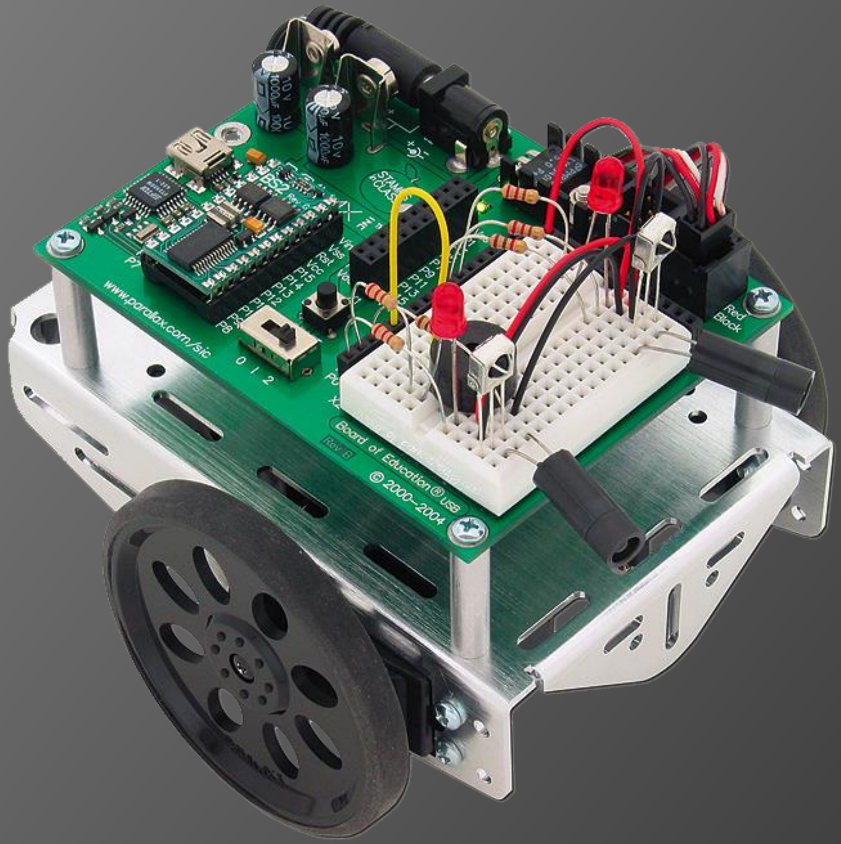
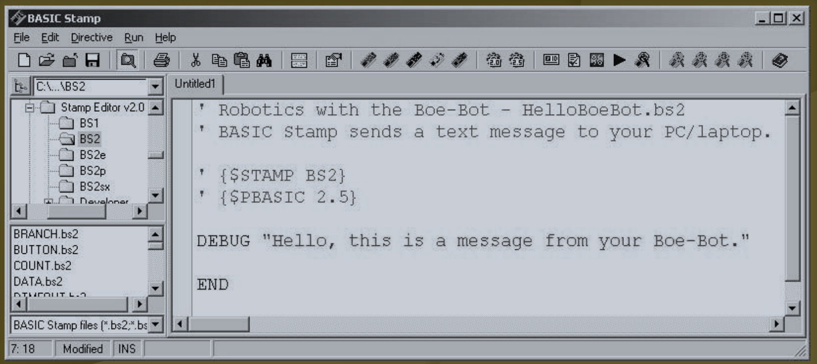


Introduction

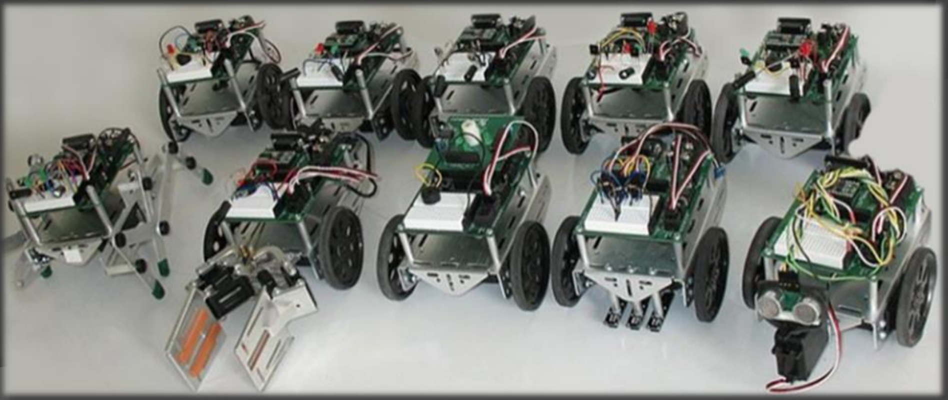
BOE-BOT



```
' Robotics with the Boe-Bot - HelloBoeBot.bs2
' BASIC Stamp sends a text message to your PC/laptop.

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

DEBUG "Hello, this is a message from your Boe-Bot."

END
```

Overview

- Introduction
- Resources
- Why Study Robots
- Selecting a “bot”
- Potential Choices
- The Boe-bot
- Board of Education
- The BASIC Stamp
- Microprocessors
- Microcontrollers
- Programming
- BASIC Stamp Editor
- PBASIC Programming
- Commands
- Sample Programs
- Light Emitting Diode
- Resistors
- Breadboard
- Controlling 2 LED’s
- Pulse out Command
- Review



Introduction to Robots

- Found in auto, medical, manufacturing and space industries
- Essential characteristics:
  - **Sensing** - Input devices
  - **Movement** - Servo's
  - **Energy** - Power source
  - **Intelligence** - Programming
- Is a system that contains sensors, control systems, manipulators, power supplies and software all working together to perform a task
- Problem, posing, solving, designing, building, programming and testing a robot is a combination of physics, mechanical engineering, electrical engineering, structural engineering, mathematics and computing

Resources

- **Handouts**:
  - Selected pages from the reference items listed below:
- **Reference**:
  - For more complete documentation, the following items are available from www.parallax.com or http://mfranzen.ca/pages/cor/tej2o1_u3.html

BASIC Stamp Windows Editor Version 2.2 (5.81 MB)

PARALLAX

BASIC Stamp®

Editor / Development System

Supported Modules: BS1, BS2, BS2e, BS2sx, BS2p, BS2pe, BS2px

Version: 2.2

Copyright: 1998-2005

http://www.parallax.com

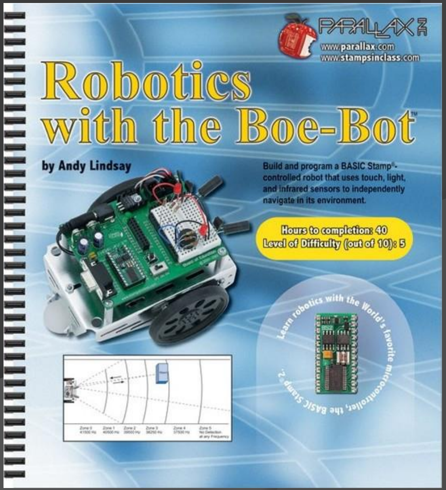
stamptech@parallax.com

BASIC Stamp Syntax and Reference Manual Version 2.1 (4.25 MB)

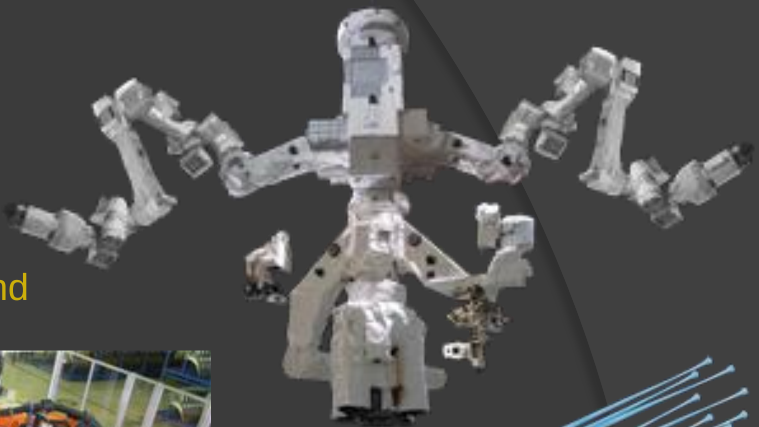
BASIC Stamp Syntax and Reference Manual

Version 2.1

Robotics with the BOE BOT Version 2.2 (4.4 MB)

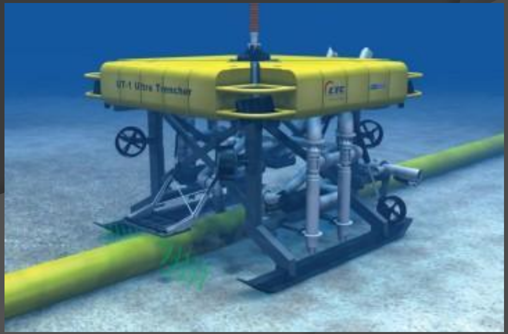

PARALLAX
www.parallax.com
www.stampsinclass.com
Robotics with the Boe-Bot™
by Andy Lindsay
Build and program a BASIC Stamp®-controlled robot that uses touch, light, and infrared sensors to independently navigate in its environment.
Hours to completion: 40
Level of Difficulty (out of 10): 5
Learn robotics with the World's favorite microcontroller, the BASIC Stamp® 2.

Why Study Robots



- ***Applications*** – There are creative uses of robots all around us in fields such as:
  - Space exploration
  - Undersea mapping
  - Manufacturing
  - Medicine
  - Navigation
  - Military applications

- ***Communicate with external devices*** – A robot is little more than a specialized computer used to read input devices (sensors) and to control output devices (motors, relays, servos, lights, sirens, etc.) It is challenging and fun to use computers to accomplish tasks and the applications are unlimited.
- ***Common usage*** – A large number of introductory engineering courses around the country include robotics projects. Students are also commonly involved in robotics-based team competitions, such as the FIRST Robotics Competition or the Sumo Bot Competition. Additionally, later engineering courses deal with more advanced robot control and applications.



Selecting a Robot

- The key component to a robot is the computer (microprocessor or microcontroller) that serves as its “brain.” Some features to consider in a microcontroller are:

1) Number of input/output ports
2) Complexity (time required for students to learn to use it)
3) Programming language
4) Power requirements
5) Cost
6) Size

Potential Choices

- Consider the following robots/microprocessors:

1) Lego Mindstorm

- Easy to use
- Limited to three inputs/outputs
- Somewhat oversimplified with all snap-together pieces

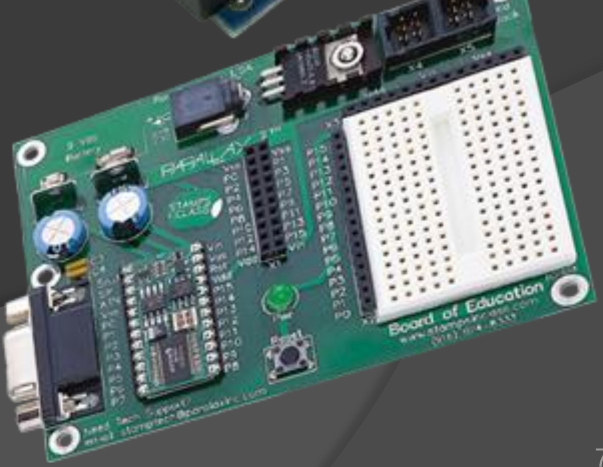
2) Arduino

- An open-source electronics prototyping platform
- ATmega328 Microprocessor, digital & analog I/O’s
- Java, processing, avr-gcc, and other open source languages

3) Parallax BASIC Stamp – ***our choice***

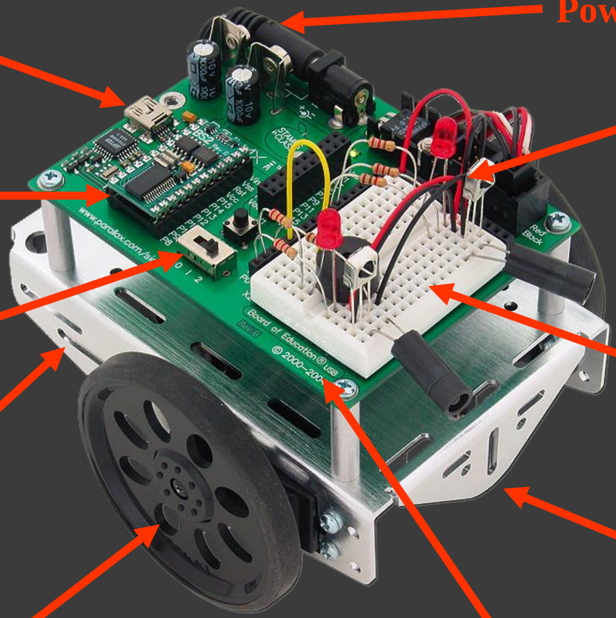
- Easy to program (using a version of BASIC)
- 16 input/outputs
- Inexpensive packaged kit
- Easy to use editor for writing and downloading programs

4)Many others are also available



The Boe-Bot

- The BOE-BOT is a robot using a Board of Education (BOE) mounted on metal frame chasse with two servo motors. We will learn to program the BOE-BOT with different sensors, functions and navigation.



USB Port for PC communications

Power connector

Assorted Electronics - input and output, wire, resistors, etc.

The BOE-BOT is equipped with BASIC Stamp 2

Programming run switch – 3 positions

Breadboard on BOE is convenient for connecting sensors or other circuits.

Chassis has numerous holes for mounting sensors.

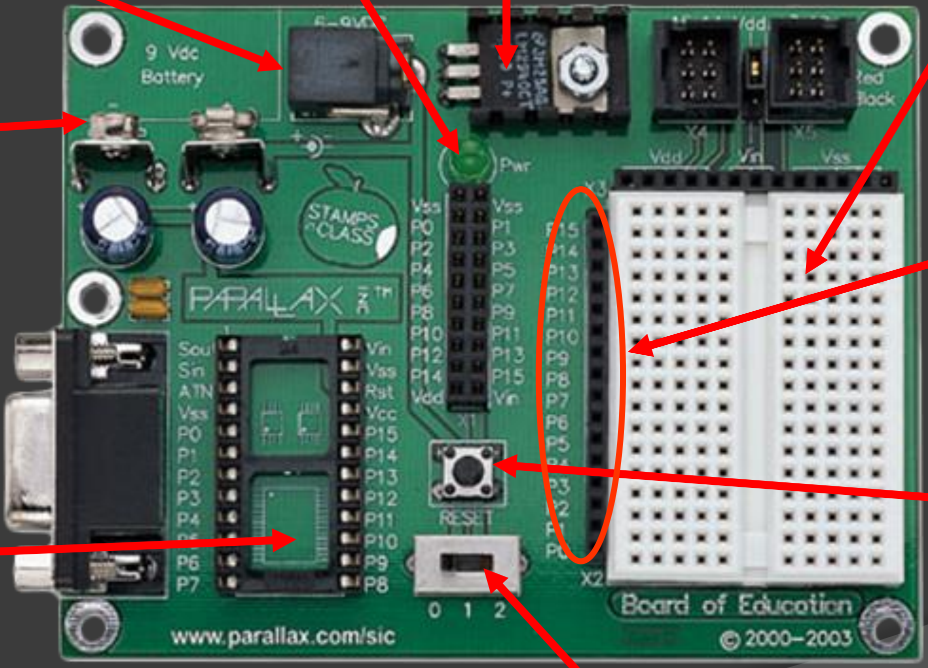
Batteries mounted under the chassis power the BOE, servos, and other devices.

Drive wheels mounted to servos. Steering is accomplished by controlling the servos

Board of Education (BOE)

The “Board of Education”

- Users could mount the BASIC Stamp on their own circuit board and provide their own connections for the computer interface, power supply, and inputs/outputs, but Parallax also sells a convenient circuit board called the “Board of Education” with such features.



Parallax “Board of Education”

DC adapter connector

Green power light

5V regulator

Breadboard (handy for experiments and other circuits)

9V Battery Connector

16 input/output connections

Computer connector (9-pin serial port or USB)

Reset button to restart a program in memory

BASIC Stamp IC Socket (plug in various types of BASIC Stamps here)

Programming switch position

The Basic Stamp

- Parallax, Inc. produces a microprocessor called the **BASIC Stamp**. It's name refers to the fact that it is about the size of a postage stamp and is programmed using a version of the language BASIC called PBASIC. There are several versions of the BASIC Stamp available, each with slightly different features as shown below:

BASIC Stamp 2

- 4,000 Instructions/second
- 20 MHz clock
- 42 PBASIC commands available

BASIC Stamp 2sx

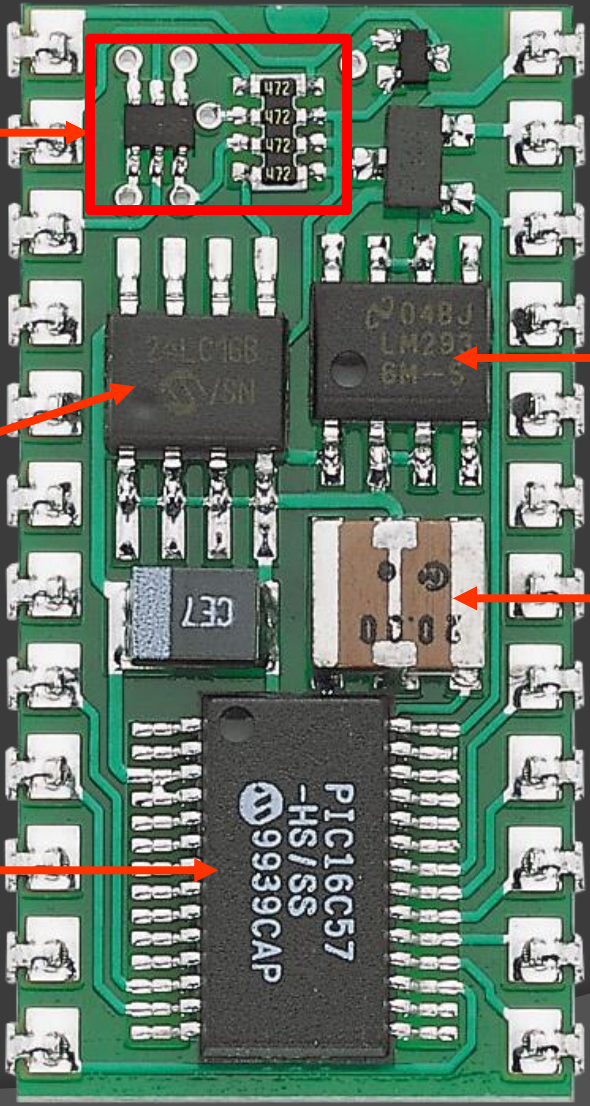
- 10,000 Instructions/second
- 50 MHz clock
- 45 PBASIC commands available

BASIC Stamp 2pe

- 6,000 Instructions/second
- 8 MHz clock
- 61 PBASIC Commands available

More information - Visit www.parallax.com and view the Stamp Comparison Chart for more extensive specifications on the different types of BASIC Stamps.

The Basic Stamp



Serial Signal Conditioning Conditions voltage signals between PC serial connection (+/-12V) and the BASIC Stamp (5V) for Programming

EEPROM Stores the your PBASIC program

Interpreter Chip Reads the BASIC program from the EEPROM and executes the instructions

5V Regulator Regulates voltage to 5V with a supply of 5.5VDC to 15VDC

Resonator Sets the speed at which instructions are processed

Microprocessor

- The “brain” of a computer system
- Generally referred to as the central processing unit (CPU), the microprocessor by itself is practically useless
- To be useful, one must have means of communicating with it using input and output devices
- One must also add memory (ROM and RAM) so that the system can be programmed.

Microcontroller

- A computer chip designed for control-oriented applications
- Unlike ordinary microprocessors, microcontrollers have built-in features that make them operate almost independent of additional circuitry
- This is possible because microcontrollers contain things like
  - memory (ROM, EPROM, RAM, etc)
  - input and output ports
  - timers
  - serial and parallel communication capability
  - analog-to-digital converters

Programming the BASIC Stamp

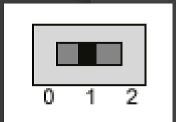
- The BASIC Stamp is programmed using the **BASIC Stamp Editor Version 2.2**. This editor is available on the computers in the H-151 lab and is also available for free download from Parallax at www.parallax.com.

Connections:

A) Connect the serial port (COM1) on the computer to the serial port on the BOE.

B) Connect the battery pack to the BOE.

C) Turn the switch on the BOE to position 1 or 2.



1) Write programs in PBASIC using the editor.

2) Download program to the BOE

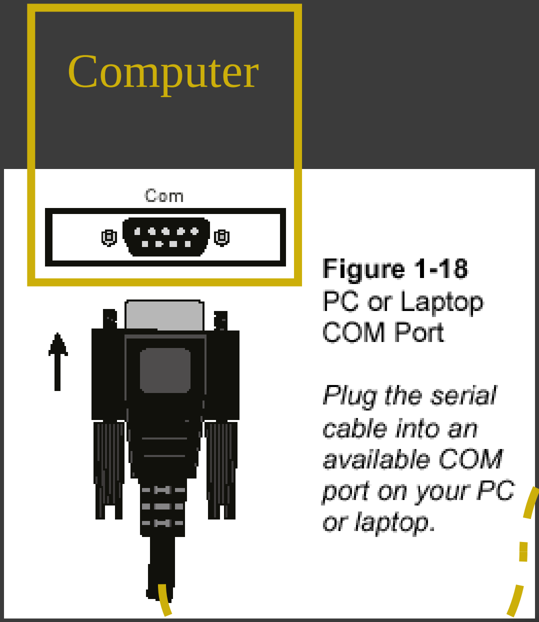
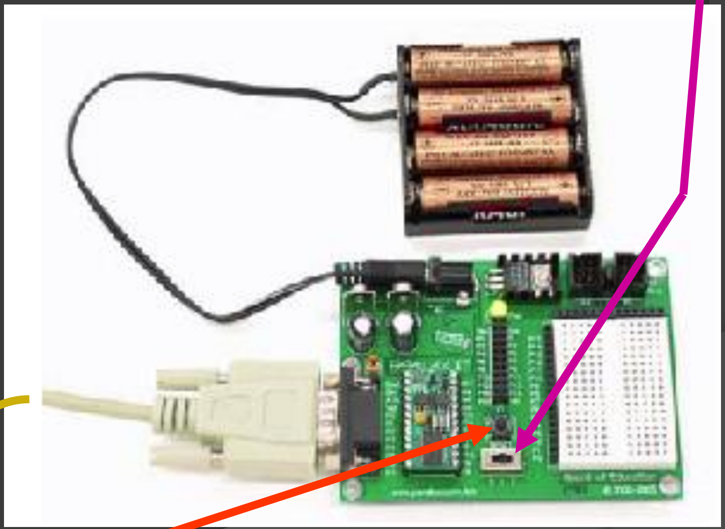


Figure 1-18
PC or Laptop
COM Port

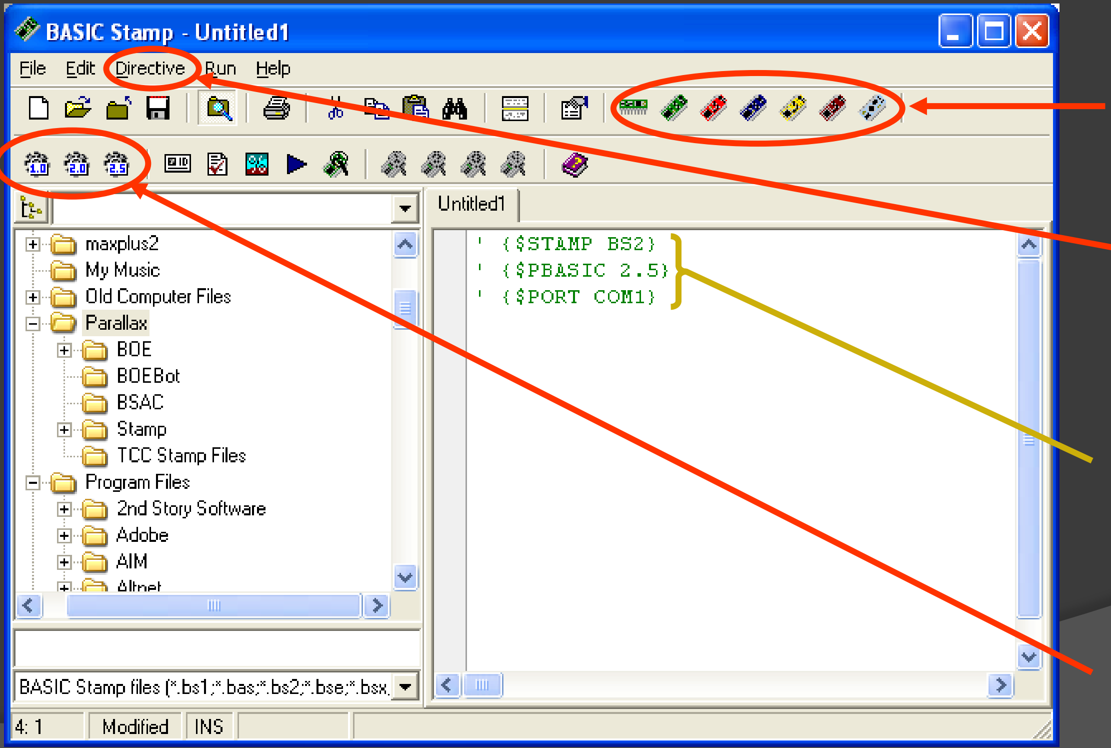
Plug the serial cable into an available COM port on your PC or laptop.



3) Press the RESET button on the BOE to run the program.

BASIC Stamp Editor Version 2.2

- Each program written with the BASIC Stamp Editor must begin with three items:
  - **Specify the version of BASIC Stamp being used.**
  - **Specify the version of the PBASIC programming language being used.**
  - **Specify the port being used to download programs (COM1)**



```
' {$STAMP BS2}
' {$PBASIC 2.5}
' {$PORT COM1}
```

Select the <u>type of Stamp.</u>

Select <u>Directive – Port – COM1</u> to specify the port.

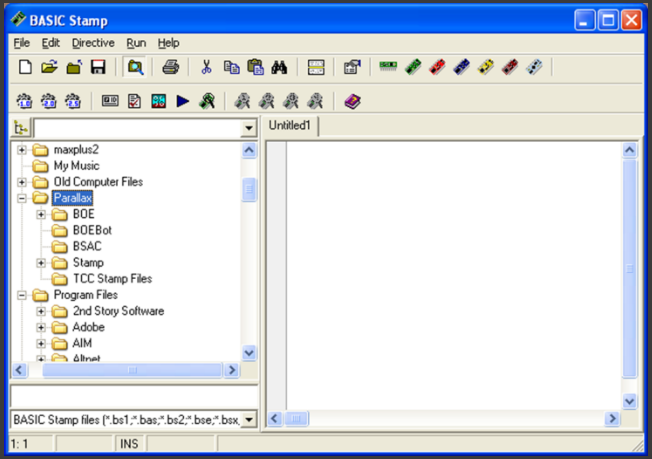
These statements are referred to as <u>compiler directives</u>.

Select the <u>Version of PBASIC.</u>

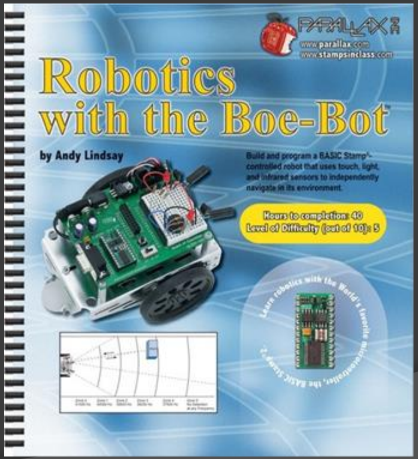
Writing PBASIC Programs

- In writing **PBASIC** programs, there are several good resources that you can use, including:

1. Examples and explanations in Robotics Version 2.2 (instructor handout or available online at www.parallax.com). For the first BOE-BOT lab, the following sections are particularly useful:
   A. Chapter 1, Activity 3 – Setting Up the Hardware and Testing the System (pp 13 - 22)
   B. Chapter 1, Activity 4 – Your First Program (pp 22 - 30)
   C. Chapter 2, Activity 2 – Tracking Time and Repeating Actions With a Circuit (pp 45 – 58)
2. Information on the usage of individual PBASIC instructions is available in BASIC Stamp Syntax and Reference Manual Version 2.1
3. Examples and explanations in this presentation
4. Web site your viewing this presentation from



BASIC Stamp files (*.bs1;*.bas;*.bs2;*.bse;*.bsx


PARALLAX
www.parallax.com
www.stampsinclass.com
Robotics with the Boe-Bot
by Andy Lindsay
Build and program a BASIC Stamp®-controlled robot that uses touch, light, and infrared sensors to independently navigate in its environment.
Hours to completion: 40
Level of Difficulty (out of 10): 5
Learn robotics with the World's favorite microcontroller, the BASIC Stamp 2

BASIC Stamp Syntax and Reference Manual

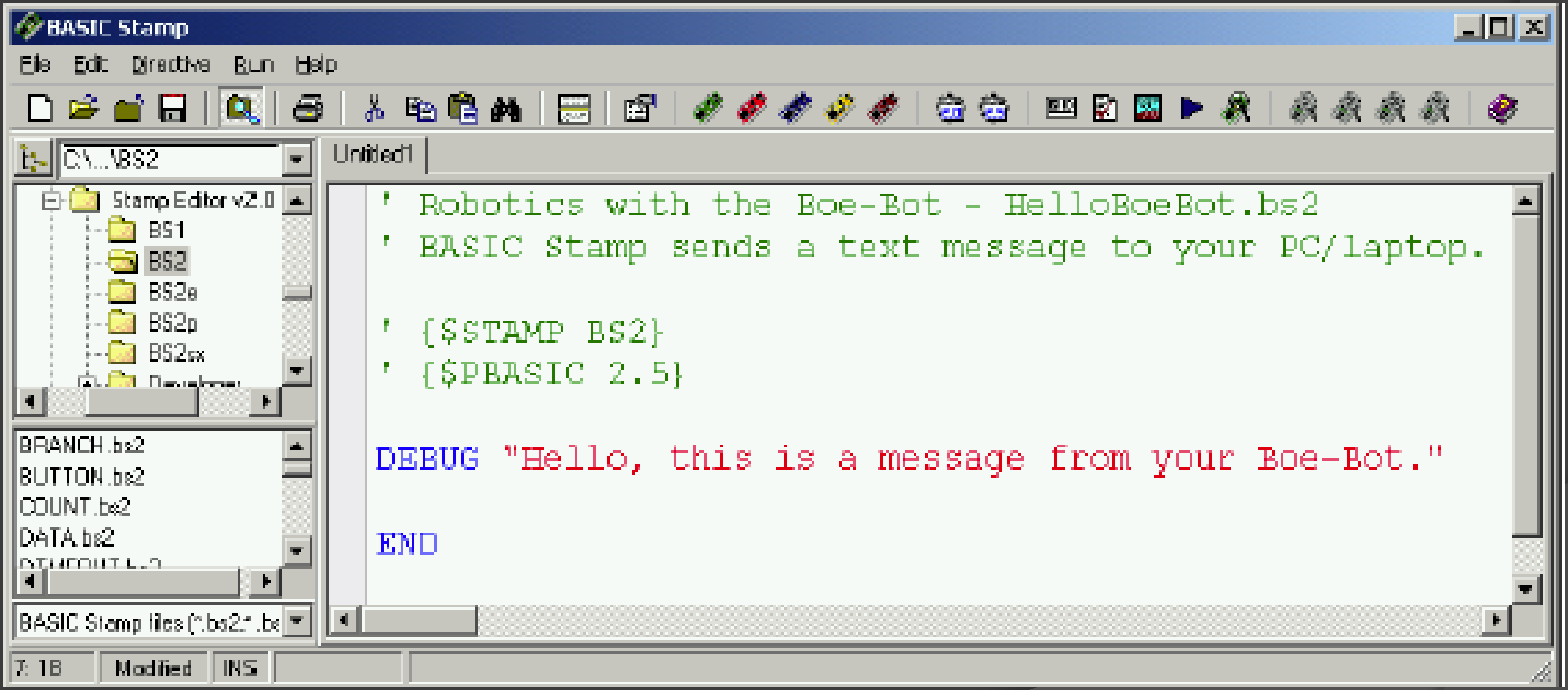
Version 2.1

PBASIC Commands

- We will gradually introduce PBASIC commands as they are needed. Shown below are a few that are needed for the first program
- DEBUG Command
  - **DEBUG** “Text” – This command is used to send a message from the Boe-Bot to the computer screen
  - **DEBUG CR**, “Text” – go to a new line(carriage return before displaying the message from the boe-bot to the computer screen
  - **DEBUG DEC X** – display the value of variable X in decimal format
- END Command
  - **END** – command to end the program. Pressing the RESET button on the Board of Education (BOE) will run the program again
- Comments
  - **‘** – Begin comments with a single quote mark ‘ and can be on the same line as other commands

Sample Program 1

- This program is shown on page 30 of Robotics, Version 2.2
- Discuss each line of the program



```
' Robotics with the Boe-Bot - HelloBoeBot.bs2
' BASIC Stamp sends a text message to your PC/laptop.

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

DEBUG "Hello, this is a message from your Boe-Bot."

END
```

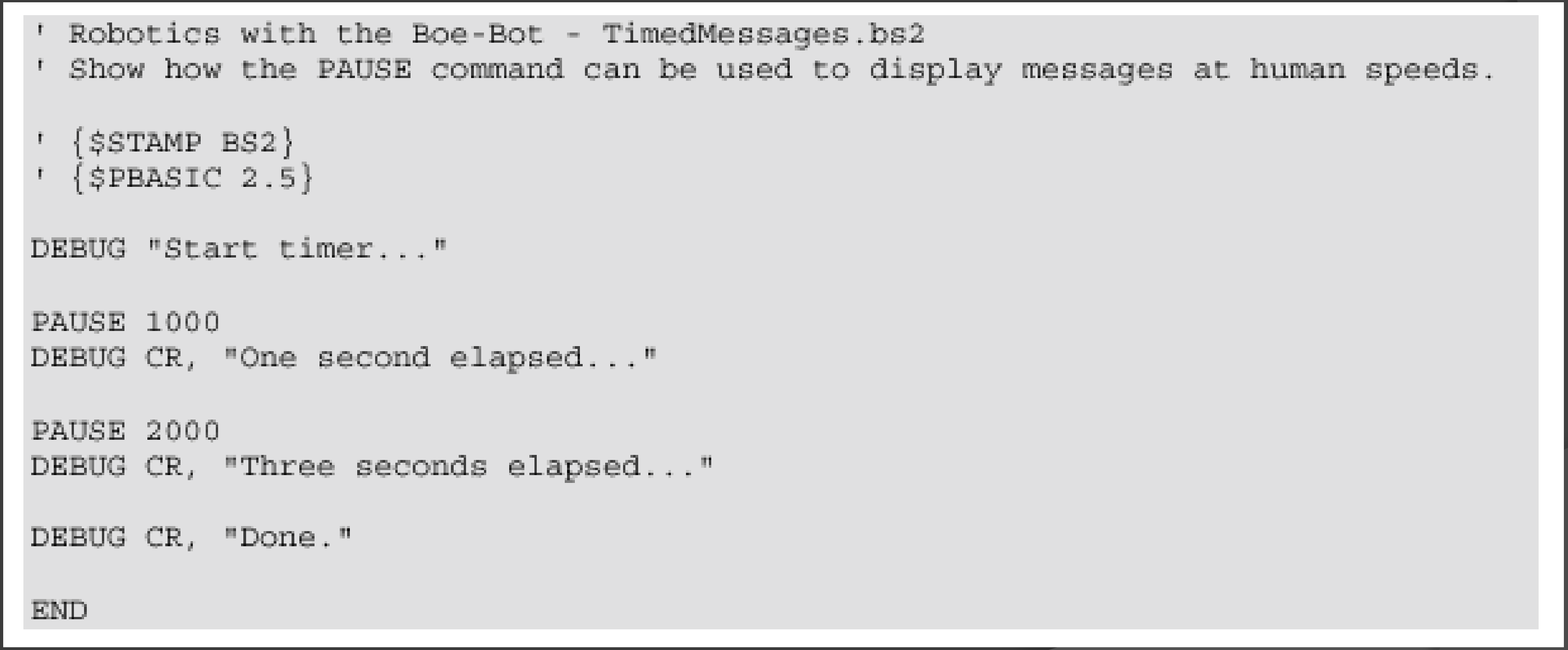
More PBASIC Commands

- HIGH and LOW Commands
  - **HIGH 3** – this command is used to set output P3 HIGH (5V). This can be used with any of the pins P0 – P15
  - **LOW 6** – this command is used to set output P6 LOW (0V). This can be used with any of the pins P0 – P15
- PAUSE Command
  - **PAUSE 500** – this command causes the BASIC Stamp to stop executing the program for N ms (milliseconds).
  - **PAUSE 500** - Pause for 500 ms (0.5 seconds)
  - **PAUSE 4000** - Pause for 4 seconds
- DO Command and LOOP Command
  - **DO** and **LOOP** are used together to mark the beginning and end of an infinite loop in a program so that the instructions in the middle of the loop will be repeated indefinitely
  - Example: (What does this do?

```
DO                      - Start of loop
  DEBUG “Hello”       }
  PAUSE 5000          }Body of loop
LOOP                    - End of loop
```

Sample Program 2

- This program is shown on page 43 of Robotics, Version 2.2
- Discuss each line of the program



```
' Robotics with the Boe-Bot - TimedMessages.bs2
' Show how the PAUSE command can be used to display messages at human speeds.

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

DEBUG "Start timer..."

PAUSE 1000
DEBUG CR, "One second elapsed..."

PAUSE 2000
DEBUG CR, "Three seconds elapsed..."

DEBUG CR, "Done."

END
```

LED’s

- **LEDs** – A simple device that we can control with the BASIC Stamp is an **LED** or light-emitting diode. LED’s are simple lights easily controlled using the 5V output from the BASIC Stamp. You find LED’s on all sorts of electronic equipment. You computer monitor and keyboard probably uses small LED’s that light to let you know when the power is on. See pages 49-53 in Robotics – Version 2.2 for more information

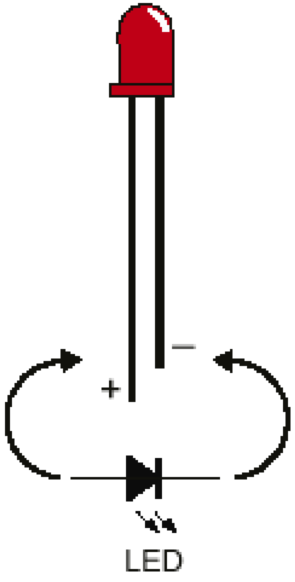


Figure 2-3
LED Part Drawing and Schematic Symbol

Part drawing (above) and schematic symbol (below).

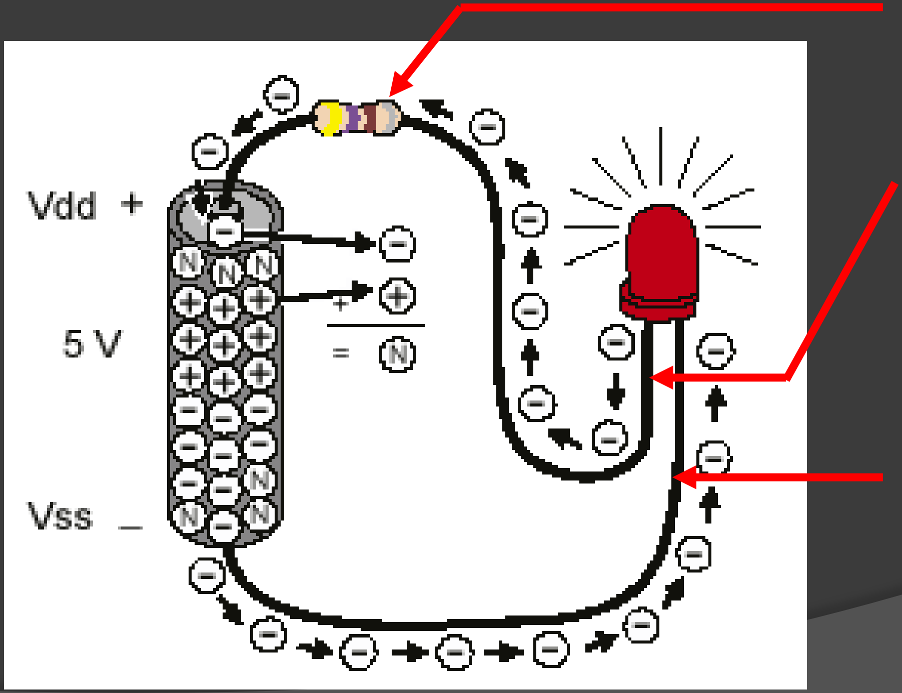
The LED part drawings in later pictures will have a + next to the anode leg.

Polarity – An LED has a positive terminal (anode) and a negative terminal (cathode) so you must be careful to place it in a circuit in the correct direction (correct polarity). You can tell which terminal is which in two ways:

- The lead for the anode (+) is typically longer.
- There is a flat side to the LED by the cathode (-).

LED Brightness

- The brightness of an LED is related to the amount of current passing through it (or the number of electrons passing though it). A device called a **resistor** is generally placed in series with the LED to limit the amount of current. When 5V is used to light the LED, resistor values from 200 ohms to 470 ohms are commonly used (the LED is brighter with 200 ohms than with 470 ohms). Using no resistor will result in too much current and will often destroy the LED. The resistor is often referred to as a **current-limiting resistor**



Current limiting resistor

LED anode (+)
connected to positive
(+) side of battery

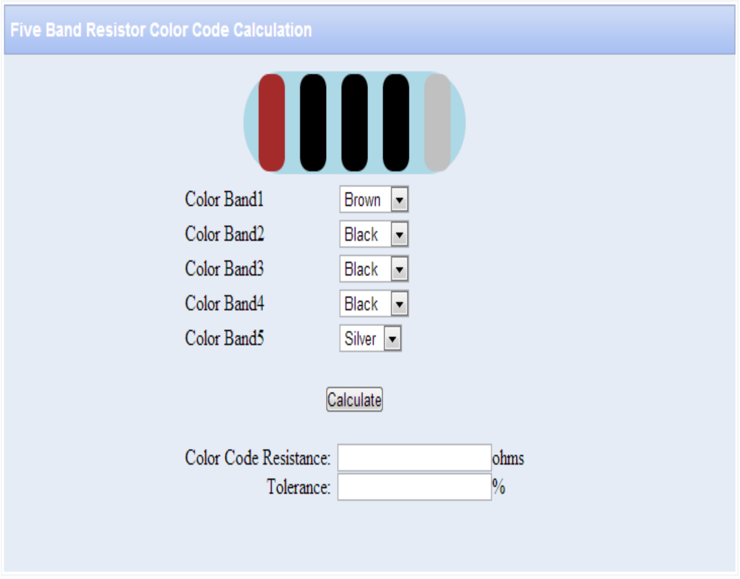
LED cathode (-)
connected to negative (–)
side of battery

Resistors

- An electrical device that resists the flow of current (electrons) though it. Resistance is measured in ohms (Ω). A 500 Ω resistor provides more resistance than a 100 Ω resistor, so less current flows through the device (given the same voltage).
- Note that resistors are not polarized like LEDs. So it doesn't matter which end you use when connecting a resistor.

Resistor Colour Code

- Carbon resistors typically have 4 color bands that indicate their value and tolerance. You can determine the value of resistance and tolerance using the table in Appendix C of Robotics, Version 2.2
- You can also use a handy online Resistor Color Code Calculator shown to the right. It is available at:
- http://easycalculation.com/physics/electromagnetism/five-band-resistor-colorcode.php



Five Band Resistor Color Code Calculation

Color Band1	Brown
Color Band2	Black
Color Band3	Black
Color Band4	Black
Color Band5	Silver

Color Code Resistance: ohms

Tolerance: %

Breadboard

- The Boe-bot comes with a convenient circuit board called a **prototype board** or a **breadboard** where circuits can be easily connected to the BASIC Stamp. See Appendix D in Robotics - Version 2.2 for more information

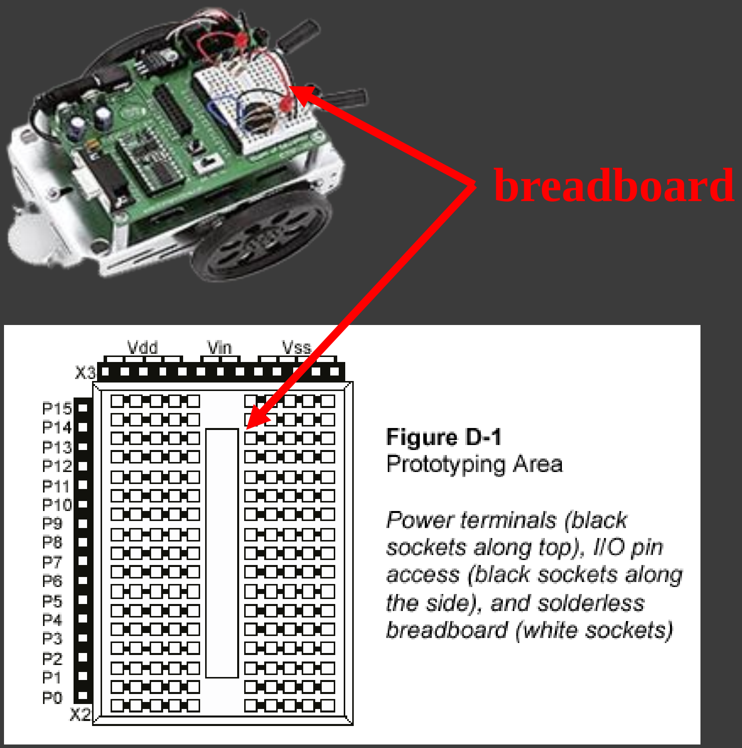


Figure D-1
Prototyping Area

Power terminals (black sockets along top), I/O pin access (black sockets along the side), and solderless breadboard (white sockets)

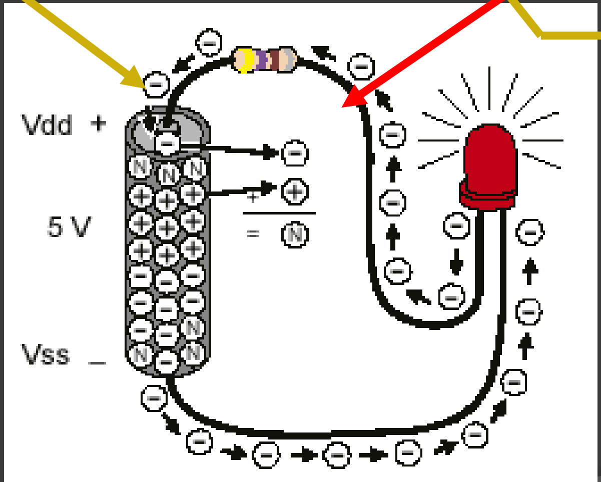
- In addition to giving a handy place to build circuits, the breadboard gives ready access to:
- **P0 – P15**: The 16 input/output pins on the BASIC Stamp
- **Vdd**: The 5V connection on the BASIC Stamp. This is like the positive (+) terminal of a 5V battery.
- **Vss**: The “ground” or 0V connection on the BASIC Stamp. This is like the negative (-) terminal of a 5V battery.

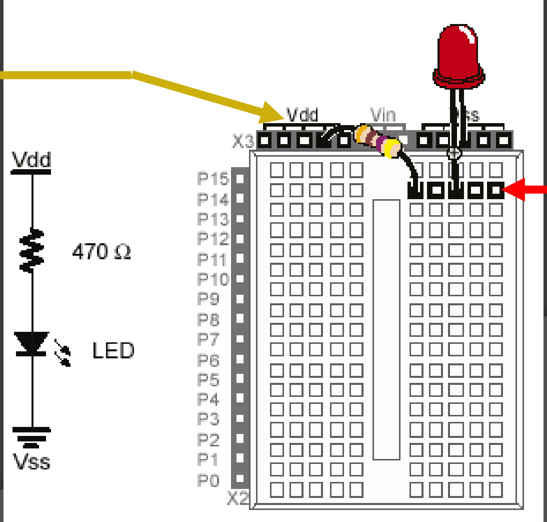
Using a Breadboard

- The rules for using a breadboard are fairly simple. Note that each series of 5 holes shown below is connected by a black line (which is added here just for illustration). These holes are connected together at the bottom of the breadboard. So the basic rule is: ***Connect two items together by plugging them into the same row.***
- **Example** – Illustrate how to connect the LED circuit below on a breadboard.

Resistor connected to positive terminal of 5V battery, so connect it to Vdd.

Resistor connected to LED, so put them in the same row.





Controlling Two LED's with the BASIC Stamp

- Suppose that we want to use two output pins from the BASIC Stamp, P12 and P13, to turn on and off two LEDs. We could wire the circuits on the breadboard as shown below. Be sure that the cathodes (-) of each LED are connected to Vss and not the anodes.

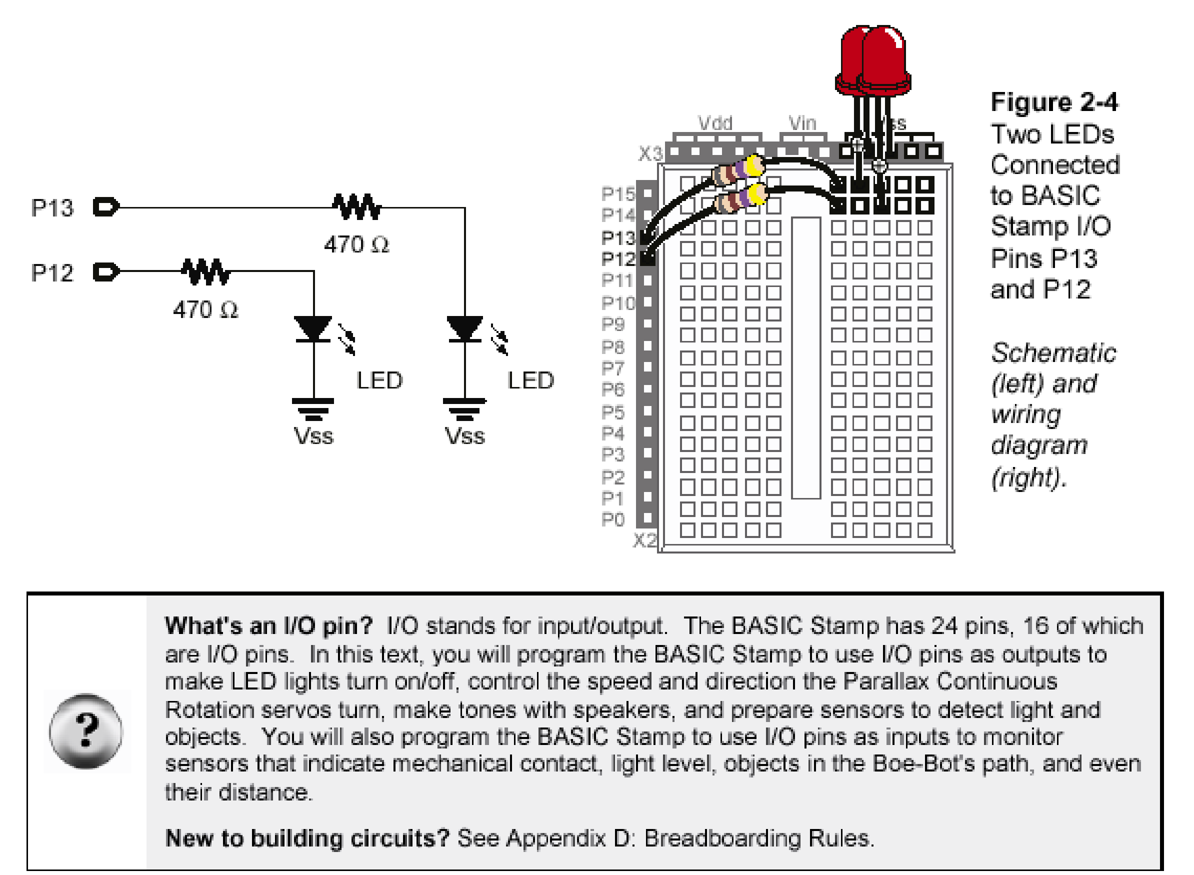


Figure 2-4
Two LEDs Connected to BASIC Stamp I/O Pins P13 and P12

Schematic (left) and wiring diagram (right).

What's an I/O pin? I/O stands for input/output. The BASIC Stamp has 24 pins, 16 of which are I/O pins. In this text, you will program the BASIC Stamp to use I/O pins as outputs to make LED lights turn on/off, control the speed and direction the Parallax Continuous Rotation servos turn, make tones with speakers, and prepare sensors to detect light and objects. You will also program the BASIC Stamp to use I/O pins as inputs to monitor sensors that indicate mechanical contact, light level, objects in the Boe-Bot's path, and even their distance.

New to building circuits? See Appendix D: Breadboarding Rules.

Sample Program 3

- This program is shown on page 50 of Robotics, Version 2.2
- Assume that LEDs and resistors are connected to pins P12 and P13 as shown below. What does the program do? Discuss possible revisions to the program

```
' Robotics with the Boe-Bot - HighLowLed.bs2
' Turn the LED connected to P13 on/off once every second.

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

DEBUG "The LED connected to Pin 13 is blinking!"

DO
  HIGH 13
  PAUSE 500
  LOW 13
  PAUSE 500
LOOP
```

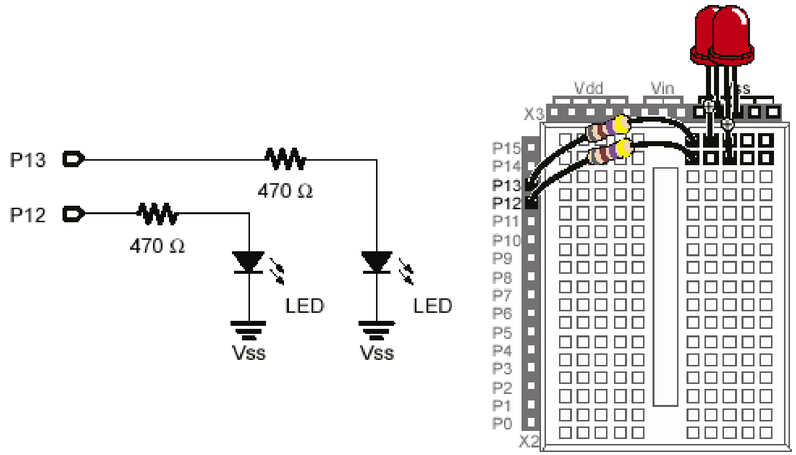
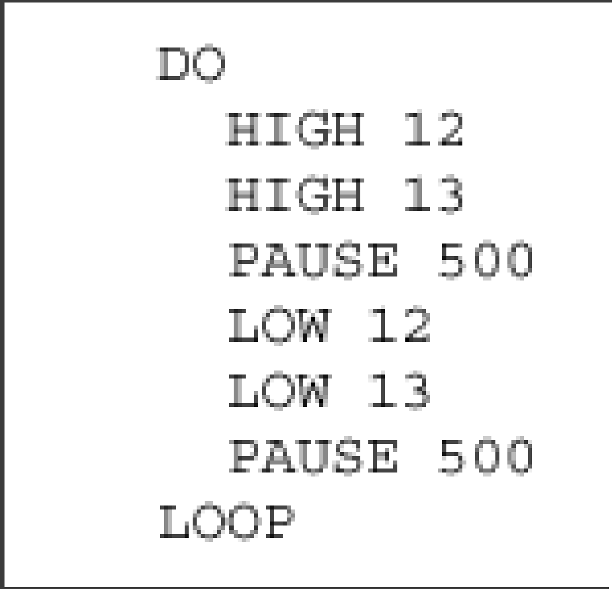


Figure 2-4
Two LEDs Connected to BASIC Stamp I/O Pins P13 and P12

Schematic (left) and wiring diagram (right).

Sample Program 4

- This program is shown on page 53 of Robotics, Version 2.2
- Assume that LEDs and resistors are connected to pins P12 and P13 as shown below. What does the program do? Discuss possible revisions to the program.



```
DO
   HIGH 12
   HIGH 13
   PAUSE 500
   LOW 12
   LOW 13
   PAUSE 500
LOOP
```

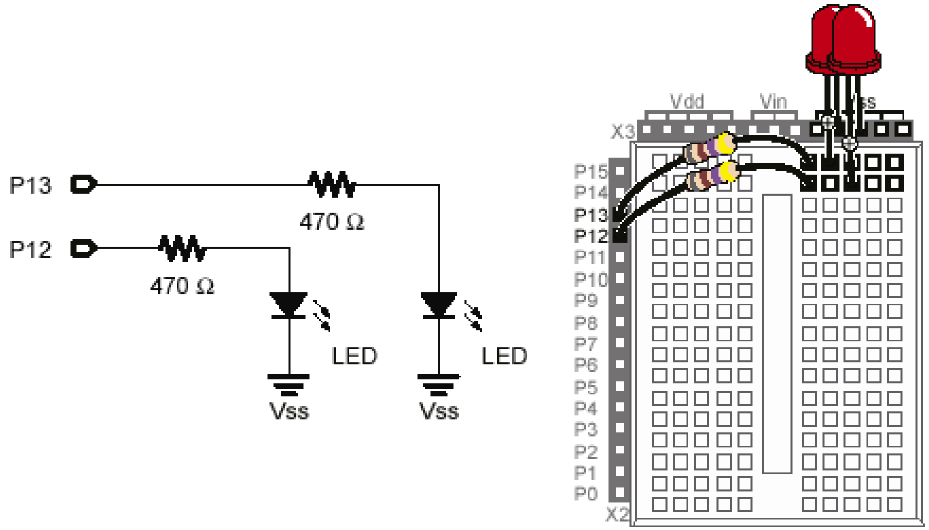


Figure 2-4
Two LEDs Connected to BASIC Stamp I/O Pins P13 and P12

Schematic (left) and wiring diagram (right).

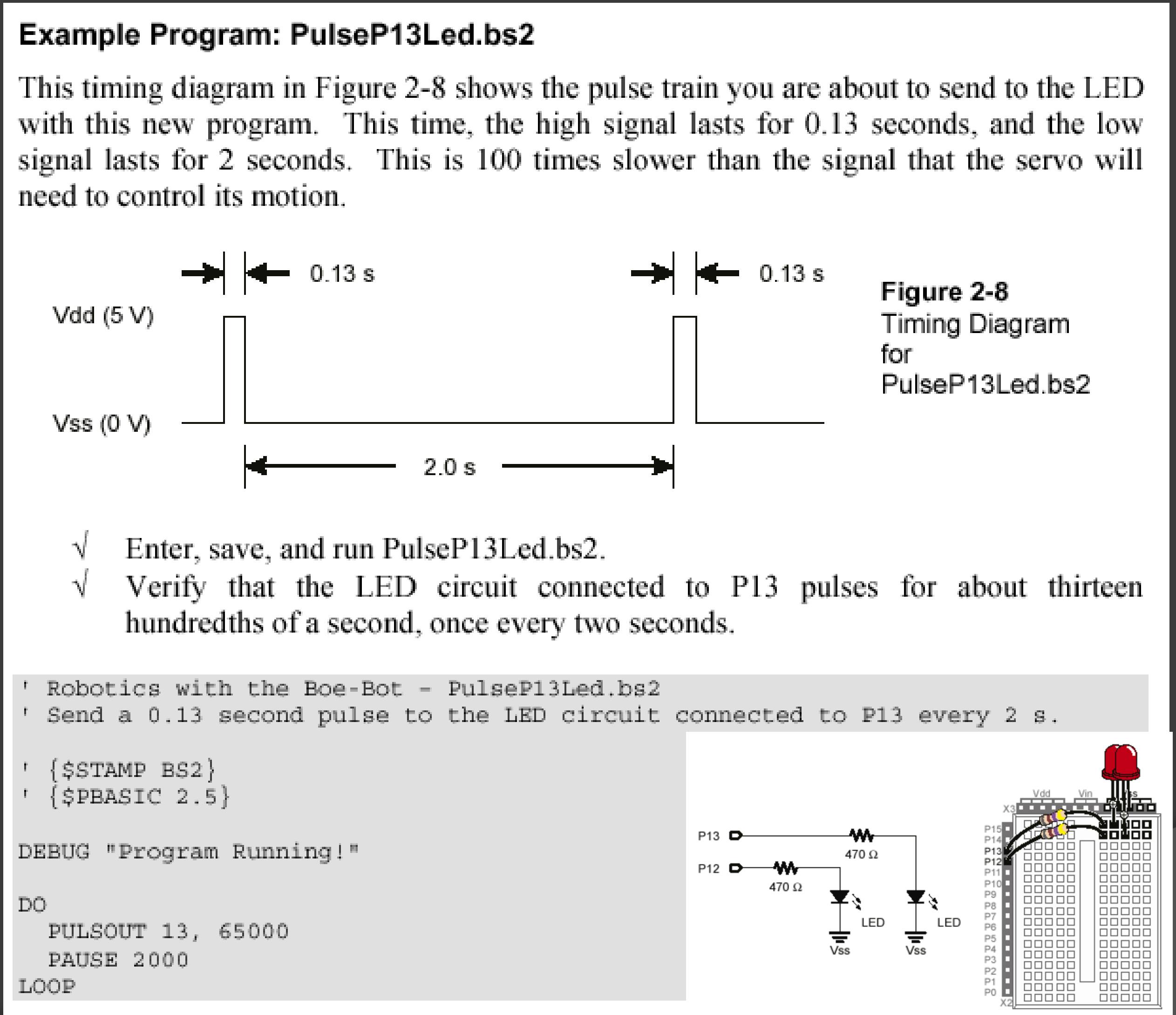
More PBASIC Commands

- The PULSOUT command described below is very important for controlling servos (which we will cover later), but it can also be used for turning LEDs on and off.
- PULSOUT
  - PULSOUT Pin Duration – this command is used to set the specified Pin HIGH for a time equal to Duration multiplied by 2 μs (2 millionths of a second, or 2 microseconds). The max value for duration is 65535 (or for 65535x2 μs = 0.13107 seconds)
- Suppose that an LED (and resistor) is connected to output pin P4:
  - PULSOUT 4 8 – the LED is lit for 8x2 μs = 16 μs
  - PULSOUT 4 100 – the LED is lit for 100x2 μs = 0.2 ms (100 * 0.000002s = 0.0002)
  - PULSOUT 4 5000 – the LED is lit for 5000x2 μs = 10 ms (5000 * 0.000002s = 0.01)
- Note that the following two programs will accomplish the same thing:

```
DO
   HIGH 13
   PAUSE 50
   LOW 13
   PAUSE 50
LOOP
```

```
DO
   PULSOUT 13 25000        'P13 is set HIGH for 25000x2 μs = 50 ms
   PAUSE 50                 'Pause for 50 ms
LOOP
```

Sample Program 5



Example Program: PulseP13Led.bs2

This timing diagram in Figure 2-8 shows the pulse train you are about to send to the LED with this new program. This time, the high signal lasts for 0.13 seconds, and the low signal lasts for 2 seconds. This is 100 times slower than the signal that the servo will need to control its motion.

0.13 s
0.13 s
Vdd (5 V)
Vss (0 V)
2.0 s

Figure 2-8
Timing Diagram for PulseP13Led.bs2

- √ Enter, save, and run PulseP13Led.bs2.
- √ Verify that the LED circuit connected to P13 pulses for about thirteen hundredths of a second, once every two seconds.

```
' Robotics with the Boe-Bot - PulseP13Led.bs2
' Send a 0.13 second pulse to the LED circuit connected to P13 every 2 s.

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

DEBUG "Program Running!"

DO
  PULSOUT 13, 65000
  PAUSE 2000
LOOP
```

Review

- Robots, types, and choices
- Boe-bot types and make-up
- Microprocessors and controllers
- The BASIC Stamp
- Robot to PC Interfacing
- Programming PBASIC
- Basic Commands
- Breadboard Connections
- Sample Programs
- Electronic Components
- Circuit Wiring