROL FAILURE
4FH	NO ERROR PRESENT	SYSTEM OPERATING PROPERLY
E0H	CONVERTER MCU ROM CHECKSUM ERROR	HARDWARE FAULT, CONTACT SILICON ENGINES

14230 CONVERTER RESPONSE CODES
FIGURE 10.2.1.

10.3. KEYWORD RESPONSE PACKET

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

- REQUEST FOR KEYWORD *(Sec. 9.11)*

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

If the 14230 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].

BYTE NO.	HEX VALUE	RESPONSE PACKET VALUE
6	*	MOST SIGNIFICANT KEYWORD BYTE
7	*	LEAST SIGNIFICANT KEYWORD BYTE
8	Σ	CHECKSUM BYTE
9	/Σ	INVERTED CHECKSUM BYTE
* VALUE DEPENDS ON ECU KEYWORD		

KEYWORD RESPONSE PACKET
FIGURE 10.3.1.

10.4. SYSTEM ID RESPONSE PACKET

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

- REQUEST FOR SYSTEM ID

This response packet includes the two-byte System ID received by the 14230 Converter, from the ECU, during the most recent alternate slow initialization sequence. *(See Sec. 11.9 for information on the alternate slow initialization sequence.)* The System ID is formatted as two hexadecimal bytes.

If the 14230 Converter has not yet initialized the attached ECU with the alternate slow init method, the System ID returned will be 0000H.

BYTE NO.	HEX VALUE	RESPONSE PACKET VALUE
6	*	MOST SIGNIFICANT SYSTEM ID BYTE
7	*	LEAST SIGNIFICANT SYSTEM ID BYTE
8	Σ	CHECKSUM BYTE
9	/Σ	INVERTED CHECKSUM BYTE
* VALUE DEPENDS ON ECU SYSTEM ID		

SYSTEM ID RESPONSE PACKET
FIGURE 10.4.1

10.5. REVISION LEVEL RESPONSE PACKET

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

- REQUEST 14230 CONVERTER REVISION LEVEL *(Sec. 9.15)*

This response packet includes a one-byte hex value that shows the 14230 Converter revision level.

BYTE NO.	HEX VALUE	RESPONSE PACKET VALUE
6	*	REVISION LEVEL (HEX VALUE)
7	Σ	CHECKSUM BYTE
8	/Σ	INVERTED CHECKSUM BYTE
* VALUE DEPENDS ON CONVERTER REVISION LEVEL		

REVISION LEVEL RESPONSE PACKET
FIGURE 10.5.1.

11. INITIALIZATION DETAILS

11.1. 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.

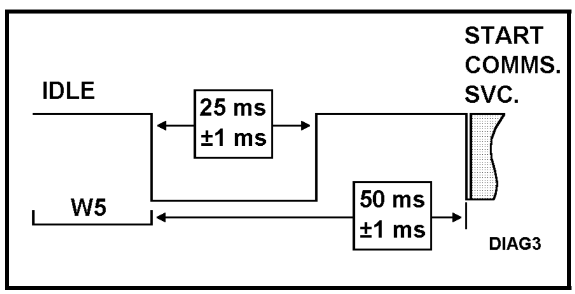
11.2. FAST INITIALIZATION

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 Sec. 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 Sec. 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 11.2.1

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 Sec. 11.4.)*

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.

11.3. 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 11.3.1

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 *(see Sec. 11.9)*.

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.

11.4. 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 11.4.1

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 11.4.2

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 11.4.3

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 11.4.4

11.5. COMPLETION OF FAST INITIALIZATION

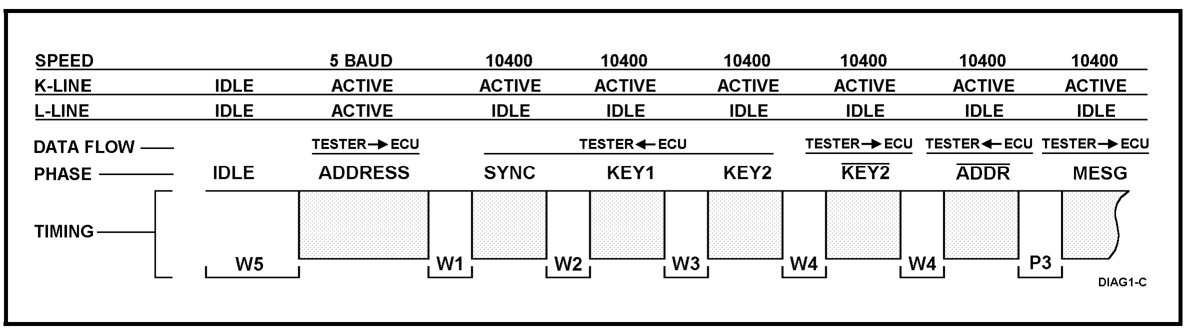
The MCU within the 14230 Converter handles the entire Fast Initialization sequence—including generation of the Wake-Up Pattern, sending the Start Communications Service packet to the ECU at 10,400 bps (typical), and receiving the Start Communications Positive Response packet at 10,400 bps (typical) from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

If initialization was successful, the 14230 Converter sends the PC a status code response packet at 19,200 bps that includes the code 4FH, no error *(Sec. 10.2)*. The 14230 Converter then raises its DSR line to signal that it is time to leave Initialization Mode, and enter Run Mode. The 14230 Converter expects the PC to respond by raising the DTR line.

If initialization was unsuccessful, the 14230 Converter sends the PC a status code response packet at 19,200 bps that includes an error code indicating the problem encountered *(Sec. 10.2)*. The 14230 Converter keeps its DSR line low to show that there is more work to do in Initialization Mode.

11.6. 5 BAUD INITIALIZATION SEQUENCE

The 5-baud initialization protocol is illustrated in the following drawing (except in the case of Alternate Method of slow initialization which is described in section 11.9) . Each step is explained below, and then the time periods W1-W5 and P3 are defined.



5-BAUD INITIALIZATION SEQUENCE
FIGURE 11.6.1

1. **Idle state:** When a diagnostic tester is connected to the ISO 9141 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 <u>0</u>10101010<u>1</u> binary (start and stop bits <u>underlined</u>), providing an alternating bit pattern that an automotive diagnostic tester can use to determine the bit rate.

5. **High-speed data rate:** The 14230 Converter communicates at a fixed high-speed data rate of 10,400 bps, unless programmed for a different speed *(see Sec. 9.6).*

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 11.6.2

   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 11.6.3

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 11.6.4

11.7. 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 11.7.1

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.

11.8. 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 at 19,200 bps that includes the code 4FH, no error *(Sec. 10.2)*. The 14230 Converter then raises its DSR line to signal that it is time to leave Initialization Mode, and enter Run Mode. The 14230 Converter expects the PC to respond by raising the DTR line.

If initialization was unsuccessful, the 14230 Converter sends the PC a status code response packet at 19,200 bps that includes an error code indicating the problem encountered *(Sec. 10.2)*. The 14230 Converter keeps its DSR line low to show that there is more work to do in Initialization Mode.

11.9. ALTERNATE METHOD OF SLOW INITIALIZATION

If the lowercase ‘p’ or ‘f’ is used in the command for slow initialization *(Sec. 9.9)*, then this is an alternate, completely different method of 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 SE 14230 RS232 Message Center because ordinary message control does not wait after every byte sent for an echo. Therefore, in order to use the Model 9002 14230/RS-232 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 “Z” (Query System ID) command.

The format of the data sent back and forth is very similar to FIGURE 11.9.1 except that after Key 1 and Key 2 are conveyed, the 2 bytes of the System ID are also conveyed.

12. HIGH-SPEED PROTOCOL

12.1. COMPLETION OF INITIALIZATION

When initialization is complete, the system shifts to Run Mode. The 14230 Converter sets the DSR line high, and the PC sets the DTR line high. Both the DTR and DSR lamps on the 14230 Converter should be green.

12.2. OPERATION IN RUN MODE

While in Run Mode, the 14230 Converter receives data bytes from the PC at 19,200 bps, and retransmits them immediately to the ECU at the ISO-14230 high-speed data rate, typically 10,400 bps.

The 14230 Converter receives data bytes from the ECU at 10,400 bps (typical), and retransmits them immediately to the PC at 19,200 bps.

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

12.3. ERRORS IN RUN MODE

The 14230 Converter does not check for any Keyword Protocol 2000 data or timing errors when it is in Run Mode. It serves mainly as a baud-rate conversion device while in Run Mode.

However, the 14230 Converter does check continually for abnormal conditions on the ISO-14230 K-line while in Run Mode. If an error is detected—for example, a K-line short to VBATT or ground—then the 14230 Converter deasserts its DSR line, requesting the PC to leave Run Mode and return to Initialization Mode.

When the PC responds by deasserting DTR, the 14230 Converter is in Initialization Mode. It sends a Status Response Packet back to the PC that includes an error code specifying the problem *(Sec. 10.2)*.

12.4. RUN-MODE TIME-OUT FUNCTIONS

While in Run-Mode, the 14230 Converter monitors activity on the K-line. If there is no K-line traffic for over 5.0 seconds, the 14230 deasserts its DSR line, requesting the PC to leave Run Mode and return to Initialization Mode.

The PC can keep the 14230 Converter in Run Mode by periodically sending TESTER PRESENT packets [ISO-14230-3 §6.4] whenever the data link is otherwise idle.

13. ISO-14230 MESSAGE CENTER—CONFIGURATION

13.1. FUNCTIONALITY

The **ISO-14230 Message Center** software is provided with the 14230 Converter. When installed in a personal computer running Windows®, this program allows you to send messages through the 14230 Converter to a connected ECU. This program is compatible with Windows 95, 98, ME, XP, and Vista.

Windows Vista is supported by ISO-14230 Message Center revision 1.9.8 or later. Please contact Silicon Engines if you need assistance in updating older versions.

13.2. INSTALLATION

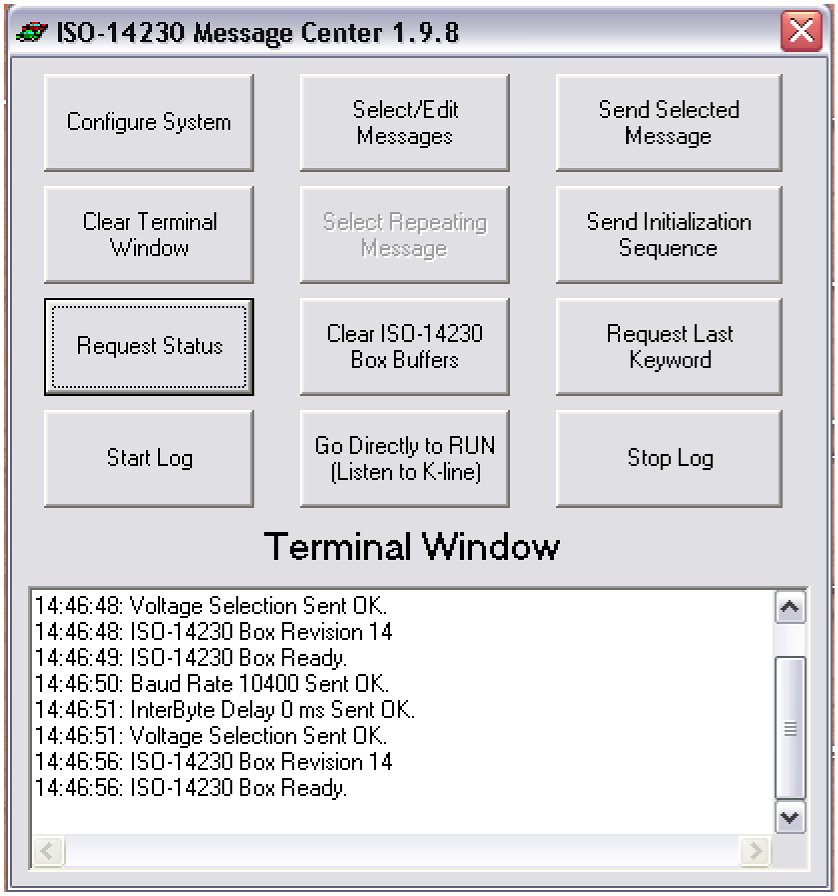
The **ISO-14230 Message Center** software is provided on a CD that comes with the product. To install the software, insert the CD into your computer's CD drive. Run the program SETUP.EXE in the root folder of the CD. This program will install the software on your computer.

13.3. DEVELOPING CUSTOM APPLICATIONS

The CD that comes with the product also contains full source code for the ISO-14230 Message Center. The source code is written in Microsoft Visual Basic 6.0. This can be used as a starting point to develop custom application programs using the 14230 Converter.

13.4. MAIN MESSAGE CENTER SCREEN

The **Main Message Center** screen appears below. It consists of 12 function buttons, and a **Terminal Window**. Brief time-stamped messages appear in the **Terminal Window** to show messages flowing between the PC, the 14230 Converter, and the connected ECU.



MAIN MESSAGE CENTER SCREEN
FIGURE 13.4.1

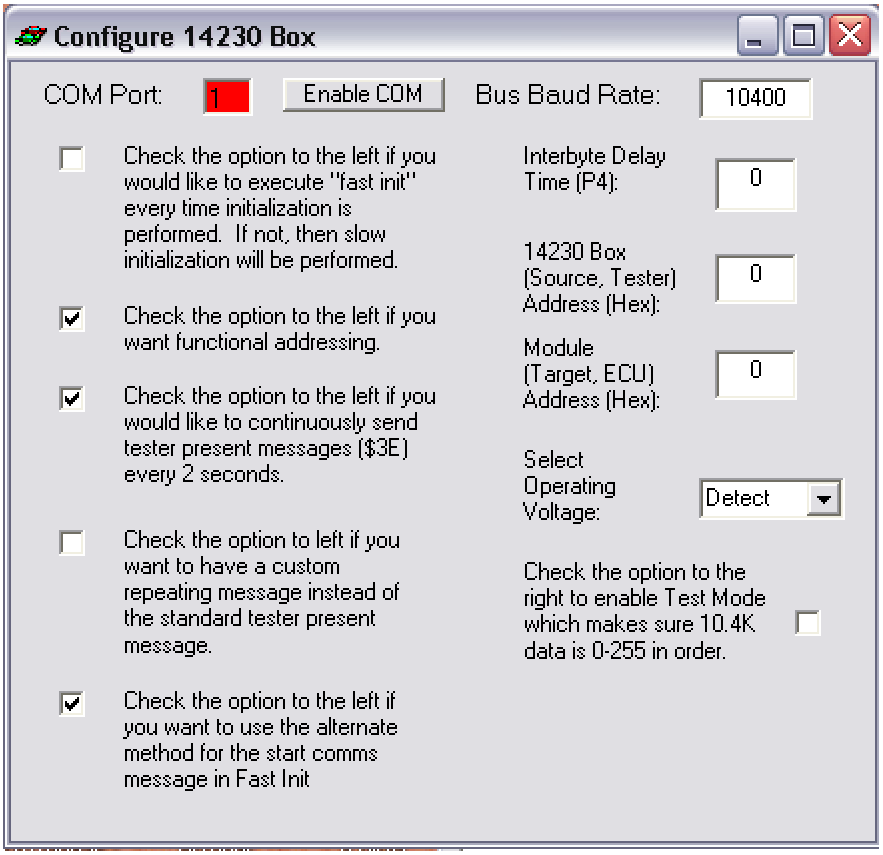
In this chapter, Part 13, we will discuss basic how to set up the system for basic functions.

In Part 14, we will discuss sending and receiving messages.

In Part 15, we will discuss advanced options.

13.5. CONFIGURE SYSTEM

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



CONFIGURATION SCREEN
FIGURE 13.5.1.

13.6. SELECTING THE PROPER COM PORT

The 14230 Converter connects to the PC over an RS-232 serial data port. Each RS-232 interface connector on the PC is assigned a communications port number—COM 1, COM 2, etc.

When the ISO-14230 Message Center software is first started up, the COM port is disabled.

In the box provided in the **Configuration Screen**, first enter the number of the COM port that matches the RS-232 serial port that you are using for the 14230 Converter.

Then click the ENABLE button.

COM port enabling is supported by PC software revision 1.9.7 or later. Please contact Silicon Engines if you need assistance in updating older versions.

13.7. SELECTING THE INITIALIZATION MODE

The 14230 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 9 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 Converter will use fast initialization. Otherwise it will use 5-baud initialization. You can also check the box for functional addressing versus physical addressing, and/or alternate method of initialization for tailoring the fast initialization process to the format that the specific ECU will recognize.

13.8. GO DIRECTLY INTO LISTENER MODE

This function is supported by 14230 Converter firmware Rev. 10 or later, and PC software revision 1.6 or later. Please contact Silicon Engines if you need assistance in updating older versions.

The 14230 Converter can optionally bypass initialization and jump directly into high-speed Run Mode. *(For technical details, see Sec. 9.10 above.)*

To do this, use the PC mouse to point to the **Go Directly to RUN (Listen to K-line)** box in the **Main Message Center Screen**.

13.9. SPECIFYING THE ISO-14230 HIGH-SPEED DATA RATE

Normally the 14230 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 Converter also allows you to specify non-standard high-speed data rates. Enter the desired baud rate in the box provided. Valid entries are from 1,000 to 10,417 bps.

13.10. SPECIFYING THE INTER-BYTE DELAY

Normally the 14230 Converter will communicate with the ECU with 5 millisecond delays between bytes. However the 14230 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 Sec. 9.7.*

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

When the 14230 Converter has been configured for 5-baud initialization and for functional addressing, the 14230 Converter will transmit the ECU address using 8N1 data format, during the 5-baud initialization sequence. *(For details, see Sec. 11.6.)*

13.12. SPECIFYING THE OPERATING VOLTAGE

The 14230 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. It changes its internal K-line and L-line load resistors to match the operating voltage. *(See Sec. 6.7 for details.)*

In the box marked **Select Operating Voltage**, select one of three options: **12V**, **24V,** or **Detect** (the 14230 Converter detects the voltage and switches automatically).

13.13. 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.

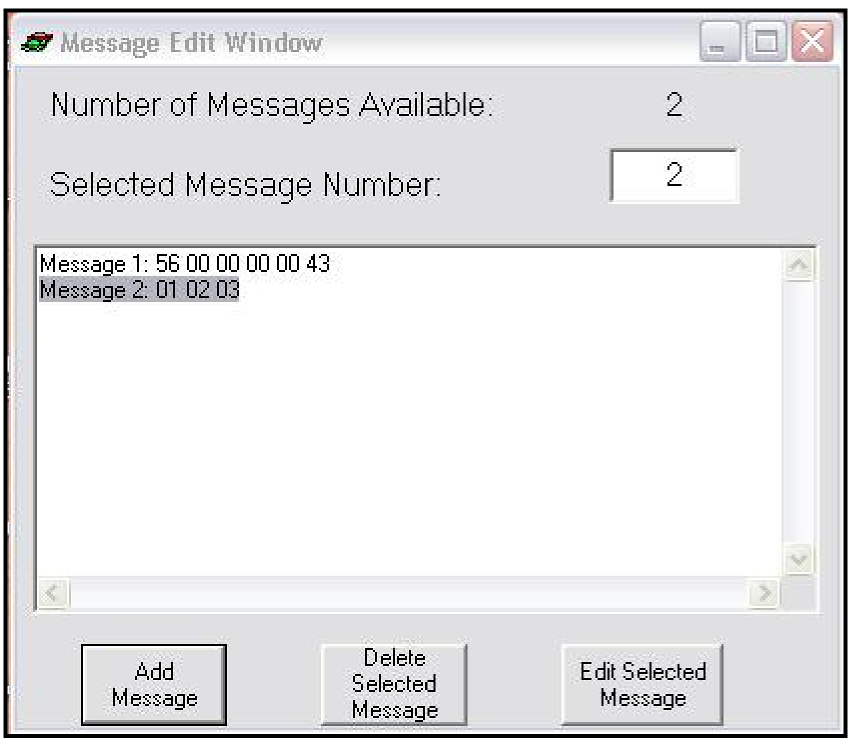
13.14. TROUBLESHOOTING INITIALIZATION PROBLEMS

- **Fast init/5-baud:** If the ECU does not respond using fast initialization, try 5-baud 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 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 as well.
- **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.
- **Confirm light patterns:** Confirm that the lamp patterns on the face of the 14230 Converter match the expected lamp patterns *(see Secs. 8.12-8.15)*.

14. ISO-14230 MESSAGE CENTER—MESSAGE CONTROL

14.1. 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.

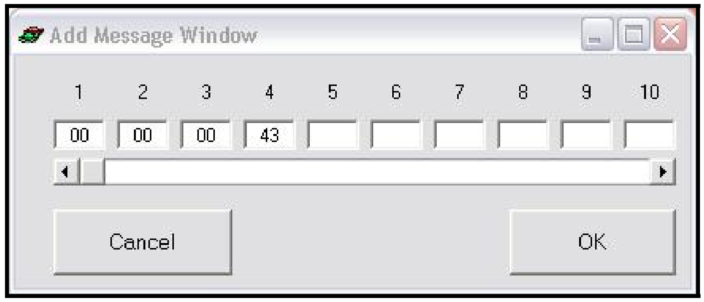


MESSAGE EDIT WINDOW
FIGURE 14.1.1.

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

14.2. 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 14.2.1.

14.3. 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**.

14.4. 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.

15. ISO-14230 MESSAGE CENTER—ADVANCED OPTIONS

15.1. REQUEST STATUS

In the **Main Message Center** screen**,** check the box **Request Status** to cause the PC to send a request to the 14230 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 Sec. 10.2.)*

15.2. REQUEST LAST KEYWORD

In the **Main Message Center** screen**,** check the box **Request Last Keyword** to cause the PC to send a request to the 14230 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 Sec. 10.3.)*

15.3. CLEAR BUFFERS IN 14230 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 Converter to clear its internal data storage buffers. This may be necessary in the event of an error in communications with a connected ECU.

15.4. TESTER PRESENT MESSAGES

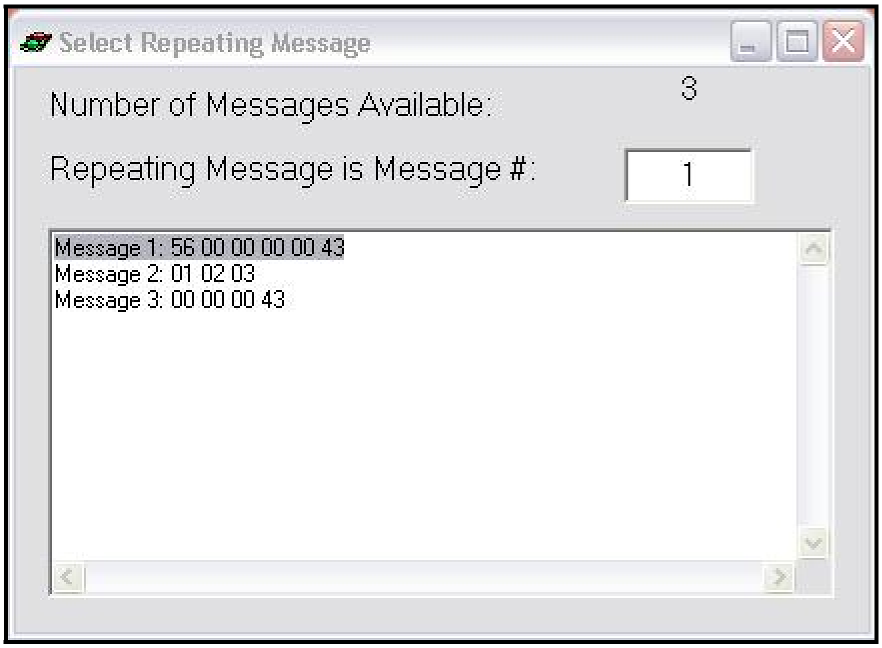
In the **Configuration Screen,** check the box for **Tester Present Messages** to send Tester Present messages (hex 3E) every two seconds to the ECU. These messages are typically generated by a diagnostic tester to keep the addressed ECU awake. Many ECUs automatically disconnect and enter low-power sleep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

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.*

15.5. SENDING A CUSTOM REPEATING MESSAGE

In the **Configuration Screen,** check the box for **Custom Repeating Message** to send your own message (in place of **Tester Present**) 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 Repeating Message** button on the **Main Message Center** screen. This brings up the following screen:



SELECTING THE REPEATED MESSAGE
FIGURE 15.5.1.

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.

15.6. 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 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.

15.7. GENERATING A LOG FILE

This function is supported by PC software revision 1.6 or later. Please contact Silicon Engines if you need assistance in updating older versions.

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.

☑

16. REFERENCES

16.1. 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.

16.2. 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).

1. ISO-9141, *Road Vehicles—Diagnostic Systems—Requirements for Interchange of Digital Information*, 1989-10-1.
2. ISO-9141-2, *Road Vehicles—Diagnostic Systems—Part 2: CARB Requirements for Interchange of Digital Information*, 1994-02-01.
3. ISO-9141-2, *Road Vehicles—Diagnostic Systems—Part 2: CARB Requirements for Interchange of Digital Information, Amendment 1*, 1996-12-01.
4. ISO-9141-3, *Road Vehicles—Diagnostic Systems—Part 3: Verification of the Communication Between Vehicle and OBD II Scan Tool*, 1998-12-15.
5. ISO-14230-1, *Road Vehicles—Diagnostic Systems—Keyword Protocol 2000—Part 1: Physical Layer*, 1999-03-15.
6. ISO-14230-2, *Road Vehicles—Diagnostic Systems—Keyword Protocol 2000—Part 2: Data Link Layer*, 1999-03-15.
7. ISO-14230-3, *Road Vehicles—Diagnostic Systems—Keyword Protocol 2000—Part 3: Application Layer*, 1999-03-15.
8. ISO-14230-4, *Road Vehicles—Diagnostic Systems—Keyword Protocol 2000—Part 4: Requirements for Emission-Related Systems*, 2000-06-01.

16.3. 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 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*, 1999 edition.
2. J1962, *Diagnostic Connector*, February 1998.
3. J1850, *Class B Data Communications Network Interface*, May 2001.

17. REVISION HISTORY

17.1. REVISION K

Revision K of this document applies to Model 9002 firmware Rev. 18 or later, and PC software revision 1.9.11 or later. If you have older versions, please contact Silicon Engines for assistance in updating your equipment or visit the Silicon Engines web-site for the latest product CD.

1. **Fig. 9.9.1, Slow Init Command:** Added information on invoking alternate method of slow initialization.
2. **Sec 9.16: Request System ID Command:** Added information on the request for System ID command.
3. **Sec 10.4: Response to System ID Command:** Added information on the System ID response.
4. **Sec 11.6: Slow Init Details:** Added that this applies to non-alternate method only.
5. **Sec 11.9: Alternate Slow Init Details:** Added.

17.2. REVISION J

Revision J of this document applies to Model 9002 firmware Rev. 14 or later, and PC software revision 1.9.8 or later. If you have older versions, please contact Silicon Engines for assistance in updating your equipment.

1. **Sec. 13.1, Functionality:** Added information on Windows Vista support.
2. **Sec. 13.4, Main Message Center Screen:** Updated screen illustration to latest version.
3. **Sec. 13.5, Configure System:** Updated screen illustration to latest version.
4. **Sec. 13.6, Selecting the Proper COM Port:** Updated to explain the COM port ENABLE function.

17.3. REVISION H

Revision H of this document applies to Model 9002 firmware Rev. 14 or later, and PC software revision 1.9.2 or later. If you have older versions, please contact Silicon Engines for assistance in updating your equipment.

1. **Page 1, Title Page:** Added Model 9002 designation, updated illustration to newer label that shows Model 9002 designation.
2. **Page 2, Company Information**: Updated Silicon Engines email address, physical address, phone and fax number, web site address.
3. **Sec. 1.3, Speed Conversion:** Made it clear that the 14230 Converter is flexible in baud rate selection, and gave range of valid baud rates.
4. **Sec. 3.1, Top panel:** Updated top panel illustration to newer version that includes the Model 9002 designation.
5. **Sec. 8.1, Status Lamp:** Gave explicit explanation of the meaning when the Status Lamp is off.
6. **Sec. 8.12-8.15, Lamp Patterns:** Gave more verbose explanations of DTR, DSR, RTS, and CTS lamp colors in Figures 8.12.1, 8.14.1 and 8.15.1.
7. **Figure 9.8.1, Alternate Fast Init Method:** Gave details on how to invoke new alternate method of initialization.
8. **Sec. 11.3, Start Communications Request:** Gave details on how alternate method of initialization modifies the Start Communications Request packet.
9. **Figure 11.4.2, Figure 11.4.4, Start Communications Positive Response, Format #2, #4:** Change text “Packet Length – 16” to “Length Byte”.
10. **Add Sec 13.3, Developing Custom Applications:** Gave the reader of this manual some ideas about how to develop their own custom applications.
11. **Figures 13.4.1, 13.5.1:** Updated these pictures.
12. **Sec. 13.7, Selecting the Initialization Mode:** Included details on alternate methods of initialization (such as physical versus functional) in this paragraph.
13. **Sec. 13.14, Troubleshooting Initialization Problems:** Changed title of this section from “Troubleshooting Installation Problems” to “Troubleshooting Initialization Problems.” Added bullet on alternate method of fast init. Added text about getting the Tester address correct, which can also be important. Added bullets on L line and Light Patterns as well.

17.4. REVISION G

1. **Sec. 9.1, Initialization Alternatives:** Added wording about listener mode operation.
2. **Sec. 9.10, Go to Listener Mode:** Added this section.
3. **Sec. 10.2, Status Code Response Packet:** Added GO TO LISTENER MODE to list of commands that stimulate a response packet to be returned.
4. **Sec. 13.3, Main Message Center Screen:** Updated screen picture to version 1.6. Changed number of function buttons from 9 to 12.
5. **Sec. 13.7, Go Directly into Listener Mode:** Added this section.
6. **Sec. 15.7, Generating a Log File:** Added this section.

17.5. REVISION F

1. **Sec. 9.7, Set Inter-Byte Delay (P4):** Added this section.
2. **Sec. 13.3, Main Message Center Screen:** Updated to Version 1.5 screen.
3. **Sec. 13.4, Configure 14230 Box:** Updated to Version 1.5 screen, with added box for Inter-Byte Delay
4. **Sec. 13.9, Specifying the Inter-Byte Delay:** Added this section.

17.6. REVISION E

1. **Sec. 1.3, Speed conversion:** Changed to show that the ISO-14230 high-speed data rate is *typically* 10,400 bps. With new 14230 Converter software, it can be any value from 1,000 bps to 10,400 bps.
2. **Sec. 6.2, Serial Data Format:** ISO-14230 high-speed data rate is *typically* 10,400 bps.
3. **Sec. 6.5, K-Line Logic:** ISO-14230 high-speed data rate is *typically* 10,400 bps.
4. **Sec. 8.7, DTR:** ISO-14230 high-speed data rate is *typically* 10,400 bps.
5. **Sec. 8.15, Lamp Patterns:** Run Mode: Changed “10,400 bps” to “ISO-14230 high speed.”
6. **Sec. 9.6, Set Baud Rate:** Added this section.
7. **Sec. 9.9, Request for Keyword:** Corrected description of command in table.
8. **Sec. 10.2, Status Code Response Packet:** Added baud rate command to list.
9. **Sec. 11.2, Fast Initialization:** ISO-14230 high-speed data rate is *typically* 10,400 bps.
10. **Sec. 11.2, Fast Initialization:** ISO-14230 high-speed data rate is *typically* 10,400 bps.
11. **Sec. 11.3, Start Communications Request:** ISO-14230 high-speed data rate is *typically* 10,400 bps.
12. **Sec. 11.5, Completion of Fast Initialization:** ISO-14230 high-speed data rate is *typically* 10,400 bps.
13. **Sec. 11.6, 5-Baud Initialization Sequence, para. 2, 5-baud address:** Added information on different 5-baud data formats for physical vs. functional addressing.
14. **Sec. 11.6, 5-Baud Initialization Sequence, para. 5, High-speed data rate:** Changed to show that 10,400 standard data rate can be altered by a PC command.
15. **Sec. 11.6, 5-Baud Initialization Sequence, paras. 7 & 8:** ISO-14230 high-speed data rate is *typically* 10,400 bps.
16. **Sec. 12.1, Operation in Run Mode:** ISO-14230 high-speed data rate is *typically* 10,400 bps.
17. **Parts 13-16:** Name of PC software changed from *ISO-14230 Message Center* to *ISO-14230 Message Center*.
18. **Sec. 13.3, Main Message Center Screen:** Updated main screen image to version 1.4.
19. **Sec. 13.4, Configure system:** Updated configuration screen image to version 1.4 screen, which includes baud rate selection box.
20. **Sec. 13.8, Specifying the ISO-14230 High-Speed Data Rate:** Added this section.
21. **Sec. 13.9, Specifying the Address Mode:** Added description of 5-baud initialization data formats.

17.7. REVISION D

1. **Sec. 4.3, Figure 4.3.2:** Added connection of SHIELD to pin 7 of DB9F connector within 14230-OBD Cable.
2. **Sec. 5.2, Figure 5.2.1:** Added typical supply currents.
3. **Chapters 13-15:** Added these chapters to describe the 14230 Message Center software package.

17.8. REVISION C

1. **Sec. 11.4:** Corrected error in Figures 11.4.1 and 11.4.2, START COMMUNICATIONS POSITIVE RESPONSE, Formats #1 and #2. Bytes 2 and 3 were reversed. Byte 2 should be TESTER ADDRESS, Byte 3 should be ECU ADDRESS.

17.9. REVISION B

1. **Sec. 5.1:** Added information about powering from 14230-OBD Cable.
2. **Secs. 8.7, 8.8:** Clarified interaction of DTR and DSR lines.
3. **Sec. 9.1:** Emphasized that initialization must first take place for communications to proceed.
4. **Sec. 9.3**: Clarified that the 14230 Converter responds to each PC command with a response packet.
5. **Sec. 10.2**: Changed name of section to STATUS CODE RESPONSE PACKET. Clarified that packet is sent in response to a specified list of PC commands.
6. **Sec. 10.3:** Added this section to clarify format of KEYWORD RESPONSE PACKET.
7. **Sec. 10.4:** Added this section to clarify format of REVISION LEVEL RESPONSE PACKET.
8. **Sec. 11.2:** Added information on the START COMMUNICATIONS SERVICE POSITIVE RESPONSE frame.
9. **Sec. 11.3:** Changed to indicate that the format byte may be either 81H or C1H.
10. **Sec. 11.4:** Added section to cover START COMMUNICATIONS SERVICE POSITIVE RESPONSE.
11. **Sec. 11.5:** Added section to clarify procedure at end of Fast Initialization.
12. **Sec. 11.8:** Added section to clarify procedure at end of 5-Baud Initialization.
13. **Secs. 12.1, 12.2, 12.3, and 12.4:** Rewritten and expanded to clarify operation in high-speed mode.

17.10. REVISION A

Initial release.

☑