

13 February, 2000

AVT-716 Firmware version 5.0

Introduction

Significant changes were implemented in the AVT-716 firmware version 5.0. Most changes concern operations in KWP mode (Key Word Protocol 2000; ISO 14230).

This document describes those changes in detail. The user should consult the "Master Commands and Responses" document; revision 22, for details on the new commands.

To be consistent with the KWP 2000 specifications; the AVT-716 is considered an off-board tester or 'client'.

Change #1

Baud rate of serial communications link with host computer.

The settings of the two baud rate jumpers on the AVT-716 #1 board have been changed. The new settings are shown in the following table.

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

Change #2

Support for high speed communications on the K-line during non-initialization operations.

The 5x 26 command has been implemented to permit switching the baud rate of the K-line to and from 115.2 kbaud. High speed K-line operations are only permitted during 'regular' communications; i.e. not during any of the initialization sequences. If the baud rate is set to 115.2 kbaud prior to commanding an initialization sequence, the K-line baud rate will be returned to that value required to conduct initialization.

Switching to high speed K-line operations is nearly instantaneous and without regard to transmitting or receiving a message on the K-line. The user is cautioned that switching speeds during K-line communications may disrupt any transmission or reception that is in-progress.

Note: This is the baud rate of the K-line only and has nothing to do with the speed of communications with the host computer.

Command summary

- 51 26: Status query.
- 52 26 00: Disable high speed K-line communications (default).
This returns the K-line baud rate to that determined by the counter/timer load.
- 52 26 01: Enable high speed K-line communications at 115.2 kbaud.

Response summary

- 62 26 00: High speed K-line communications disabled.
- 62 26 01: High speed K-line communications enabled.

Change #3

Modification of time parameter P4_MIN.

The time interval between tester sourced bytes is determined by time parameter P4_MIN. In high speed mode, a long P4_MIN would inhibit the full available bandwidth.

The 5x 27 command has been implemented to permit the user to change time P4_MIN.

Command summary

- 51 27: Status query.
- 53 27 xx yy: Set P4_MIN to \$xx yy;
where \$xx yy is in 0.5 millisecond increments.

Response summary

- 63 27 xx yy: P4_MIN time interval in 0.5 millisecond increments.

Note that the available range is \$00 00 to \$07 CE.

Note that the user is cautioned to disable the real time clock feature (command: 52 0E 00) of the AVT-716 during KWP operations in order to obtain maximum accuracy and operational dependability.

Change #4

Disable transmitted checksum.

The 5x 25 command has been implemented to permit the user to enable or disable the automatic calculation and inclusion of the transmitted checksum.

When enabled (default) the 716 unit automatically computes and appends the checksum to every transmitted message. When disabled, the 716 unit will not compute or append the checksum.

Note that for every received message a checksum is computed and compared to the last byte received. The checksum error bit in the received status byte is cleared or set, depending on the outcome of that test. The 5x 25 command does not affect that function or operation. (If the user wants to see the received checksum, the 5x 01 command is still available.)

Command summary

51 25:	Status query.
52 25 00:	Enable automatic checksum computation and inclusion. (default).
52 25 01:	Disable the automatic checksum function.

Response summary

62 25 00:	Automatic checksum function is enabled.
62 25 01:	Automatic checksum function is disabled.

Change #5

Loss of arbitration.

The AVT-716 transmit routines have been updated to detect the occurrence of a ‘loss of arbitration’ during transmission of a message.

Due to the design of the physical layer, it is possible for two nodes to simultaneously transmit a message. If one node transmits a bit ‘low’ while the other node is attempting to transmit a ‘high’ bit, the ‘low’ bit will win arbitration and will be permitted to keep transmitting.

The AVT-716 uses an industry standard UART for transmission of each message byte. Following transmission of a byte, the received byte (an echo) is compared to the byte transmitted. If the two bytes don’t match, the following occurs:

- A ‘loss of arbitration’ is flagged by setting bit #4 in the received status byte of the message currently being received.
- The transmission is terminated immediately.
- The ‘loss of arbitration’ counter is incremented.
- If the ‘loss of arbitration’ counter is less than three, the message is queued again for transmission at the next available opportunity.
- If the ‘loss of arbitration’ counter is equal to 3, the message is discarded and the status message \$01 \$10 is transmitted to the host.

Note that if a ‘loss of arbitration’ is detected, the received message status byte bit #4 is set and the message being received is allowed to complete. The fact that bit #4 is set only indicates to the host that the transmit message lost arbitration to ‘this’ message and will be queued again for transmission.

Only reception of the status byte \$01 \$10 indicates that the transmit message failed completely and has been discarded.

Note that arbitration can only be implemented on a byte-by-byte basis, not a bit-by-bit basis.

Change #6

Received message processing by format/length byte.

In prior versions of the AVT-716 firmware, the detection of the end of a message from the network was determined solely by the expiration of time interval P2_MIN. In an environment where messages were not separated by a sufficiently long P2_MIN time interval, the AVT-716 receive function could disrupt message reception (message fragments and/or message concatenation).

[That method was a carry over from ISO 9141 communications methodology.]

The KWP 2000 specification calls for the message length to be embedded in the message itself.

The 5x 28 command has been implemented that enables or disables a function where received messages are processed according to the format/length byte(s) in the message being received.

The user should consult the specification (ISO 14230-2) for detailed information on the construction of a proper message with length information embedded.

Command summary

51 28:	Status query.
52 28 00:	Disable received message format byte processing (default).
52 28 01:	Enable received message format byte processing.

Response summary

62 28 00:	Received message format byte processing is disabled.
62 28 01:	Received message format byte processing is enabled.

Note: The format byte processing function supports messages with A1 A0 bits set to 01 (CARB exceptions).

Note: The time processing function is enabled at all times. The user should set P2_MIN (command: 53 02 xx yy) to a sufficiently long time interval to be sure to 'catch' messages that are in error or don't conform to the expected format definitions.

Change #7

Incoming command processing.

The method by which commands received from the host are queued, parsed, and processed has been changed. The new method should improve AVT-716 unit response and overall performance.

The changes completely transparent to the user and do not require any action on the part of the user.

Change #8

New error codes.

The \$21 \$49 error code has been added to indicate the reception of a message of length 00.

The \$21 \$4A error code has been added to indicate the reception of a message of length 01. This is illegal since all messages have a checksum; thus a minimum length message would be 02.