ADVANCED VEHICLE TECHNOLOGIES, Inc. AVT inc.

AVT-931

Dual J1850 INTERFACE (VPW and PWM)

PC/104 Unit

User's Manual

1509 Manor View Road
Davidsonville, MD 21035
410-798-4038 (voice) 410-798-4308 (fax)

Revision 'A1'

June 1998

TABLE OF CONTENTS

1. INTRODUCTION 3

1.1 SPECIFICATIONS 3
1.2 DEFINITIONS 4

2. INSTALLATION 4

2.1 HARDWARE CONFIGURATION 4
2.2 HARDWARE INSTALLATION 5
2.3 TESTING THE AVT-931 5

3. INTERFACE BOARD INTRODUCTION 6

4. INTERFACE BOARD DESCRIPTION 6

5. INTERFACE OPERATION 9

5.1 COMMUNICATIONS INTRODUCTION 9
5.2 VPW MODE 9
5.2.1 Command Header Byte 10
5.2.2 Response Header Byte 11
5.2.3 Transmit message format 12
5.2.4 Receive message format 12
5.2.5 Match Function 13
5.2.6 Received Message Status Byte Definitions 14
5.2.7 Examples 14
5.3 PWM MODE 14
5.3.1 Command Header Byte 15
5.3.2 Response Header Byte 16
5.3.3 'Look Alike' mode 16
5.3.4 Transmit message format 'Look Alike' mode Off 17
5.3.5 Transmit message format 'Look Alike' mode On 17
5.3.6 Receive message format 'Look Alike' mode Off 18
5.3.7 Receive message format 'Look Alike' mode On 18
5.3.8 Monitor Mode 18
5.3.9 HBCC Initialization Parameters 19
5.3.10 Examples 20
5.4 ERROR CODES 21
5.5 HARDWARE REGISTER #2 DEFINITION 22
5.6 ISA BUS MEMORY MAP 22

6. REFERENCES 23

6.1 TECHNICAL SUPPORT 23
6.2 INTERFACE INFORMATION 23
6.2.1 Interface Board P1 24
6.2.2 Interface Board P2 24

7. COMPANY OVERVIEW 25

1. INTRODUCTION

This manual covers the AVT-931 Dual J1850 hardware interface unit. It provides technical information on using the interface, connections to it, communications with it, and other related information. If using the AVT Controller software, please refer to the "AVT-xxx Controller Software User's Manual."

The AVT-931 Dual J1850 Interface implements both VPW and PWM versions of SAE specification J1850. The board is designed to be compatible with the PC/104 form factor specifications.

The Society of Automotive Engineers (SAE) has adopted a specification known as J1850 "Class B Data Communications Network Interface." This specification describes two forms of a multiplex bus structure intended for use in a vehicle. The two forms of this multiplex bus are known as Pulse Width Modulation (PWM) and Variable Pulse Width (VPW). The AVT-931 Dual J1850 Interface implements both versions of the J1850 standard (Simultaneous operations are not permitted).

The AVT-931 Dual Interface provides the following functions:

- Isolated electrical interface between the subject vehicle and the host computer.
- Protocol conversion and communications between the vehicle J1850 multiplex bus and the host computer via the ISA bus (in a PC/104 form factor).
- Passive network traffic monitor.

The AVT-931 conforms to the Ford Motor Company Standard Corporate Protocol (SCP) which defines network traffic management, message construction, and other protocol issues. (The SCP specifies operations in PWM mode.) The AVT-931 is fully compatible with any similarly equipped Ford Motor Company or Mazda product.

When in VPW mode of operation the AVT-931 permits both transmit and receive operations at 4 times the normal speed (4X mode). This mode of operation is not supported by the J1850 specification. 4X mode may, however, be required to be compatible with General Motors' Class 2 system.

1.1 Specifications

The following describes the AVT-931 hardware interface. Once installed as part of a PC/104 computer system 'stack' the AVT-931 interface can be accessed from MS-DOS (directly) and from other operating systems (using an appropriate device driver).

Host Computer:

- PC/104 form factor computer system or ISA bus with appropriate adapter card.

AVT-931 Dual Interface Board:

- Overall size: 3.8 x 4.3 x 0.5 inches
- Weight: 3.5 oz.
- +5 VDC from host computer.
- +12 volts from either host computer or the subject vehicle.
  (The two sources are diode isolated.)
- Power dissipation: 1.2 watts (nominal).

1.2 Definitions

The following terms are used in this manual.

- Class 2: General Motors standard for in-vehicle network communications.
- DLC: Data Link Controller (Motorola / Delco Electronics).
  This is the VPW interface device.
- HBCC: Hosted Bus Controller Chip (Motorola / Ford Motor Company).
  This is the PWM interface device.
- SCP: Standard Corporate Protocol (Ford Motor Company).
- All numbers used in this manual are hexadecimal digits (0 .. 9 and A .. F) and are preceded with a dollar sign ($) for clarity.

2. Installation

The AVT-931 Dual J1850 Interface board is designed for installation as a part of a PC/104 computer 'stack.' It is an 8-bit ISA board but is usually equipped with the 16-bit connector. Additionally, the AVT-931 is usually outfitted with stackthrough connectors.

Before installing the AVT-931 interface board in the host computer 'stack,' the board's address should be set. A test of communications between the AVT-931 and the host may then be conducted to verify proper operations.

2.1 Hardware Configuration

The AVT-931 interface board is mapped into the memory space of the host computer. (It is not mapped into host I/O space.) The interface board occupies only 16 bytes of memory space (of which only 3 bytes are actually used).

Before installing the AVT-931 into the host computer, the base address of the Interface board must be set. This is done by the proper selection of the two DIP switches (8-bits each) found on the board.

Note that the AVT-931 base address is the same as the segment address of the board. (In a PC compatible machine only 20 address lines are available through the 8-bit ISA bus connection.) The AVT-931 base address equates to address lines [A19..A4]. This is the same as a segment address. The bottom four address lines [A3..A0] are decoded and are used to select the specific on-board peripherals (e.g. hardware register and FIFOs).

The factory default base (segment) address of the AVT-931 interface is $B600. This is in the space normally occupied by a monochrome video adapter (in a PC-AT compatible machine). Since most machines use a color VGA or SVGA display this address space is usually empty.

Before installing the AVT-931 determine the segment address of the address space in which the board is to be installed. (The segment address is obtained by taking the flat address and dropping the last hex digit. Example: The factory default board address is $B6000. The segment address is $B600.) The DIP switches should then be set to the desired segment address.

NOTE: A DIP switch setting of OFF equals an address line logic '1.'

DIP Switch	Address line		Factory Default $B600
SW1-8	A19	MSB	Off '1'
SW1-7	A18		On '0'
SW1-6	A17		Off '1'
SW1-5	A16		Off '1'
SW1-4	A15		On '0'
SW1-3	A14		Off '1'
SW1-2	A13		Off '1'
SW1-1	A12		On '0'
SW2-8	A11		On '0'
SW2-7	A10		On '0'
SW2-6	A9		On '0'
SW2-5	A8		On '0'
SW2-4	A7		On '0'
SW2-3	A6		On '0'
SW2-2	A5		On '0'
SW2-1	A4	LSB	On '0'

As many AVT-931 interface boards may be installed in a PC/104 'stack' as desired and can be accommodated. Each AVT-931 installed should have a unique base address assigned prior to installation.

2.2 Hardware Installation

Installation of an AVT-931 into a PC/104 computer 'stack' is dependent on the host system into which the AVT-931 is being installed. As a result, no generic installation instructions can be provided here. The AVT-931 only uses the 8-bit ISA bus connection (PC/104 connector J1). The 16-bit connector (PC/104 connector J2) can be removed if desired. Both J1 and J2 are equipped as stackthrough connectors. These may be replaced to make the board compatible for use on the end of the 'stack.'

2.3 Testing the AVT-931

After the AVT-931 Dual Interface has been installed, proper operation of the hardware can be observed by commanding the unit to start and reading FIFO #2. When the AVT-931 is started it immediately comes up in VPW mode, executes a basic operational check, and then reports the results to the host. A successful initialization report is: `$91 $07` and can be read from FIFO #2. This report indicates that the unit is running in VPW mode and has completed its initialization and startup test.

If an error is encountered during initialization, an error message is written into FIFO #2. The error message will be of the form `$2x $yy`. These error codes are listed in Section 5.4.

3. Interface Board Introduction

The AVT-931 Dual J1850 Interface board is a four layer printed circuit board designed in the PC/104 form factor. The board communicates with the rest of the computer through the ISA bus. The board also receives +5 and +12 VDC power from the host through the bus connection.. The interface board may be connected (via an appropriate cable) to a vehicle OBI-II connector via the on-board connector P2, a 16-pin right angle pin header.

The AVT-931 Dual J1850 Interface performs all data and protocol conversion functions permitting communications between the host and the subject vehicle. For the VPW mode of operation the AVT-931 utilizes the Motorola MC68HC57 Data Link Controller. This device includes the VPW bus transceiver function. For the PWM mode of operation the AVT-931 utilizes the Motorola HBCC device and a Ford PWM bus transceiver design.

The AVT-931 provides an isolated electrical interface between the control computer and the vehicle. There is no electrical connection between the vehicle and the host computer. All signals and power supplies are isolated. This was done to minimize the possibility of damage to the host computer in the event a voltage or signal transient originates on the vehicle.

The AVT-931 utilizes an embedded 68HC11 core microcontroller to perform the necessary control and communications functions. The interface is capable of operating in either VPW mode or PWM mode. The unit is not designed to support simultaneous operations.

When in VPW mode of operation the AVT-931 is always listening to, or monitoring, the J1850 bus. All bus traffic is reported to the host computer. Transmit operations occur only when initiated by the operator from the control computer.

Operations in PWM mode are different from VPW mode. When operating in PWM mode the HBCC device must be initialized with a unique node address (and other parameters). The HBCC is initialized automatically by the microcontroller when PWM mode is enabled and during a reset. Refer to Section 5.3.9 for a listing of HBCC initialization parameters. These parameters may be changed by the user through the host computer.

When in PWM mode of operation the HBCC device performs input message filtering. Therefore, only bus traffic that is destined for the specified node address is sent to the host computer and displayed. To view all bus traffic while in PWM mode, the HBCC device can be placed into monitor mode. When in monitor mode the AVT-931 becomes a passive PWM network monitor, all bus activity is reported to the host, and no transmit operations are permitted. Refer to Section 5.3.8.

4. Interface Board Description

A block diagram of the AVT-931 interface board is shown in Figure 1. The heart of the unit is the MC68HC711KA2 microcontroller. This device utilizes a 68HC11 core with a bus speed of 4 MHz. The operation firmware is contained in an on-chip EPROM.

Other interface functional blocks include two FIFOs, an HBCC device (the PWM message interface), a PWM bus transceiver, a DLC device (the VPW byte level interface), a bus multiplexer, the electrical isolation of the host computer from the vehicle, and a hardware control/status register.

Two FIFOs are included on the AVT-931 and are configured as a set of bi-directional mail boxes for message passing between the microcontroller and the host computer. Each FIFO is 4K bytes deep and fully asynchronous. They facilitate communications between the host and the microcontroller while minimizing the risk of lost data and eliminating the requirement for real-time response from the host.

The electrical isolation barrier is depicted on the block diagram. The AVT-931 interface derives its operating power (+5 VDC and +12 VDC) from the host computer. Additionally, a connection exists between the interface board and the subject vehicle unswitched battery positive (V_{BATT}). This supply is diode isolated and electrically isolated from the host +12 VDC supply. Such an arrangement permits normal interface operations even when not connected to a vehicle. When the AVT-931 is connected to a vehicle the V_{BATT} connection provides a common positive supply for all network nodes.

When the host computer releases the RESET line for the microcontroller, the microcontroller begins executing the code stored in its on-chip EPROM. The unit will start up in VPW mode and can be switched to PWM mode through a simple software command.

Refer to Section 5.5 for a description of the hardware control register that the host uses to control the status of the AVT-931 Interface unit.

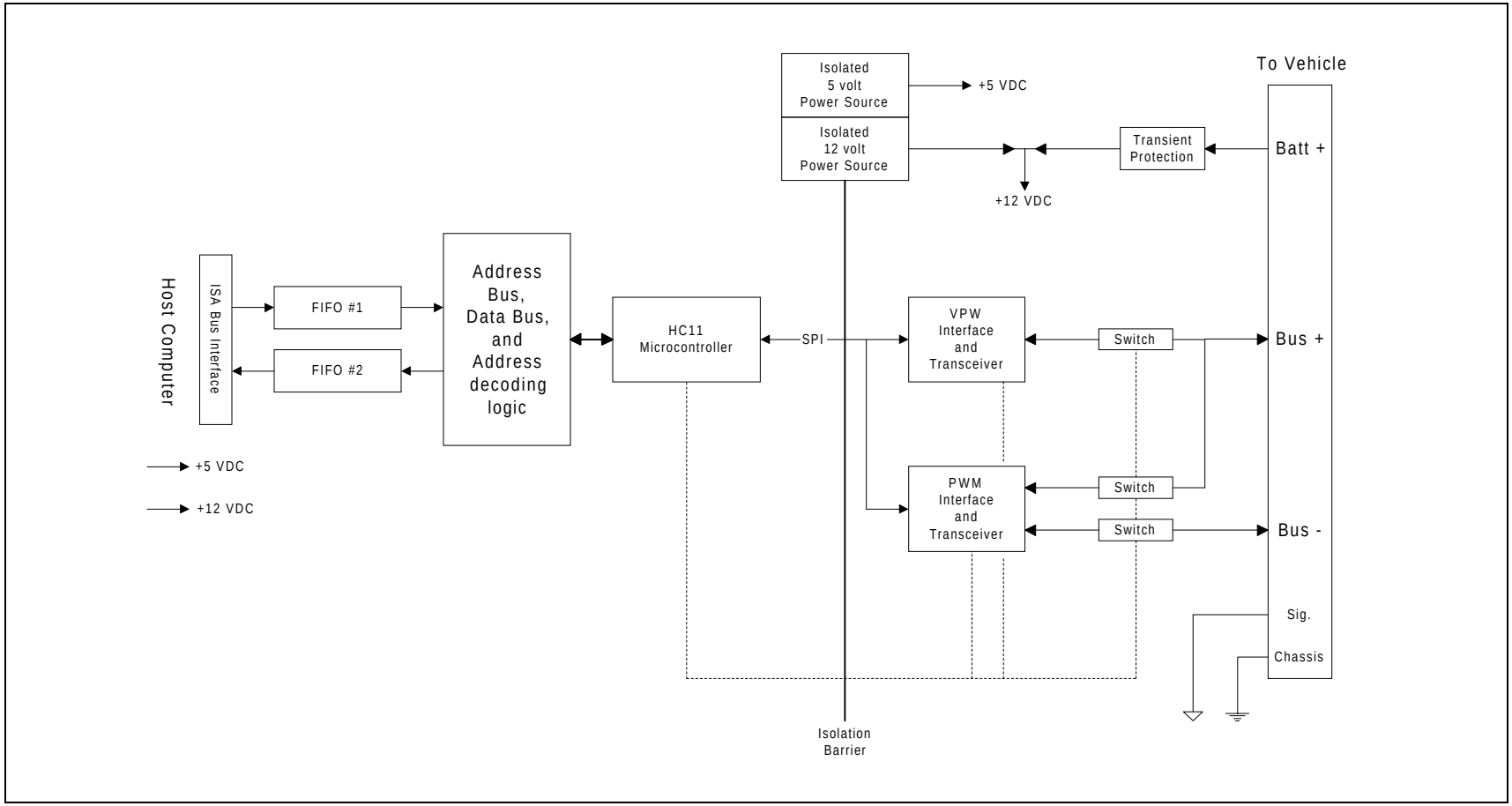

Figure 1 AVT-931 Hardware Interface Unit Block Diagram

5. Interface Operation

The following describes the use of the AVT-931 interface. It is assumed that the board is configured properly and the software has been installed on the host PC.

5.1 Communications Introduction

All communications between the AVT-931 interface and the vehicle network are in conformance with SAE Standard J1850 and all related standards and recommended practices. Additionally, when in the PWM mode of operation, the AVT-931 is compliant with the Ford SCP. The user is referred to those documents for more detailed information.

All communications between the AVT-931 and the host computer are conducted through the ISA bus and conform to accepted ISA bus operations.

The interface board is mapped into the memory space (not I/O space) of the host computer. The interface board occupies 16 bytes of memory space (only 3 locations are actually used). The base address of the interface board is determined by the setting of the two 8-bit DIP switches on the board.

Refer to Section 2.1 for instructions on how to set the DIP switches.

The structure and protocol of communications between the AVT-931 and the host computer are stated in the following sections. All data is transferred in packets. The size of each data packet varies from 1 byte to 16 bytes (inclusive). The first byte in each data packet is the header byte and is used to convey information only between the interface board and the host computer.

The header byte is divided into the upper nibble and lower nibble. The upper nibble indicates what information the data packet is conveying. The lower nibble is the count of the number of bytes that follow the header byte. The meaning of the upper nibble of the header byte depends on which direction the data packet is moving; whether to or from the host computer.

Messages from the host computer to the AVT-931 are known as Commands. Messages from the AVT-931 to the host computer are known as Responses.

Improvements are always being implemented in Advanced Vehicle Technologies products. As a result, the lists of commands and responses in this manual may become out of date. The most recent list of commands and responses for all AVT's products can be found on the web site and is available for download. The web address or URL is listed in Section 6.1.

5.2 VPW Mode

The following sections describe commands, responses, error codes, and other related information while in the VPW mode of operation.

5.2.1 Command Header Byte

This byte is prepended onto a packet transmitted by the control computer to the AVT-931.

Low nibble, bits b3 - b0: Byte count (how many bytes to follow); may be zero when the message is only a status or error message.

High nibble, bits b7 - b4:

0: This is a valid packet for transmission. (The null byte, $00, is ignored.)

1:

2: Reset command.
- $21 $01: Reset the HBCC device.
- $21 $03: Reset the DLC device.
- $21 $04: Reset both FIFOs.

3: Match Table.
- $30: Match status request, report match table contents.
- $31 $7B: Match function off, clear table.
- $32 $xx $yy: Match table entry. $xx - byte position. $yy - byte value.

4:

5: Operations Commands.
- $53 $20 $xx $yy:
  Transmit a block of data. xx yy - data block byte count.
  Refer to Block transfer supplement document for specific instructions on how to conduct data block transfers.
- $51 $21: Request “No-echo” status.
- $52 $21 $00: Disable “No-echo” function.
- $52 $21 $01: Enable “No-echo” function.
- $51 $31: Request “Short to Ground” status.
- $51 $32: Request “Short to High” status.
- $52 $33 $00: Command DLC to exit standby or sleep mode.
- $52 $33 $01: Command DLC to enter standby or sleep mode.
- $51 $34: Command DLC to abort transmission and flush transmit FIFO.
- $53 $35 $xx $yy: Direct communications with the DLC device.
  $xx - Transmit Data or Configuration byte.
  $yy - Command byte.
- $51 $36: Send a Break symbol onto the bus.

6:

7: Microcontroller write to RAM.
- $73 $xx $yy $zz: Write to address xx yy with data byte zz.

8: Microcontroller read from RAM.
- $83 $xx $yy $zz: Read from address xx yy. Read zz bytes.

9:

A:

B: $B0: Request firmware version number.

C: 4x mode command.
- $C0: Request 4x mode status.
- $C1 $00: 4x mode off.
- $C1 $01: 4x mode on.

D: $D0: Request operational mode.

E: Mode switch command.

$E1 $33: Switch to VPW mode.
$E1 $CC: Switch to PWM mode.

F: Re-start microcontroller.
$F1 $A5. (Note: This is not a true reset command.)

5.2.2 Response Header Byte

This byte is prepended onto a packet transmitted by the AVT-931 to the control computer.

Low nibble, bits b3 - b0: Byte count (how many bytes to follow); may be zero when the message is only a status or error message.

High nibble, bits b7 - b4:

0: Valid packet received from the bus.
$0x $yy $zz ...: x - byte count to follow; yy - received message status byte;
zz ... - message bytes.

1:

2: Error message, error byte(s) follow.

3: Command message error, error header byte follows ($31 $xx).

4: Match function:
$40 - function off.
$41 $B7 - table full, entry ignored
$42 $xx $yy - table entry report. $xx - byte position. $yy - byte value.

5:

6: Operations Reports
$62 $20 $00: Exited block transfer mode.
$62 $20 $01: Ready to transmit data in block transfer mode.
$62 $21 $00: “No-echo” mode disabled.
$62 $21 $01: “No-echo” mode enabled.
$62 $31 $00: No “Short to Ground” detected.
$62 $31 $01: “Short to Ground” detected and active.
$62 $32 $00: No “Short to High” detected.
$62 $32 $01: “Short to High” detected and active.
$62 $33 $00: DLC commanded to exit standby or sleep mode.
$62 $33 $01: DLC commanded to enter standby or sleep mode.
$61 $34: DLC commanded to abort transmission.
$63 $35 $xx $yy: Direct communications with DLC device.
$xx - Status byte.
$yy - Receive Data or Completion Code.

7:

8: Read microcontroller RAM.
$8x $yy $zz $ww ... : RAM address yyzz; ww ... memory contents.

9: Board status information.
$91 $03: HBCC initializations complete.
$92 $04 $xx: Firmware version report; xx - firmware version number.
$91 $06: PWM operations.
$91 $07: VPW operations.
$91 $08: DLC initializations complete.

A:

B:

C: 4x mode status response.

$C1 $00: 4x mode off.
$C1 $01: 4x mode on.

D:

E:

F: Used only in block transfer mode.
One “F0” indicates the start of a received block of data.
Fifteen “F0’s” will follow at the end of a received block of data.

5.2.3 Transmit message format

To transmit a message onto the network the message must be built by the operator and then sent to the AVT-931 interface. The DLC device on the board will automatically append the transmit CRC.

It is up to the user to determine and know the proper protocol and messaging strategy that is implemented on the vehicle to which the AVT-931 is attached.

Any message destined for transmission must be preceded by a byte whose upper nibble is ‘0’ (zero) and lower nibble is the byte count of the message. The message bytes then follow immediately.

Once a message is accepted by the AVT-931 for transmission it is sent to the DLC as soon as possible for transmission onto the bus. If the message loses arbitration, the DLC will attempt to re-transmit the message at the next opportunity (bus idle, IFS). The DLC will continue to attempt transmission of the message indefinitely.

In the event the AVT-931 attempted transmission of a message and subsequently lost arbitration, the received message status byte will have bit 4, the lost arbitration bit, set. This only indicates that the AVT-931 received a message that won arbitration over the message queued for transmission. The received message is valid.

If the message is properly transmitted a received message status byte is sent to the control computer. The properly transmitted message echo is received by the interface but is not passed to the control computer. (Example: a message is sent to the AVT-931 for transmission, at some later time the AVT-931 will report: $01 $60 which indicates that the message was transmitted correctly and that the message was received from itself.)

5.2.4 Receive message format

Messages received from the network are assembled into the original byte sequence. The received CRC is calculated and checked to be equal to $C4. Both transmitted and received CRC bytes are then discarded. A received message status byte is constructed and prepended onto the message. A header byte is then prepended onto the message (ahead of the received status byte).

As an example the byte sequence `A7 B6 C5` plus CRC byte is transmitted by a node. The transmitted message is received by the interface and the following byte sequence is sent to the control computer: `04 00 A7 B6 C5`.

The byte $04 indicates that it is a received message and that four bytes follow.
The byte $00 is the received message status byte and indicates that no errors were found.
(Received message status byte, bit definitions are listed below.)
The message bytes then follow. Note that the CRC byte is stripped.

5.2.5 Match Function

A coarse filtering mechanism for messages received from the bus is provided by the AVT-931 interface unit firmware. If the match table is cleared (on power-up, reset, or $31 $7B command) all messages received from the network are passed to the host.

When at least one entry is made to the match table, all messages received from the network are checked against the match table. If a match is found the message is passed to the host. If no match is found, the message is discarded, and the host is not notified.

A match table entry is made using the $32 $xx $yy command. The $xx value is the byte position and the $yy value is the byte value. This filtering mechanism is more easily explained by example.

- It is desired to receive all messages (at the host) where the third byte of the message is equal to $F1.
- Send the command $32 $03 $F1 to the AVT-931 interface.
- Receive the response $42 $03 $F1.
- To verify the table entry send the command $30.
- The response will be $42 $03 $F1.
- The only network messages passed to the host will now be of the form: $zz $xx $F1 $... Note that at the host the message will be $rr $ss $zz $xx $F1 $... where $rr is the header byte, $ss is the received message status byte, and the message follows.

The match table can hold ten entries where an entry consists of a byte position and a byte value. The byte position refers to where in the network message the match byte is to be compared. The first byte of the message has a byte position value of one.

Ordering of the match table is not important. All table entries are checked until a match is found or the end of the table is encountered. If a match table entry specifies a byte position that doesn't exist for the message being checked (the message is shorter than the table entry), that table entry is not checked.

5.2.6 Received Message Status Byte Definitions

The received message status byte always follows the header byte, even if the status byte is the result of transmitting a message.

Bit	Definition
0:	CRC error.
1:	Incomplete message (incorrect number of bits).
2:	Break received.
3:	IFR data.
4:	Lost arbitration.
5:	Transmission successful.
6:	From this device.
7:	Bad message.

5.2.7 Examples

To illustrate the construction and decoding of messages between the control computer and the AVT-931 interface, several examples are provided here.

Example #1: Want to request the current operational mode.
Command string: `D0`.
Explanation: The AVT-931 will respond with: `91 07`. The '`9`' indicates a board response, the '`1`' indicates one byte follows, and the `07` indicates VPW mode (Motorola device).

Example #2: want to send a message out on the bus.
Command string: `04 32 89 AC 5F`.
The message `32 89 AC 5F +crc` is transmitted onto the bus.
The AVT-931 will respond with: `01 60`. The '`0`' indicates a received message and the '`1`' indicates only one byte, which is the received message status byte. The '`60`' indicates that bits `6` and `5` are set which means the received message was from this device and the transmission was successful. (Messages transmitted by the AVT-931 are received by the AVT-931, are checked, but are not echoed back to the controller. Only a status byte is sent to the controller to indicate the status of the transmitted message.)

Example #3: received a message from the bus.
Response string: `05 00 56 B4 AC 8F`.
The '`05`' indicates that this is a bus message and 5 bytes follow. The '`00`' is the receive status byte and indicates that no errors were detected and that the message is not from this device. The following four bytes, '`56 B4 AC 8F`', are the actual message bytes (the transmitted and received CRC bytes are stripped). Refer to Section 5.2.6 for received message status byte bit definitions.

5.3 PWM Mode

The following sections describe commands, responses, error codes, and other related information while in the PWM mode of operation.

5.3.1 Command Header Byte

This byte is prepended onto a packet transmitted by the control computer to the AVT-931.

Low nibble, bits b3 - b0: Byte count (how many bytes to follow); may be zero when the message is only a status or error message.

High nibble, bits b7 - b4:

0: This is a valid packet for transmission by the HBCC. (Null byte '00' is ignored.)

1: 'Look-alike' mode.
- $10: Look alike mode status request.
- $11 $00: Look alike mode off.
- $11 $01: Look alike mode on.

2: Reset command, must be followed by a byte specifying the device to be reset.
- $01: Reset the HBCC device.
- $03: Reset the DLC device.
- $04: Reset FIFOs.

3: Match Table.
- $30: Match status request, report match table contents.
- $31 $7B: Match function off, clear table.
- $32 $xx $yy: Match table entry. $xx - byte position. $yy - byte value.

4:

5: Write HBCC.
- $53 $01 $xx $yy: xx -register address; yy - data value
- $53 $02 $xx $yy: xx -RAM address; yy - data value

6: Read HBCC.
- $63 $01 $xx $yy: xx -register address; yy - number of bytes to read.
- $63 $02 $xx $yy: xx -RAM address; yy - number of bytes to read.

7: Microcontroller write to RAM.
- $73 $xx $yy $zz: Write to address xx yy with data byte zz.

8: Microcontroller read from RAM.
- $83 $xx $yy $zz: Read from address xx yy. Read zz bytes.

9:

A:

B: $B0: Request firmware version number.

C:

D: $D0: Request operational mode.

E: Mode switch command.
- $E1 $33: Switch to VPW mode.
- $E1 $CC: Switch to PWM mode.

F: Re-start microcontroller.
$F1 $A5. (Note: This is not a true reset command.)

5.3.2 Response Header Byte

This byte is prepended onto a packet transmitted by the AVT-931 to the control computer.

Low nibble, bits b3 - b0: Byte count (how many bytes to follow); may be zero when the message is only a status or error message.

High nibble, bits b7 - b4:

0: Valid packet received from the bus.
$0x $yy $zz ...: x - byte count to follow; yy - message number;
zz ... - message bytes.

1: 'Look-alike' mode.
11 00: Look alike mode off.
11 01: Look alike mode on.

2: Error message, error byte(s) follow.

3: Command message error, error header byte follows ($31 $xx).

4: Match function:
$40 - function off.
$41 $B7 - table full, entry ignored
$42 $xx $yy - table entry report. $xx - byte position. $yy - byte value.

5:

6: Read HBCC.
$6x $01 $yy $zz: yy - register start address; zz ... - values.
$6x $02 $yy $zz: yy - RAM start address; zz ... - values.

7:

8: Read microcontroller RAM.
$8x $yy $zz $ww ... yy zz - RAM address; ww... - memory contents.

9: Board status information.
$91 $03: HBCC initializations complete.
$92 $04 $xx: Firmware version report; xx - firmware version number.
$91 $06: PWM operations.
$91 $07: VPW operations.
$91 $08: DLC initializations complete.

A: Message transmitted OK.
$Ax $yy ... ; yy - acknowledger's address.

B:

C: Acknowledgments (monitor mode).
$Cx $yy ...: yy ... - acknowledger's addresses.

D:

E:

F:

5.3.3 'Look Alike' mode

The AVT-931 utilizes two different devices and methods of interfacing to the two busses (VPW and PWM). As a result, messages transmitted and received when in PWM mode are different than if the same message was transmitted or received in VPW mode. Therefore, a 'Look Alike' mode was developed for use while in PWM operations. This mode is defaulted to the Off state.

When 'Look Alike' mode is enabled, the construction of a message is such that it 'looks' like its VPW counterpart. Each of the following sections describes message construction when 'Look Alike' is enabled (on) or disabled (off).

5.3.4 Transmit message format 'Look Alike' mode Off

To transmit a message onto the network the message must be built and then passed to the AVT-931 interface. The HBCC (on the AVT-931) will automatically insert the source address and append the CRC.

All messages destined for transmission onto the network must be constructed as indicated here. All packets for transmission must have a byte count from 2 to 9 (inclusive). Therefore the header byte will have a value of $02 to $09 (inclusive). The bytes following the header byte must be in the following form.

Byte	Description	
00:	Header byte	lower nibble is byte count to follow.
01:	Priority/Type	Priority: 0 - F / Type: 0 - 4 (message type 5 is not implemented as an automatic transmit feature)
02:	Target ID	$00 - $FF
03 - 09:	Data bytes	(optional, 0 to 7 data bytes depending on message type)

5.3.5 Transmit message format 'Look Alike' mode On

To transmit a message onto the network the message must be built and then passed to the AVT-931 interface. The HBCC (on the AVT-931) will automatically append the CRC.

All messages destined for transmission onto the network must be constructed as indicated here. All packets for transmission must have a byte count from 3 to 10 (inclusive). Therefore the header byte will have a value of $03 to $0A (inclusive). The bytes following the header byte must be in the following form.

Byte	Description	
00:	Header byte	lower nibble is byte count to follow.
01:	Priority/Type	Priority: 0 - F / Type: 0 - 4 (message type 5 is not implemented as an automatic transmit feature)
02:	Target ID	$00 - $FF
03:	Source ID	(e.g. a scan tool would be $F1).
04 - 0A:	Data bytes	(optional, 0 to 7 data bytes depending on message type)

An additional function of the 'Look Alike' mode is to check the Priority/Type byte. If this byte has a value of $68 then its value is changed to $61. Next bit #3 is of the Priority/Type byte is cleared. (These changes are useful when viewed in the context of OBD-II standardized messages. Please refer to SAE documents J1979 and J2178 for more details.)

5.3.6 Receive message format 'Look Alike' mode Off

Messages received from the network are assembled into a packet by the AVT-931 interface and then passed to the control computer. These messages will have the following form and will have a byte count less than or equal to $0D.

Byte	Description	
00:	Header byte	lower nibble is byte count to follow.
01:	Message number	according to FMLT order $00: node to node $FF: network monitor
02:	Priority/Type	
03:	Target Specifier	
04:	Source Address	
05 to n-1	Data bytes	
n	CRC	
(n <= $D)		

5.3.7 Receive message format 'Look Alike' mode On

Messages received from the network are assembled into a packet by the AVT-931 interface and then passed to the control computer. These messages will have the following form and will have a byte count less than or equal to $0C. With 'Look Alike' mode enabled the CRC byte is dropped when the message is forwarded to the host.

Byte	Description	
00:	Header byte	lower nibble is byte count to follow.
01:	Message number	according to FMLT order $00: node to node $FF: network monitor
02:	Priority/Type	
03:	Target Specifier	
04:	Source Address	
05 to n	Data bytes	
(n <= $C)		

5.3.8 Monitor Mode

The HBCC device is capable of being placed into a 'monitor mode.' When in monitor mode all bus traffic is observed and passed to the host; including IFR message acknowledgments.

Enter Monitor Mode

To place the HBCC into monitor mode, send the following command to the AVT-931: `$53 $01 $02 $16` and receive the response: `$63 $01 $02 $16`.

Exit Monitor Mode

To exit monitor mode and place the HBCC device back into normal operations send the following command to the AVT-931:
`$53 $01 $02 $72` and receive the response: `$63 $01 $02 $72`.

5.3.9 HBCC Initialization Parameters

Upon startup of the AVT-931 interface in PWM mode, the HBCC device is initialized and a local loopback test is conducted. If the local loopback test passes successfully, the HBCC device operational parameters are then set to default parameters and the HBCC network drivers are enabled.

In the event the HBCC local loopback test fails for any reason an error code is sent to the control computer and the HBCC network drivers are disabled.

A number of HBCC registers and RAM locations are initialized when the AVT-931 starts. The following is a list of these registers and RAM locations along a brief explanation of the initialization status and the value written to each location. The user should consult the HBCC User's Guide for detailed information on each of the registers, RAM, and individual bit definitions.

Name	Address	Default Value	Description
UIMR User Interrupt Mask Register	Register $07	$FF	All user interrupt disabled
RCR Receive Control Register	Register $01	$04	Receive OK interrupt mask Receive error interrupt mask Receiver overrun interrupt mask Unable to acknowledge interrupt mask Network fault interrupt mask
HCR HBCC Control Register	Register $02	$72	Enable network driver A Enable network driver B 41.6 kbps never sleep
TCR Transmit Control Register	Register $00	$0x	Transmit OK interrupt mask Transmit error interrupt mask Critical transmit error interrupt mask
NAR Node Address Register	Register $10	$F1	OBD-II diagnostic tool address
FMLT Function Message Lookup Table	RAM $10	$5A	OBD-II diagnostic message.
	RAM $11	$5B	OBD-II diagnostic message.
	RAM $12	$6A	OBD-II diagnostic message.
	RAM $13	$6B	OBD-II diagnostic message.
FRMLT Function Read Message Lookup Table	RAM $14	$04	
	RAM $15	$06	

FRDR1 Function Read Data Register #1	Register $08	$B2	
FRDR2 Function Read Data Register #2	Register $09	$B4	
FRDR3 Function Read Data Register #3	Register $0A	$B6	
FRMLTP Function Read Message Lookup Table Pointer	Register $11	$14	
SUR Start of User's RAM	Register $12	$16	

5.3.10 Examples

To illustrate the construction and decoding of messages between the control computer and the AVT-931 interface, several examples are shown for illustration.

Example #1: Want to write a byte to a register in the HBCC device.
Command string: `53 01 02 16`.
Explanation: header byte `53`, `5` is the command for HBCC write and `3` is the number of bytes to follow. `01` is the register specifier. `02` is the address of the register to be written in the HBCC. `16` is the value of the byte to be written into HBCC register at address `02`. (This is the command to put the HBCC device into the network monitor mode.)

Example #2: Want to make an OBD-II engine temperature query with 'Look Alike' off.
Command string: `04 61 6A 01 05`.
Explanation: header byte `04`, `0` indicates that the following bytes are for transmission onto the network by the HBCC and 4 indicates that 4 bytes follow. `61` indicates a priority 6 message of type 1 (broadcast). `6A` indicates that the message is only destined to those nodes that recognize function 6A messages (diagnostic message). The last two bytes '01' and '05' are the actual message data bytes (mode 1 PID 5; refer to SAE J1979 for more information).

Example #3: Want to make an OBD-II engine temperature query with 'Look Alike' on.
Command string: `05 68 6A F1 01 05`.
Explanation: header byte `05`, `0` indicates that the following bytes are for transmission onto the network by the HBCC and 5 indicates that 5 bytes follow. `68` indicates a priority 6 message and type 8 (with 'Look Alike' on this byte is replaced with $61). `F1` is the source ID. `6A` indicates that the message is only destined to those nodes that recognize function 6A messages (diagnostic message). The last two bytes '01' and '05' are the actual message data bytes (mode 1 PID 5; refer to SAE J1979 for more information).

Example #4: Send the message of example #2 and receive the response: `22 09 40`.
Explanation: header byte `22`, 2 indicates HBCC error, 2 indicates that two bytes follow. `09` is the error code for the HBCC and indicates that an interrupt register 2 (IR2) error occurred. The last byte is the contents of IR2. A value of `40` in IR2 indicates that a transmit error limit expired.

This error indicates that no node responded with confirmation of reception of the message (no node answered).

Example #5: Send the message of example #2 and receive the response: `41 C4`. Explanation: header byte `41`, 4 indicates that the message was transmitted OK (at least one node responded) and the 1 indicates one byte follows. The byte `C4` is the address of the node that responded to the transmitted message (node C4 responded affirmatively to receipt of the transmitted message).

5.4 Error Codes

These error codes are sent as a separate byte following the byte containing the flag and count.

$00: Reserved.
$01: HBCC: Initialization error, response not = 01.
$02: HBCC: No message received OK flag, from loopback test.
$03: HBCC: No message transmitted OK flag, from loopback test.
$04: HBCC: Test message byte #1 error, from loopback test.
$05: HBCC: Test message byte #3 error, from loopback test.
$06: HBCC: No IRQ detected.
$07: Reserved.
$08: HBCC: IR1 error, IR1 byte follows.
$09: HBCC: IR2 error, IR2 byte follows.
$0A: HBCC: IR3 error, IR3 byte follows.
$0B: No buffers available, error bits follow.
$0C: Reserved.
$0D: Transmit message too short.
$0E: Transmit message too long.
$0F: Reserved.

$10: DLC: Initialization failure, no interrupt.
$11: DLC: Initialization failure, status byte error.
$12: DLC: Initialization failure, test byte #1 error.
$13: DLC: Initialization failure, test byte #2 error.
$14: DLC: Initialization failure, test byte #3 error.
$15: DLC: Initialization failure, not ‘completion code only’.
$16: DLC: Initialization failure, completion code incorrect.
$17: DLC: Receive FIFO overflow.
$18: DLC: No buffers available.
$19: VPW: Short to Ground detected (DLC NETF bit).
$1A: VPW: Short to High detected.
$1B: DLC: Break received.

5.5 Hardware Register #2 Definition

Hardware Register #2 on the interface board can only be accessed by host memory read and write operations via the ISA bus. The register is read/write capable, however only bit #7, "!Microcontroller Reset," can be written. All other bits are read only.

Bit	Function	State	
0:	! FIFO #1 Empty Flag	active low	read only.
1:	! FIFO #1 Full Flag	active low	read only.
2:	not used, always reads a '1'.		
3:	! FIFO #2 Empty Flag	active low	read only.
4:	! FIFO #2 Full Flag	active low	read only.
5:	not used, always reads a '1'.		
6:	not used, always reads a '1'.		
7:	! Microcontroller Reset	active low	read /write.

Upon power-up this register is cleared. It is the responsibility of the host to write the correct value into the register for the desired operations.

Through this register the host controls the reset line for the AVT-931 microcontroller, and thus controls whether the interface board is 'on' or 'off.' The state of the reset line is determined by writing the appropriate value into bit #7 of the register. All other bits are read only and not affected by writes.

To stop the AVT-931

write $00 into Hardware Register #2
bit #7 is cleared, the microcontroller is halted.

To start the AVT-931

write $80 into Hardware Register #2
bit #7 is set, the microcontroller commences operations.

5.6 ISA Bus Memory Map

The AVT-931 Dual J1850 Interface board is a memory mapped device and is accessed via memory operations through the ISA bus. The base or segment address of the board is determined by the setting of DIP switches on the board. The board occupies 16 bytes of memory space, only 3 bytes are actually used. The board contains Hardware Register #2, FIFO #1, and FIFO #2; each of which occupy one address. The address of each of these functions is listed relative to the base address of the board. All unused addresses are not accessible and are not listed.

Address	Function	
$00	Hardware Register #2	read / write
$01	FIFO #1	write only
$02	FIFO #2	read only

When reading or writing the FIFOs the user should be aware of the respective full and empty flags for the FIFOs. If a read is attempted from an empty FIFO #2 an invalid data byte will be obtained. If a write is attempted to a full FIFO #1 the data will be discarded. No other means are provided to indicate that data was lost or invalid data was read.

6. References

This section contains reference information on the J1850 specification, the HBCC device, the Standard Corporate Protocol, and related technical information. These documents should be consulted for additional or detailed information. Note that Advanced Vehicle Technologies, Inc. is not a source for these documents.

1. SAE Standard J1850 "Class B Data Communications Network Interface." Available from the SAE, 400 Commonwealth Drive, Warrendale, PA 15096-0001; phone: 724-776-4970.
2. Ford Motor Company "Hosted Bus Controller Chip User's Guide."
3. Ford Motor Company "SCP Protocol Definition and Interface Requirements." Document number: ES-F7LC-12K529-BA.
4. Motorola MC68HC56 / MC68HC57 Data Link Controllers Technical Data. Publication: MC68HC56/57/D.

6.1 Technical Support

The user may contact Advanced Vehicle Technologies, Inc. for assistance in any of the areas covered here. When calling please be prepared to identify yourself, tell us the serial number of your hardware, and the version number of the firmware you are running.

Advanced Vehicle Technologies, Inc. is located in Maryland and is open from 0800 hrs. to 1800 hrs. Eastern Time. If calling after hours, please leave a message and we will return your call as quickly as possible.

You may also fax your questions to us. We will either fax an answer back or call you, at your request. If faxing your question, please include as much relevant information about your question or problem as you can.

We can be contacted by phone at:

Voice:	410-798-4038
Fax:	410-798-4308
or by E-mail at:	avt-inc@ari.net

Our Home Page on the World Wide Web can be reached at:
http://www2.ari.net/avt-inc/

6.2 Interface Information

The following sections contain information about the AVT-931 Dual J1850 Interface board, the connectors, and the jumpers on the board.

6.2.1 Interface Board P1

P1 on the AVT-931 Interface Board contains spare or unused microcontroller I/O ports as well as host computer +5 volts and ground. This connection is provided for future test and expansion options. Connector P1 is a 20 position right angle pin header. The pins are on a 0.1 inch grid and are 0.025 inch square. This header is compatible with a ribbon cable IDC connector such as AMP # 111918-4.

Pin Number	Name	Pin Number	Name
1	+5 VDC	11	VRL
2	Ground	12	PA7
3	/XIRQ	13	PD5
4	PE2	14	PH2
5	PE3	15	PH3
6	PE4	16	XOUT
7	PE5	17	RXD
8	PE6	18	TXD
9	PE7	19	Ground
10	VRH	20	E-CLOCK

6.2.2 Interface Board P2

P2 on the AVT-931 Interface Board is the connection to the vehicle (normally through the vehicle's OBD-II connector). Connector P2 is a 16 position right angle pin header. The pins are on a 0.1 inch grid and are 0.025 inch square. This header is compatible with a ribbon cable IDC connector such as AMP # 111918-3.

Only the indicated pins are connected, all others are 'not connected.'

Pin Number	Name
3	J1850 Bus + [OBD-II connector pin #2].
4	J1850 Bus - [OBD-II connector pin #10].
7	Vehicle ground (isolated from host computer ground) [OBD-II connector pin #4].
9	Vehicle ground (isolated from host computer ground) [OBD-II connector pin #5].
10	Vehicle unswitched battery (isolated from host computer) [OBD-II connector pin #16].

7. Company Overview

Advanced Vehicle Technologies, Inc. is dedicated to providing affordable hardware, software, and technical support to the developers and users of vehicle based multiplex networks.

AVT, Inc. also offers other vehicle multiplex bus products including:

- AVT-715 Dual J1850 Interface. Stand alone unit with serial connection to a host computer. Also available is a Chrysler CCD add-on board for the AVT-715 unit.
- AVT-716 Multiple Interface (J1850 VPW, PWM, ISO 9141, Keyword Protocol 2000, and GM 8192 UART). Stand alone unit with serial connection to a host computer.
- AVT-921 Dual J1850 Interface for ISA bus. A 3/4 sized ISA bus board.
- AVT-815 J1850 VPW Trainer / Development kit.

Contact the factory or review our World Wide Web site for information on these products and our latest offerings.

The engineering staff at AVT, Inc. is experienced with multiplex bus standards including: J1850 VPW and PWM and ISO-9141 and 9141-2. Members of the staff are available to provide assistance on the use of any of AVT's products.

AVT engineering staff members are available to provide dedicated engineering support for a customer project. Through a simple contractual arrangement, a customer is able to 'tap' into AVT's knowledge and experience.

Information on any of the products or engineering support that Advanced Vehicle Technologies can provide is available by calling, faxing, or writing.

Advanced Vehicle Technologies, Inc.
1509 Manor View Road, Davidsonville, MD 21035
410-798-4038 (voice)
410-798-4308 (fax)

Home page: http://www2.ari.net/avt-inc/
E-mail: avt-inc@ari.net

Revision Record:

A1: Original release.