

Excerpts Adapted from BASIC Stamp Editor Help v2.5; Getting Started with Stamps in Class



Getting Started with Stamps in Class

This is the place to begin if you are setting up your BASIC Stamp 2 and Board of Education or BASIC Stamp HomeWork Board to prepare for the activities in any of our Stamps in Class tutorials, including *What's a Microcontroller?* and *Robotics with the Boe-Bot*.

The activities in this document, which were removed from the Help file for translation, will guide you through the following tasks:

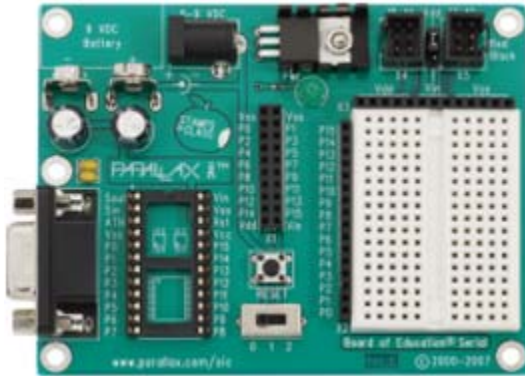
- Identify what BASIC Stamp development board you are using.
- Connect your BASIC Stamp development board to your computer.
- Test the programming connection.
- Write and test your first PBASIC program for your BASIC Stamp.
- Write a PBASIC program that uses ASCII code.
- Turn off the power to your BASIC Stamp development board when you are done.

You will navigate from one task to the next using the arrows and the page numbers shown. On each page, follow the instructions indicated by checkmarks:

► Go to Identify Your Board on page 2 to begin!

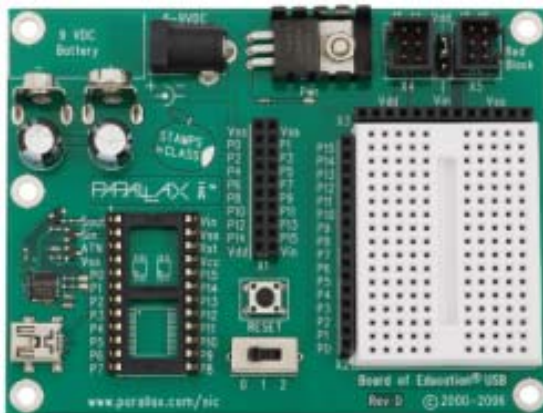
Identify Your Board

✓ Let's begin by identifying the type of board you have.



Board of Education - Serial

► Go to Board of Education - Serial on page 3.



Board of Education - USB

► Go to Board of Education - USB on page 12.

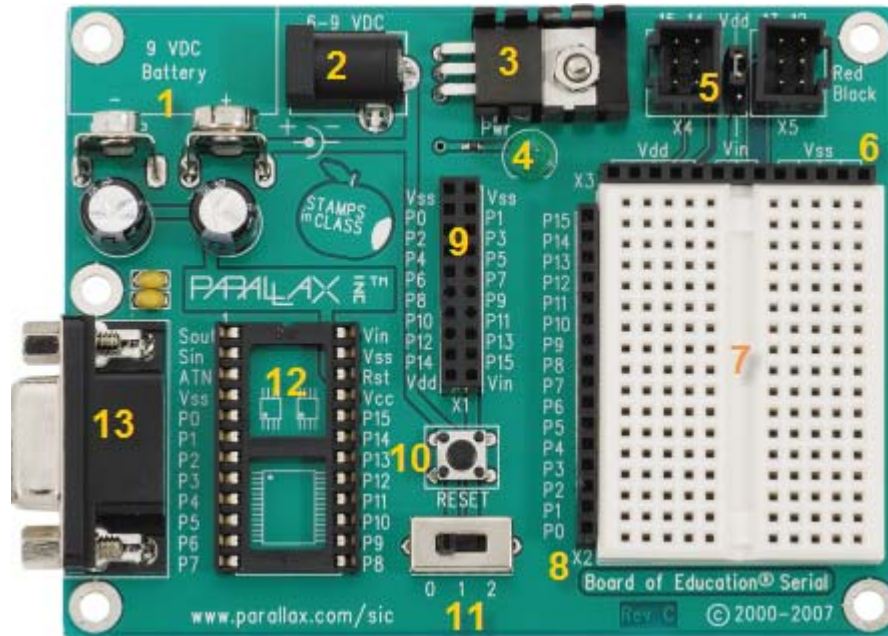


BASIC Stamp HomeWork Board

► Go to HomeWork Board on page 16.

Board of Education - Serial

- ✓ Take a look at your board to get familiar with its parts.



1. **9V Battery Clip:** You can use alkaline or rechargeable 9 volt batteries. The battery clip and barrel jack are intentionally positioned so you cannot use both at once.
2. **Barrel Jack:** This accepts a 2.1 mm center-positive barrel plug from a 6-9 V wall-mount supply or from a battery pack. You cannot use the barrel jack and a 9 volt battery at the same time.
3. **Voltage regulator:** Supplies regulated 5 V (up to 1 amp of current) for sockets and pins labeled Vdd. Vdd sockets are convenient for supplying 5 V to circuits you will build on the breadboard area.
4. **Power Indicator LED:** This LED will light up when power is supplied to your board and the power switch is in position 1 or 2.
5. **Servo headers (X4 and X5) and Power Select Jumper:** These each have two 3-pin connectors that bring power, ground, and I/O pin access together so you can easily plug in servos or other 3-pin devices. The power connection is pre-set to Vdd (+5 V) but you can set it to Vin (the board's supply voltage) by moving the shorting block on the jumper between the headers. Each 3-pin row is labeled with an I/O pin number above it. The 12, 13, 14, and 15 signal lines for the servo headers are also accessible as P12, P13, P14, and P15 I/O pin sockets on the X1 and X2 headers. This can be useful for building a servo signal indicator light on the breadboard as you may do in some Stamps in Class activities. For independent projects, keep these shared connections in mind, especially to avoid inadvertently connecting circuits with conflicting functions to the same I/O pin.

6. **Power header (X3):** The sockets labeled Vdd connect to +5 VDC, Vin connects directly to the power supplied to the board by the battery clip or barrel jack, and Vss connects to 0 V (ground).
7. **Breadboard:** The breadboard has metal clips that run underneath the white plastic board in a horizontal fashion. Each strip connects a 5-socket group, with two groups to each row, separated by a center trench. Wires or legs of components plugged into the same 5-socket group will be electrically connected. Components with many legs (such as pushbuttons or ICs), are placed in the middle of the board so that half of the legs are on the left side and half are on the right side of the trench. Note: Always disconnect power before building or modifying circuits!
8. **I/O Pin Access Header (X2):** The BASIC Stamp module's 16 I/O pins, labeled 0 to 15, are connected to this header. Its location adjacent to the breadboard makes it convenient for connecting circuits to I/O pins. Keep in mind that I/O pin access is also brought to the X4, X5, and X1 headers, so be careful not to build conflicting breadboard circuits if you are using these other headers as well.
9. **AppMod header (X1):** The AppMod header provides power, I/O pins, Vdd, Vin, and Vss access for any devices that are designed to use this 2x10 socket. Examples include the LCD Terminal AppMod (#29121), CMUcam (#30051), Easy Bluetooth Module (#30085), and Say It voice recognition module (#30080).
10. **Reset Button:** The reset button can be used to restart your BASIC Stamp without having to cycle the power. This saves wear-and-tear on the power switch for simple program restarts. Some advanced programming techniques use the reset button and the BASIC Stamp EEPROM program and data storage as a way to toggle between different program functions.
11. **3-Position Power Switch:** The leftmost position (0) is OFF – all power is disconnected. Always place the switch in this position when adding or changing components on the breadboard. The middle position (1) provides Vin (unregulated battery or power supply voltage) to the regulator, the BASIC Stamp socket, and to the connectors marked “Vin.” This switch position also makes Vdd (5 volts) available to Vdd sockets on the breadboard and AppMod connectors. The rightmost position (2) also provides power to the servo connectors X4 and X5. Especially if your program causes a robot with servos connected to X4/X5 to start moving immediately, you can keep the 3-position switch in position (1) while loading the program, then switch to position (2) when you are ready for the robot to start moving.
12. **Socket for BASIC Stamp:** This socket is compatible with all 24-pin BASIC Stamp modules. It connects the BASIC Stamp to the programming connector, power, the power indicator LED, reset button, and all I/O pin headers.
13. **Serial Programming Connector:** This is a female DB9 socket for programming and for two-way serial communication between the BASIC Stamp and your computer. You can connect directly to a serial port on your computer, or use a USB to Serial Adapter to connect to a USB port.

Important! Check the Revision Letter of your Board of Education

It'll either be below the breadboard or in the upper-right corner, and might be Rev A, Rev B, Rev C, etc... Make a note of this revision because certain Stamps in Class activities will utilize different circuits, depending on which revision you use.

- ▶ If you have a Board of Education - Serial Rev C, shown above, you don't need any extra instructions. Go ahead to Using a USB to Serial Adapter with your Board of Education? on page 7.
- ▶ If you have a Board of Education - Serial Rev A, the label will be in the upper right corner of the board. Read the
- ▶
- ▶ Board of Education - Serial Rev A Special Instructions on page 5 before continuing.
- ▶ If you have a Board of Education - Serial Rev B, the label will be in the upper right corner on top of the board, just below and to the right of the X5 servo header. Read the Board of Education - Serial Rev B Special Instructions on page 6 before continuing.

Board of Education - Serial Rev A Special Instructions

Servo Ports - There Are None!

The Board of Education Rev A does not have servo ports. However, 3-pin connections for servos and other devices can be built on the breadboard. For an example of connecting a standard servo to the Board of Education Rev A, see the instructions in the Servo Circuit Connections for Older Boards pdf at www.parallax.com/go/WAM. If you wish to supply a 3-pin device, such as a Parallax Continuous Rotation Servo, with a 6-volt power supply via a breadboard circuit, Parallax carries 4 AA cell battery holder with tinned leads (#753-00001) for this purpose.

3-Position Switch - There Isn't One!

To disable power on the Board of Education Rev A, you must disconnect the power supply or battery at its connector.

- ▶ Go ahead to Using a USB to Serial Adapter with your Board of Education? on page 7.

Board of Education - Serial Rev B Special Instructions

Servo Ports - There is no Servo Power Selection Jumper!

The Board of Education Rev B has servo headers (labeled X4 and X5), but it does not have a Servo Power Selection Jumper. On this board, the X4 and X5 headers are permanently connected to Vin. When connecting servos or other devices to the X4 and X5 headers, be sure to use a power supply that does not exceed the voltage tolerance of the servo or device. For example, the Parallax Continuous Rotation Servo requires a 6 V (maximum) supply, so it should not be plugged into the servo headers if using a 9 V battery or a 7.5 V or 9 V wall supply, or the life of the servo could be shortened. A 4 AA cell battery pack with a compatible barrel jack connector is available from Parallax (#700-00038).

3-Position Switch - There Isn't One!

To disable power on the Board of Education Rev B, you must disconnect the power supply or battery at its connector.

Vdd on AppMod Header - Limited Current Budget

Most of the connections labeled Vdd are supplied by the Board of Education Rev B's 5 V, 1 A voltage regulator. However, the socket labeled Vdd on the X1 AppMod header is instead supplied by the small voltage regulator on the BASIC Stamp module. The BASIC Stamp module's voltage regulator only has a 50 mA current budget for its processor, certain circuits built on the breadboard and connected to I/O pins, and any devices drawing current from its Vdd pin. The socket labeled Vdd on the X1 AppMod header draws current from the BASIC Stamp module's Vdd pin, so an AppMod or other device that draws current from this pin has to stay within the 50 mA current budget, which in many applications may turn out to be only a few milliamps. Before using an AppMod with the Board of Education Rev B, verify that it gets its power from the AppMod Header's Vin socket, and not Vdd.

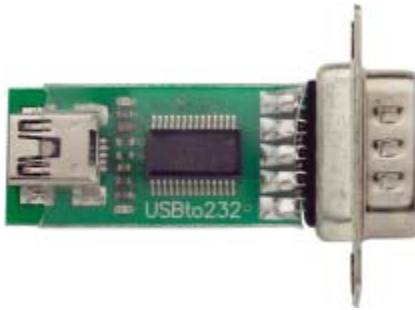
► Go ahead to Using a USB to Serial Adapter with your Board of Education? on page 7.

Using a USB to Serial Adapter with your Board of Education?

Will you be using a USB port on your computer, and a USB to Serial Adapter, with your Board of Education?

YES!

- ▶ Go to Board of Education with USB to Serial Adapter - Setup on page 8



NO!

- ▶ Go to Board of Education Serial - Setup on page 9.

Board of Education with USB to Serial Adapter - Setup

- ✓ DO NOT connect power to your board until instructed to do so!

Affixing the Rubber Feet

Your board should come with four adhesive rubber feet:



- ✓ Remove each rubber foot from its adhesive, and affix it to the underside of your board. There are circles to show correct placement.

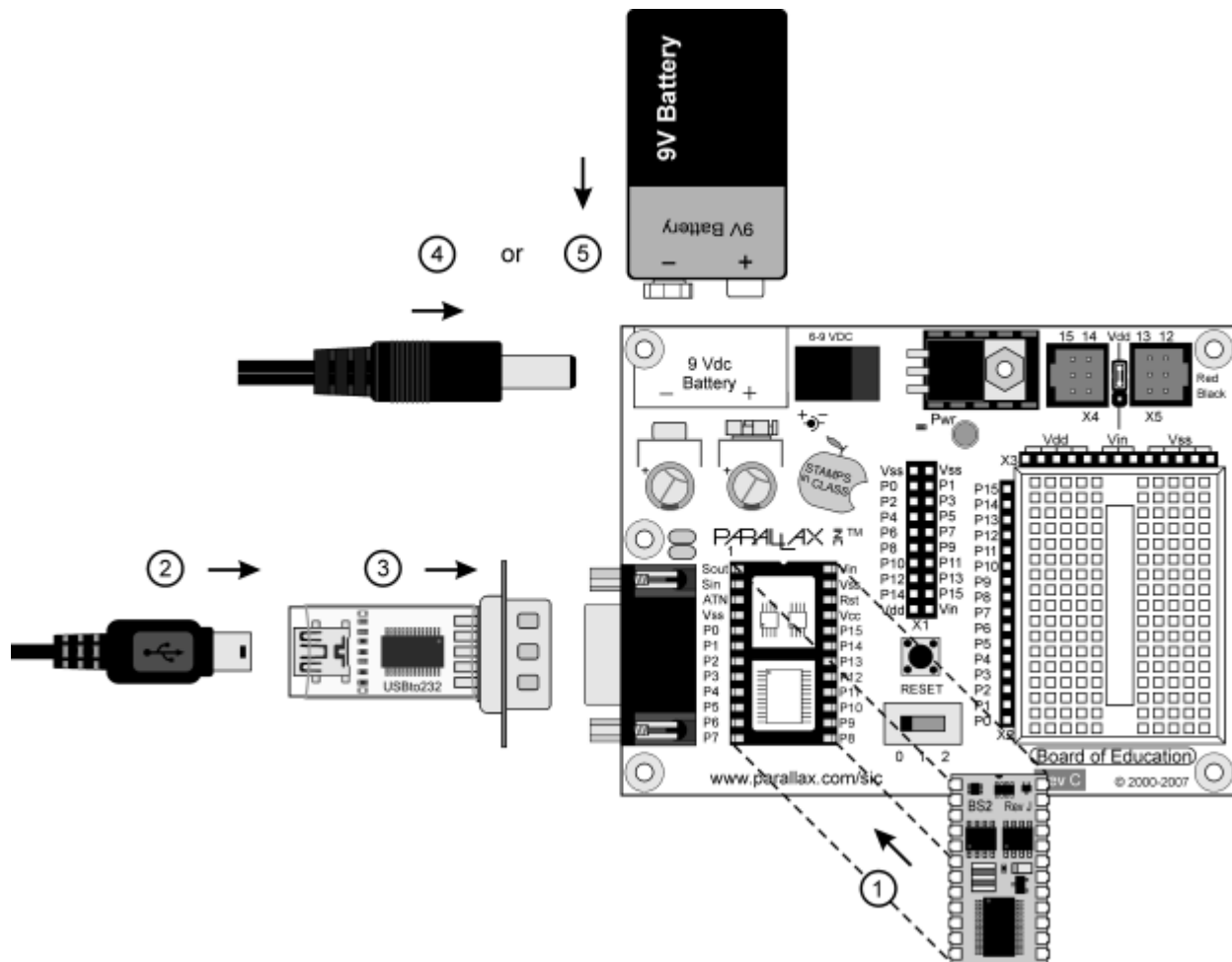


Making the Board Connections

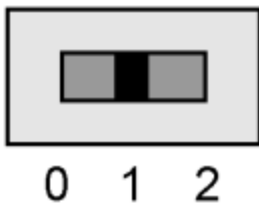
- ✓ Use the silkscreen pictures that show the three prominent chips on the BASIC Stamp to determine how to orient it before inserting it into its socket. There is also a small semi-circular notch at in the top-center of the BASIC Stamp module that indicates which way is “up.” Match this reference notch to the notch in the socket.
- ✓ After correctly orienting the 24-pin BASIC Stamp module, gently insert it into its socket as shown at (1) in the picture below.
- ✓ Make sure that each pin is seated in its corresponding socket hole, then apply firm downward pressure with your thumb. The module should sink about $\frac{1}{4}$ of an inch (~6 mm) into the socket, and only the flared portion of each BASIC Stamp leg should be visible above the socket.
- ✓ Visually inspect the legs to make sure each pin is in its socket and that no pins are folded underneath the module.
- ✓ Connect the larger "A" end of the USB cable to a USB port on your computer.
- ✓ Connect the smaller "Mini B" end of the USB cable to the USB connector on the USB to Serial Adapter, as shown in (2).
- ✓ Connect the male DB9 connector on the USB to Serial Adapter to the female DB9 socket on the board (3).
- ✓ Connect one of the two power sources (see Power Supplies on page 41).
 - a 6 to 9 V wall-mount power supply with a 2.1 mm center positive barrel jack (4);

- OR -

- a 9 V alkaline or rechargeable battery (5).



- ✓ Set the 3 - position switch to Position 1 to turn on power to your board.



You are ready to test your programming connection!

► Go to Testing the Connection on page 23.

Board of Education Serial - Setup

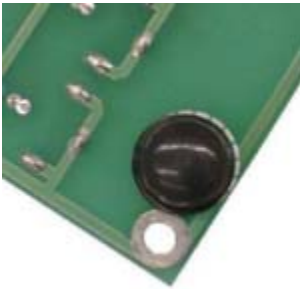
- ✓ DO NOT connect power to your board until instructed to do so!

Affixing the Rubber Feet

Your board should come with four adhesive rubber feet:

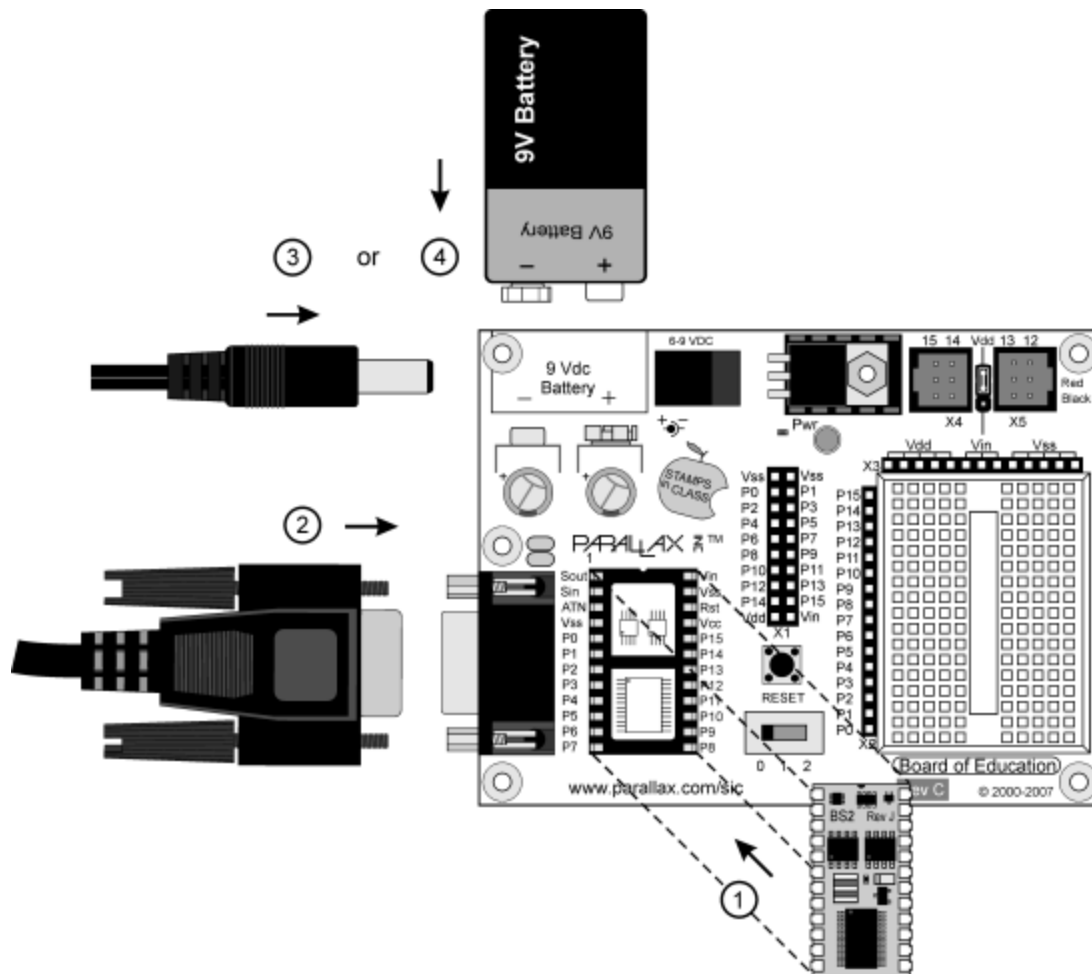


- ✓ Remove each rubber foot from its adhesive, and affix it to the underside of your board. There are circles to show correct placement.

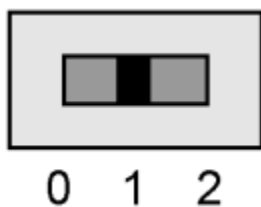


Making the Board Connections

- ✓ Use the silkscreen pictures that show the three prominent chips on the BASIC Stamp to determine how to orient it before inserting it into its socket. There is also a small semi-circular notch at in the top-center of the BASIC Stamp module that indicates which way is “up.” Match this reference notch to the notch in the socket.
- ✓ After correctly orienting the 24-pin BASIC Stamp module, gently insert it into its socket as shown at (1) in the picture below.
- ✓ Make sure that each pin is seated in its corresponding socket hole, then apply firm downward pressure with your thumb. The module should sink about $\frac{1}{4}$ of an inch (~6 mm) into the socket, and only the flared portion of each BASIC Stamp leg should be visible above the socket.
- ✓ Visually inspect the legs to make sure each pin is in its socket and that no pins are folded underneath the module.
- ✓ Connect the female end of the serial cable to the serial port on your computer (it is a 9-pin male port).
- ✓ Connect the male end of the serial cable to the DB9 connector on the board, as shown in (2).
- ✓ Connect one of the two power sources (see Power Supplies on page 41).
 - 6 to 9 V wall-mount power supply with a 2.1 mm center positive barrel jack (3);
 - OR -
 - a 9 V alkaline or rechargeable battery (4).



- ✓ Set the 3 - position switch to Position 1 to turn on power to your board.

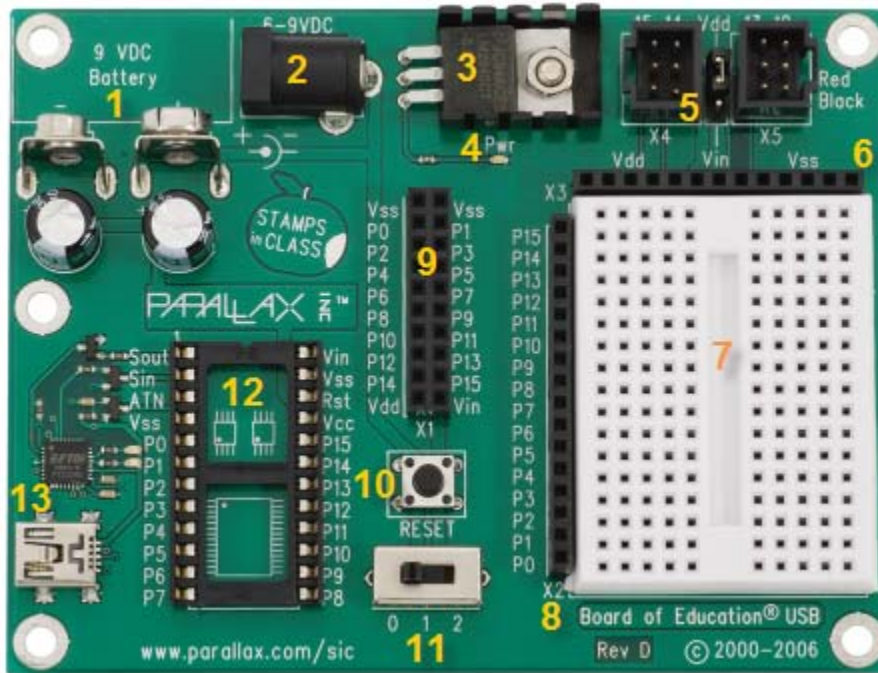


You are ready to test your programming connection!

► Go to Testing the Connection on page 23.

Board of Education - USB

- ✓ Take a look at your board to get familiar with its parts.



1. **9V Battery Clip:** You can use alkaline or rechargeable 9 volt batteries. The battery clip and barrel jack are intentionally positioned so you cannot use both at once.
2. **Barrel Jack:** This accepts a 2.1 mm center-positive barrel plug from a 6-9 V wall-mount supply or from a battery pack. You cannot use the barrel jack and a 9 volt battery at the same time.
3. **Voltage regulator:** Supplies regulated 5 V (up to 1 amp of current) for sockets and pins labeled Vdd. Vdd sockets are convenient for supplying 5 V to circuits you will build on the breadboard area.
4. **Power Indicator LED:** This LED will light up when power is supplied to your board and the power switch is in position 1 or 2.
5. **Servo headers (X4 and X5) and Power Select Jumper:** These each have two 3-pin connectors that bring power, ground, and I/O pin access together so you can easily plug in servos or other 3-pin devices. The power connection is pre-set to Vdd (+5 V) but you can set it to Vin (the board's supply voltage) by moving the shorting block on the jumper between the headers. Each 3-pin row is labeled with an I/O pin number above it. The 12, 13, 14, and 15 signal lines for the servo headers are also accessible as P12, P13, P14, and P15 I/O pin sockets on the X1 and X2 headers. This can be useful for building a servo signal indicator light on the breadboard as you may do in some Stamps in Class activities. For independent projects, keep these shared connections in mind, especially to avoid inadvertently connecting circuits with conflicting functions to the same I/O pin.

6. **Power header (X3):** The sockets labeled Vdd connect to +5 VDC, Vin connects directly to the power supplied to the board by the battery clip or barrel jack, and Vss connects to 0 V (ground).
7. **Breadboard:** The breadboard has metal clips that run underneath the white plastic board in a horizontal fashion. Each strip connects a 5-socket group, with two groups to each row, separated by a center trench. Wires or legs of components plugged into the same 5-socket group will be electrically connected. Components with many legs (such as pushbuttons or ICs), are placed in the middle of the board so that half of the legs are on the left side and half are on the right side of the trench. Note: Always disconnect power before building or modifying circuits!
8. **I/O Pin Access Header (X2):** The BASIC Stamp module's 16 I/O pins, labeled 0 to 15, are connected to this header. Its location adjacent to the breadboard makes it convenient for connecting circuits to I/O pins. Keep in mind that I/O pin access is also brought to the X4, X5, and X1 headers, so be careful not to build conflicting breadboard circuits if you are using these other headers as well.
9. **AppMod header (X1):** The AppMod header provides power, I/O pins, Vdd, Vin, and Vss access for any devices that are designed to use this 2x10 socket. Examples include the LCD Terminal AppMod (#29121), CMUcam (#30051), Easy Bluetooth Module (#30085), and Say It voice recognition module (#30080).
10. **Reset Button:** The reset button can be used to restart your BASIC Stamp without having to cycle the power. This saves wear-and-tear on the power switch for simple program restarts. Some advanced programming techniques use the reset button and the BASIC Stamp EEPROM program and data storage as a way to toggle between different program functions.
11. **3-Position Power Switch:** The leftmost position (0) is OFF – all power is disconnected. Always place the switch in this position when adding or changing components on the breadboard. The middle position (1) provides Vin (unregulated battery or power supply voltage) to the regulator, the BASIC Stamp socket, and to the connectors marked “Vin.” This switch position also makes Vdd (5 volts) available to Vdd sockets on the breadboard and AppMod connectors. The rightmost position (2) also provides power to the servo connectors X4 and X5. Especially if your program causes a robot with servos connected to X4/X5 to start moving immediately, you can keep the 3-position switch in position (1) while loading the program, then switch to position (2) when you are ready for the robot to start moving.
12. **Socket for BASIC Stamp:** This socket is compatible with all 24-pin BASIC Stamp modules. It connects the BASIC Stamp to the programming connector, power, the power indicator LED, reset button, and all I/O pin headers.
13. **USB Programming Connector:** This is a USB Mini B socket and USB to serial (RS232) circuitry for programming and for two-way serial communication between the BASIC Stamp and your computer. The required USB drivers for Windows were included in the BASIC Stamp Editor Software installer; see the www.parallax.com/basicstampsoftware page for more information.

► Go to Board of Education USB - Setup on page 14.

Board of Education USB - Setup

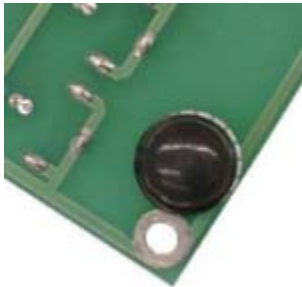
- ✓ DO NOT connect power to your board until instructed to do so!

Affixing the Rubber Feet

Your board should come with four adhesive rubber feet:

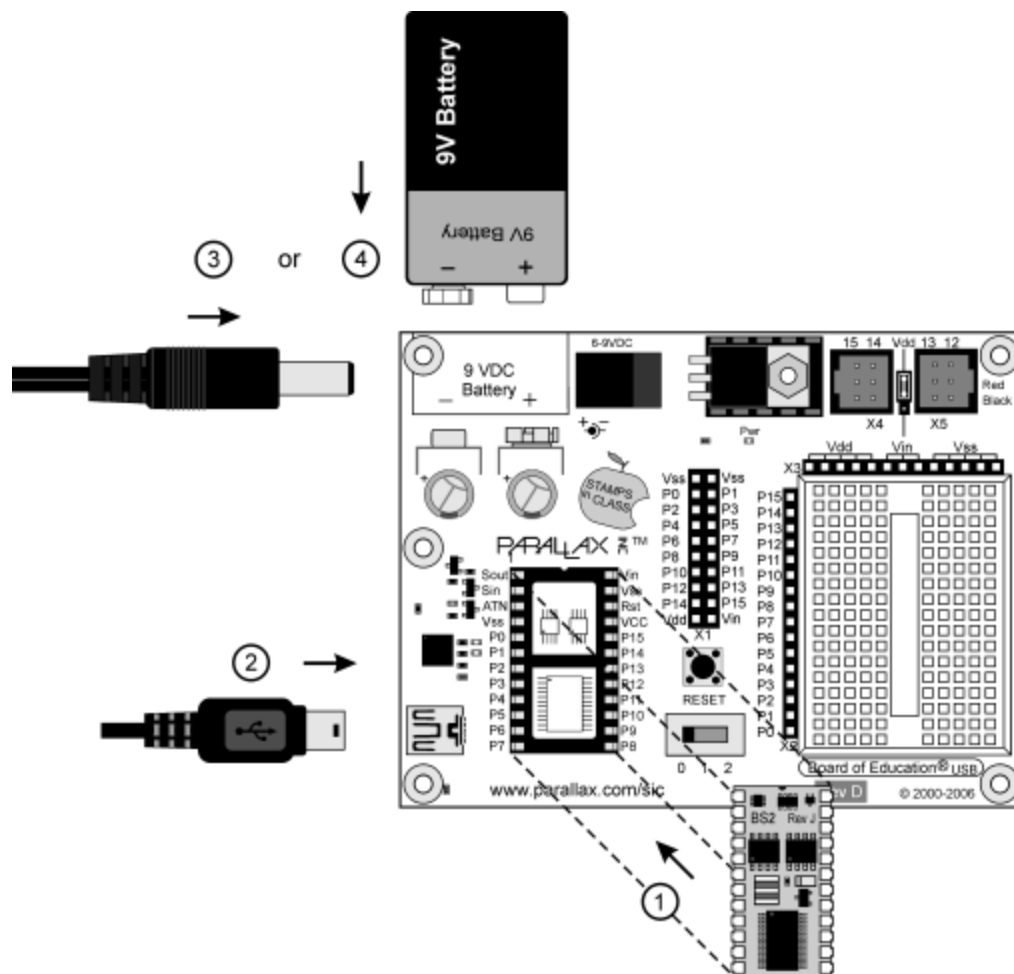


- ✓ Remove each rubber foot from its adhesive, and affix it to the underside of your board. There are circles to show correct placement.

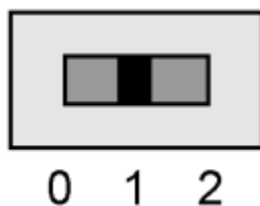


Making the Board Connections

- ✓ Use the silkscreen pictures that show the three prominent chips on the BASIC Stamp to determine how to orient it before inserting it into its socket. There is also a small semi-circular notch at in the top-center of the BASIC Stamp module that indicates which way is “up.” Match this reference notch to the notch in the socket.
- ✓ After correctly orienting the 24-pin BASIC Stamp module, gently insert it into its socket as shown at (1) in the picture below.
- ✓ Make sure that each pin is seated in its corresponding socket hole, then apply firm downward pressure with your thumb. The module should sink about $\frac{1}{4}$ of an inch (~6 mm) into the socket, and only the flared portion of each BASIC Stamp leg should be visible above the socket.
- ✓ Visually inspect the legs to make sure each pin is in its socket and that no pins are folded underneath the module.
- ✓ Connect the larger "A" end of the USB cable to a USB port on your computer.
- ✓ Connect the smaller "Mini B" end of the USB cable to the USB connector on the board, as shown in (2).
- ✓ Connect one of the two power sources (see Power Supplies on page 41).
 - 6 to 9 V wall-mount power supply with a 2.1 mm center positive barrel jack (3);
 - OR -
 - a 9 V alkaline or rechargeable battery (4).



- ✓ Set the 3 - position switch to Position 1 to turn on power to your board.

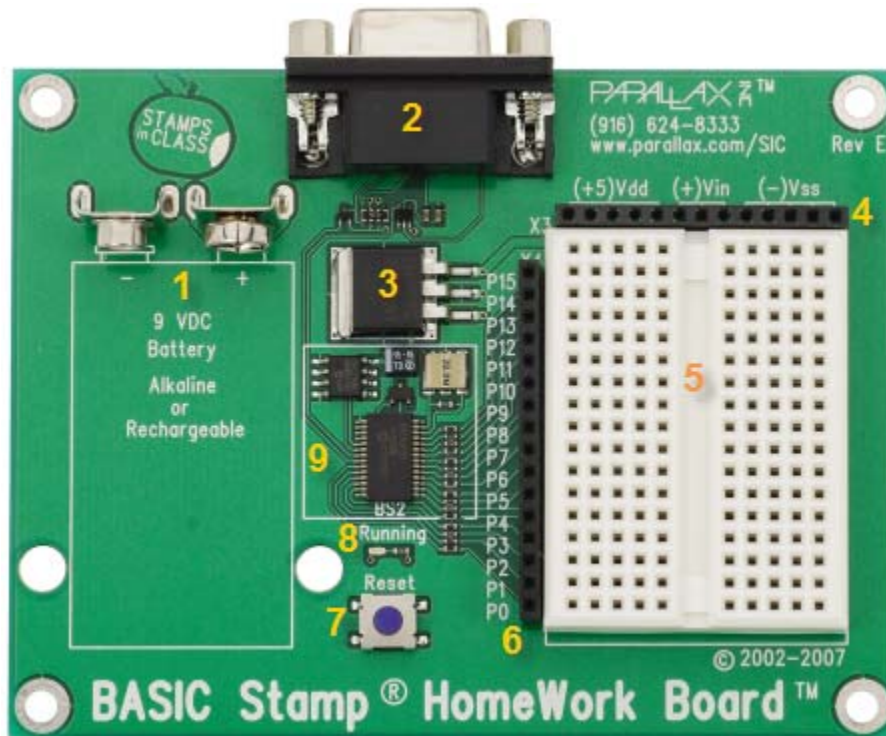


You are ready to test your programming connection!

► Go to Testing the Connection on page 23.

HomeWork Board

- ✓ Take a look at your board to get familiar with its parts.



1. **9V Battery Clip:** You can use alkaline or rechargeable 9-volt batteries. Disconnect the battery to turn off power to your board
2. **Serial Programming Connector:** This is a female DB9 socket for programming and for two-way serial communication between the BASIC Stamp and your computer. You can connect directly to a serial port on your computer, or use a USB to Serial Adapter to connect to a USB port.
3. **Voltage regulator:** Supplies regulated 5 V (up to 500 milliamps) to the BASIC Stamp and sockets labeled Vdd for circuits you will build on the breadboard area.
4. **Power header (X3):** The sockets labeled (+5)Vdd connect to +5 VDC, (+)Vin connects directly to the power supplied to the board by the battery clip (+9 VDC typical), and (-)Vss connects to 0 V (ground).
5. **Breadboard:** The breadboard has metal clips that run underneath the white plastic board in a horizontal fashion. Each strip connects a 5-socket group, with two groups to each row, separated by a center trench. Wires or legs of components plugged into the same 5-socket group will be electrically connected. Components with many legs (such as pushbuttons or ICs), are placed in the middle of the board so that half of the legs are on the left side and half are on the right side of the trench. Note: Always disconnect power before building or modifying circuits!

6. **I/O Pin Access Header (X4):** The BASIC Stamp module's 16 I/O pins, labeled 0 to 15, are connected to this header, so you can conveniently connect to your breadboard circuits. On this board, there are 220-ohm resistors placed between the header and the BASIC Stamp I/O pins to help prevent damage in case of a wiring mistake.
7. **Reset Button:** The reset button can be used to restart your BASIC Stamp without having to cycle the power. This saves wear-and-tear on the battery clip for simple program restarts. Some advanced programming techniques use the reset button and the BASIC Stamp EEPROM as a way to toggle between different program functions.
8. **Running Indicator LED:** This LED will light up when a BASIC Stamp program is running. It is not a power indicator LED.
9. **BASIC Stamp 2:** The components of a BASIC Stamp 2 module are built directly onto the board. It connects the BASIC Stamp to the programming connector, power, the running indicator LED, reset button, and I/O pin header.

Important! Check the Revision Letter of your HomeWork Board!

✓ Look for the revision letter on your Board of Education:

- ▶ If you have a HomeWork Board - Rev C, D, or E, you don't need any extra instructions. Go ahead to Using a USB to Serial Adapter with your HomeWork Board? on page 18.

If you have a HomeWork Board Rev B, read the

- ▶ HomeWork Board Rev B Special Instructions on page 17 before continuing.

HomeWork Board Rev B Special Instructions

Vdd - Limited Current Budget

The BASIC Stamp HomeWork Board Rev B's voltage regulator supplies current to the embedded BASIC Stamp, certain circuits built on the breadboard and connected to I/O pins, and any devices drawing current from the Vdd sockets. The current budget for the Board of Education Rev B board's voltage regulator is 50 mA. Most devices featured in Stamps in Class texts only draw a few extra milliamps of current and will work fine when powered by the Board of Education Rev B board's Vdd Pins. However, there are a few exceptions, including servos, pumps, and heating elements. These devices should not be connected to Vdd, or it could damage the board's voltage regulator. Instead, these higher current devices should be connected to Vin. The voltage supplied to Vin should in turn be limited to a level that is compatible with the device (servo, pump, heating element).

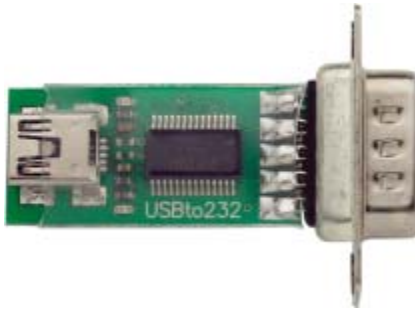
- ▶ Go ahead to Using a USB to Serial Adapter with your HomeWork Board? on page 18.

Using a USB to Serial Adapter with your HomeWork Board?

Will you be using your computer's USB port and a USB to Serial Adapter with your HomeWork Board?

YES!

- ▶ Go to HomeWork Board with USB to Serial Adapter - Setup on page 19



NO!

- ▶ Go to HomeWork Board - Setup on page 21.

HomeWork Board with USB to Serial Adapter - Setup

- ✓ DO NOT connect power to your board until instructed to do so!

Affixing the Rubber Feet

Your board should come with four adhesive rubber feet:

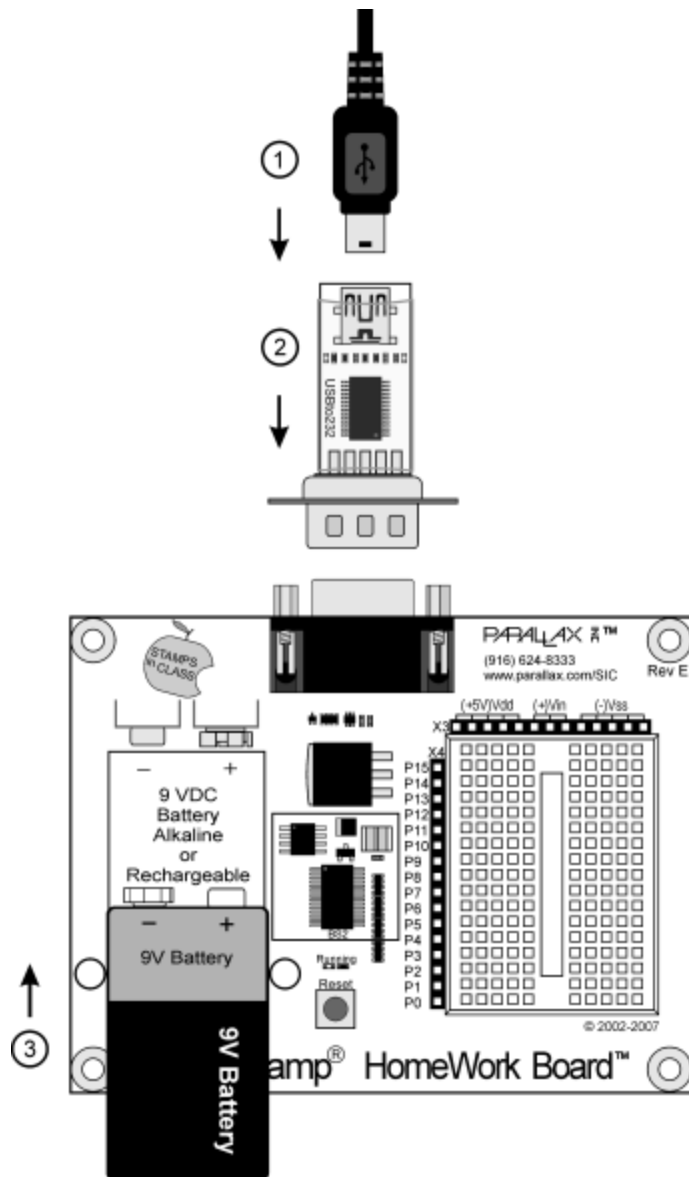


- ✓ Remove each rubber foot from its adhesive, and affix it to the underside of your board near each corner. Make sure not to cover up the plated holes, you might need these holes later for mounting your board in a project box or on a robot.



Making the Board Connections

- ✓ Connect the larger "A" end of the USB cable to a USB port on your computer.
- ✓ Connect the smaller "Mini B" end of the USB cable to the USB connector on the USB to Serial Adapter, as shown in (1).
- ✓ Connect the male DB9 connector on the USB to Serial Adapter to the female DB9 socket on the board (2).
- ✓ Connect a 9 V alkaline or rechargeable battery to the battery clip (3). (
- ✓ Power SuppliesSee Power Supplies on page 41 for more about batteries and battery replacer devices.)



You are ready to test your programming connection!

► Go to Testing the Connection on page 23.

HomeWork Board - Setup

- ✓ DO NOT connect power to your board until instructed to do so!

Affixing the Rubber Feet

Your board should come with four adhesive rubber feet:

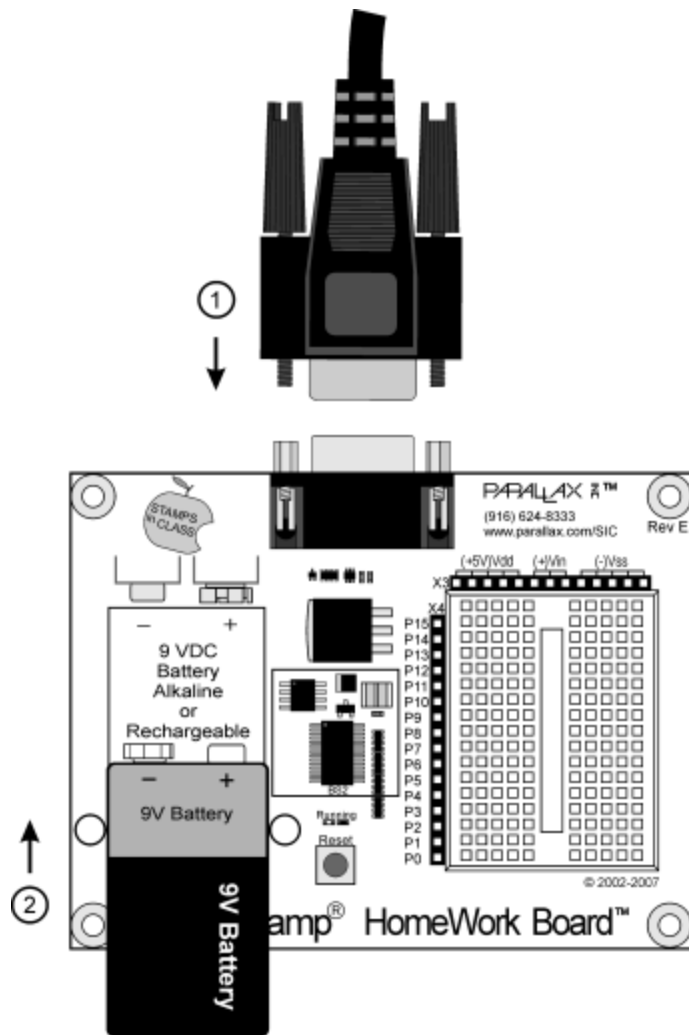


- ✓ Remove each rubber foot from its adhesive, and affix it to the underside of your board near each corner. Make sure not to cover up the plated holes, you might need these holes later for mounting your board in a project box or on a robot.



Making the Board Connections

- ✓ Connect the female end of the serial cable to the serial port on your computer (it is a 9-pin male port).
- ✓ Connect the male end of the serial cable to the DB9 connector on the board, as shown in (1).
- ✓ Connect a 9 V alkaline or rechargeable battery to the battery clip (3). (
- ✓ Power SuppliesSee Power Supplies on page 41 for more about batteries and battery replacer devices.)



You are ready to test your programming connection!

► Go to Testing the Connection on page 23.

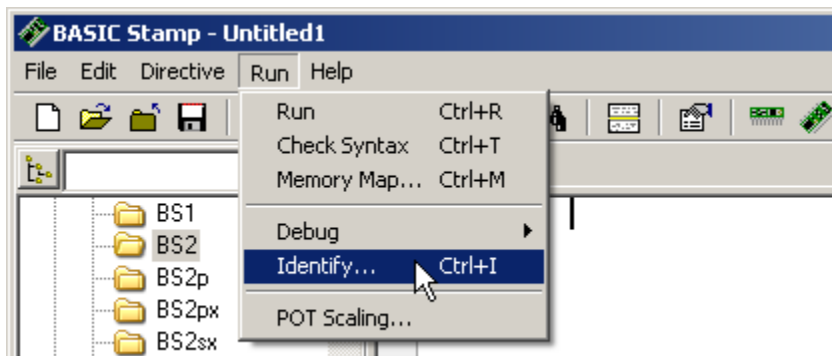
Testing the Connection

At this point, you should have:

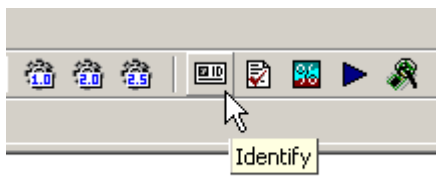
- your BASIC Stamp on your development board (either built-in or inserted into its socket),
- your development board connected to your computer,
- power connected to your development board (and turned on if there is a switch).
- If not, go back to Identify Your Board on page 2, make those connections, and then return here.

The Run/Identify Test

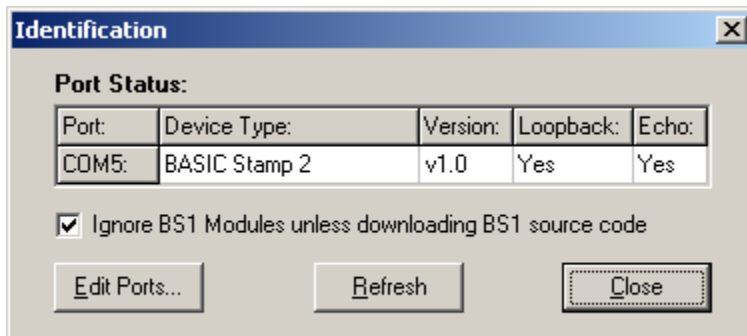
- ✓ To make sure your BASIC Stamp module can communicate with your computer, click the Run menu, then select Identify.



- ✓ Or, you can use the ID icon on the toolbar:



An identification window will appear similar to the one shown here. This example shows that a BASIC Stamp 2 has been detected on COM5.



- ✓ Check the Identification window to make sure your BASIC Stamp module has been detected on one of the COM ports.

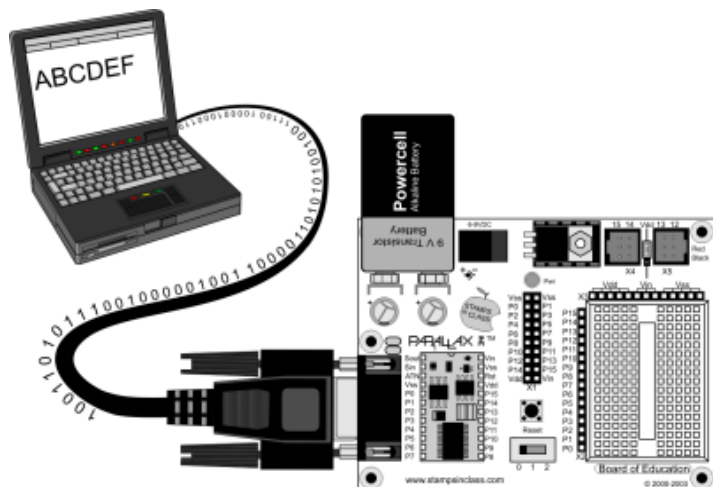
- ✓ If it has been detected, then you are ready to program your BASIC Stamp.
- ✓ Click the NEXT arrow at the top or bottom of the page.

What if it didn't work?

- ▶ If the Run -> Identify test DID NOT locate the BASIC Stamp on any COM port, you will need to go to Connection Troubleshooting on page 44.

First Program

The first program you will write and test will tell the BASIC Stamp to send a message to your computer. The figure below shows how it sends a stream of ones and zeros to communicate the text characters displayed by your computer. These ones and zeros are called binary numbers. The BASIC Stamp Editor software has the ability to detect and display these messages as you will soon see.



First Program

In this Help tutorial, the program listings that you will type into the BASIC Stamp Editor and download to the BASIC Stamp module will be shown with a blue background like this:

Example Program: FirstProgram.bs2

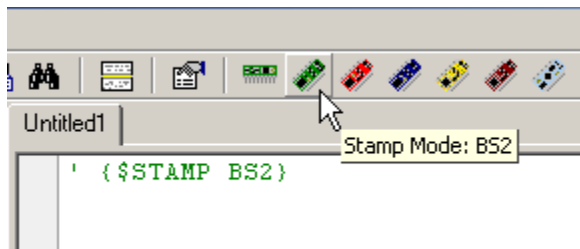
```
' Stamps in Class - FirstProgram.bs2
' BASIC Stamp sends message to Debug Terminal.

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

DEBUG "Hello, it's me, your BASIC Stamp!"
END
```

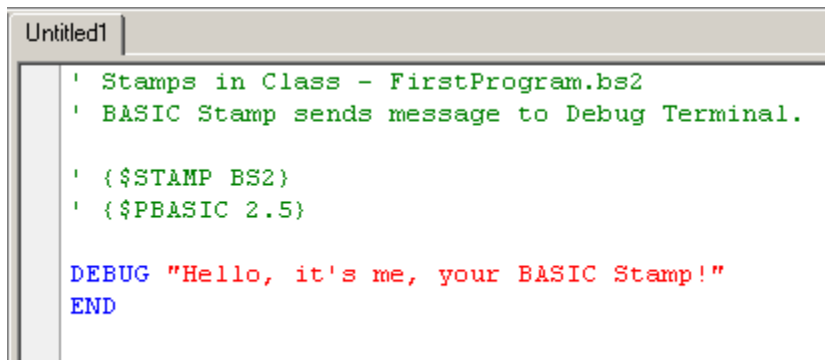
You will enter this program into the BASIC Stamp Editor. Some lines of the program are created automatically by clicking buttons on the toolbar. Other lines are made by typing them in from the keyboard.

- ✓ Begin by clicking the BS2 icon (the green diagonal chip) on the toolbar. If you hold your cursor over this button, its flyover help description “Stamp Mode: BS2” will appear.
- ✓ Next, click on the gear icon labeled “2.5.” It’s flyover help description is “PBASIC Language: 2.5”.



Tip: ALWAYS use these toolbar buttons to add these two lines as the beginning of every program! Compiler directives use braces { }. If you try to type in these parts of your program, you may accidentally use parentheses () or square brackets []. If you do this, your program will not work.

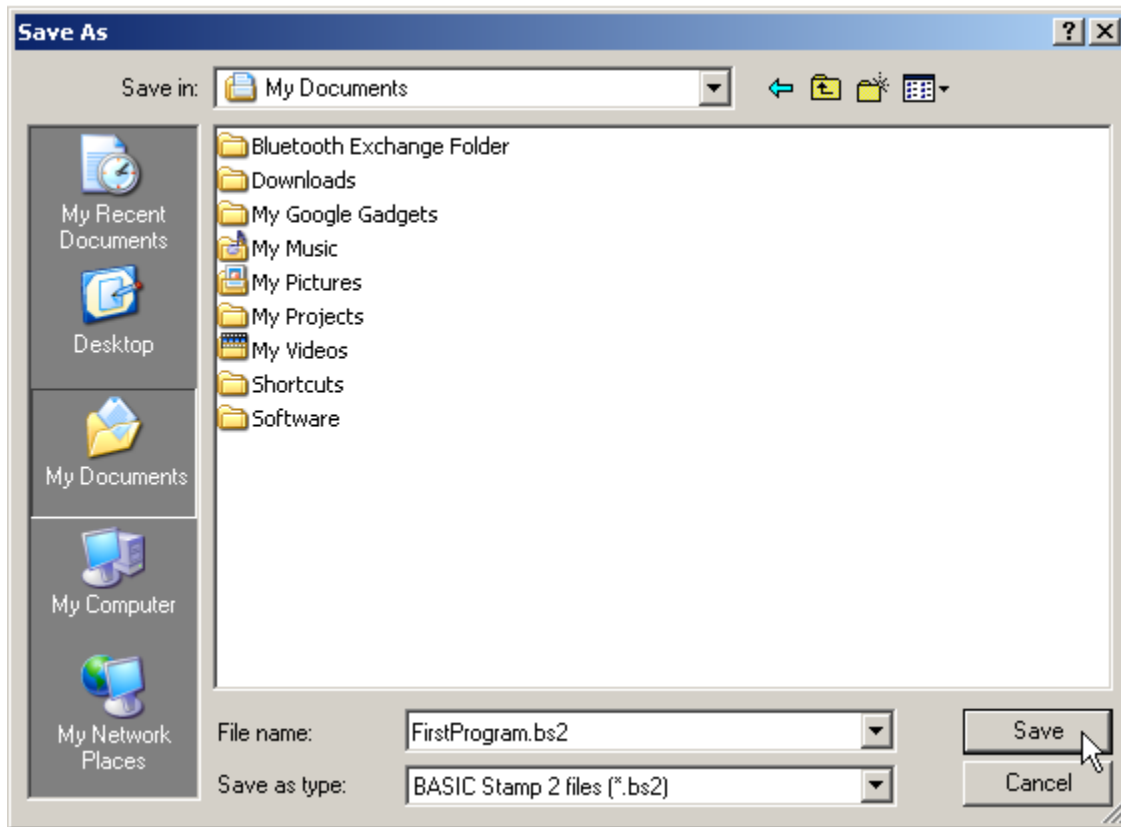
- ✓ Type in the remaining lines of the program exactly as shown:



- ✓ Save your work by clicking File and selecting Save.

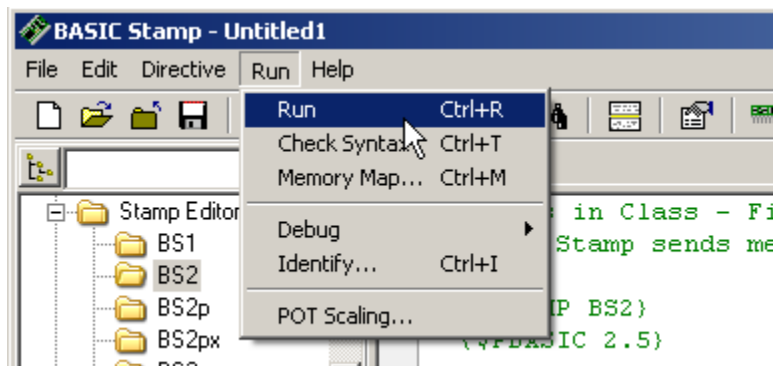


- ✓ Enter the name "FirstProgram" into the File name field near the bottom of the Save As window.
- ✓ Click the Save button.



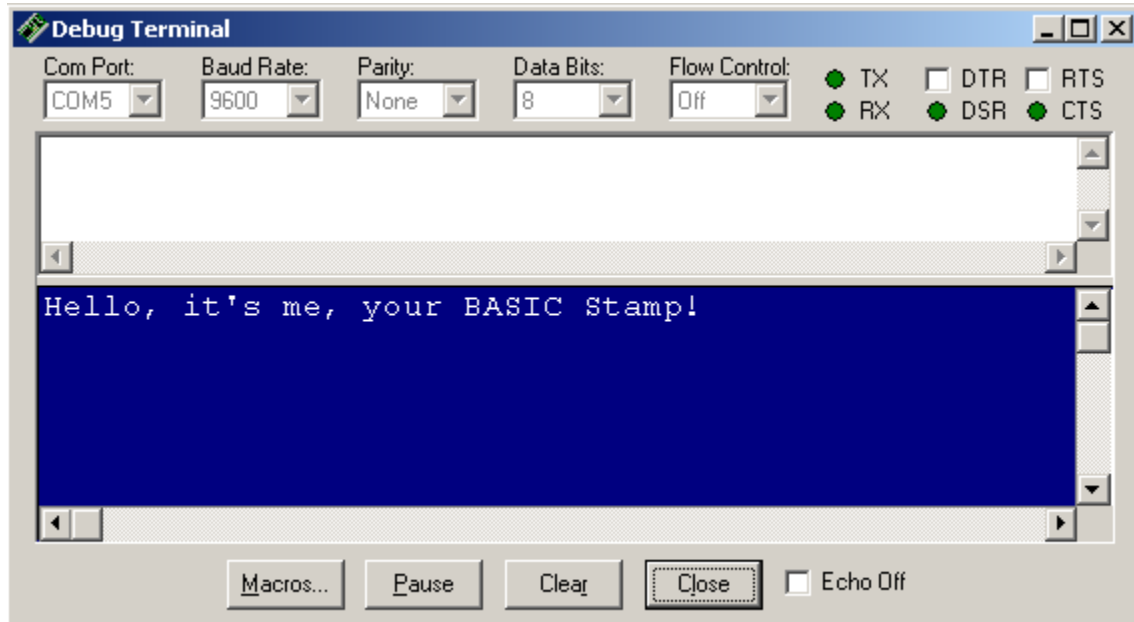
Tip: The next time you save, the BASIC Stamp Editor will automatically save to the same filename (FirstProgram.bs2) unless you tell it to save to a different filename by clicking File and selecting Save As (instead of just Save).

- ✓ Click Run, and select Run from the menu that appears.



A Download Progress window will appear briefly as the program is transmitted from your computer to your BASIC Stamp module. The figure below shows the Debug Terminal that should appear when the download is complete. You can prove to yourself that this is a message from the BASIC Stamp by pressing and releasing the Reset button on your board. Every time you press and release it, the program will re-run, and you will see another copy of the message displayed in the Debug Terminal.

- ✓ Press and release the Reset button. Did you see a second “Hello...” message appear in the Debug Terminal?



The BASIC Stamp Editor has shortcuts for most common tasks. For example, to run a program, you can press the ‘Ctrl’ and ‘R’ keys at the same time. You can also click the Run button. It’s the blue triangle that looks like a music player’s Play button. The flyover help (the Run hint) will appear if you point at the Run button with your mouse. You can get similar hints to find out what the other buttons do by pointing at them too.



How FirstProgram.bs2 Works

The first two lines in the example are called comments. A comment is a line of text that gets ignored by the BASIC Stamp Editor, because it’s meant for a human reading the program, not for the BASIC Stamp module. In PBASIC, everything to the right of an apostrophe is normally considered to be a comment by the BASIC Stamp Editor. The first comment tells which book the example program is from, and the program’s filename. The second comment contains a handy, one-line description that explains what the program does.

```
' Stamps in Class - FirstProgram.bs2  
' BASIC Stamp sends message to Debug Terminal.
```

Although comments are ignored most of the time, the BASIC Stamp Editor does search through comments for special directives. Every program in this Getting Started section of help will use these two directives:

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

The first directive is called the \$STAMP Directive, and it tells the BASIC Stamp Editor that you will be downloading the program specifically to a BASIC Stamp 2 module. The second directive is called the \$PBASIC directive, and it tells the BASIC Stamp Editor that you are using version 2.5 of the PBASIC programming language. These special comments are called compiler directives, and they are enclosed in braces { } not parentheses (). You should always use the toolbar icons to place these compiler directives in your program to avoid typing errors. Also, entering the compiler directives by hand may not activate the syntax highlighting in the BASIC Stamp Editor. That function is what causes various letters, characters and words in your program to appear in different colors and capitalization schemes. Syntax highlighting makes your programs easier to read, understand, and correct if there are any bugs in them. A command is a word you can use to tell the BASIC Stamp do a certain job. The first of the two commands in this program is called the DEBUG command:

```
DEBUG "Hello, it's me, your BASIC Stamp!"
```

This is the command that tells the BASIC Stamp to send a message to the PC using the serial cable. The second command is the END command:

```
END
```

This command is handy because it puts the BASIC Stamp into low power mode when it's done running the program. In low power mode, the BASIC Stamp waits for either the Reset button to be pressed (and released), or for a new program to be loaded into it by the BASIC Stamp Editor. If the Reset button on your board is pressed (or if you disconnect and reconnect your power supply), the BASIC Stamp will re-run the program you loaded into it. If a new program is loaded into it, the old one is erased, and the new program begins to run.

Your Turn – Delays with PAUSE, DEBUG Formatters, and Control Characters

In *What's a Microcontroller?* and *Robotics with the Boe-Bot*, the first command you will likely see in the example programs that display messages in the Debug Terminal is a 1-second delay, typically with the command PAUSE 1000. The PAUSE command delays the program for a certain number of milliseconds. Milliseconds are thousandths of a second and are typically abbreviated ms. So, PAUSE 1000 delays the program for 1000 thousandths of a second, which is one second.

- ✓ Modify the program by inserting PAUSE 1000 immediately above the DEBUG command.

```
PAUSE 1000
```

- ✓ Your code should then look like this:

```
' Stamps in Class - FirstProgram.bs2  
' BASIC Stamp sends message to Debug Terminal.  
  
' {$STAMP BS2}  
' {$PBASIC 2.5}
```

```
PAUSE 1000
DEBUG "Hello, it's me, your BASIC Stamp!"
END
```

- ✓ Run the modified program and verify that it delays for a second before displaying the Hello message.

For comparison, you can disable the PAUSE command by commenting it. In other words, add an apostrophe to its left so that it reads ' PAUSE 1000. By removing the apostrophe and re-running the program, you can then test how the program behaves without the PAUSE.

- ✓ Try it.

Tip: Inserting a one second delay before the BASIC Stamp transmits messages to the Debug Terminal ensures that the Windows operating system cannot possibly mistake the BASIC Stamp for a plug-and-play-serial device like a mouse or keyboard. This can happen if the BASIC Stamp is running a program that immediately transmits messages to the Debug Terminal when it gets connected to a USB port. It can also happen if the same program is running as the computer boots while it is connected to a serial or USB port. The PAUSE 1000 ensures that this case of "mistaken microcontroller identity" won't happen because it waits longer than the 0.7 second window that PCs give serial plug-and-play devices to identify themselves.

DEBUG Formatters and Control Characters

A DEBUG formatter is a code-word you can use to make the message the BASIC Stamp sends look a certain way in the Debug Terminal. DEC is an example of a formatter that makes the Debug Terminal display a decimal value. An example of a control character is CR, which is used to send a carriage return to the Debug Terminal. The text or numbers that come after a CR will appear on the line below characters that came before it. You can modify your program so that it contains more DEBUG commands along with some formatters and control characters. Here's an example of how to do it:

- ✓ Modify the comments at the beginning of the program so they read:

```
' Stamps in Class - FirstProgramYourTurn.bs2
' BASIC Stamp sends messages to Debug Terminal
```

- ✓ Add these three lines between the first DEBUG command and the END command:

```
DEBUG CR, "What's 7 X 11?"
DEBUG CR, "The answer is: "
DEBUG DEC 7 * 11
```

- ✓ Save the changes you made by clicking File and selecting Save As. A good name would be FirstProgramYourTurn.bs2
- ✓ Check your work against the example program shown here.
- ✓ Run your modified program. You will have to either select Run from the Run menu again, or click the Run button.

```

FirstProgram.bs2  FirstProgramYourTurn.bs2
' Stamps in Class - FirstProgramYourTurn.bs2
' BASIC Stamp sends messages to Debug Terminal

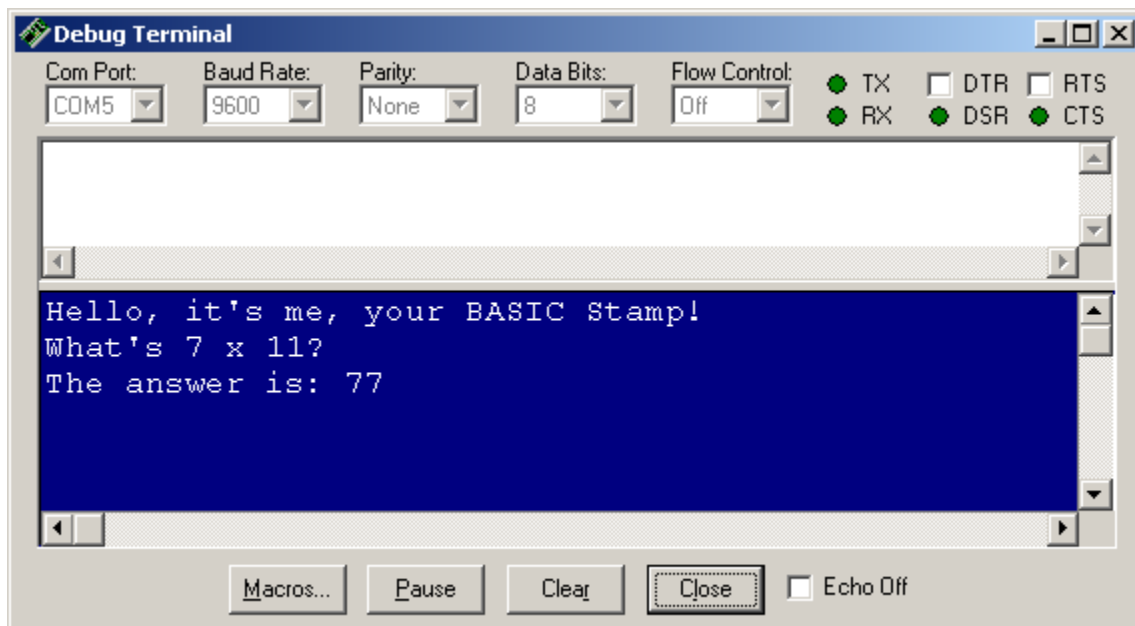
' {$STAMP BS2}
' {$PBASIC 2.5}

PAUSE 1000

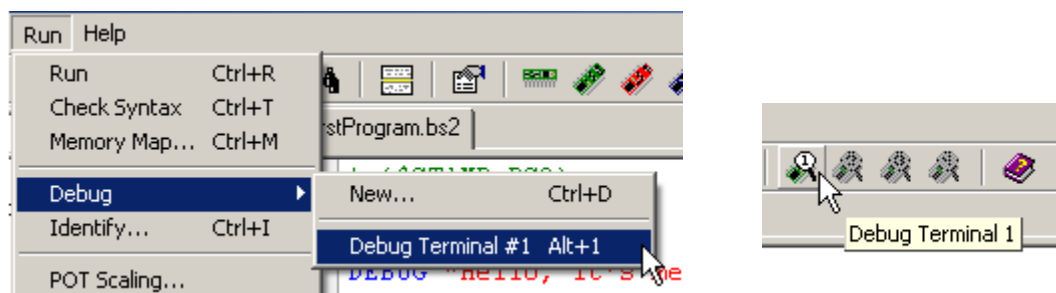
DEBUG "Hello, it's me, your BASIC Stamp!"
DEBUG CR, "What's 7 X 11?"
DEBUG CR, "The answer is: "
DEBUG DEC 7 * 11

```

- ✓ Check your Debug Terminal - does it now look like this?
- ✓ If not, correct your program and re-run it until you get the results you expect.



Tip: Sometimes the Debug Terminal gets hidden behind the BASIC Stamp Editor window. You can bring it back to the front by using the Run menu as shown, the Debug Terminal 1 shortcut button on the toolbar, or the F12 key on your keyboard.

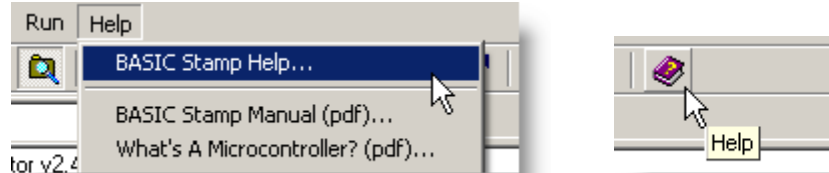


Looking Up Answers

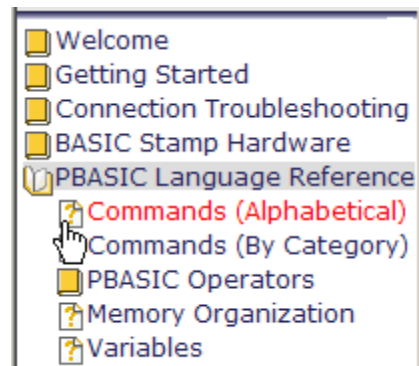
The activities you just finished introduced two PBASIC commands: DEBUG and END. You can find out more about these commands and how they are used by looking them up, either in the BASIC Stamp Help or in the BASIC Stamp Manual.

Exploring the Help Resources

- ✓ Open the BASIC Stamp Editor Help with the Help menu or the Help icon



There are many ways to find resources in the Help. If you want to know the syntax and usage details of a particular PBASIC command, you can click on "PBASIC Language Reference" in the Table of Contents to see the full list.



Next to each command is a set of icons representing which BASIC Stamp models support that command. All commands available to the BASIC Stamp 2 have this icon: . The syntax showing how to use the command is given to the right of the icons.

• DATA							<code>{Symbol} DATA Dataltem {, Dataltem, ...}</code>
• DEBUG							<code>DEBUG OutputData {, OutputData}</code>
• DEBUGIN ^{2.5}							<code>DEBUGIN InputData</code>

Clicking on a command will take you to a page full of information, and a link to example programs. Here is the DEBUG command's page:

DEBUG

 1  2  2E  2SX  2P  2PE  2PX

[DEBUG Examples](#)

Syntax: **DEBUG** *OutputData* {, *OutputData*}

Function

Display information on the PC screen within the BASIC Stamp editor program. This command can be used to display text or numbers in various formats on the PC screen in order to follow program flow (called debugging) or as part of the functionality of the BASIC Stamp application.

You can also learn more about a command using the Search menu. If you enter a term in the field and hit the Search button, you will see a list of topics where the search term appears:

[Table of Contents](#)

[Search](#)

[Favorites](#)

→

DEBUG Search 

Rank	Title
1	DEBUG
2	Binary Operators
3	DEBUG Examples

Clicking an item in the list will take you that page, and each instance of the search term on that page will be highlighted:

Search

DEBUG Search 

Rank	Title
1	DEBUG
2	Binary Operators
3	DEBUG Examples
4	SERIN
5	FOR...NEXT
6	LOOKDOWN
7	Unary Operators
8	First Program

DEBUG Examples

 1  2  2E  2SX  2P  2PE  2PX

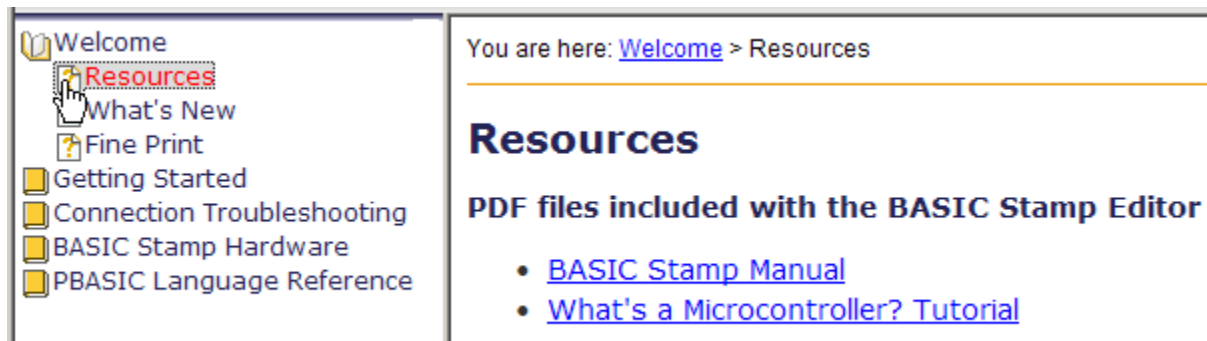
```
' DEBUG.BAS
' Demonstrates the use of the BS1 DEBUG instruction. The BS1 DEBUG defaults
' to decimal output and provides the name of the variable unless directed
' otherwise.
```

You can save the search string by clicking the magnifying glass icon, and then the search string will show up in your Favorites list:



Additional Resources

Some helpful books are included with the BASIC Stamp Editor software, as PDF files. You can find the Resources page link under Welcome in the Table of Contents. The Resources page contains links that will open PDF files in your computer's PDF viewer, if you have one.



Your Turn

- ✓ Use the Help file's features to research the DEBUG and END commands.
- ✓ If you have a PDF viewer, research DEBUG and END in the BASIC Stamp Manual file.
- ✓ Now go on to the next activity: Introducing ASCII Code on page 35.

Introducing ASCII Code

In the First Program activity, you used the DEC formatter with the DEBUG command to display a decimal number in the Debug Terminal. But what happens if you don't use the DEC formatter with a number? If you use the DEBUG command followed by a number with no formatter, the BASIC Stamp will read that number as an ASCII code.

Programming with ASCII Code

ASCII is short for American Standard Code for Information Interchange. Most microcontrollers and PC computers use this code to assign a number to each keyboard function. Some numbers correspond to keyboard actions, such as cursor up, cursor down, space, and delete. Other numbers correspond to printed characters and symbols. The numbers 32 through 126 correspond to those characters and symbols that the BASIC Stamp can display in the Debug Terminal. The following program will use ASCII code to display the words "BASIC Stamp 2" in the Debug Terminal.

Example Program – ASCIIName.bs2

✓ Enter and run ASCIIName.bs2.

Tip: Remember to use the toolbar icons to place Compiler Directives into your programs!

- '{\$STAMP BS2} - Use the diagonal green electronic chip icon.
- '{\$PBASIC 2.5} - Use the gear icon labeled 2.5.

```
' Stamps in Class - ASCIIName.bs2
' Use ASCII code in a DEBUG command to display "BASIC Stamp 2."

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

PAUSE 1000
DEBUG 66,65,83,73,67,32,83,116,97,109,112,32,50

END
```

How ASCIIName.bs2 Works

Each number in the DEBUG command corresponds to one ASCII code symbol that appeared in the Debug Terminal.

```
DEBUG 66,65,83,73,67,32,83,116,97,109,112,32,50
```

66 is the ASCII code for capital "B", 65 is the code for capital "A" and so on. 32 is the code for a space between characters. Notice that each code number was separated with a comma. The commas allow the one instance of DEBUG to transmit each symbol as a separate value. This is much easier to type than 12 separate DEBUG commands.

Your Turn – Exploring ASCII Code

- ✓ Save ASCIIName.bs2 as ASCIIRandom.bs2
- ✓ Pick 12 random numbers between 32 and 126.

- ✓ Replace the ASCII code numbers in the program with the numbers you chose.
- ✓ Run your modified program to see what you get!

This document has an ASCII Chart with the numbers and their corresponding symbols. You can look up the corresponding code numbers to spell your own name.

- ✓ Save ASCIIRandom.bs2 as YourASCIIName.bs2
- ✓ Using the ASCII Chart beginning on page 37 as a reference, modify the program to spell your own name.
- ✓ Run the program to see if you spelled your name correctly.
- ✓ If you did, good job, and save your program!

ASCII Chart

This chart includes the first 128 ASCII characters. Note that the control codes (lowest 32 ASCII characters) have no standardized screen symbols. The characters listed for them are just names used in referring to these codes. For example, to move the cursor to the beginning of the next line of a printer or terminal often requires sending line feed and carriage return codes. This common pair is referred to as "LF/CR."

Dec	Hex	Char	Name / Function	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	NUL	Null	32	20	space	64	40	@	96	60	`
1	01	SOH	Start Of Heading	33	21	!	65	41	A	97	61	a
2	02	STX	Start Of Text	34	22	"	66	42	B	98	62	b
3	03	ETX	End Of Text	35	23	#	67	43	C	99	63	c
4	04	EOT	End Of Transmit	36	24	\$	68	44	D	100	64	d
5	05	ENQ	Enquiry	37	25	%	69	45	E	101	65	e
6	06	ACK	Acknowledge	38	26	&	70	46	F	102	66	f
7	07	BEL	Bell	39	27	'	71	47	G	103	67	g
8	08	BS	Backspace	40	28	(	72	48	H	104	68	h
9	09	HT	Horizontal Tab	41	29	)	73	49	I	105	69	i
10	0A	LF	Line Feed	42	2A	*	74	4A	J	106	6A	j
11	0B	VT	Vertical Tab	43	2B	+	75	4B	K	107	6B	k
12	0C	FF	Form Feed	44	2C	,	76	4C	L	108	6C	l
13	0D	CR	Carriage Return	45	2D	-	77	4D	M	109	6D	m
14	0E	SO	Shift Out	46	2E	.	78	4E	N	110	6E	n
15	0F	SI	Shift In	47	2F	/	79	4F	O	111	6F	o
16	10	DLE	Data Line Escape	48	30	0	80	50	P	112	70	p
17	11	DC1	Device Control 1	49	31	1	81	51	Q	113	71	q
18	12	DC2	Device Control 2	50	32	2	82	52	R	114	72	r
19	13	DC3	Device Control 3	51	33	3	83	53	S	115	73	s

20	14	DC4	Device Control 4	52	34	4	84	54	T	116	74	t
21	15	NAK	Non Acknowledge	53	35	5	85	55	U	117	75	u
22	16	SYN	Synchronous Idle	54	36	6	86	56	V	118	76	v
23	17	ETB	End Transmit Block	55	37	7	87	57	W	119	77	w
24	18	CAN	Cancel	56	38	8	88	58	X	120	78	x
25	19	EM	End Of Medium	57	39	9	89	59	Y	121	79	y
26	1A	SUB	Substitute	58	3A	:	90	5A	Z	122	7A	z
27	1B	ESC	Escape	59	3B	;	91	5B	[	123	7B	{
28	1C	FS	File Separator	60	3C	<	92	5C	\	124	7C	
29	1D	GS	Group Separator	61	3D	=	93	5D	]	125	7D	}
30	1E	RS	Record Separator	62	3E	>	94	5E	^	126	7E	~
31	1F	US	Unit Separator	63	3F	?	95	5F	_	127	7F	delete

When You're Done

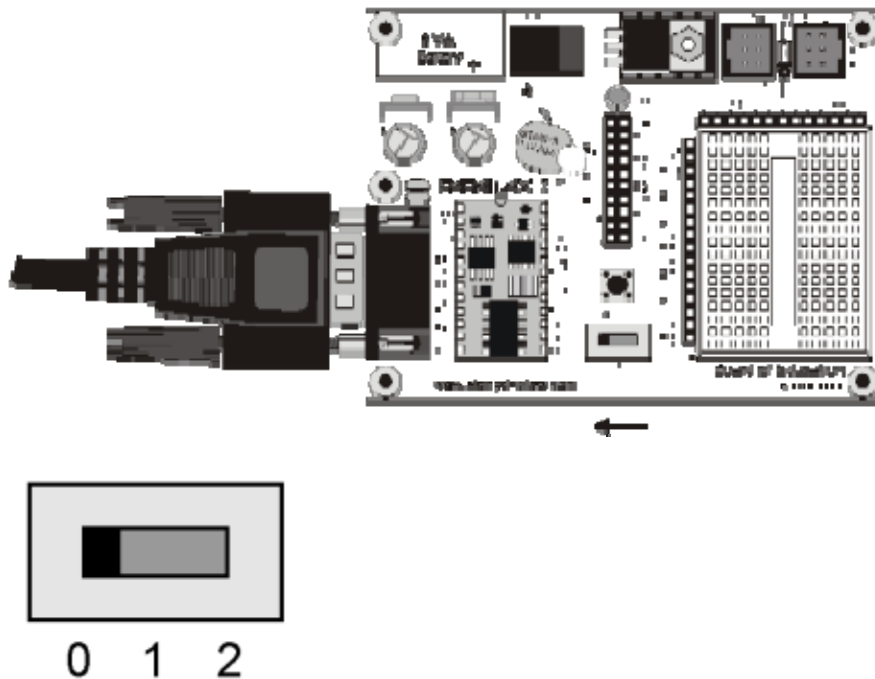
It's important to disconnect the power from your BASIC Stamp and Board of Education (or HomeWork Board) whenever you leave it unattended or when you build or modify circuits on its breadboard. First, your batteries will last longer if the system is not drawing power when you're not using it. Second, soon you will build circuits on the Board of Education or HomeWork Board prototyping area.

Caution! Circuit prototypes should never be left unattended with a battery or power supply connected. Always disconnect the power from your Board of Education or HomeWork Board before you walk away, even if you only plan on leaving it alone for a minute or two.

Disconnecting Power

For the Board of Education

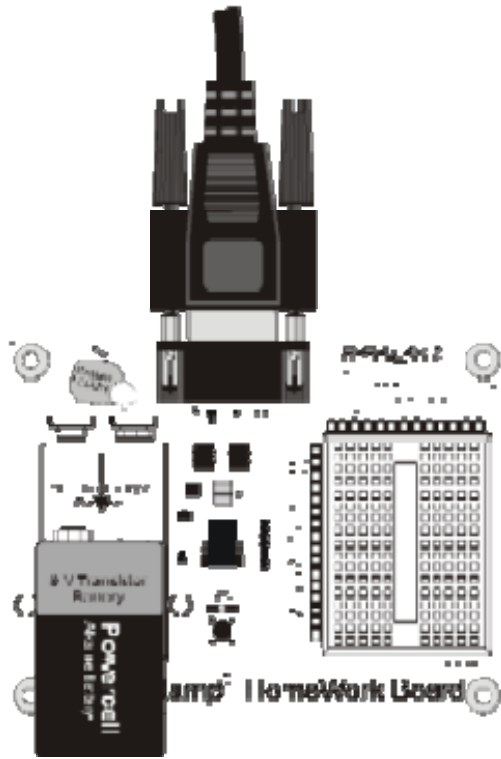
With the Board of Education - Serial (Rev C), or Board of Education USB, disconnecting power is easy. Just move the 3-position switch to position-0 by pushing it to the left as shown below. (If you have an older Board of Education- Serial which does not have a 3-position switch, unplug the battery or power supply plug.)



Caution! Do not remove the BASIC Stamp module from its socket, unless it needs to be replaced with another module! Every time the BASIC Stamp is removed and re-inserted into the socket, you risk damaging it. You do not need to remove it for storage.

For the HomeWork Board

Disconnecting the BASIC Stamp HomeWork Board's power is easy too. If you are using the BASIC Stamp HomeWork Board, disconnect the battery as shown below.



Your Turn

- ✓ Disconnect the power to your board now.
- ✓ If you are following the directions in a Stamps in Class text, you are ready to return to the text now.

Power Supplies

Parallax BASIC Stamp Development Boards may have either a barrel jack connector, 9 V battery clip connector, or both (as does the Board of Education.) This page gives options for both types of power connectors.

For Boards with a Barrel Jack Connector

The two most common power supply options used with this connector are wall-mount DC power supply, and battery holders.

Wall-mount DC Power Supplies

- Plug: it must have a 2.1 mm center-positive barrel plug. Look for this symbol, indicating the positive connection is in the center of the plug:



- Output current: wall-mount power supplies must have a DC (direct current) output. Look for a statement that indicates VDC output or uses this symbol: **≡**. For example:
- OUTPUT: 7.5 VDC 1000 mA
- OUTPUT: 7.5 V **≡** 1.4 A
- Output voltage: The power supply's output voltage must be compatible with the voltage rating on your board. For example, the Board of Education requires 6 to 9 VDC.
- Input voltage and current: Wall-mount power supplies must be compatible with the frequency and amplitude of the AC power at the wall outlet you are using. This may vary depending on the country you live in. In the US and Canada, the input should be compatible with 120 VAC, 60 Hz.

Parallax carries power supplies compatible with BASIC Stamp development boards. For example, the 7.5 V, 1 amp supply (#750-00009) is compatible with the Board of Education. Go to www.parallax.com/go/powersupplies for more information.



Battery Holders

Like wall-mount supplies, battery holders must have a 2.1 mm center-positive plug. However, many battery holders don't have a symbol indicating whether or not the plug is center-positive. Use a voltmeter to test any battery holder of unknown origins. A 4 AA cell battery holder with a 2.1 mm center-positive plug is available from Parallax (#700-00038); this is the one included in the [Boe-Bot Robot Kit](#).



For Boards with Battery Clip Connector

9-Volt Batteries

9 V Alkaline or rechargeable batteries with a snap-connector are fine. For best results, rechargeable batteries should be rated for 100 mAh (milliamp hours) or higher. Not all chargers work with all types of batteries. Make sure your charger is recommended for the type of battery you are using. Follow all battery and charger instructions and caution statements.

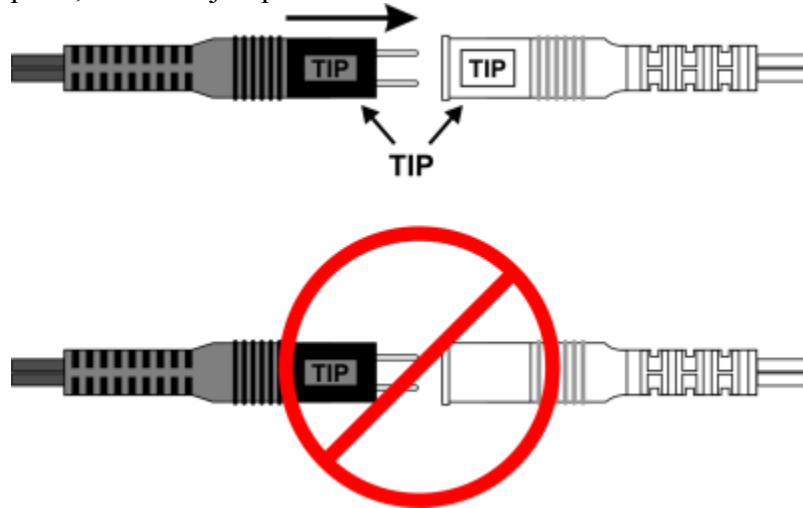
9 V Battery Extension

The figure below shows a DC supply and a 9 V battery extension that can be used with battery clip connectors, such as the one on the BASIC Stamp HomeWork Board. Read the CAUTION statement below the picture.



Beware of Universal Adapters and Reversed Supply Terminals!

The picture below shows a common mistake that should be avoided with universal adapters. Many of these allow you to reverse the terminals on the 9 V battery extension. Although it cannot hurt the BASIC Stamp, Board of Education, or HomeWork Board, it can destroy a Parallax Standard Servo connected to Vin in a matter of seconds. The only system that can protect the servo from this mistake is a Board of Education that has a jumper between the servo ports, with that jumper set to Vdd.



Beware of "Battery Replacers"

Many battery replacers are designed to supply appliances with low current draw. With current ratings in the neighborhood of 10 mA, their output capacities are insufficient for many of the activities in the Stamps in Class texts. For example, two LEDs connected to 220-ohm resistors draw a total of 14.5 mA, and that definitely won't work with a "battery replacer."

Note: It's pretty easy to tell when a circuit is drawing more current than the supply can deliver because the Power LED on the Board of Education flickers and/or goes dim. On the HomeWork Board, the "Running" LED may flicker if the power drops low enough to cause the BASIC Stamp to reset and restart the program.

Connection Troubleshooting

This is the "triage" area for troubleshooting the programming connection between your computer and your BASIC Stamp development board. Let's rule out the most common causes of communications failure first:

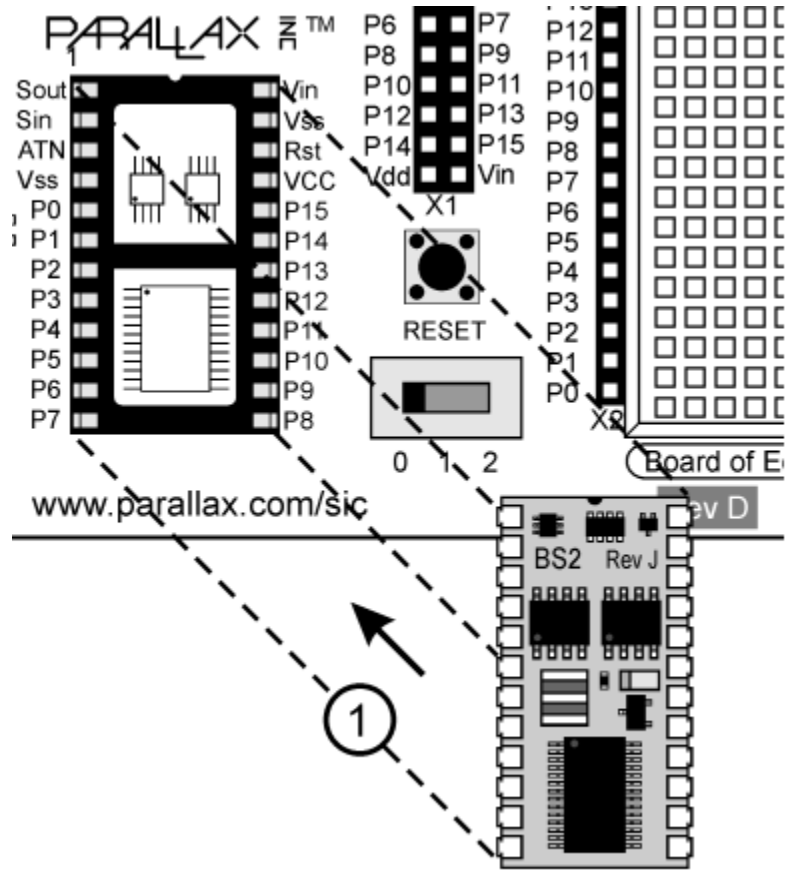
1. **Loose connections:** Double-check the connections between your computer and your programming cable, the Parallax USB to Serial Adapter if you are using one, and the development board.
2. **Misconnected BASIC Stamp:** If you are using a Board of Education, double-check how to Insert the BASIC Stamp module on page 45 .
- ✓ **No power:** double-check the power supply that you have connected to your board. (See
3. **Power Supplies** on page 41 for options). If your board has a power switch, make sure it is switched on and the power indicator is lit. If you are using a 9V battery, try a fresh one. If you are using a battery pack, make sure the batteries are all inserted into the pack correctly; try fresh batteries. If you are using a wall-mount supply, make sure the wall outlet is not turned off at a wall switch. If you have a voltmeter, use it to measure the voltage supplied by the battery pack or wall-mount to rule out defective power supply hardware.
4. **Software version:** make sure you are using the latest version of the BASIC Stamp Editor Software. This is Version 2.5. You may check for a later version at www.parallax.com/basicstampsoftware.
5. **USB drivers:** If you are using a Parallax USB connection, you may need to update your USB drivers. See the www.parallax.com/usbdriers page for help.
6. **Beware of non-Parallax USB to Serial Adapters:** Not every brand of USB to Serial Adapter supports the timing precision needed for BASIC Stamp programming. Some known-incompatible brands are Belkin and Gigaware. Keyspan #USA19-HS is known to be compatible. You can order an inexpensive Parallax USB to Serial Adapter (#28030) from www.parallax.com.
- ✓ Okay, if that didn't solve the problem, see the Connection Troubleshooting section of the BASIC Stamp Editor Help.

Contact Parallax Tech Support:

- Forums: forums.parallax.com
- Web: www.parallax.com/support
- Email: support@parallax.com
- Telephone (Toll Free in Continental US): 1-888-997-8267
- Outside the Continental US: 1-916-624-8333

Insert the BASIC Stamp module

- ✓ Use the silkscreen pictures that show the three prominent chips on the BASIC Stamp to determine how to orient it before inserting it into its socket. There is also a small semi-circular notch at in the top-center of the BASIC Stamp module that indicates which way is “up.” Match this reference notch to the notch in the socket.
- ✓ After correctly orienting the 24-pin BASIC Stamp module, gently insert it into its socket as shown at (1) in the picture below.
- ✓ Make sure that each pin is seated in its corresponding socket hole, then apply firm downward pressure with your thumb. The module should sink about $\frac{1}{4}$ of an inch (~6 mm) into the socket, and only the flared portion of each BASIC Stamp leg should be visible above the socket.
- ✓ Visually inspect the legs to make sure each pin is in its socket and that no pins are folded underneath the module.



Caution! Do not remove the BASIC Stamp module from its socket, unless it needs to be replaced with another module! Every time the BASIC Stamp is removed and re-inserted into the socket, you risk damaging it. You do not need to remove it for storage.