ADVANCED VEHICLE TECHNOLOGIES, Inc. AVT inc.

AVT-715

Dual J1850 INTERFACE (VPW and PWM)

RS-232/422 Unit

User's Manual

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

Revision 'C'

July 1997

TABLE OF CONTENTS

1. INTRODUCTION

This manual covers the AVT-715 hardware interface. It provides technical information on using the interface, connections to it, communications with it, and other related information. If using the AVT Controller or Capture software, please refer to the:

"AVT-XXX Controller Software User's Manual."

The AVT-715 ready-to-use package consists of the hardware interface board housed in a rugged enclosure; the Controller software; a serial cable; an OBD-II compatible cable; a User's Manual for the hardware interface; and a User's Manual for the Controller Software. Connecting the AVT-715 interface to a subject vehicle is quick and easy and no tools are required. Also necessary for the operation and use of the AVT-715 interface, but not provided, is a host computer with either an RS-232 or an RS-422 serial port.

The AVT-715 OEM module consists of the AVT-715 interface board only. All connections to the interface board are identified in Section 6.3.

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-715 interface implements both versions of the J1850 standard.

The AVT-715 implementation of the PWM interface conforms to the Ford Motor Company Standard Corporate Protocol (SCP) which defines network traffic management, message construction, and other protocol issues. The AVT-715 is fully compatible with any similarly equipped Ford Motor Company or Mazda product.

The AVT-715 Dual J1850 Interface unit provides the following functions:

- Isolated electrical interface between the subject vehicle and the control computer.
- Protocol/data conversion between the vehicle J1850 multiplex bus and the serial data link to the control computer.
- Passive network traffic monitor.
- Performs all functions of an active SCP qualified network node.

1.1 Specifications

Enclosure:

- Overall size (inches): 6.7 wide x 2.2 high
  x 4.75 deep (5.4 including switch and connectors).
- Weight: 17 oz.

OEM board:

- Overall size (inches): 5.6 wide x 4.0 deep x 1.0 high.
- Weight: 5 oz.

- +12 volts DC (nominal) from subject vehicle.
- Input voltage range: +7.0 to +24.0
- Power dissipation: 1.8 watts (nominal).

1.2 Definitions

The following terms are used in this manual.

- HBCC: Hosted Bus Controller Chip (Motorola / Ford Motor Company).
- SCC: Serial Communications Controller 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

Prior to using the AVT-715 the type and baud rate of the serial link between it and the host computer must be properly set. The interface must then be connected to both the subject vehicle and the control computer.

2.1 Hardware Configuration

The serial communications link between the AVT-715 interface and the control computer can be configured for either RS-232 or RS-422 operation . Additionally, the data rate can be configured for 9.6k, 19.2k, 38.4k, or 57.6k baud rate. These configuration selections are set by jumpers and connectors on the interface board.
(Note: documentation by other manufacturers may make reference to 56k baud. This is usually the same as what we refer to as 57.6k baud, the actual baud rate.)

The factory default settings for the AVT-715 interface are for RS-232 operation at a 9.6k baud rate.

To select RS-232 operation:

Locate jumpers JP3 and JP4 on the interface board.
Place the jumper across pins 2 and 3 on JP3
Place the jumper across pins 2 and 3 on JP4
Connect the internal serial cable to P1 on the interface board.
(The internal serial cable is a ribbon cable connected to the enclosure mounted connector P2, it's the narrower of the two cables.)

To select RS-422 operation:

Locate jumpers JP3 and JP4 on the interface board.
Place the jumper across pins 1 and 2 on JP3
Place the jumper across pins 1 and 2 on JP4
Connect the internal serial cable to P2 on the interface board.
(The internal serial cable is a ribbon cable connected to the enclosure mounted connector P2, it's the narrower of the two cables.)

The baud rate is determined by installing or removing the jumpers on JP1 and JP2 on the AVT-715 interface board. Refer to Table 1 to determine the settings of JP1 and JP2 for the desired baud rate.

Baud Rate	JP1	JP2
9.6k	In	In
19.2k	Out	In
38.4k	In	Out
57.6k	Out	Out

Table 1 Baud Rate Selection

2.2 Hardware Connections

Connecting the AVT-715 to the subject vehicle is accomplished via the OBD-II compatible cable available from AVT, Inc. (Note: this cable is not supplied with the OEM module.) The cable assembly has a DA-15S connector on one end and an OBD-II compatible connector on the other end.

The DA-15S connector is mated to the DA-15P connector on the AVT-715 enclosure. The other end of the cable assembly is mated to the OBD-II connector in the subject vehicle. (Refer to SAE Standard J1962 for a description of the OBD-II connector and its location in a vehicle.) The AVT-715 interface receives operating power from the subject vehicle through this cable assembly as well as power and signal grounds. Additionally, the cable provides J1850 bus connections between the interface and the subject vehicle.

The AVT-715 interface is connected to the control computer via the supplied 9 pin cable assembly. (Note: this cable is not supplied with the AVT-715 OEM module.) The cable is compatible with a PC-AT standard 9 pin RS-232 connection. We cannot guarantee it, but direct connection of the AVT-715 interface enclosure to a PC-AT 9 pin RS-232 connection using this cable (or any straight through wired cable) should work without requiring any adapters.

No other connections or installation is required.

2.3 Testing the AVT-715

Once the hardware connections have been made, proper operation of the hardware can be observed.

- Launch the AVT-715 Controller software or a similar communications program. *Note that the AVT-715 communicates with the host computer using binary bytes (not ASCII digits).*
- Set the parameters of the communications program appropriately (as determined by the configuration of the AVT-715 interface).
- Turn on the AVT-715 interface unit.
- Observe on the interface board that the green (power) LED is lit.
- Observe that the red (operations) LED is blinking fast.

- Observe on the computer that the following is received from the interface: `91 05`. *This is the 'mode of operation' report where $91 indicates that the message is a 'board status report' with one byte to follow. The $05 indicates that the board status is VPW mode of operation. Refer to Sections 5.4 and 5.5 for commands to and responses from the AVT-715 interface.*

  Note: The Controller software only displays and accepts hex digits, no dollar signs.
- At this point the AVT-715 hardware is operational.

3. Interface Introduction

The AVT-715 Dual J1850 Interface is available housed in a rugged polycarbonate enclosure with internal cable assemblies and power switch. The two connections to the enclosure are to the subject vehicle and the control computer.

The interface provides an isolated electrical interface between the control computer and the vehicle. The AVT-715 interface is connected directly to the vehicle and derives its operating power from the vehicle. The serial interface to the control computer is electrically isolated from the vehicle electrical and electronic systems. This is done to prevent damage to the control computer due to spikes or surges from the vehicle electrical system. It is not recommended that the control computer be connected directly to the vehicle electrical system.

The AVT-715 hardware interface performs the data and protocol conversion functions. When in VPW mode the interface utilizes the Harris HIP7030A0 J1850 VPW microcontroller device along with the Harris HIP7020 bus transceiver. When in PWM mode the AVT-715 utilizes the Motorola/Ford HBCC device and a Ford bus transceiver design.

The serial communications interface function of the interface is provided by a Philips Serial Communications Controller (SCC) device. The RS-232 and RS-422 line drivers and receivers are contained in an electrically isolated section of the AVT-715 interface board. Signals are coupled through optical isolators and power is supplied by an isolated DC-DC converter.

The AVT-715 interface uses an embedded 68HC05 core microcontroller (part of the HIP7030A0 device) to perform the necessary internal control and communications functions. The interface is capable of operating in either VPW or PWM mode. The selection of the mode of operation is determined by software commands issued from the control computer.

When in VPW mode the interface is always listening to, or monitoring, the J1850 bus. All bus traffic is reported to the control computer by the interface. 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 for proper operation, including a unique node address. The HBCC is initialized automatically by the microcontroller on the interface when powered up or reset. Refer to Section 5.5.8 for a listing of the initialization parameters for the HBCC device. These parameters may be changed by the user from the control computer.

When in PWM mode of operation the HBCC device performs input message filtering. Therefore only bus traffic that is destined for the address of the interface is passed to the control computer and displayed. To view all bus traffic while in PWM mode, the HBCC device can be placed into monitor mode by the Controller software. When in this mode the AVT-715 interface becomes a passive PWM network monitor and will not permit transmissions.

4. Interface Description

A block diagram of the AVT-715 interface is shown in Figure 1. The heart of the unit is the HIP7030A0 microcontroller. This device utilizes a 68HC05 core with a bus speed of 5 MHz. The operation firmware is contained in an EPROM that is organized into two pages. One code page is for VPW operations only while the other code page is for PWM operations only. Software commands from the control computer control the selection of the operational mode of the microcontroller, and hence the interface.

The microcontroller utilizes several peripheral devices: two FIFO's each 4 Kbytes deep, the HBCC device, and a serial communications controller.

The two FIFO's are used by the microcontroller as buffers during transmit and receive operations between itself and the control computer. A serial communications controller is used to implement the standard serial data link, including RTS/CTS hardware handshaking. The serial communications controller is interfaced to the control computer through a signal isolation block and either an RS-232 or an RS-422 transceiver block.

The HBCC device implements the J1850 PWM scheme that is Ford SCP compliant. The HBCC is connected to the subject vehicle bus through a bus transceiver block.

The HIP7030A0 device contains an on-board symbol encoder/decoder functional block that implements the J1850 VPW scheme. This VPW functional block is interfaced to the subject vehicle through a Harris HIP7020 J1850 VPW bus transceiver device.

The control computer serial data link is electrically isolated from the rest of the interface and the vehicle through an isolated DC-to-DC power converter and a signal isolation block.

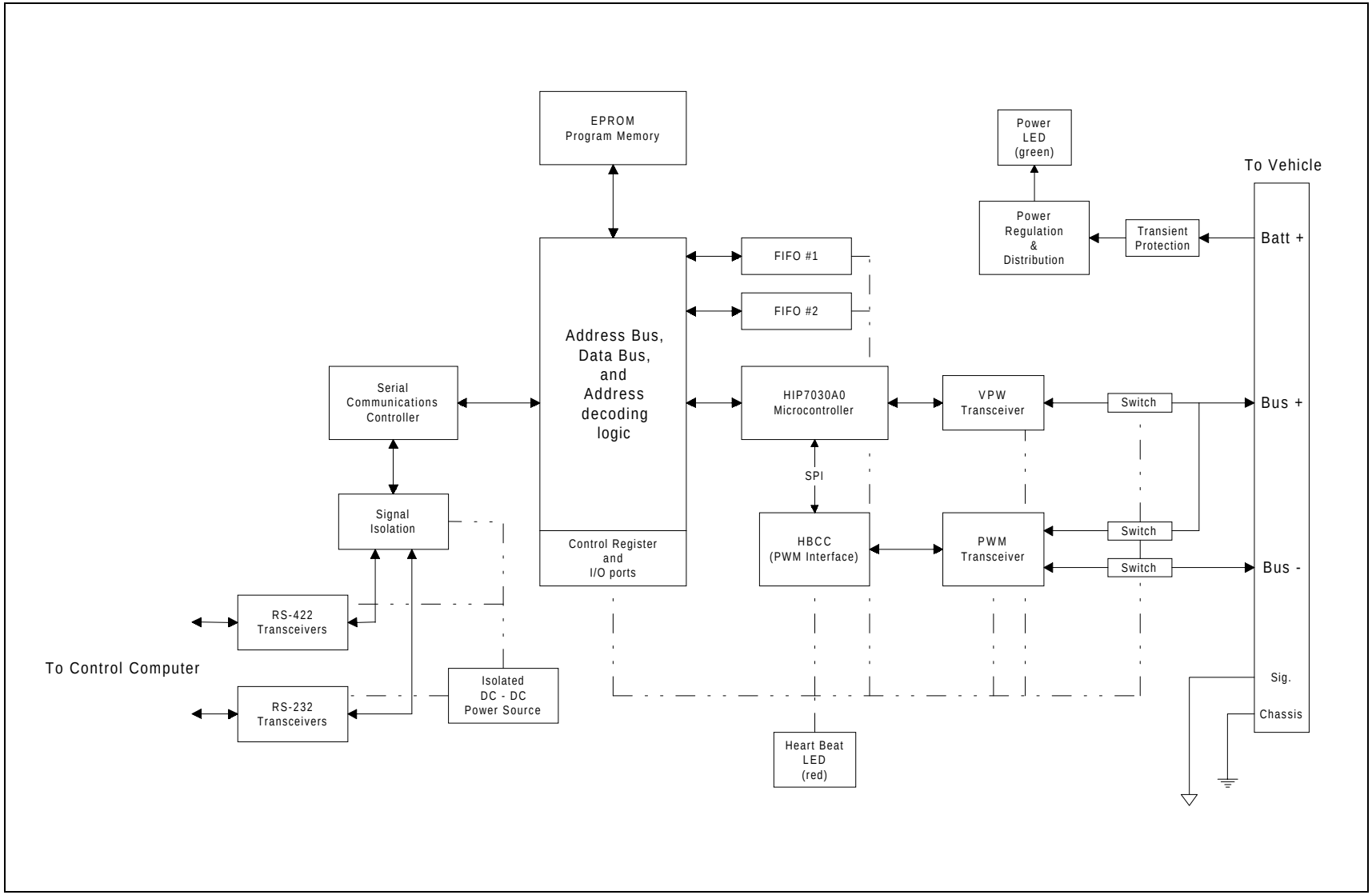

Figure 1 AVT-715 Hardware Interface Unit Block Diagram

5. AVT-715 Operation

The following describes the use of the AVT-715 Dual J1850 Interface. It is assumed that the interface is properly connected to the subject vehicle and to the control computer. Furthermore, it is assumed that the AVT-715 Controller software (or similar communications software) has been installed on the control computer, that it is running, and that the test described in Section 2.3 has been completed successfully.

5.1 Indicators

The AVT-715 has two indicator LED's on the board, one green and one red.

The green LED is connected to the +5 VDC supply for the board and provides a quick indication that power is available for normal operation. If the green LED should fail to light, check the power source from the subject vehicle, and check fuse F1 on the AVT-715 board.

The red LED is a heartbeat indicator. The microcontroller toggles the state of the red LED every 52 milliseconds. As a result the red LED flashes noticeably during normal operations. If a problem with the microcontroller should occur the LED will either go to a full ON or full OFF state. This should be readily apparent and be indicative of an abnormal condition.

5.2 Communications, Electrical

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

All communications between the AVT-715 interface and the control computer conform, at the physical interface, to either EIA-RS-232 or RS-422 standards (as selected by the user). Communications between the interface unit and the control computer follow industry standard serial communications protocol. There is one start bit, eight data bits (least significant bit first), no parity, and one stop bit. Additionally, hardware handshaking using the RTS/CTS signal lines is used.

When in RS-232 mode the output data line, TXD, idles low (-8 volts) and the RTS signal line idles high (+8 volts). Likewise, the AVT-715 expects RXD to idle low and CTS to idle high. (All voltages measured with respect to the communications isolated ground.) The user should be aware of potential communications errors that may occur when using RS-232 at the higher data rates or in an electrically noisy environment.

When in RS-422 mode the output data lines idle such that TXD+ is high (+4 volts) and TXD- is low (0 volts). The RTS+ signal idles low (0 volts) and RTS- is high (+4 volts). Likewise the AVT-715 expects RXD+ to idle high (RXD- is low) and CTS+ is low (CTS- is high). (All voltages measured with respect to the communications isolated ground.)

5.3 Communications, Messages

The structure and protocol of communications between the control computer and the interface 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 control computer and the microcontroller in the interface.

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 control computer.

Messages from the control computer to the AVT-715 interface are known as Commands. Message from the interface to the control computer are known as Responses.

5.4 VPW Mode

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

5.4.1 Command Header Byte

This byte is prepended onto a packet transmitted by the control computer to the interface. Only those commands shown are valid. All others are reserved and should not be used.

Low nibble, bits b3 - b0: Byte count, how many bytes to follow. May be zero.

High nibble, bits b7 - b4:

0: This is a valid packet for transmission (the null byte '00' is ignored).
1: Reserved.
2: Reset command, must be followed by a byte specifying the device to be reset.
    $01: Reset the HBCC device.
    $02: Reset the SCC device.
    $04: Reset FIFO #1.
    $05: Reset FIFO #2.
3: Match function:
    $30 - status request; report match table entries.
    $31 $7B - function off, clear table.
    $32 $xx $yy - table entry. $xx - byte position. $yy - byte value.
4: Reserved.
5: Reserved.
6: Reserved.
7: HIP7030A0 write RAM (address byte, data byte, follow).
8: HIP7030A0 read RAM (address byte, requested bytes [<=$0E], follow).
9: Hardware Register #1 write (data byte follows).
A: Hardware Register #1 read.
B: Request firmware version number.
C: Reserved.
D: Request operational mode.
E: Switch to PWM mode, must be followed by $CC. [$E1 $CC]
F: Hardware unit Re-Start command, must be followed by $A5. [$F1 $A5]

5.4.2 Response Header Byte

This byte is prepended onto a packet transmitted from the interface to the control computer.

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

High nibble, bits b7 - b4:

- 0: This is a valid packet received from the bus.
- 1: Reserved.
- 2: Error message, error byte(s) follow (refer to Section 5.7).
- 3: Command message error, error header byte follows. ($31 $xx).
- 4: Match function response:
  - $40 - function off.
  - $41 $B7 - table full, entry ignored.
  - $42 $xx $yy - table entry report. $xx - byte position. $yy - byte value.
- 5: Reserved.
- 6: Reserved.
- 7: HIP7030A0 read RAM error, requested byte count greater than $0E.
- 8: HIP7030A0 read RAM (address byte, data bytes, follow).
- 9: Board status information byte follows:
  - $02: SCC initializations complete.
  - $03: HBCC initializations complete.
  - $04: Firmware version number follows.
  - $05: VPW operation (Harris).
  - $06: PWM operation (Motorola).
- A: Message status information. Received message status byte follows. ($A1 $10 - lost arbitration, message flushed.)
- B: Reserved.
- C: Reserved.
- D: Hardware Register #1 read, data byte follows.
- E: Errors: $E0: buffer open failure; $E1: SENDEC error, SEDSR follows.
- F: Reserved.

5.4.3 Transmit message format

To transmit a message onto the network the message must be built by the operator and then sent to the interface unit. The microcontroller (HIP7030A0) in the interface will automatically append the transmit CRC.

It is up to the user to determine and know the proper protocol that is implemented on the vehicle to which the AVT-715 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-715 interface for transmission the proper transmit CRC byte is appended and transmission of the message commences as soon as possible. (J1850 requires that a minimum IFS [Inter-Frame Separation] time interval has elapsed on the bus before any node may begin transmission of a new message.)

If, while attempting to transmit a message, the message loses arbitration the winning received message will have the 'lost arbitration' bit set in the received status byte. (Refer to Section 5.4.6 for more details on the received message status byte.)

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 sent to the control computer. (Example: a message is sent to the interface for transmission, at some later time the interface will report: $01 $60 which indicates that the message was transmitted correctly and that the message was received from itself.)

The AVT-715 interface will attempt to transmit a message three times. If the message loses arbitration three times the message is discarded and a message transmission failure is reported to the control computer. The arbitration failure code is $A1 $10

5.4.4 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). This byte sequence is then sent to the host computer.

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 passed 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 in the next section.)
The message bytes then follow. Note that the CRC bytes are stripped.

5.4.5 Match Function

A coarse filtering mechanism for messages received from the bus is provided by the AVT-715 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-715 interface. 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.

5.4.6 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 (message too short or too long).

5.4.7 Examples

To illustrate the construction and decoding of messages between the control computer and the AVT-715 interface several examples are shown for illustration.

Example #1: Want to request the current operational mode.
Command string: `D0`.
The interface responds with: `91 05`. The '`9`' indicates a board response, the '`1`' indicates one byte follows, and the `05` indicates VPW mode.

Example #2: want to send a message out on the bus.
Command string: `04 32 89 AC 5F`.
The interface responds 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 interface are received by the interface, are checked for errors, but are not echoed back to the controller. Only a status byte is passed to the controller to indicate the status of the transmitted message.)

5.5 PWM Mode

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

5.5.1 Command Header Byte

This byte is prepended onto a packet transmitted by the control computer to the interface. Only those commands shown are valid. All others are reserved and should not be used.

Low nibble, bits b3 - b0: Byte count, how many bytes to follow. May be zero.

High nibble, bits b7 - b4:

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

1: 'Look Alike' mode.
$10: mode request.
$11 $00: mode off.
$11 $01: mode on.

2: Reset command, must be followed by a byte specifying the device to be reset.
$01: reset the HBCC
$02: reset the SCC
$04: reset FIFO #1
$05: reset FIFO #2

3: Match function:
$30 - function off.
$31 $B7 - table full, entry ignored.
$32 $xx $yy - table entry report. $xx - byte position. $yy - byte value.

4: Reserved.

5: HBCC write register (address byte, data byte, follow).

6: HBCC read register (address byte, requested bytes [< = $0E], follow).

7: HBCC write RAM (address byte, data byte, follow).

8: HBCC read RAM (address byte, requested bytes [< = $0E], follow).

9: Hardware Register #1 write (data byte follows).

A: Hardware Register #1 read.

B: Request firmware version number.

C: Reserved.

D: Request operational mode.

E: Switch to VPW mode, must be followed by $33. [$E1 $33]

F: Hardware unit Re-Start command, must be followed by $A5. [$F1 $A5]

5.5.2 Response Header Byte

This byte is prepended onto a packet transmitted by the interface unit to the control computer.

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

High nibble, bits b7 - b4:

- 0: This is a valid packet received from the HBCC.
- 1: 'Look Alike' mode response.
  - $11 $00: mode off.
  - $11 $01: mode on.
- 2: Error message, error byte(s) follow (refer to Section 5.7).
- 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: HBCC read register error, requested byte count greater than $0E.
- 6: HBCC read register (address byte, data bytes, follow).
- 7: HBCC read RAM error, requested byte count greater than $0E.
- 8: HBCC read RAM (address byte, data bytes, follow).
- 9: Board status information byte follows:
  - $02: SCC initializations complete.
  - $03: HBCC initializations complete.
  - $04: Firmware version number follows.
  - $05: VPW operation (Harris).
  - $06: PWM operation (Motorola).
- A: Message transmitted OK, acknowledgment bytes follow.
- B: Reserved.
- C: Monitor mode, acknowledgers list.
- D: Hardware Register #1 read, data byte follows.
- E: Reserved.
- F: Reserved.

5.5.3 'Look Alike' mode

The AVT-715 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 (to be compatible with previous versions of unit firmware).

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).

5.5.4 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-715 device. The HBCC (internal to the AVT-715) 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)

5.5.5 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-715 device. The HBCC (internal to the AVT-715) 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.)

5.5.6 Receive message format 'Look Alike' mode Off

Messages received from the network are assembled into a packet by the AVT-715 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)		

5.5.7 Receive message format 'Look Alike' mode On

Messages received from the network are assembled into a packet by the AVT-715 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)		

5.5.8 HBCC Initialization Parameters

When the AVT-715 interface is powered up, 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 passed to the control computer and the HBCC network drivers are disabled.

A number of HBCC registers and RAM locations are initialized when the AVT-715 interface 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.

UIMR (user interrupt mask register):	All user interrupts disabled $FF	
RCR (receive control register):	Receive OK interrupt mask Receive error interrupt mask Receiver overrun interrupt mask Unable to acknowledge interrupt mask Network fault interrupt mask $04	
HCR (HBCC control register):	Enable network driver A Enable network driver B 41.6 kbps never sleep $72	
TCR (transmit control register):	Transmit OK interrupt mask Transmit error interrupt mask Critical transmit error interrupt mask $0x	
NAR (node address register):	$F1	
FMLT (function message lookup table):	$5A $5B $6A $6B	(RAM addr: $10) (RAM addr: $11) (RAM addr: $12) (RAM addr: $13)
FRMLT (function read message lookup table):	$04 $06	(RAM addr: $14) (RAM addr: $15)
FRDR1 (function read data register #1): FRDR2 (function read data register #2): FRDR3 (function read data register #3):	$B2 $B4 $B6	(reg. addr: $08) (reg. addr: $09) (reg. addr: $0A)
FRMLTP (function read message lookup table pointer):	$14	
SUR (start of user's RAM):	$16	

5.5.9 Examples

To illustrate the construction and decoding of messages between the control computer and the AVT-715 interface, several examples are shown for illustration.

Example #1: Want to write a byte to a register in the HBCC device.
Command string: `52 02 16`.
Explanation: header byte `52`, `5` is the command for HBCC write register and `2` is the number of bytes to follow. `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).

5.6 Hardware Register Definition

The AVT-715 interface board contains one hardware register that is used by the microcontroller to access several physical devices on the board. The user may access the hardware register through the $9x and $Ax commands. The following are the bit definitions for the hardware register.

Bit	Function
0:	! VPW Loopback enable (active low); read/write.
1:	VPW bus relay (active high); read/write.
2:	PWM bus + relay (active high); read/write.
3:	PWM bus - relay (active high); read/write.
4:	not defined; read/write.
5:	not defined; read/write.
6:	jumper JP1, baud rate select bit 0; read only.
7:	jumper JP2, baud rate select bit 1; read only.

5.7 Error Codes

These error codes are passed as a separate byte following the byte containing the flag and count. Only the codes used are shown.

$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.
$08: HBCC: IR1 error, IR1 byte follows.
$09: HBCC: IR2 error, IR2 byte follows.
$0A: HBCC: IR3 error, IR3 byte follows.
$0B: No buffer available, error bits follow.
$0D: Transmit message too short.
$0E: Transmit message too long.
$0F: SCC: FIFO #1 overflow.
$31: Received a Watchdog time-out. (AVT-716, PWM mode only).

6. Technical Information

6.1 Reference Documentation

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.

1. SAE Standard J1850 “Class B Data Communications Network Interface.”
   SAE Standards J1979, J2178, and many others.
   All are available from the SAE, 400 Commonwealth Drive, Warrendale, PA 15096-0001; phone: 412-776-4970.
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. Ford Motor Company “SCP Diagnostic Message and Dialogue Requirements.” Document number: ES-F7LC-12K529-CA.
5. Automotive Multiplexing Technology.
   Papers from the 1995 SAE World Congress. SAE publication SP-1070.
   Papers from the 1996 SAE World Congress. SAE publication SP-1137
   Papers from the 1997 SAE World Congress. SAE publication SP-1224
   All are available from the SAE, 400 Commonwealth Drive, Warrendale, PA 15096-0001; phone: 412-776-4970

6.2 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/

6.3 AVT-715 Interface Board Information

Information about the enclosure is contained in Section 6.4.

The following sections contain information about the AVT-715 interface board, the connectors, and the jumpers on the board.

A description of all versions of the firmware for the AVT-715 interface board is kept up to date and posted on the web site. It can be accessed from the home page under the AVT-715 product description section.

6.3.1 Interface Board P1

P1 on the interface board is the RS-232 serial port connection. The connector is a 10 position header with pins on 0.100 inch centers. The header is compatible with an AMP #111634-1 a 10 position IDC ribbon cable connector.

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

Pin Number	Name		
2		connected to pin #7 (for DTR/DSR loop)	
3	TXD	Transmit Data	output
4	CTS	Clear To Send	input
5	RXD	Receive Data	input
6	RTS	Ready To Send	output
7		connected to pin #2 (for DTR/DSR loop)	
9	Signal ground	Isolated from vehicle ground	

6.3.2 Interface Board P2

P2 on the interface board is the RS-422 serial port connection. The connector is a 10 position header with pins on 0.100 inch centers. The header is compatible with an AMP #111634-1 a 10 position IDC connector.

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

Pin Number	Name	Function / Description

1	Signal ground	Isolated from vehicle ground	
2	RTS -	Ready To Send (inverted)	output
3	CTS +	Clear To Send	input
4	RTS +	Ready To Send	output
5	CTS -	Clear To Send (inverted)	input
6	TXD +	Transmit data	output
7	RXD +	Receive data	input
8	TXD -	Transmit data (inverted)	output
9	RXD -	Receive data (inverted)	input

6.3.3 Interface Board P3

P3 on the interface board is the connection to the vehicle's OBD-II port. The connector is a 16 position header with pins on 0.100 inch centers. The header is compatible with an AMP #111811-3 a 16 position IDC connector.

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

Pin Number	Name
3	J1850 Bus +
4	J1850 Bus -
5	Ground
7	Ground
9	Ground
10	+12 VDC supply (vehicle power)

6.3.4 Interface Board JP1 and JP2

Jumpers JP1 and JP2 on the interface board are used to select the serial communications port baud rate. Set the jumpers for the desired baud rate in accordance with the instructions in Section 2.1.

6.3.5 Interface Board JP3 and JP4

Jumpers JP3 and JP4 are used to select which serial communications port is active, RS-232 or RS-422. The jumpers on JP3 and JP4 should be set the same. They should be installed across pins 1 and 2 for RS-422 operation or across pins 2 and 3 for RS-232 operation. Refer to Section 2.1.

6.3.6 Interface Board JP5

JP5 enables or disables the high side supply voltage to the entire AVT-715 interface board. It is a 2 position header with pins on 0.100 inch centers and is compatible with AMP #640441-2 2 position IDC connector. When the interface board is installed in the supplied enclosure, JP5 is connected to the front panel toggle switch. When the AVT-715 is delivered as an OEM module JP5 has a jumper installed.

6.3.7 Fuse F1

Fuse F1 is a 500 milliamp fast blow fuse designed to protect the AVT-715 in the event of reverse voltage application or a voltage surge significant enough to trip the input transient voltage suppressor.

The fuse is a Schurter # MSF 125 034.4216.

6.4 AVT-715 Enclosure Information

The following sections contain information about the AVT-715 interface board mounted in the provided enclosure. The AVT-715 board is available mounted in a rugged polycarbonate enclosure. Also included in the enclosure are two internal cable assemblies and a locking toggle switch for unit power.

6.4.1 Connector P1

Connector P1 on the AVT-715 enclosure is a DA-15P connector and will mate to any industry standard DA-15S connector. It is connected to the subject vehicle OBD-II connector through the supplied cable assembly. Only the signal lines that the AVT-715 interface board uses are indicated here. In accordance with the standard there may be other signals on the OBD-II connector in the vehicle. The AVT-715 does not connect to or utilize signals or lines other than those indicated here. All other pins on this connector are 'not connected.'

Pin Number	Name
2	J1850 Bus +
3	Ground
4	Ground (chassis)
5	Ground (signal)
10	J1850 Bus -
13	Unswitched vehicle battery positive.

6.4.2 Connector P2

Connector P2 on the AVT-715 enclosure is a DE-9S connector and will mate to any industry standard DE-9P connector. Depending on the configuration of the AVT-715 interface board, this connector is either an RS-232 or an RS-422 communications port. Through this connection the control computer communicates with the AVT-715 interface and, hence, the subject vehicle. Hardware handshaking (RTS/CTS) is required for proper operation of the AVT-715 interface.

All signals and directions indicated are relative to the interface.

RS-232 Configuration

Pin Number	Name	Function / Description	
1	CD	not used	
2	TXD	Transmit Data	output
3	RXD	Receive Data	input
4	DTR	connected to DSR, pin #6	
5	Signal ground	Isolated from vehicle ground	
6	DSR	connected to DTR, pin #4	
7	CTS	Clear To Send	input
8	RTS	Ready To Send	output
9	RI	not used	

RS-422 Configuration

Pin Number	Name	Function / Description	
1	Signal ground	Isolated from vehicle ground	
2	CTS +	Clear To Send	input
3	CTS -	Clear To Send (inverted)	input
4	RXD +	Receive data	input
5	RXD -	Receive data (inverted)	input
6	RTS -	Ready To Send (inverted)	output
7	RTS +	Ready To Send	output
8	TXD +	Transmit data	output
9	TXD -	Transmit data (inverted)	output

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-1850-1 A J1850 VPW Development System.
- AVT-716 Triple Interface (J1850 VPW, PWM, and ISO 9141-2).
- AVT-921 Dual J1850 Interface (8-bit ISA bus board).

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 base.

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:

B2: Changed description of red LED operation. It toggles state every 52 milliseconds. (Incorrectly stated previously at 250 milliseconds.)

B3: Corrected the pinout for connector P2 on the HIU (enclosure) when configured for RS-422 mode of operation.

C1: Major revision. Removed all references to AVT Controller and Capture software. A separate User's Manual has been created for those software applications. Included the contents of supplement #2.

C2: Corrected the word "course" to "coarse."