

Programmable Voltage Reference Command Set

This document describes the use of a serial interface to control the PVR.

Serial interface set up: The PVR has an FTDI connector that can be used for communication. The protocol is standard UART at 115200-8-N-1. That is, 115200 baud, 8 data bits, no parity and one stop bit. All messages to and from the PVR are readable ASCII text so as to allow easy debug. The messages can be generated or received / monitored by a terminal emulator program, however no line editing (backspace, etc.) is supported.

When the PVR powers up it sends out the message "Voltage Reference" followed by a carriage return character.

Anytime the PVR reference voltage is manually changed using the rotary encoder or button the PVR sends out a message containing the new reference voltage value followed by a carriage return character.

Message format: All commands sent to the PVR should be in the form "C>NNNN<nl>" (do not include the quotes) where:

C is a single character indicating the command to be done. Valid values are '#', '!', 'W', 'R', 'U', 'N'.

NNNN is four characters defining a decimal number. Each N is a digit character ('0' thru '9'). Together these form a number. The valid range for the number is defined in the context of the command.

<nl> is a single termination character. Valid values for <nl> are the carriage return character (0x0D) or the linefeed character (0x0A). If your terminal emulator sends both the extra will be ignored.

Command validation: If the message is less than six characters the message will be discarded. Beyond six characters will be ignored. If the message does not begin with one of the defined command characters the message will be discarded. If the message does not end with one of the <nl> characters the message will be discarded.

Note: Validation is NOT done on the N characters. Decimal conversion is done assuming each N is a valid digit character. If non-digit characters are given the decimal value will be generated based on their ASCII value.

Commands:

#: Sets the PVR reference value to NNNN millivolts, plus the internally stored calibration offset. Only the bottom 12 bits of the reference value are used to program the DAC. A value of zero can be set with this command though the PVR will not actually output zero volts. Values greater than 4095 will be treated as NNNN modulo 4096.

!: Sets the PVR reference value to NNNN millivolts, without adding calibration offset. Only the bottom 12 bits of the reference value are used to program the DAC. A value of zero can be set with this command though the PVR will not actually output zero volts. Values greater than 4095 will be treated as NNNN modulo 4096.

W: Writes a value to EEPROM memory. The previous PVR reference value is used as the EEPROM address. The value to be written is created from NNNN. Only the lower 8 bits are used, therefore NNNN should be restricted to the range "0000" to "0255".

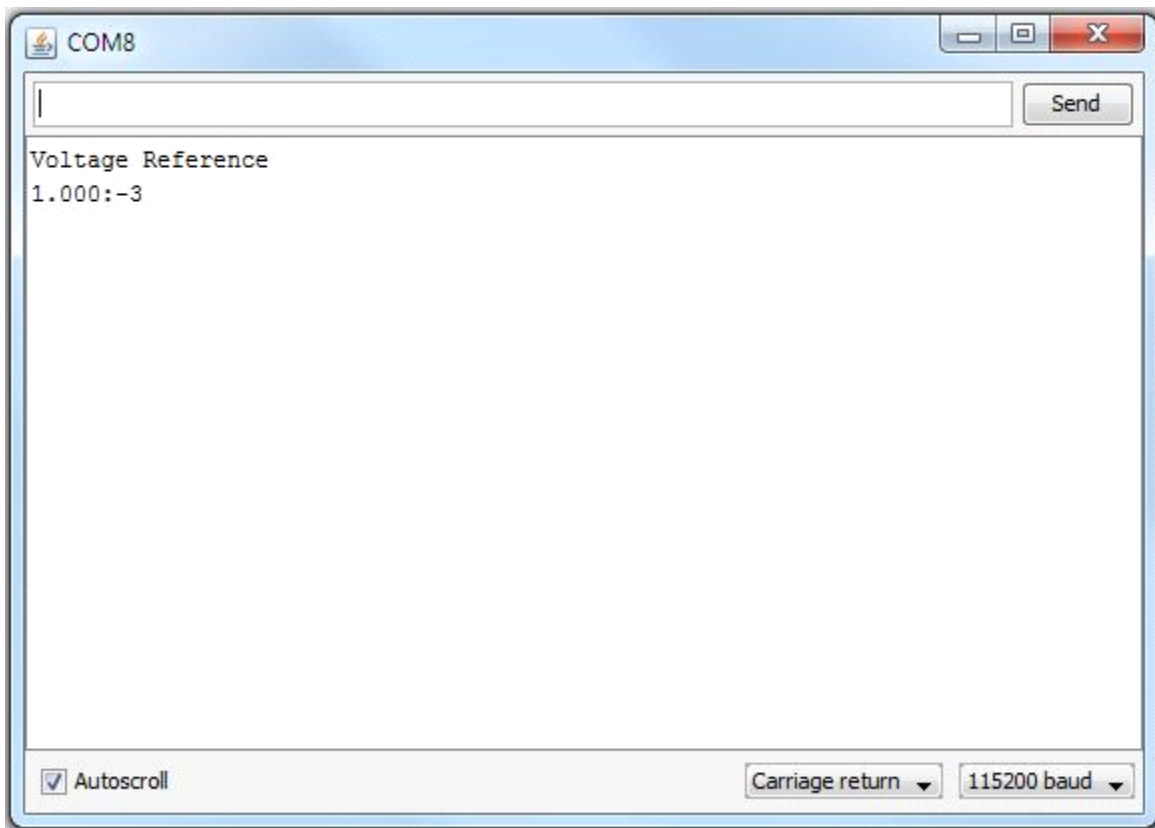
R: Returns the compressed table of calibration offsets as a series of strings. Each string represents one EEPROM memory location starting from zero.

U: Sets an immediate offset value to be used with the current reference value instead of using the offset from the calibration memory.

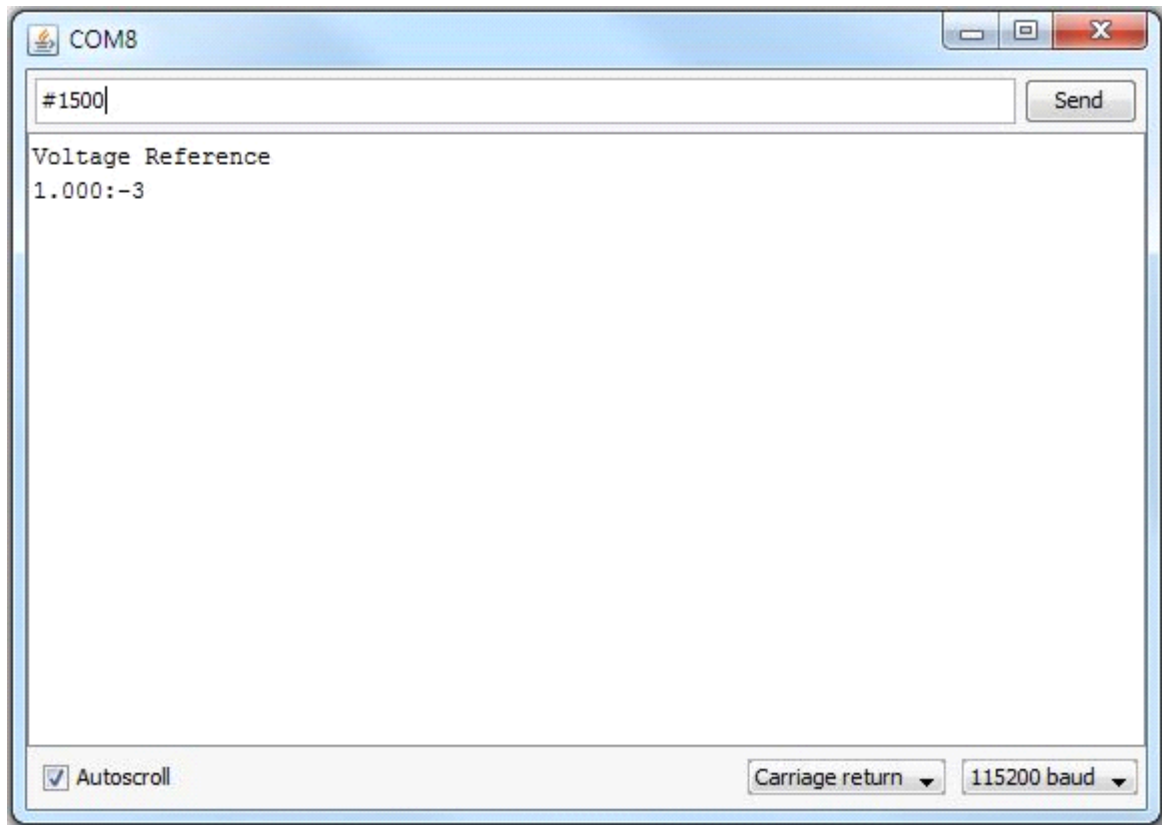
N: Returns behavior to look up offsets from the calibration memory.

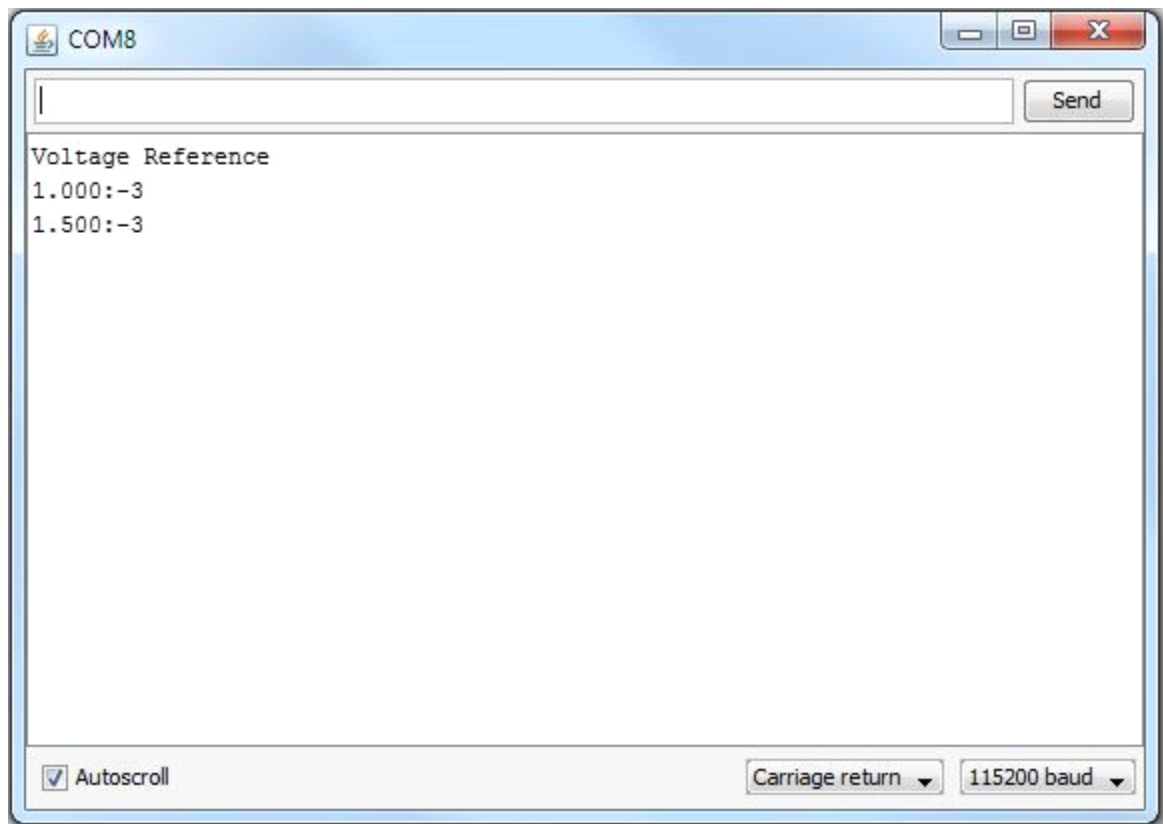
Examples: All examples were created from the Arduino IDE Serial Monitor window.

After restarting PVR

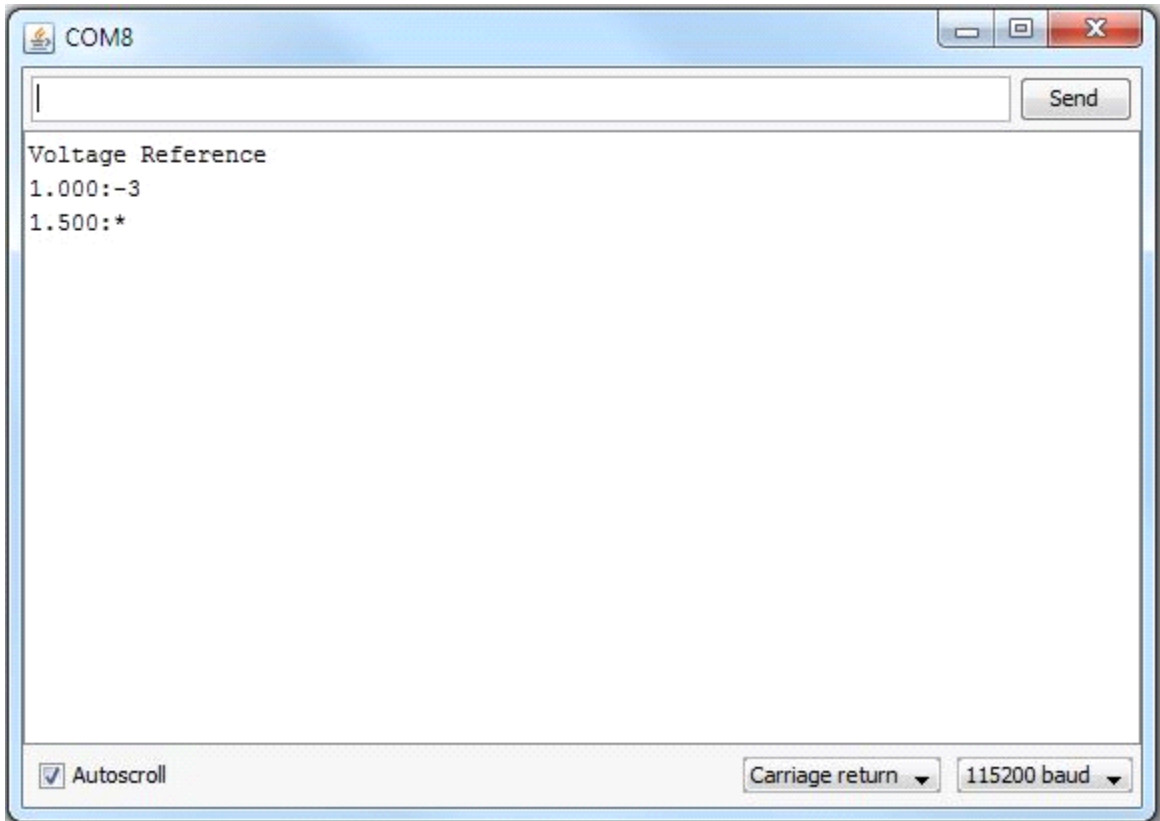
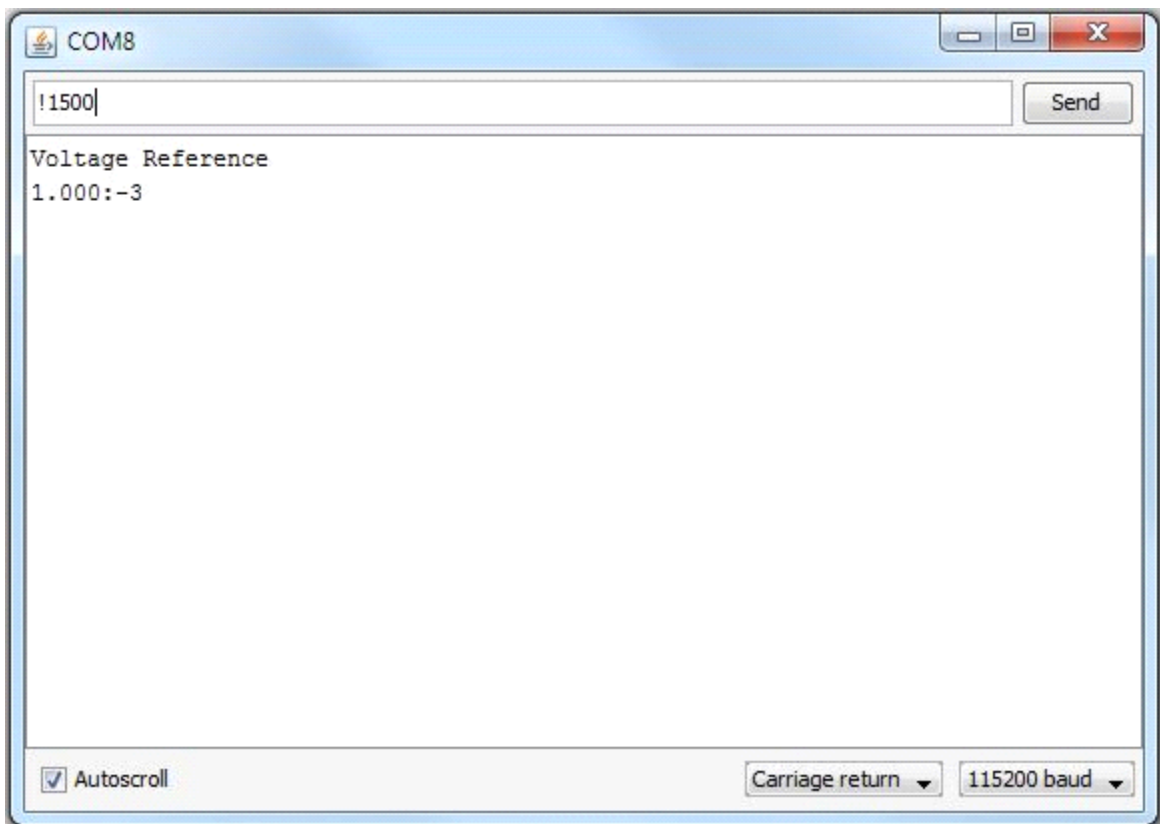


Setting the PVR to 1.500 volts, calibrated:



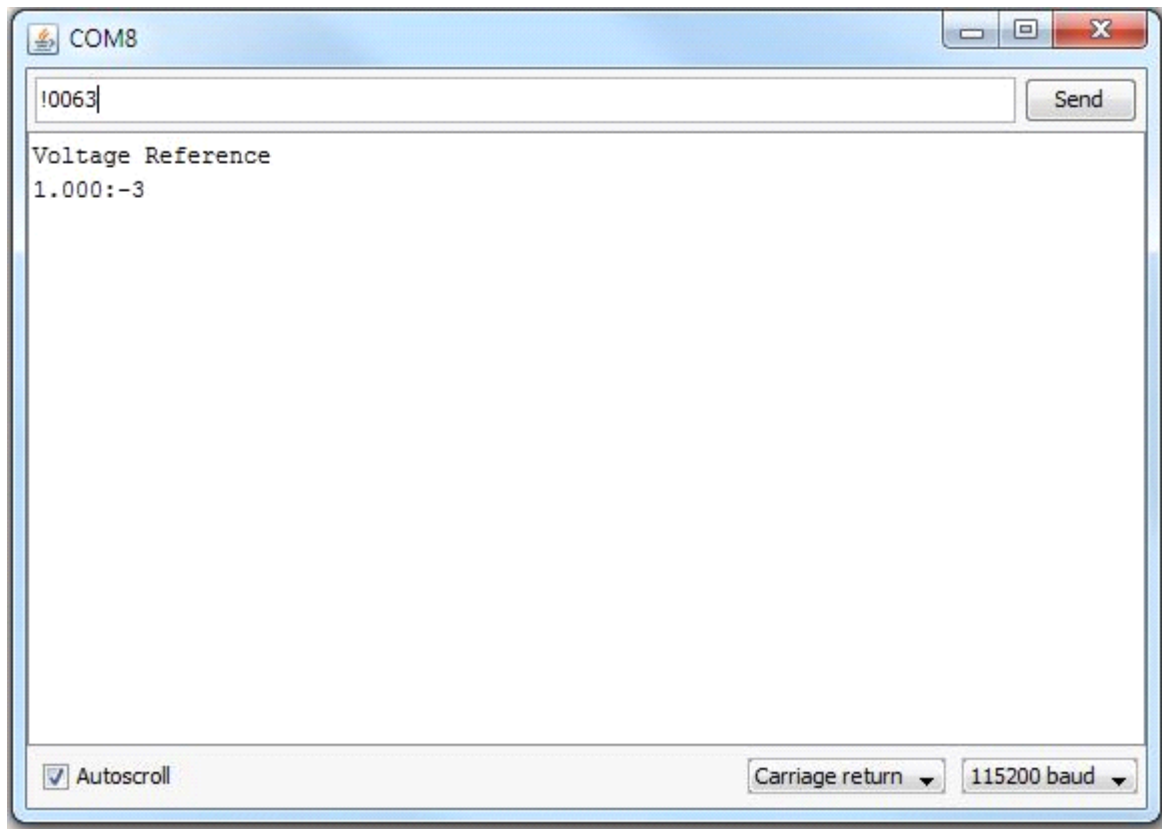


Setting the PVR to 1.500 volts, uncalibrated (PVR output may be incorrect):

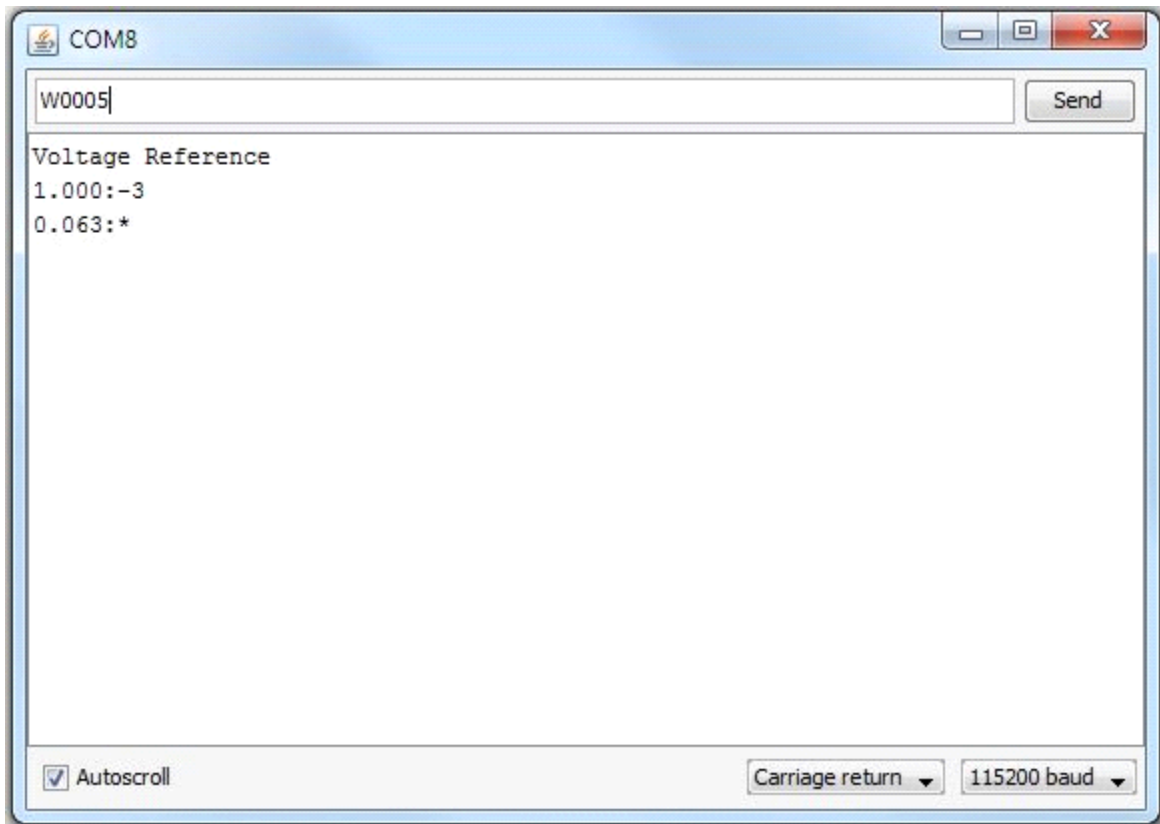


Writing the value 5 to EEPROM location 63:

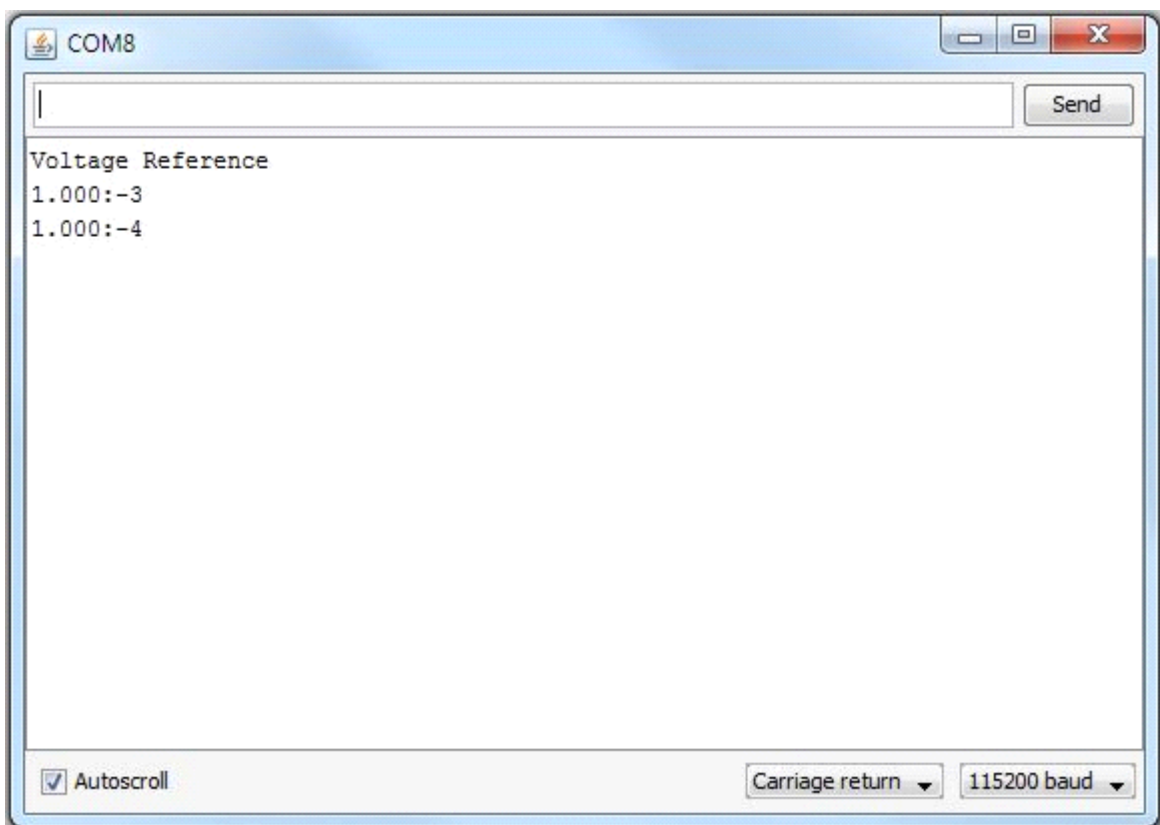
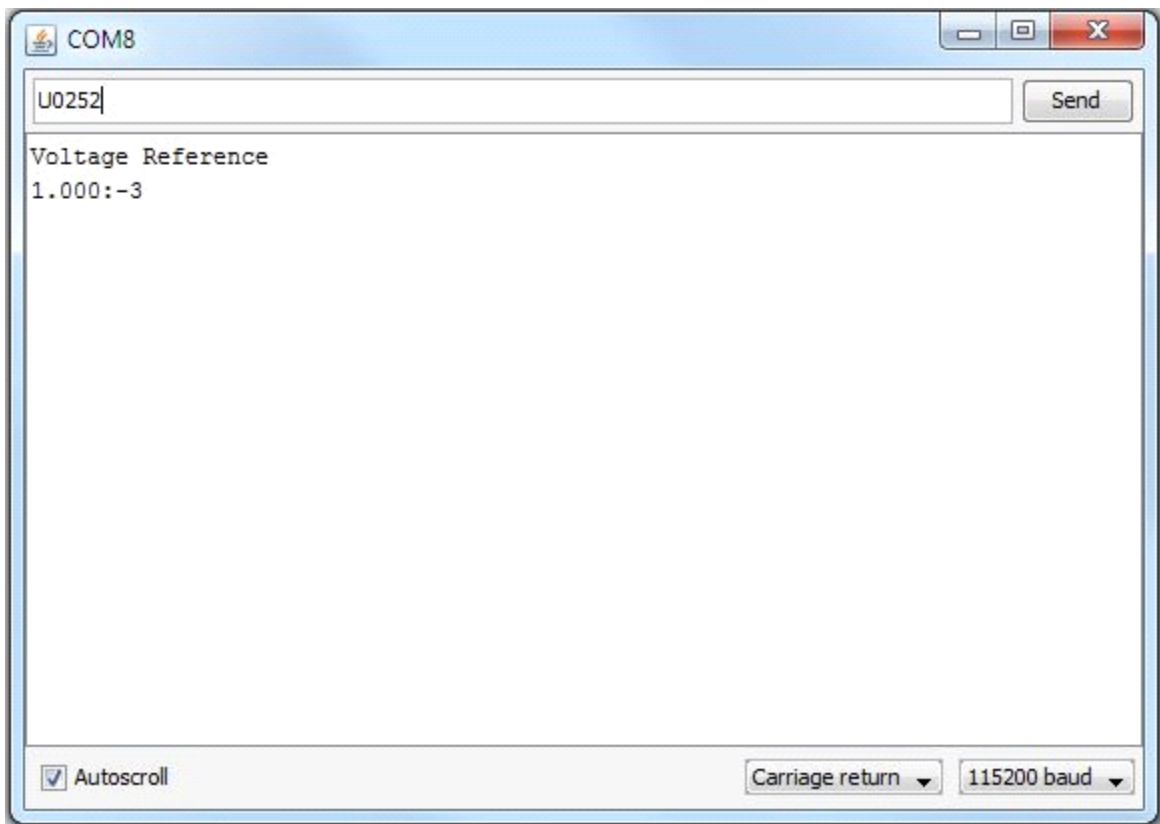
This requires two steps. The first step sets up the address by setting the reference value to 63.



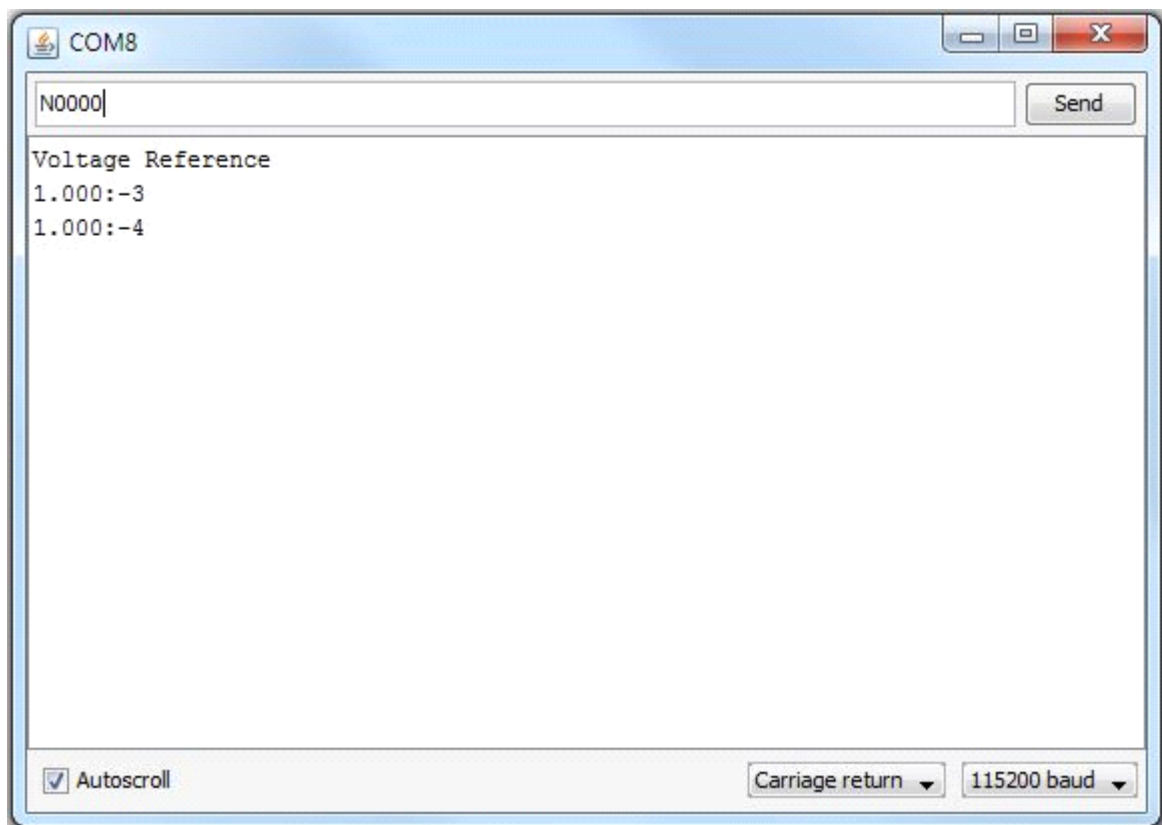
The second step writes data value.

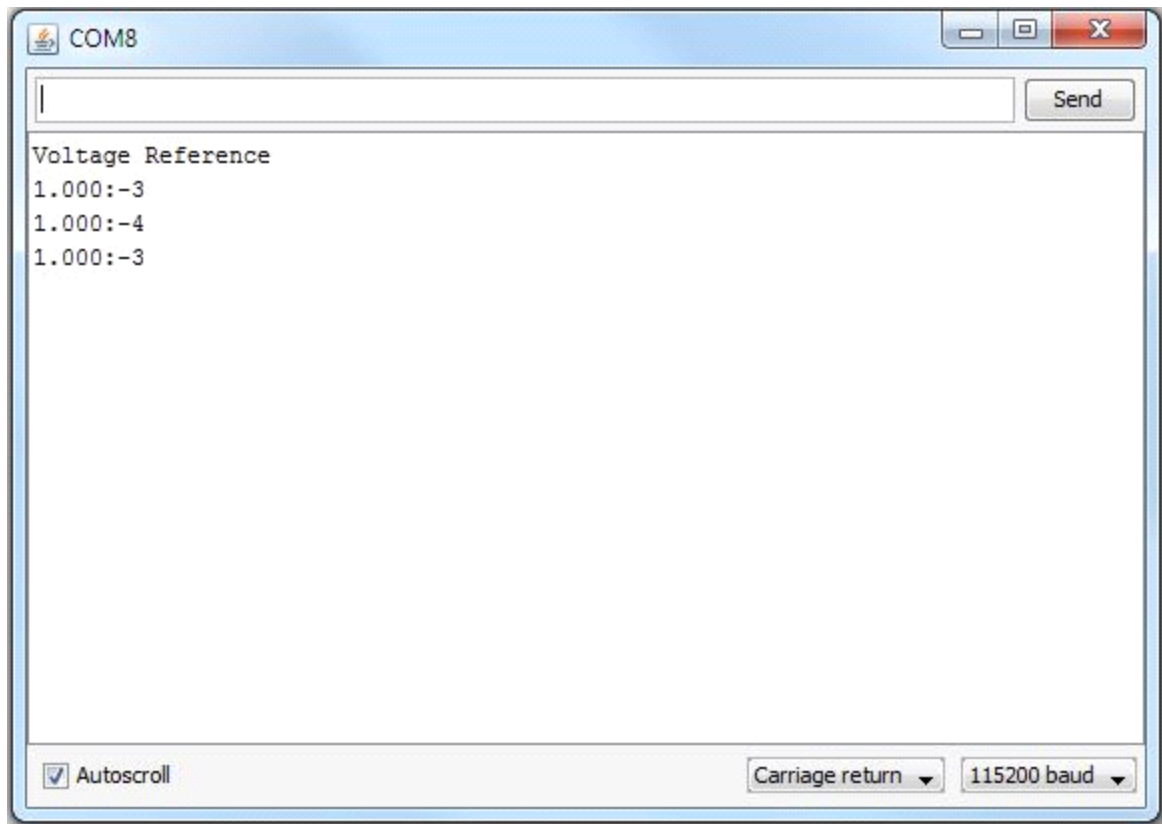


Set the PVR to always use an offset value of -4:



Set the PVR to use the offsets from the calibration table:





Print out the calibration table:

