AttoBASIC Version History What's New (and fixed!) by version

VERSION 2.31

New (or improved) features:

1. Added support to calculate, with exact precision, the answer to life, the universe and everything. Don't Panic, the answer correlates to that provided with infinite majesty and calm by *Deep Thought*.
2. Added support for the DALLAS/MAXIM 1-Wire® protocol.
3. Added support for the nRF24L01(+) series of 2.4GHz ISM band RF transceivers.
4. Added CRC-8 command to support the DALLAS/MAXIM 1-Wire® ROM commands.
5. Modified the RESTore command to return the size of the data held in the DATA statement buffer. This makes it easier to determine a payload size when using the nRF24L01 routines since each payload received could be between 1 and 32 bytes in length.

Bug Fixes:

1. VPG command now returns the proper page of the desired constant. MCU builds of differing RAM sizes caused page cross-over issues, which resulted in erroneous data being returned.

VERSION 2.30

(Unreleased)

New (or improved) features:

1. Added support for the AT90USB1286 MCU, which is found on the PJRC *Teensy++ 2.0* product. This MCU has 128KB of FLASH, 4KB of EEPROM and 8KB of RAM.
2. Added support for the Atmega2560 MCU, which is found on the ARDUNIO *Mega 2560* product. This MCU has 256KB of FLASH, 4KB of EEPROM and 8KB of RAM.
3. Added support for the TEENSY LOADER application so that the TEENSY++2.0 specific HEX file, named "*ATTOBASICV230_usb1286-16MHZ-teensypp20.hex*", can be programmed using the TEENSY LOADER under Linux or WINDOWS®.
4. Added the number of "USED" bytes to the "FREE" command.
5. Modified the LSL and LSR commands to accept an optional 2nd command line parameter to specify the number of bits to shift by.
6. Added the SWAP command to return a value with the high and low nibbles swapped.
7. Added the NBH and NBL commands, which return the high and low nibbles of a parameter.
8. Updated the USB Serial I/O routines to PJRC's Version 1.7 and modified for use as a stand-alone library with MCU's having more than 64KB of FLASH memory, which requires implementing the RAMPZ register and the ELPM instruction.
9. Added the FILL command as part of the DEBUG routines. FILL is for filling RAM with a preset value.
10. Added the DATA, READ and RESTore commands, which stores up to 8 bytes of data for use in storing constants for access in a program.
11. Added new binary conversion function; using the " ` " (tick), which acts like the " $ " but takes an 8-char ASCII binary number and converts to a single 8-bit number.
12. Added new command IDUMP to dump the contents of the MCU's I/O.
13. New feature of GOSUB-RETURN; can now nest up to 4 levels deep.
14. New feature of FOR-NEXT loop; can now nest up to 4 levels deep.

15. When using USB builds (Mega32U4 and 90usb1286), where there is no host to initialize the USB interface within a preset timeout period (currently 32 seconds) then the USB interface is left un-initialized and AttoBASIC continues in a stand-alone mode just like it’s USART counterparts. This allows the self-start feature to be used without the need for a USB host’s intervention. Any serial I/O executed within a self-started program will be ignored.
16. When using USB Serial I/O builds (Mega32U4 and 90usb1286), where there is a host to initialize the USB interface, AttoBASIC will behave one of the following two ways. This is different from prior versions that required the user to type a character before AttoBASIC would respond.
17. Await assertion of the DTR signal before emitting the sign-on message
18. If after the preset timeout period (currently 32 seconds) the DTR signal is not asserted, it will shut down the USB interface and carry on as in line 15 above.
19. The self-start feature was disabled in all pre-assembled HEX file builds. It is now enabled in all pre-assembled HEX file builds. Be sure the SelfStrt pin is not grounded unless one wishes to execute the program saved in file number “0” (in EEP), if any, upon reset.
20. Changed INIT routine to always set the system clock prescaler to the value defined in FCLK_PS (see Defs_Constants.inc), which defaults to 1x in all pre-assembled HEX file builds. This is partly due to the fact that TEENSY++ 2.0 defaults to 8x divider upon power-up but also gives the user a means to insure proper operation at lower power supply voltages by setting the AVR core’s clock speed.
21. Enabled LUFA's BootloaderDFU.c to check for PORTF4 (JTAG TCK) pin shorted to GND, which will invoke the DFU bootloader upon power-up or hardware/software reset. This is for platforms running with a Mega32U4 or USB1286 where a JTAG port may available on the target PC board. PORTF4 is the JTAG TCK signal, which is pin 1 of the ATMEL 10-pin JTAG header. Placing a shorting jumper between pin 1 and pin 2 will short TCK to GND to invoke the bootloader.
22. Changed the order of the DUMp and EDUmp routines to print memory from bottom to top and added byte index header to more easily read the printout. Also added key check for ^S (control-S) to suspend output until another key is pressed.
23. Added the ability for the PWM command to use two channels on OC1A and OC1B.
24. PWM output is now only on OC1A/B pins as OC0A/B is no longer supported. For Mega88/168/328 devices, only OC1A is available unless the SPI and DataFile routines are disabled.
25. Added support for easier access to certain system internal variables by symbolic name. They are intended to be used with as the page number when accessing the internal variables with the PEEK and POKE commands. Ex: “PRINT PEEK VPG@RTC0” returns the low byte of the Real-time counter’s 32-bit internal counter. See the application note for examples on using symbolic names in your program.
    - The "VPG" command returns the page in RAM where the internal variables are stored.
    - The "@" command is a prefix that designates that the following three (3) characters are a symbolic name.
    - Added the symbolic names “DHI”, “DHD”, “DTI” and “DTD” to support the DHT sensor command set.
    - Added the symbolic name "RTCn", where “n” is a number between 0 to 3, designating which of the four Real-time Counter register bytes one wishes to access.
    - Added the symbolic name "DFAn", where “n” is a number from 0 to 1, designating which of the two Data File sequential address counter bytes one wishes to access.
26. Added the “EEW” command to write a byte to the EEPROM. The format is the same as the “POKE” command. A special “bulk erase” feature of the EEW command is invoked when the address

provided is E2END + 1. I.e. writing to 0x0400 on a Mega328 will bulk erase the EEPROM because the last address is 0x03FF.

27. Added the “EER” command to read a byte from the EEPROM. The format is the same as the “PEEK” command.
28. Modified SAVe routine to also check PCHI/PCLO when saving programs to EEPROM. This speeds execution when saving to devices with large amounts of EEPROM.
29. Modified LOAd routine to check for null character in the data stream when loading programs from EEPROM, which signifies the end of a program. This speeds execution when loading from devices with large amounts of EEPROM.
30. Implemented EEPROM File Sysytem (EFS).
    - Changed the “SAVE” command to accept a file number to write to.
    - Changed the “LOAD” command to accept a file number to read from.
    - Added new command “ERA” to erase data from a file number in the EFS.
    - Added new command “REM” for “REMARK”. This command is intended to be used as the 1st line of a program (when the EFS has been enabled) to more easily identify the program SAVed in each file slot. CAVEAT: all characters contained in the REM program line eat memory, use sparingly.
    - Added new command “CAT” to list the file size, 1st program line of each SAVed program and the total bytes used and available.
    - Added new command “INIT” to initialize the EFS volume before then 1st use. It should be used when a target MCU has EEPROM contents left over from another project or has fully erased EEPROM contents.
    - Added file system support to the “RUN” command. If “RUN” is typed without a parameter then RUN will run the program currently in memory. If “RUN“ is typed with a parameter, which is a valid file number, then the program is first loaded into program memory (the current program overwritten) then run.

Bug Fixes:

1. This bug may not have affected MCU's with smaller RAM sizes. The RAMFILL macro's pointers in "NEWPROGRAM" routine are destroyed due to ISR's overwriting them.
2. This may not have been an issue on some builds. ADR now enables the ADC and initiates the 1st conversion. ADC now handles channels > 31 on MCU's that support it.
3. Results of the ADC command can now be assigned to a variable. Prior versions were unable to assign the result of an ADC measurement to a variable.
4. When using USB Serial I/O builds (Mega32U4 and 90usb1286), the RST command did not shutdown the USB interface before initiating a system reset. USB shutdown is now executed followed by a 1 second delay to insure the host properly disconnects.
5. On USART serial I/O builds, corrected DHT sensor checksum and timeout errors due to spurious interrupts when using the CP2102 UART to USB bridge, which spews nulls into the receiver causing a flood of interrupts during the reading of the sensor.
6. The TWR command improperly interpreted a slave's DACK response to a master’s designation of the “last byte to receive”, as a bus error.
7. The ICP function did not use TIMR3 on the M32U4 as was intended. Corrected and reduced the coding size as well.

8. If the SPM command was not issued prior to using the SPW, SPR and SPS commands then an infinite software loop occurs waiting for the SPI interface to send data. This is more to prevent users from inadvertently issuing those commands before issuing the SPM command.
9. Corrected SPI port is initialization so that the SPI_SS pin is set high.
10. Verified proper support and operation of the DS routines.
11. The printout of the DUMp and EDUmp routines has been corrected. For EDUmp", the last byte of the EEPROM is now printed (address 0x000”. For DUMp, the memory address of “0x00FF” is no longer printed.
12. DHT READY flag updating was dependent on the setting of the RTI command and should have been independent of RTI's setting.
13. Control-S (suspend output) and Control-C (abort) are now recognized during a DUMp and EDUmp.
14. DHT routine will abort program execution upon a DHT checksum or non-response error. Changed "DHTRead" routine to zero both temperature and humidity readings as a sign that there was a checksum or non-response error detected from the DHT sensor. A message is also printed to the console to inform the user.
15. Correct SLP routine so that all valid interrupt source flags are cleared before entering sleep mode. In some cases, after using the watchdog as an interrupt source, the flag was left set causing the SLP command to prematurely end.
16. Corrected sign-on header string generation, whereas certain builds of Mega88 and 168 would corrupt a line of text due to the assembler’s “word padding”.
17. Control-C now reliably breaks program execution when using USB Serial I/O. Also, control-C only need be pressed once to break execution.
18. The “backspace” key was not properly handled as a control-H.
19. Modified the STK500V2 bootloader source code to recognize the WDT as a valid reset source and jump to the application code if a WDT reset has occurred. This issue affects the RST command on the ARDUINO Mega 2560 boards and any Atmega2560’s using the STK500V2 bootloader supplied with the ARDUINO IDE.

VERSION 2.22

New (or improved) features:

1. Added support for the DHTxx series of low-cost temperature and humidity sensors. Reference the Command List for version 2.22 for details in using the new commands and any idiosyncrasies therein. Note that if the PWM feature is enabled and the PWM output is currently emitting a signal then using the DHT or DHH commands will cause a 7mS interruption in the PWM output.
2. Added support for the use of the AVR’s System Clock Prescaler. AttoBASIC calculates the proper run frequency based on the original FCLK setting divided by the FCLK_PS setting. Note that the pre-assembled versions use a 1x prescaler value but one can assemble a specific version using any other prescaler value. This feature was added mostly as an aid in testing at different run frequencies without having to change crystals to do so.
3. Added assembly time support to select the proper serial baud rate with the lowest bit error when AttoBASIC is assembled to run at 1MHz and 2MHz. The normal baud rate of AttoBASIC is 19.2K baud but changes to 9600 baud at 2MHz and 4800 baud at 1MHz.
4. Improved TWI support by adding conditional statements to correctly calculate the maximum available TWI bit clock value based on the system clock frequency. Also added conditional assembly to the

"TWIcommand" routine to support 400K and 100K selection if the system clock allows it, otherwise, only the maximum rate detected is allowed for both the “TWI 0” and “TWI 1” commands. In other words, at bit rates of 100K or less, the “TWI 0” and “TWI 1” select the same bit rate.

Bug Fixes:

1. HELP command formatting has been corrected when using AttoBASIC on a Mega32U4 with USB serial enabled. All other build flavors are unaffected.

Version 2.21

New (or improved) features:

1. None.

Bug Fixes:

1. In bootloader versions, fixed the DDS bug where the DDS port bit toggled with a random duty cycle instead of at the frequency defined in variables X,Y and Z. Version 2.20

Version 2.20

New (or improved) features:

1. Line numbers are now printed along with the error message.
2. The new "comment" character "#", allows the use of comments in source code. Upon receipt of the "#" sign from the terminal, the line interpreter will ignore all characters after the "#" until a CR is found. This aids in commenting programs when one is editing and uploading them via a serial terminal. Its use is allowed in the print string command.
3. Added 32-bit Real-time Counter with new commands; "RTI", "RTR" and "RTP".
4. Added "OSC" command to set and read the internal oscillator's calibration register.
5. Added a "HELP" command to list the commands available on that particular build.
6. Added support to the "PRINT" command to print strings embedded in quotes. Supports sending a CR/LF combination by using the "~" character embedded within the string.
7. Added "SLP N" instruction. When executed, the CPU enters the power-down mode. The routine is exited once the "waking event" has occurred.
8. Added "XB[p] [n]" command to toggle bit "n" on port "p".
9. Added "RST" command, which invokes the watchdog timer in "System Reset" mode to cause hardware reset of AttoBASIC.
10. Added "Data File" interface routines. Presently, the routines support the MICROCHIP 25AAxxx series of Serial EEPROM devices using the SPI interface. Address range is16 bits or up to 65,536 storage locations. The intent behind this added functionality is for use as a data-recorder when used with the "SLP" instruction. Eventually, these routines could be used with SD memory cards as they also use the SPI interface.
11. Added specific commands for the "AVR Data Recorder". If enabled, conditional assembly is performed to enable selecting channel #'s, setting gain and the AD536's output type (TRMS or dBV).

12. New "LDD" command loads the default capture and record program for the Data Recorder.
13. New "ACI" command to enable/disable analog comparator interrupts. This is mainly for use with the "SLP 0" command. ACI only supports inputs on AIN+ and AIN-.
14. New "ACS" command to select source to the analog comparator's "-" input. Either AIN-, ADC0 or ADC1.
15. "ACO" command now selects the AIN(-) source based on the value selected by the "ACS" command.
16. The "ACO" command now returns the actual state of the ACO bit, which was inverted in prior versions.
17. New "DIG" command to directly control the N-MOSFET on the AVR Data Recorder.
18. New modulus operator, "%".
19. The "hex value" command, "$" now detects 1 or 2 characters. Prior versions required entering two characters even if the MSD was "0".
20. The "ADC" command now averages sixteen (16) 10-bit samples before converting to an 8-bit result.
21. The "DUMP" command now dumps all of RAM contents instead of just PROGRAM memory.
22. The "SIZE" command now prints in bytes instead of pages and characters.
23. The "not equal" test operator, "<>" has been renamed to "!=". Done to conform to "C conventions".
24. Test operators "=", "!=", "<" and "<" now return a "1" for a "true" condition or a "0" for a false condition.
25. The detection logic of the "IF" command supports the newly revised test operators.
26. Certain operators have been changed to single-character commands; AND = "&", OR = "|", XOR = "^", and NEG to "!". This change reduces the amount of program memory needed for the command while also conforming to "C conventions". The original commands are still available.
27. The sign-on and "NEW" messages now display the CPU Clock frequency, which works for integer frequencies of 1MHZ+, no fractionals.
28. Re-ordered the "LOAD" command to execute the "NEW" command first, print "loading from EEPROM" then execute the "FREE" command.
29. Added "BIGRAM" conditional assembly to allow larger program storage. Caveat: Programs larger than EEPROM cannot be saved and will be truncated! This option is good for when one wishes to save and load programs via a terminal emulator.

Bug Fixes:

1. Corrected bug in the "PEEK" command, which added erroneous data to the stack.
2. Corrected bug in the "DDS" command that falsely detected whether the command line parameter was a "0" or a "1".
3. The "subtract" operator issued an "arithmetic underflow" when an operation resulted in a half-carry. This was the result of adding the "AOV" command in version 2.00.
4. Corrected error message string pointer overflow problem, which in some build flavors resulted in "garbage characters" being printed instead of an actual "error message".
5. When the "self-start" feature is activated, any console I/O would be block because global interrupts were not enabled after the serial I/O hardware was initialized and the "self-start" routine executed. The user can now BREAK a program executing via the "self-start" feature.
6. For Mega168/328 using the Optiboot loader; modified the pre-processing script to correctly fill unused bytes between addresses at the end of the HEX image. There are two bytes required to be placed as the last two bytes in the HEX file but were placed right after the code causing incorrect operation of the boot-loader.

7. In version 2.1x, the "ADC" command used a "default" value when a channel number was not supplied with the command. Assignment of the ADC value to a variable would return "0" instead of the actual value provided by the user. The ability to use a "default" channel with the ADC command has been removed and must be specified with the ADC command.
8. Corrected an ADC bug whereas the Mega328 was not actually supported in conditional assembly builds.
9. The line parsing routine improperly inserted and deleted lines. This was caused in version 2.11 while optimizing relative branches within the routines.
10. On the Mega32U4, added conditional compile directives in "usb_serial.h" to properly set the PLL for using 8MHz or 16MHz CPU clock. Version 2.1x did not set this up properly and using the USB_Serial I/O routines at 8MHz yielded an "unrecognized USB device".

Version 2.11

New (or improved) features:

1. None.

Bug Fixes:

1. The storage of programs in program memory does not exceed 1st page (256 bytes).

Version 2.10

New (or improved) features:

1. Added support for ATmega328, which many ARDUINO's use.
2. Added the BLDR command, which always assumes that the target MCU contains a valid bootloader. When executed, an error message is displayed if a boot-loader is not found. Otherwise, control is passed to the bootloader at the address specified by the BOOTSZ[1:0] fuse bits.
3. PEEK and POKE now support access to the entire 64KB data space using an optional parameter that specifies the page number.
4. If the "Self-Start" feature is enables then port pin’s pull-up, checks the level of the pin then disables the pull-up. This allows for an ARDUINO users to use AttoBASIC without needing to be concerned about the "SS" feature loading a program from E2P when not desired.
5. Added support for the on-chip USB controller for USB Serial I/O and USB LUFA DFU Boot-loader as now supported for ATmega32U4’s running at 8MHz and 16MHz.
6. Changed sign-on message to display the software version code, then MCU type.
7. Code size reduced by implementing internal formatting tokens as part of the message strings stored in the PROGRAM memory space.

Bug Fixes:

1. Corrected bug in "IBx" command where if the bit tested was high, a "0" was returned and versa-vise.

AttoBASIC Version History
What's New (and fixed!) by version

VERSION 2.00

New (or improved) features:

1. Original port of AttoBASIC from Dick Cappels' AttoBASIC for the ATmega163 to then ATmega88/168.
2. Added support for ATmega32U4 sith USART serial I/O only. USB functionality not yet implemented.
3. Added commands to support TWI/I2C interface.
4. Added commands to support the SPI interface.
5. Added RND command to generate 8-bit random #.
6. Added '*' and '/' commands for 8-bit multiply and divide
7. Added AOV command to enable or ignore arithmetic underflow and overflow conditions.
8. Added new commands to support input capture on ICP1 using a selectable input capture gate time.
9. Added command separator for program lines, which is the semicolon (";"). This provides for multiple commands to be on the same program line (up to 52 characters). This is helpful for using the TWI commands for example.
10. Added assembly-time feature to enable/disable support of various functional modules so that a user can enable/disable certain command features if not needed. See DEFS_CONSTANSTS.INC file.
11. Changed UART receive handler to use the AVR's "sleep" mode until a character is received. Note: any interrupt wakes from sleep but the 'recvchar' routine checks for data in RX buffer. If no data is ready, it goes back to sleep. This allows for other interrupt-driven
12. routines (such as DDS output) to function as well and keeps the MPU running in a low-power state when waiting for user input.
13. Added D[irect] D[igital] S[ynthesis] feature, which emits a square wave on a pre-designated port pin at a frequency up to 25KHz.
14. Added 'ID[p]" command to return the value of the DDR[p] register.
15. Input line parsing correctly handles ^H and BACKSPACE so that the character to be erased is actually erased from the terminal screen.
16. Input line parsing ignores the linefeed characters (0x0A) when they are seen. This allows for uploading programs using a terminal emulation program.
17. Added size check to EEP save routine so that if a program in memory exceeds the available EEP space, a message is displayed indicating so.
18. Enhanced the ADC command so that the ADC command now supports channels all channels available on the target MCU.
19. Added the ADR command to set the ADC's reference to internal or external.
20. Added EDUMP command to dump the contents of EEP.
21. Added VDUMP command to dump the contents of the 26 variables [A..Z].

Bugs:

Probably lots!

Have fun!