All Things Digital

Amateur Radio for the 21st Century

006

Robert C. Mazur, VA3ROM
E: va3rom@gmail.com
W: www.va3rom.com

Article first published in the January-February 2013 issue of The Canadian Amateur

MICROCONTROLLER UNITS (MCUs): PART DEUX

In my previous column (Nov-Dec 2012), the MCU and PICAXE were introduced (additional PICAXE articles and programs are available from my website). But there are many *TCA* readers who may have limited electronics/computer programming experience and the PICAXE can be a bit intimidating since you need to build (solder) many of the kits, and the training material requires a higher level of electronics knowledge, but Parallax's BASIC Stamp MCU system and training courses can help overcome these problems. The BASIC Stamp should have been the first article, but I started with the PICAXE and had no prior experience with the Stamp until just recently.

THE BASIC STAMP

The very popular Parallax Inc. BASIC Stamp MCU (the original was the size a postage stamp) is programmed in the BASIC programming language and comes in various versions with increasing power and complexity (much the same as the PICAXE). It is widely used in the U.S. school system (Stamps in the Classroom) as part of the STEM (Science Technology Engineering and Mathematics) program. Parallax's BASIC Activity Kit comes with a HomeWork (HW) board, BASIC Stamp (BS2) module, all the necessary electronic components for the experiments and projects, and the *What's a Microcontroller?* manual (also available as a free PDF download from the Parallax website). The course takes about 40 hours to complete and is a totally self-contained, all-in-one package (Figure 1, next page).

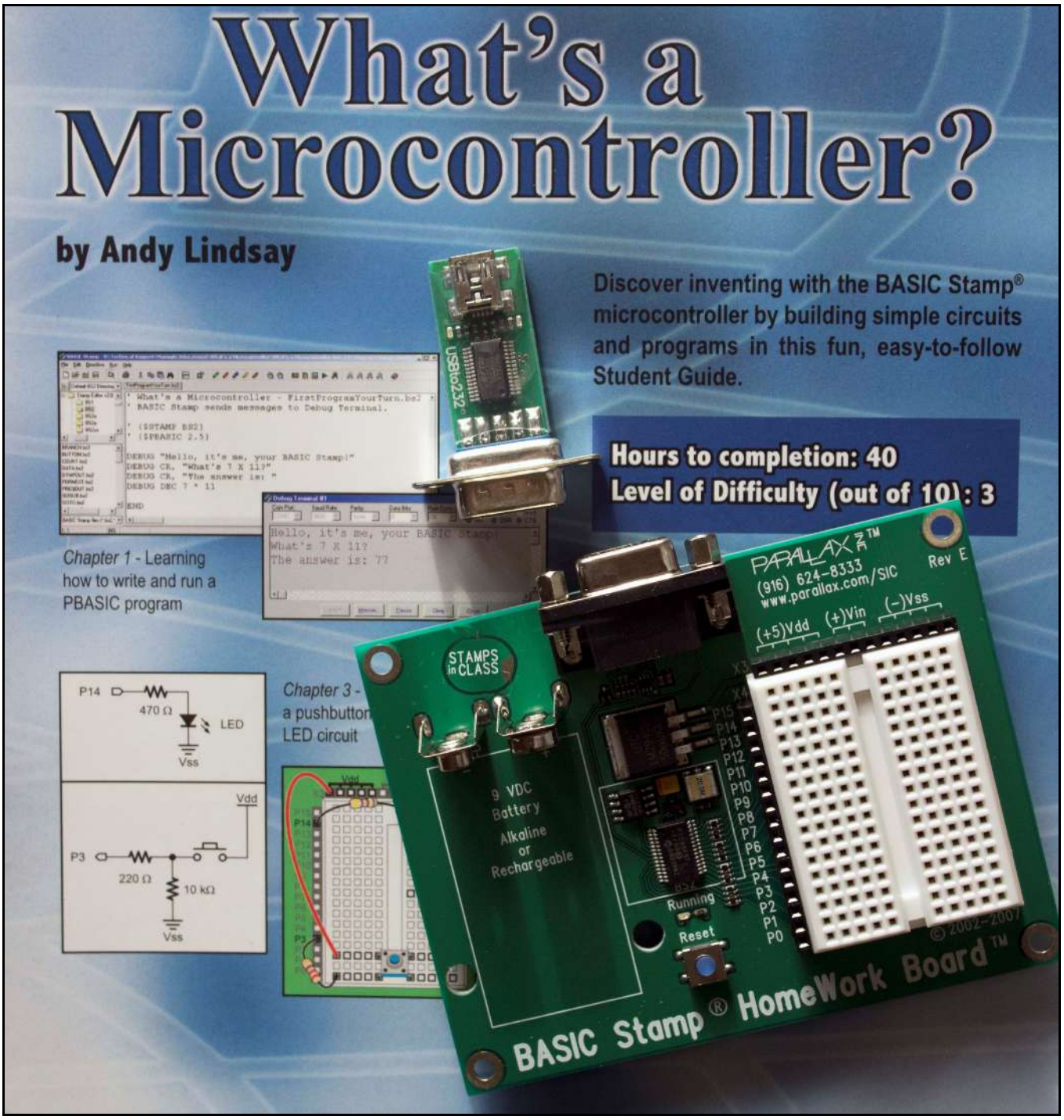


FIGURE 1: Parallax *What's a Microcontroller?* BASIC Stamp Course

Interestingly, the BASIC Stamps (BS 1 and BS2) are made by Microchip Technology Inc. (it's a really a PIC variant) and there are different Stamp MCUs (Spin, Javelin, Propeller, etc.) made by a company called Ubicom. Most of Parallax's products are simple plug-and-play assemblies (no soldering required), and their best known product is the Boe-Bot (Board of Education Robot) which only needs a small screwdriver and pliers to assemble (Figure 2, next page).

The BASIC Stamp course assists educators in teaching electronics and computer programming using real-world applications in a self-contained package specifically designed for classroom use by young students, and this type of approach may appeal to adult students, also!

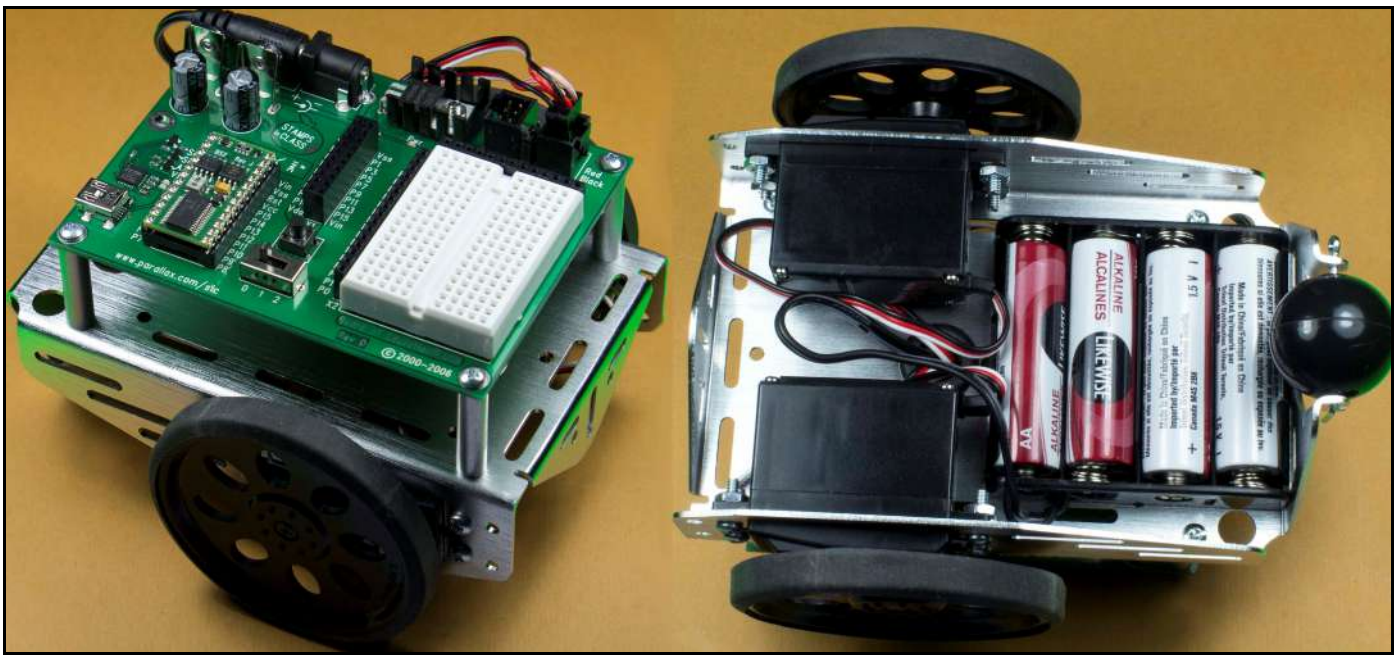


FIGURE 2: The Parallax BOE-BOT Robotics Training Platform

COMPARED TO THE PICAXE

With the Stamp, you don't require any knowledge of the chip's architecture when writing your BASIC programs. For example, you define a program variable or constant (value) simply by giving it a name and a data type (or value), and you don't need to know (or care) where in memory it's stored. The PICAXE, on the other hand, requires you to assign a variable or constant (value) to a specific memory location depending on the storage space required for the variable or constant (value); this means you need a very good understanding of the PICAXE's memory structure—even for simple programs!

Examples of Variable/Constant Declarations

Parallax BASIC			**PICAXE BASIC**		
Counter	VAR	BYTE	symbol	Counter	B0
Pulses	VAR	WORD	symbol	Pulses	W1
PI	CON	314	symbol	PI	314

The above side-by-side program variable and constant declarations comparisons both accomplish the same thing. With Parallax BASIC you define a variable (or constant) name and its data type, but with PICAXE BASIC, you MUST specify the variable (or constant) name AND its memory assignment (location), except constants such as PI.

PICAXE memory location B0 can only hold 1 byte of data (from 0 to 255) and it's paired with byte memory location B1, and the two combine to create memory location W0 which can store one "word" of integer data ranging from 0 to 65535. However, once you allocate memory location B0 to a byte variable, you can't declare a word variable using memory location W0 (only another byte variable B1); therefore, the variable Pulses must be allocated to another unused WORD memory location called W1 (memory byte pair B2 and B3). So, are you totally confused, now?

Note: *The PICAXE and Stamp MCUs can only perform integer arithmetic (no decimals or negative values). To use a constant like PI (3.1415926), we need to divide by 100 to get the proper integer answer (rounded).*

Another important difference is with the Stamp because it doesn't have built-in ADC (Analogue-to-Digital Conversion) inputs like the PICAXE does, so you must use an external resistor/capacitor circuit and the RCTIME command, or you can use an external ADC chip or comparator Op-Amp (Operational Amplifier). It's still possible to do things like read analog voltages from an LDR (Light Dependent Resistor) or a potentiometer, but it's just not as convenient as compared to the PICAXE. *See References and Resources, Microcomputers for Ham Radio, Part 1—BASIC Stamp which has sample ADC circuits.*

YOUR FIRST BASIC STAMP PROGRAM!

By tradition, the first computer program written is called "HELLO, WORLD!" (See chapters 1 and 2 of the *What's a Microcontroller?* manual). You require the BASIC Stamp Activity Kit, the free Program Editor, and a USB programming cable, plus a 9V DC battery to power the board. You then type in the program, and when you click the RUN button, your program is loaded or "burned" into the BS2's program memory and executed automatically; this should flash the LED (Light Emitting Diode) on and off for as long as power is supplied to the board.

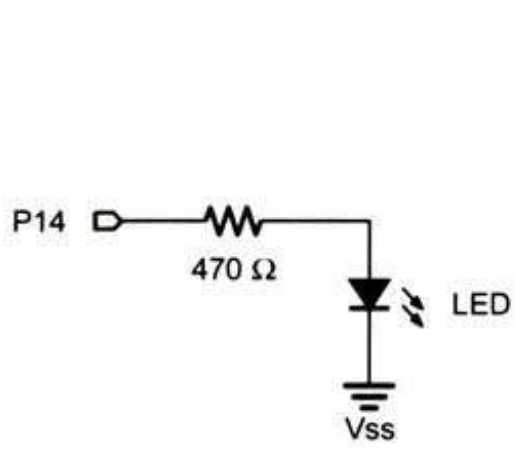
Figure 3 (next page) shows the BASIC program code along with the circuit built on the kit's mini-solderless breadboard.

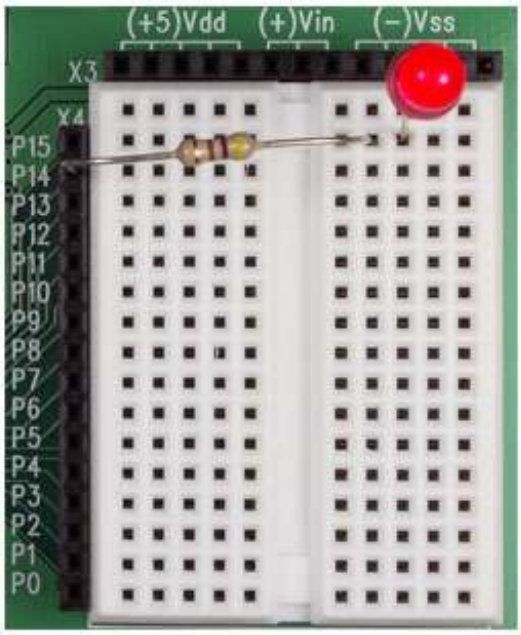
FIGURE 3: FLASH LED PROGRAM AND CIRCUIT

\Stamp Editor v2.5.3\BS2\LEDOnOff.bs2

LEDOnOff.bs2

```
' {$STAMP BS2}
' {$PBASIC 2.5}

' Program: LEDOnOff.BS2
' Purpose: Blinks an LED connected to BASIC Stamp pin (port) P14

' ---------------------------------------------------------------

DO              ' do everything that follows below (start of loop)

  HIGH 14       ' turn LED on
    PAUSE 500   ' wait one-half second (500 milliseconds)

  LOW 14        ' turn LED off
    PAUSE 500   ' wait one-half second (500 milliseconds)

LOOP            ' do it again (loop forever or until power disconnected!)
```





At the end of each chapter is a summary along with questions and exercises. For example:

1. Which resistor would allow **more** current to flow in a circuit, a 470 ohm or a 1000 ohm?
2. For how long would a PAUSE 10000 last?
3. Modify the program to flash the LED on/off 4 times per second.

As you go deeper into the course and gain more experience the chapter questions and exercises won't be as easy, and this is the idea behind the "Stamps in the Classroom" educational products and courses. Parallax has developed several other courses and you can download the various free PDF manuals to see which appeal to you. For example: the *Robotics with the Boe-Bot* course is an excellent introduction to the exciting and fun world of robotics using MCUs.

MY FINAL

Well, that's just the tip of the iceberg—the rest is up to you! If you're interested in exploring MCUs, robotics and programming then have a serious look at the Parallax BASIC Stamp product line. Most programs and circuits using either the Stamp or PICAXE can be converted to work on other MCU platforms (and vice versa).

In the next column, we'll look at Winlink 2000 (WL2K) Global Radio Email Messaging System. You can use your Amateur Radio transceiver(s) to send and receive messages without needing the Internet, Wi-Fi, or landlines!—73

REFERENCES AND RESOURCES

BASIC Stamp http://www.parallax.com/tabid/295/Default.aspx
BASIC Stamp Support Index http://www.emesystems.com/BS2index.htm

Parallax http://www.parallax.com

Stamps in Class Mini Projects http://learn.parallax.com/basicstamp

Microcontrollers and Amateur Radio
http://www.qcwa70.org/presentations1.htm

Microcomputers for Ham Radio, Part 1—BASIC Stamp
http://tinyurl.com/b6bmhzo

What's a Microcontroller? http://www.parallax.com/downloads

SOME CANADIAN MCU & ELECTRONIC SUPPLIERS

Cana Kits http://www.canakit.com
Robot Shop http://www.robotshop.com/ca
SOLARBOTICS http://www.solarbotics.com

ADDITIONAL ARTICLES

Introducing Microcontrollers to Your Amateur Radio Club *TCA, Jan-Feb 2010, (p. 43-47).*