Introduction To The PICAXE

By Darrell Davis KT4WX - ARRL Technical Specialist
6350 Mills Road, Fort Meade, FL 33841-9584
Email: kt4wx@arrl.net Phone (863) 245-9923

ABSTRACT

The purpose of this paper is to introduce the amateur radio operator to designing microcontroller based projects with the PICAXE microcontroller. Being the PICAXE microcontroller is programmed in BASIC programming language, the ease of learning to use the device for the newcomer as well as the radio amateur who has programmed in BASIC programming in times past, is much easier. This paper will explain what PICAXE microcontrollers are and how to get started in using them in project designs.

Keywords: PIC, PICAXE, Microcontroller, BASIC, Programming.

I. INTRODUCTION TO THE PICAXE

History of the PICAXE Microcontroller: The PICAXE microcontroller was first introduced in 1999 by Revolution Education Ltd. (also known by the abbreviation Rev-Ed), a British company. The PICAXE microcontroller is a PIC Microcontroller pre-loaded with a BASIC interpreter developed by Rev-Ed. Since that time Rev-Ed. Has made many improvements to the BASIC interpreter and has added more PICAXE controllers to their product line as well.

Advantages of using the PICAXE versus the PIC or other microcontrollers: (1) For the amateur radio operator who is new to programming microcontrollers, the PICAXE is a great way to get started in programming with microcontrollers, because it is programmed in a customized version of BASIC and is easier for newcomers to get started with.. The PIC Microcontroller by Microchip, the AVR microcontroller by Atmel Semiconductor, or other 8 bit microcontroller lines are programmed with a derivative of the C or C++ programming language. Usually the manufacturers of these microcontrollers even supply a well designed Integrated Development Environment (IDE) program that will run on a IBM PC Compatible computer running the Windows OS. This IDE will allow the programmer to write, run simulations, debug, and program the target microcontroller, via an external programmer that is attached to the serial port or a USB of that computer. The external programmer is either supplied by the manufacturer of the microcontroller or by a third party. However many amateur radio operators are new to programming and have never learned C or C++, which is a wonderful programming language. However, C or C++ is intimidating to the newcomer and can discourage them from further attempts at building with microcontrollers.

(2) For the veteran amateur radio operator, who has used BASIC in the late 1970s through the early 1990s, this is familiar ground, and can make their start with microcontrollers much easier. Also the PICAXE is a great way to get a young person started into programming microcontrollers and once they are comfortable may wish to move up to using C o C++ in future projects.

(3) Because the PICAXE is a stand along microcontroller, it can be integrated into projects more

14

seamlessly than other microcontroller platforms that are modular in nature. However there are breakouts boards and even some modular forms of the PICAXE available for designers as well.

(4) Most PICAXE designs are available primarily in Dual In Line (DIP) packaging for those who are new to microcontrollers and are not comfortable yet with surface mount soldering or have yet acquired the skills to solder Surface Mount Technology (SMT) devices onto printed circuit boards. This also makes breadboarding new designs much easier for the newcomer as well. However all of the PICAXE line is available in SMT packaging as well for those who wish to use surface mount devices.

(5) The PICAXE BASIC Interpreter provides commands that are easy to use, to control all the underlying PIC Microcontroller general purpose I/O ports and custom ports such as the SPI and I2C serial ports, A/D converters.

(6) The PICAXE controllers are well documented in several well done manuals in PDF format.

II. PICAXE PRODUCT LINE AND BASIC SPECIFICATIONS

As of this writing, there are currently seven PICAXE based IC available for use in new designs. PICAXE microcontroller based IC has been produced in several versions. Microchip still produces the PIC Microcontroller based microcontroller itself. After fabrication and packaging (installing the IC dies into its DIP or SMT package), Microchip then programs the "bootstrap firmware" onto the PIC Microcontroller and then it is shipped out to Rev-Ed., the end customer. Then Rev-Ed ships the PICAXE controller to distributors and makes them for sale directly to customers via their website. All of the microcontroller manufacturers for large production runs from a single customer will offer this service or pre-loading the customer written program onto the target microcontroller after production before the microcontroller order(s) are shipped to the customer for incorporation into their end product. For smaller quantities, the manufacturers will sell them directly to distributors and the distributors sell to small or mid quantity customers.

Earlier versions of the PICAXE had the standard PIC microcontroller model number engraved on it. Recent and current versions of the PICAXE have the PICAXE microcontroller model number engraved on the package by Microchip when it is packaged, which for the end user leads to less confusion. If a designer were to have standard PIC microcontrollers and PICAXE controllers on his or her workbench, there would be no confusion as to which controller is which.

The DIP products are standard 300 millimeter packages with the exception of the 40 pin version which is a standard 600 millimeter package, which is common. The SMT products are in SOIC format. The DIP product will therefore work well with common breadboards making prototyping very straightforward. The DIP and SMT are produced in 8, 14, 16, 18, 20, 28 and 40 pin packages, , which is part of the DIP industry standard.

Listed below is a chart showing the specifications of the PICAXE Microcontrollers. The chart show the program memory, scratchpad RAM size, number of I/O pins available and the clock speed of the particular microcontroller. These charts I produced from data was taken from the PICAXE Manuals.

Processor	Program Memory	I/O Pins	RAM	Clock
PICAXE 08M2	2048 Bytes (2K)	6	128 Bytes	4-32 MHz
PICAXE 14M2	2048 Bytes (2K)	12	512 Bytes	4-32 MHz
PICAXE 18M2	2049 Bytes (2K)	16	256 Bytes	4-32 MHz
PICAXE 20M2	2048 Bytes (2K)	18	512 Bytes	4-32 MHz

Figure 1:PICAXE Basic Line of Microcontrollers

Processor	Program Memory	I/O Pins	RAM	Clock
PICAXE 20X2	4096 Bytes (4K)	18	256	4-64 MHz
PICAXE 28X2	4096 Bytes (4K)	22	1280	4-64 MHz
PICAXE 40X2	4096 Bytes (4K)	33	1280	4-64 MHz

Figure 2: PICAXE Advanced Line of Microcontrollers

The devices vary for 8 pin versions with 2 Kilobytes of program memory, 128 bytes of "scratchpad" of RAM ,a 4-32 MHz clock, and 6 I/O pins, all the way to the 40 pin version with 4 Kilobytes of program memory, 1280 bytes of "scratchpad" RAM, 4 -64 MHz clock, and 33 I/O pins. The Program Memory is comparable to the hard drive space on a desktop PC and the RAM is comparable to the physical memory on a desktop PC. When selecting a PICAXE for a new project, you select the one that will best fit your needs. Some may be wondering, "That is not much memory". Remember, you are dealing with what is called an embedded system in the electronics industry. Many embedded devices do not have a Graphical User Interface (GUI), or even have a display at all. Therefore most embedded systems do not require much memory to do a dedicated task, unlike a desktop PC, which has a lot more overhead.

All PICAXE microcontrollers, like there PIC microcontroller "lineage" are TTL level devices. They will operate on a supply voltage of 3-5 Volts quite satisfactorily. However do not exceed 5VDC on the supply or port pins or you can, and possibly will, damage the device. This is the most important consideration when using these microcontrollers.

III. PICAXE BASIC COMMANDS

Figure 3 and Figure 4 contain a list of the more commonly used BASIC commands that are available in the PICAXE Basic Interpreter. The PICAXE Manuals list all the available commands which are simply would take too long to list here. You may look at the entire list of commands also at the PICAXE website at http://www.picaxe.com/BASIC-Commands. Most commands are in operation/operand syntax.

COMMAND	DETAILS	COMMAND	DETAILS
Pause	Pause for a defined number of milliseconds	For/Next	Repeat a loop for number of time. Return from loop
Pauseus	Pause for a defined number of mcroseconds	If / Else	Conditionally execute program code. Alternative
Wait	Delay for a number of seconds	EndIf	End of If / Else statement
Sleep	Sleep for a period of time	Gosub / Return	Call a routine Come back from routine
Doze	Reduce pwr consumed for a short period of time	Goto	Continue program execution from label

Figure 3: PICAXE Commands – Time Delays and Variables

COMMAND	DETAILS	COMMAND	DETAILS
High	Switch an output pin on	Pwmout	Generate a continuous pulse width modulation
Low	Switch an output pin off	Pwmduty	Set duty cycle of a pulse width modulation output
Toggle	Switch an output between on and off	Input	Set a pin as in input
Sound	Generate simple sounds	Output	Set a pin as an output
Button	Detect and debounce a pushbutton switch	If Pin	Respond to the state of an input pin

Figure 4: PICAXE Commands – Digital Input and Output Commands.

IV. PICAXE PROGRAMMING AND SETUP

PICAXE Programming Setup: The PICAXE microcontrollers are all programmed with a three pin serial connection. Serial programming is ubiquitous today with virtually all microcontrollers. In the PICAXE documentation these pins are referred to as SERIN, SEROUT, and 0V. The following is the function of those pins:

- SERIAL IN   Program data going from the host PC to the PICAXE.
- SERIAL OUT   Communications data going from the PICAXE back to the host PC.
- 0V   Signal Ground

Figure 5 below illustrates how to setup the PICAXE for programming. The beauty of this setup is it allow for In Circuit Serial Programming (ICSP). With ICSP, you can upload an updated or revised program without ever removing the PICAXE from the circuit. One 10K Ohm resistor must be placed across the SERIN and 0V pins. Then between the 10K resistor and the SERIN pin, a 22K Ohm resistor must be placed as well. When building any PICAXE circuit you must include two resistors. Without these resistors the PICAXE could go into program mode by accident once the PICAXE is installed into a circuit. So be sure to include these resistors even if you do not plan to make provision for ICSP. You may program a PICAXE with a standard RS232 serial port or with a USB port.
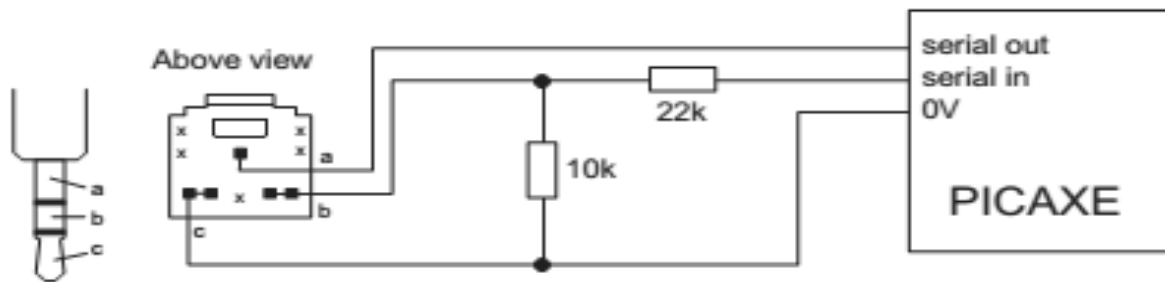

Figure 5: Serial PICAXE Configuration

Programming with an RS232 Serial Port: If you use an RS232 serial port, connection is straightforward. SERIAL IN would go to the TXD pin and SERIAL OUT would go to the RXD pin and 0V would go to Ground.

Programming with a USB Port: If you use a USB port, you will need a USB to RS232 Serial Port Adapter. The price of these adapters today is anywhere from $5-$20 USD. Also you will have to install the appropriate drivers in your computer to support the USB to Serial converter. Then you connect the output of the USB to Serial converter to the PICAXE in the same manner as a regular serial port connection.

PICAXE Setup: Remember the following precautions when setting up a PICAXE project

1. Do not power the PICAXE with more than 5VDC period!!! Supplying more than 5VDC to the PICAXE can damage or destroy the PICAXE.
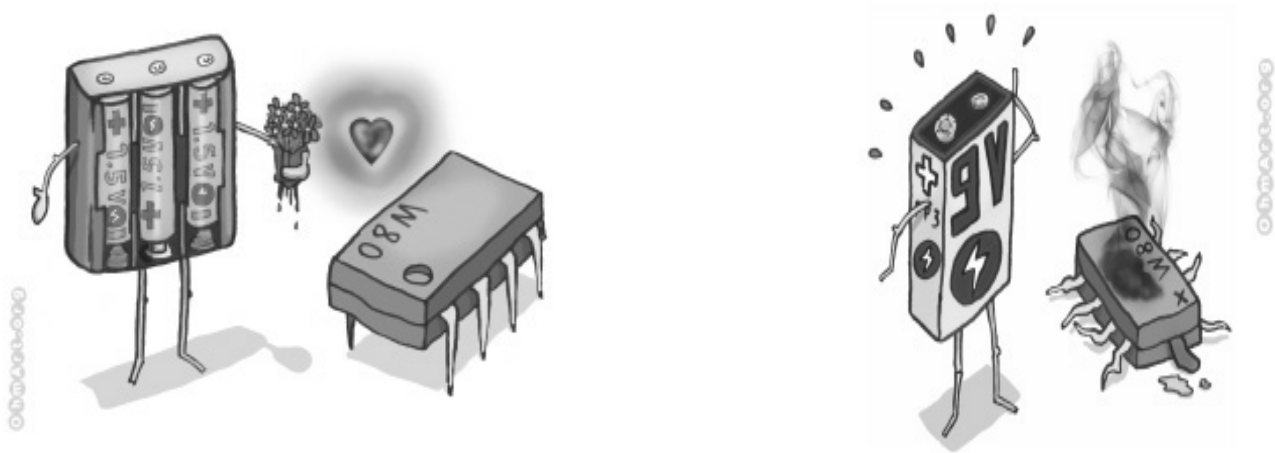

Figure 6 – A happy PICAXE running on 5VDC or less
and a fried PICAXE after 9VDC is applied to it.

2. Each output can only source or sink about 20 mA of current. The maximum that the device should source or sink is 90 mA of current. If you exceed these limits you can damage the PICAXE output ports. You can drive an LED or LED segment directly without any issues but you would not be able to power a relay coil or a motor directly. If you need to control heavy current loads you must use an interface circuit that will not load the microcontroller but will source or sink the current itself.

3. Reset pin (if equipped) must be terminated to the supply voltage through a 4.7 K Ohm or a 10 K Ohm resistor. Unreliable operations will be the result if you do not pull the reset pin to the supply voltage.

PICAXE Programming Software: Rev-Ed has produced some great software for working with the PICAXE and some of it is oriented towards young people new to programming. Here is a rundown of the programming software packages as of this writing.

- PICAXE Editor Version 6 – This is the main software for programming the PICAXE. It is a GUI based editor with a built in PICAXE Simulator. It is only available for the Windows OS.
- Blocky for the PICAXE – A visual block and text editor for PICAXE programming. It is available for the Windows, Linux, Mac, iOS. And Android operating systems.
- AXEPad – A cross platform development tool for the PICAXE. It is available for the Windows, Linux, and Mac OS.

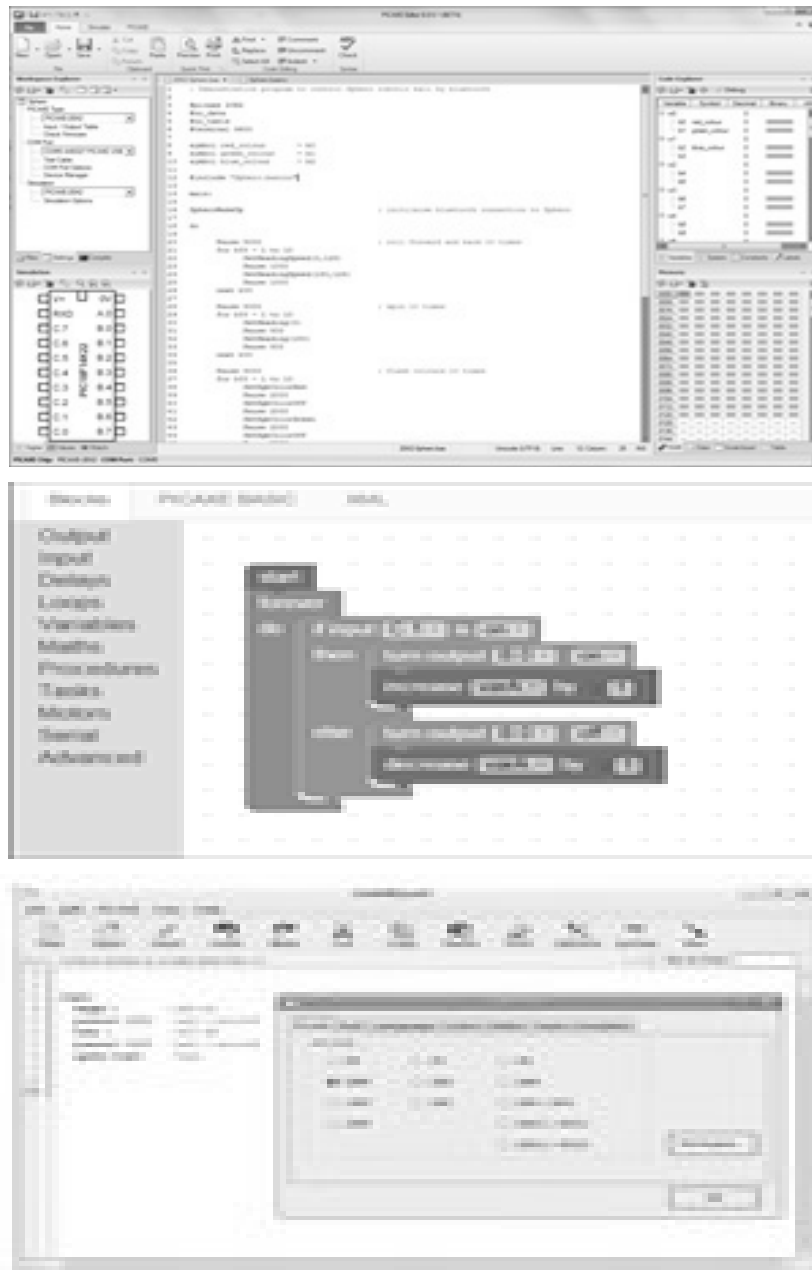PICAXE Programming Manuals: Rev-Ed has done an excellent job in producing user mauals for the PICAXE.

Figure 7: The PICAXE Programmer 6, Blocky for the PICAXE, and
AXEPAD for the PICAXE Software Screenshots.

V. SAMPLE PICAXE PROGRAM

Listed below is the equivalent of the "Hello World" program for the PICAXE. This program is taken for the PICAXE Manuals. In computer programming, the first program that a programmer often learns is how to make the computer screen say "Hello World". This program is a very simple program but illustrates the basic components of a PICAXE program and how easy it is to program these devices. The program is referred to as "Blinky".

Figure 8 shows a simple circuit for a PICAXE 08M2. The port numbers referenced are pins on the PICAXE 08M2. The following simple program turns the LED on for one second and then turns the LED off for one second and then repeats endlessly until you turn the power off or reset the PICAXE.

```
main:          high 4
               pause 1000
               low 4
               pause 1000
               goto main
```
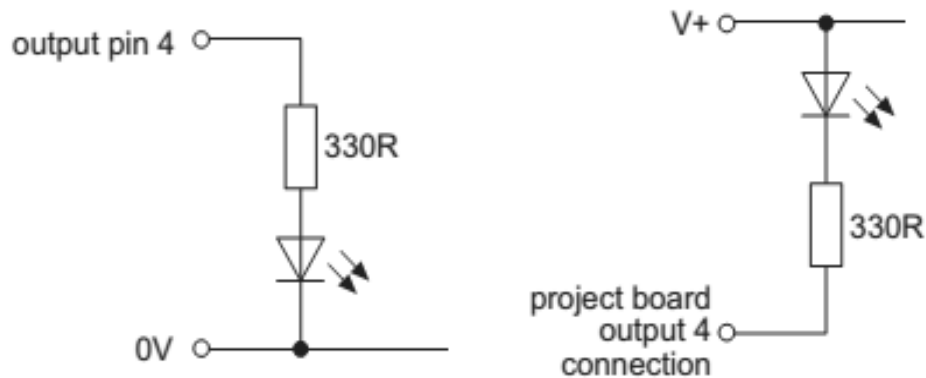
Figure 8: PICAXE BASIC Program for "Blinky"

Figure 9: PICAXE Circuit options for "Blinky"

All PICAXE programs must contain a MAIN function or routine. The MAIN Function calls upon all other functions and should continuously loop or your device will crash while running.

The MAIN functions does the following in this program: The command HIGH takes pin 4 of the PICAXE 08M2 to V+, which is the supply voltage. Then the PAUSE command pauses the PICAXE for 1000 milliseconds or 1 second. Then the command LOW takes pin 4 of the PICAXE 08M2 to 0V, which is ground. Again the PAUSE command pauses the PICAXE for 1000 milliseconds or 1 second. The GOTO commands takes the PICAXE back to the beginning and starts the MAIN program all over again.

A good microcontroller program, being it is a standalone device, will continuously run until it is turned off or the device is reset.

VI. PICAXE RESOURCES

Are you ready to dive into the world of microcontrollers with the PICAXE? The following is a list of resources to help you get started programming and using the PICAXE. If I receive good feedback from this presentation, I will endeavor to write a book titled something like, "PICAXE for Ham Radio".

BOOKS:

- Ham Radio for Arduino and PICAXE by Leigh L. Klotz Jr. WA5ZNU ISBN-13: 978-0872593244. (This book has some real good PICAXE projects that you may be interesting in building.)
- Programming and Customizing the PICAXE Microcontrollers by David Lincoln ISBN-13: 978-0071457651

INTERNET:

- PICAXE Website: http://www.picaxe.com (Well built and good material here.)
- PICAXE Programming Software: http://www.picaxe.com/Software (Goes over the various software packages available and all but one are for FREE download.)
- PICAXE Hardware Manuals: http://www.picaxe.com/Getting-Started/PICAXE-Manuals/ (There are five PDF manuals total and they are very well done: Manual #1 - Getting Started, Manual #2 - BASIC Command list, Manual #3 - PICAXE Interfacing, PICAXE #4 – Flowcharts, and Manual #5 – Blocky. They are also for FREE download.)
- PICAXE Wikipedia Page: http://en.wikipedia.org/wiki/PICAXE (Basic introduction to the PICAXE microcontroller)
- YouTube Video Tutorials:

  ◦ https://www.youtube.com/user/PICAXEdotcom (Main PICAXE YouTube Channel.) This YouTube channel was recently added.

  ◦ https://www.youtube.com/user/azadude18/videos (Produced around 2010 but are good videos to help get you started)

REFERENCES

PICAXE. Revolution Education Ltd., n.d. Web. 16 Aug. 2016. <http://www.picaxe.com>.

"PICAXE Manual." *PICAXE*. Reolution Education Ltd., n.d. Web. 16 Aug. 2016. <http://www.picaxe.com/Getting-Started/PICAXE-Manuals/>.