**Lesson 5: Figure 8**

Needed

* Boe-Bot
* Computer with BASIC Stamp Editor program
* USB cable

Today we are going to expand on our programming of Boe-Bot motion. We will introduce into our programming skills the technique of writing subroutines.

From the last class, you should have the code written for full speed forward, full speed reverse, sharp right turn, and sharp left turn. In future programs, you may want to perform the same functions. For instance, what if you had a program that was supposed to follow this flow chart:

This program requires multiple instances of full speed forward as well as full speed reverse. You could write the program to have the code for each motion of the Boe-Bot to be written each time it appears or you can use what is called a *subroutine*. Basically, if there is a code written already for a function in a program you do not have to retype it every time it appears. You can simply call on it using a GOSUB command.

**Initialize Variables**

(Defining the size of the variables used in the program)

**Go Forward Full speed for a duration**

**Go Forward Full speed for a duration**

**Turn Sharply to the Right for a duration**

**Turn Sharply to the Left for a duration**

**Go in Reverse Full speed for a duration**

**Go in Reverse Full speed for a duration**

How do you use a GOSUB?

Well you give your prewritten code a name (i.e. for full speed forward, maybe its name should be fullForward). Once you have the name for the code, instead of writing that code into your program, you would type GOSUB then the name of the code that you want to run (for forward it would be GOSUB fullForward). Then you would type the rest of your code as you would like. You can put as many GOSUB commands as you want. Now whenever you need the code you can simply call on it. This is especially helpful when something appears multiple times in a program. After you type out your full program with all the needed GOSUB commands below your program, you can paste the code that the GOSUB is calling for (for full speed forward you would type fullForward: then all the code associated with full speed forward). Note that you put a colon after the name of the code. Two things you should always remember when using the GOSUB command are:

1. Put a RETURN statement on the last line of the subroutine. This tells the Boe-Bot to go back to the next line in the main program.
2. Put an END statement after your main program and before the first subroutine, so that the Boe-Bot doesn’t plow on into the first subroutine after it finishes executing the main program.

Try it out! You already have the code for full speed forward, full speed reverse, sharp right turn, and sharp left turn. Give each of these a name and call on them accordingly as the flow chart describes.

Solution:

\*PROGRAM SOLUTION NEEDED\*

Activity – Figure 8

This activity will require your Boe-Bot to perform a figure 8 motion around two objects set a certain distance apart.

22in

Starting Location

Setup: Take two markers (this could be cups, books, keys, etc.) and place them approximately 22inches apart (the length of two sheets of paper). This will form the course that the figure 8 will follow. Have the Boe-Bot start at the bottom marker (indicated in the drawing).

Objective: Make the Boe-Bot form a figure 8 around the markers and return to the starting location.

Suggestion: Have the students make a flow chart of the program needed to form the figure 8 prior to writing the program.

Solution: