Teachers Teaching with Technology™
Professional Development from Texas Instruments

Exploring Path to STEM Projects with TI-Innovator™ Hub Using TI-84 Plus CE Graphing Calculators

Materials for Workshop Participant*

*This material is for the personal use of T^3 Instructors in delivering a T^3 workshop. T^3 Instructors are further granted limited permission to copy the participant packet in seminar quantities solely for use in delivering seminars for which the T^3 Office certifies the Instructor to present. T^3 workshop organizers are granted permission to copy the participant packet for distribution to those who attend the T^3 workshop.

*This material is for the personal use of participants during the workshop. Participants are granted limited permission to copy handouts in regular classroom quantities for use with students in participants' regular classes. Participants are also granted limited permission to copy a subset of the package (up to 25%) for presentations and/or conferences conducted by participant inside his/her own district institutions. All such copies must retain Texas Instruments copyright and be distributed as is.

Request for permission to further duplicate or distribute this material must be submitted in writing to the T^3 Office.

Texas Instruments makes no warranty, either expressed or implied, including but not limited to any implied warranties of merchantability and fitness for a particular purpose, regarding any programs or book materials and makes such materials available solely on an "as-is" basis.

In no event shall Texas Instruments be liable to anyone for special, collateral, incidental, or consequential damages in connection with or arising out of the purchase or use of these materials, and the sole and exclusive liability of Texas Instruments, regardless of the form of action, shall not exceed the purchase price of this calculator. Moreover, Texas Instruments shall not be liable for any claim of any kind whatsoever against the use of these materials by any other party.

This page intentionally left blank

Color Printed Documents	
A. Unpacking the TI-Innovator™ Breadboard Pack	
B. Introduction to Breadboarding - Breadboard Diagram	
C. TI-Innovator™ Breadboard Pack Inventory List	
D. Resistors Table	
E. TI-84 Plus CE Program Menu Screenshots	
F. Path to STEM Unit Diagrams	

Introduction to the TI-Innovator™ System

TI PROFESSIONAL DEVELOPMENT

Participants will receive an overview of the TI-Innovator System.

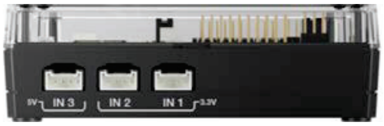
Step 1: Overview of the TI-Innovator™ Hub's features

- Inputs and outputs
- On-Board sensors
- Data Port (Micro USB Port)
- Power
- LED Indicators

Step 2: Distinguishing the difference between inputs and outputs

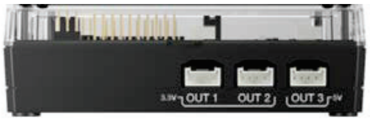
Left side - Three ports for collecting data or status from input modules:

- IN 1 and IN 2 provide 3.3 V of power
- IN 3 provides 5 V of power

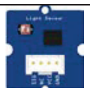


Right side – 3 ports for controlling output modules:

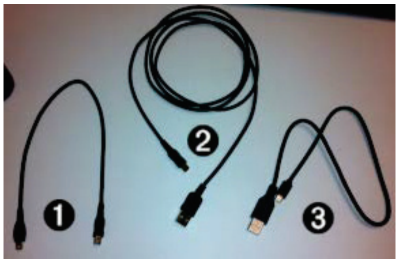
- OUT 1 and OUT 2 provide 3.3 V of power
- OUT 3 provides 5 V of power



Sample sensors include:

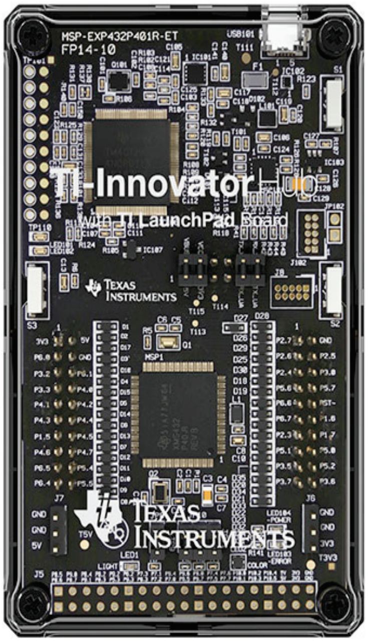
Servo Motor	Output Device	
White LED	Output Device	
Vibration Motor	Output Device	
Analog Light Sensor	Input Device	

Step 3: Cables

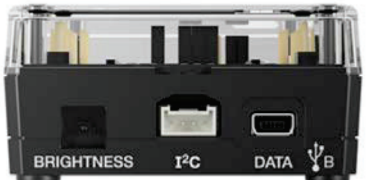


1. USB Unit-to-Unit (Mini-A to Mini-B) - Connects the Hub to a TI-84 Plus CE Graphing Calculator or a TI-Nspire™ CX Handheld.
2. USB Standard A to Mini-B - Connects the Hub to a computer running TI-Nspire™ CX Software.
3. USB Standard A to Micro - Connects the **PWR** port of the Hub to a TI approved power source required by some peripherals.

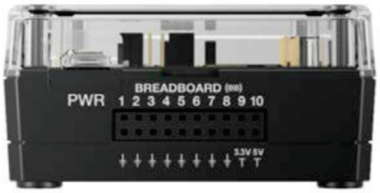
Step 4: On-Board components and indicators



1. A Light Brightness Sensor at the bottom of the Hub can be read as "BRIGHTNESS" in Hub command strings.
2. Red LED is addressable as "LIGHT" in Hub command strings.
3. Speaker (at back of Hub, not shown) labeled as SOUND is addressable as "SOUND" in Hub command strings.
4. Red-Green-Blue LED is addressable as "COLOR" in Hub command strings.
5. I^{2}C port connects to peripherals that use the I^{2}C communication protocol.
6. DATA Mini-B port, used with the appropriate cable, connects to a compatible graphing calculator or computer for data and power.



7. USB-Micro connector (PWR) for auxiliary power required by some components. Also used for updating the Hub internal software.
8. Breadboard Connector with 20 labeled pins for communication with connected components. A breadboard and jumper cables are included with the TI-Innovator™ Breadboard Pack, sold separately.



Vocabulary and Useful Terms for the TI-Innovator

TI-Innovator System: The power supply, cables, sensors, TI-Innovator Hub, and all components.
TI-Innovator Hub: The main box "Hub" for the TI-Innovator System.
LED: Light emitting diode.
Servo Motor: A motor that rotates or pushes part of a machine with a high level of precession.

This page intentionally left blank

Blinkin' Beacon

Student Activity

Name ________________

Class ________________

Objectives

This activity uses the TI-Innovator™ Hub to create a working circuit using a single Light Emitting Diode (LED).

Materials

- TI-Nspire™ CX or TI-84 CE Handheld
- TI-Innovator Hub with appropriate cable, mini A – Mini B
- 1 red or green LED

Part 1

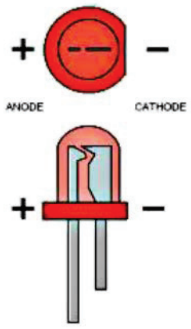
1. What is current?

2. What is a circuit?

Part 2

Set-up

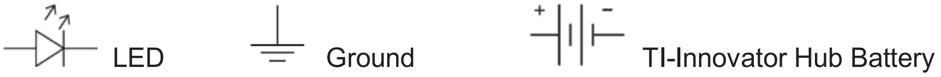
Connect the **B** end of the cable to the **B** data port of the Hub and then connect the **A** end of the cable to the handheld. When the Hub has power, a green LED power light will come on.

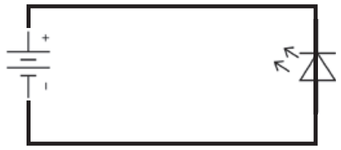


An LED has two legs, notice that one leg is shorter than the other, the short leg is called a **cathode or negative terminal.** The long leg is the **anode or positive terminal.**

Using the onboard breadboard feature of the Hub insert the LED **positive** terminal into **BB1** and the **negative** terminal into the ground port. ⏚

3. Circuit schematics are simplified and universal diagrams to illustrate an electrical circuit. Below is the diagram for the simple circuit that has been constructed. Using a different color pen or marker, mark the direction in which the current will flow.

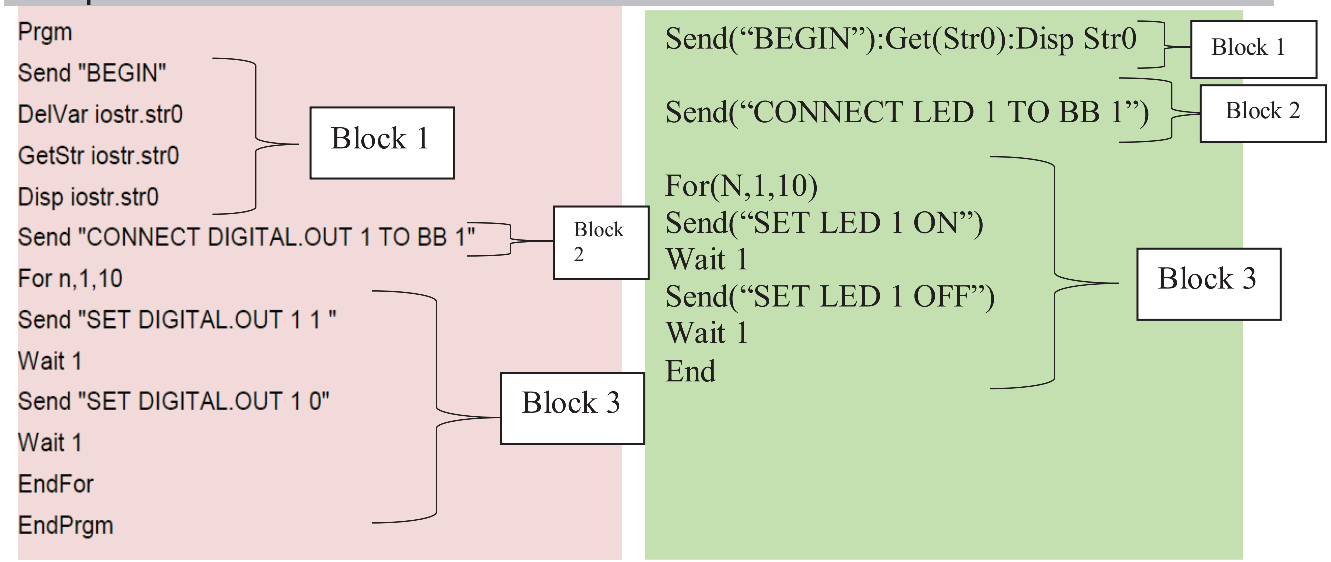




Making the LED light

On your handheld open a new program and name it Beacon. We are going to write the code to turn the LED on/off.

TI-Nspire CX Handheld Code **TI-84 CE Handheld Code**



```
Prgm
Send "BEGIN"
DelVar iostr.str0
GetStr iostr.str0
Disp iostr.str0
Send "CONNECT DIGITAL.OUT 1 TO BB 1"
For n,1,10
Send "SET DIGITAL.OUT 1 1 "
Wait 1
Send "SET DIGITAL.OUT 1 0"
Wait 1
EndFor
EndPrgm
```

```
Send(“BEGIN”):Get(Str0):Disp Str0

Send(“CONNECT LED 1 TO BB 1”)

For(N,1,10)
Send(“SET LED 1 ON”)
Wait 1
Send(“SET LED 1 OFF”)
Wait 1
End
```

Block 1
Block 2
Block 3
Block 1
Block 2
Block 3

Describe the function of each block of code:

Block 1:

____

____

____

Block 2:

____

____

____

Block 3:

____

____

____

TI-Nspire CX Code

To create a program on the TI-Nspire, from the Home screen select New Document, choose Program Editor, and then New. Name the program, and press OK. We will name this program beacon.

Block 1 –

A. Press the Menu key and choose **Hub, Manage** (if you can't find it, scroll down): **Send "BEGIN"**.

Block 2 –

A. Press the **Men**u key and choose **Hub, Send: "CONNECT – Output", DIGITAL.OUT**

B. Enter the number 1.

The number 1 will be used to make a connection between the number 1 and BB1. You can change this to a different integer value less than or equal to 8.

C. Return to the **Hub** menu and choose **Settings, TO**.

D. Return to the **Hub** menu and select **Ports** and choose **BB 1**.

Block 3 –

A. Press the **Menu** key and go to **Control, For…EndFor.**

B. After the **For** and within the commas, type n,1,10 (respectively).

C. Return to the **Hub** menu and select **Send "SET DIGITAL.OUT** ".

D. Enter the number 1 to connect to BB1.

Should be the same as the number from Block 2, Part B.

E. Add another 1 to send 3.3 volts (turn on) to BB1.

F. Enter the wait command by returning to the **Hub** command and choose **Wait.** Enter the number 1 for the time of one second.

G. Return to the **Hub** menu and select **Send "SET DIGITAL.OUT** ".

H. Enter the number 1 to connect to BB1.

Should be the same as the number from Block 2, Part B.

I. Add another 0 to send 0 volts (turn off) to BB1.

J. Enter the wait command by return to the **Hub** command and choose **Wait.** Enter the number 1 for the time of one second.

In order to store or compile the code for the Nspire to run the program, you must select Menu > Check Syntax & Store (CTRL B). This does not save the document. It compiles the code.

TI-84 CE Code

To create a program on the TI-84, press the program key (**prgm**), choose new, press enter, name the program using the alpha keys, and press enter. We will name this program BEACON.

Block 1-

A. Press the **prgm** key again to gain access to programming menus. Right arrow over to **HUB.**

B. Arrow down until you come to **manage** and press **enter**. Choose **Send "Begin.**

Block 2-

A. Press the **prgm** key and arrow over to **Hub, Send: "CONNECT – Output", LED.**

B. Enter the number 1.

The number 1 will be used to make a connection between the number 1 and BB1. You can change this to a different integer value less than 8.

C. Return to the **Hub** menu by pressing the **prgm** key and choose **Settings, TO**.

D. Return to the **Hub** menu and select **Ports** and choose **BB 1.**

TECH TIP

Pressing the **clear** key erases the entire line of code. To delete text, use the arrow keys to arrow over to what you want to delete and press the **delete** key.

Block 3 –

A. Press the **prgm** key and go to **Control, For(.**

B. After the **For(** and within commas, type n,1,10.

C. Return to the **Hub** menu and **Send ("SET LED.**

D. Enter the number 1 to connect to BB1.

E. Enter the state of the LED after the number 1 to ON.

F. Enter the wait command by returning to the **Hub** command and choosing **Wait.** Enter the number 1 for the time of one second.

G. Repeat the **Send(SET LED** command but set the state to zero 0FF.

H. Enter another **Wait** command with a time of 1 second.

I. To end the program, return to the Control menu and select **End.**

J. To execute the program press the 2nd and **MODE** keys to quit the program editor. Press the **prgm** key and notice that the BEACON program now shows up under the **prgm** menu. Arrow to the BEACON program, and press enter to select the program. This will automatically return you to the Home screen of the calculator. Press the enter key to execute the program.

Name _______________

Class _______________

Follow up questions

1. In what ways could you modify the code to do something different? (List at least three.)

2. Pick one of your modifications and create a LOGIC flow map for your code.

3. Try coding your modification. Does it work? If not, what errors were made?

This page intentionally left blank

Objectives

This activity uses the TI-Innovator™ Hub to create a working circuit using a single Light Emitting Diode (LED).

Materials

- TI-Nspire™ CX or TI-84 CE Handheld
- TI-Innovator Hub with appropriate cable, mini A – mini B
- 1 red or green LED

Part 1

1. What is current?

    Current is the flow of electrons from a positive to negative terminal.

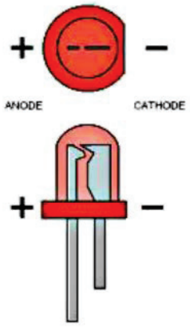
2. What is a circuit?

    A closed path (circular) which an electric current flows or may flow.

Part 2

Set-up

Connect the **B** end of the cable to the **B** data port of the Hub and then connect the **A** end of the cable to the handheld. When the Hub has power, a green LED power light will come on.

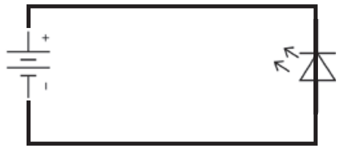


An LED has two legs, notice that one leg is shorter than the other, the short leg is called a **cathode or negative terminal.** The long leg is the **anode or positive terminal.**

Using the onboard breadboard feature of the Hub insert the LED **positive** terminal into **BB1** and the **negative** terminal into the ground port. ⏚

3. Circuit schematics are simplified and universal diagrams to illustrate an electrical circuit. Below is the diagram for the simple circuit that has been constructed. Using a different color pen or marker, mark the direction in which the current will flow.

LED
Ground
TI-Innovator Hub Battery



Making the LED light

On your handheld open a new program and name it Beacon. We are going to write the code to turn the LED on/off.

TI-Nspire CX Handheld Code

```
Prgm
Send "BEGIN"
DelVar iostr.str0
GetStr iostr.str0
Disp iostr.str0
Send "CONNECT DIGITAL.OUT 1 TO BB 1"
For n,1,10
Send "SET DIGITAL.OUT 1 1 "
Wait 1
Send "SET DIGITAL.OUT 1 0"
Wait 1
EndFor
EndPrgm
```

Block 1

Block 2

Block 3

TI-84 CE Handheld Code

```
Send("BEGIN"):Get(Str0):Disp Str0

Send("CONNECT LED 1 TO BB 1")

For(N,1,10)
Send("SET LED 1 ON")
Wait 1
Send("SET LED 1 OFF")
Wait 1
End
```

Block 1

Block 2

Block 3

Describe the function of each block of code:

Block 1:

This block of code allows for any commands, variables, or displays to be cleared within the Hub.

Block 2:

This block of code connects the LED as a digital device to the port of Breadboard 1.

Block 3:
This block of code is set in a FOR LOOP that will loop itself 10 times having the light turn on and off. The time for each cycle is 2 seconds, which would have a total time of 20 seconds.

TI-Nspire CX Code

Block 1 –

A. Press the Menu key and choose **Hub, Manage** (if you can't find it, scroll down): **Send "BEGIN"**.

Block 2 –

A. Press the **Men**u key and choose **Hub, Send: "CONNECT – Output", DIGITAL.OUT**.

B. Enter the number 1.

The number 1 will be used to make a connection between the number 1 and BB1. You can change this to a different integer value less than 8.

C. Return to the **Hub** menu and choose **Settings, TO**.

D. Return to the **Hub** menu and select **Ports** and choose **BB 1**.

Block 3 –

A. Press the **Menu** key and go to **Control, For…EndFor**.

B. After the **For** and within the commas, type n,1,10 (respectively).

C. Return to the **Hub** menu and select **Send "SET DIGITAL.OUT** ".

D. Enter the number 1 to connect to BB1.

Should be the same as the number from Block 2, Part B.

E. Add another 1 to send 3.3 volts (turn on) to BB1.

F. Enter the wait command by returning to the **Hub** command and choose **Wait.** Enter the number 1 for the time of one second.

G. Return to the **Hub** menu and select **Send "SET DIGITAL.OUT** ".

H. Enter the number 1 to connect to BB1.

Should be the same as the number from Block 2, Part B.

I. Add another 0 to send 0 volts (turn off) to BB1.

J. Enter the wait command by return to the **Hub** command and choose **Wait.** Enter the number 1 for the time of one second.

In order to store or compile the code for the Nspire to run the program, you must select Menu > Check Syntax & Store (CTRL B). This does not save the document. It compiles the code.

TI-84 CE Code

To create a program on the TI-84, press the program key (**prgm**), choose new, press enter, name the program using the alpha keys, and press enter. We will name this program BEACON.

Block 1-

A. Press the **prgm** key again to gain access to programming menus. Right arrow over to **HUB.**

B. Arrow down until you come to **manage** and press **enter**. Choose **Send "Begin**.

Block 2-

A. Press the **prgm** key and arrow over to **Hub, Send: "CONNECT – Output", LED**.

B. Enter the number 1.

The number 1 will be used to make a connection between the number 1 and BB1. You can change this to a different integer value less than 8.

C. Return to the **Hub** menu by pressing the **prgm** key and choose **Settings, TO**.

D. Return to the **Hub** menu and select **Ports** and choose **BB 1.**

TECH TIP

Pressing the **clear** key erases the entire line of code. To delete text, use the arrow keys to arrow over to what you want to delete and press the **delete** key.

Block 3 –

A. Press the **prgm** key and go to **Control, For(**.

B. After the **For(** and within commas, type n,1,10.

C. Return to the **Hub** menu and **Send ("SET LED**.

D. Enter the number 1 to connect to BB1.

E. Enter the state of the LED after the number 1 to ON.

F. Enter the wait command by returning to the **Hub** command and choosing **Wait.** Enter the number 1 for the time of one second.

G. Repeat the **Send(SET LED** command but set the state to zero 0FF.

H. Enter another **Wait** command with a time of 1 second.

I. To end the program, return to the Control menu and select **End**.

J. To execute the program press the 2nd and MODE keys to quit the program editor. Press the **prgm** key and notice that the BEACON program now shows up under the Prgm menu. Arrow to the BEACON program, and press enter to select the program. This will automatically return you to the Home screen of the calculator. Press the enter key to execute the program.

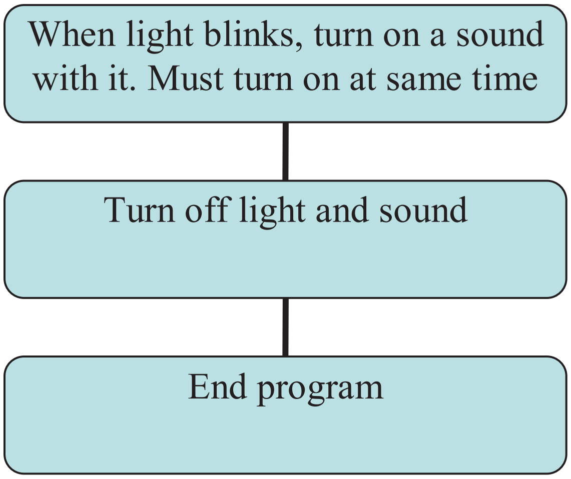
Follow up questions

1. In what ways could you modify the code to do something different? (List at least three.)

   The code could be modified to add in multiple lights, play a sound with the blink, give off an SOS signal…answers may vary.

2. Pick one of your modifications and create a LOGIC flow map for your code.

   A sample flow chart would look like: **play a sound**



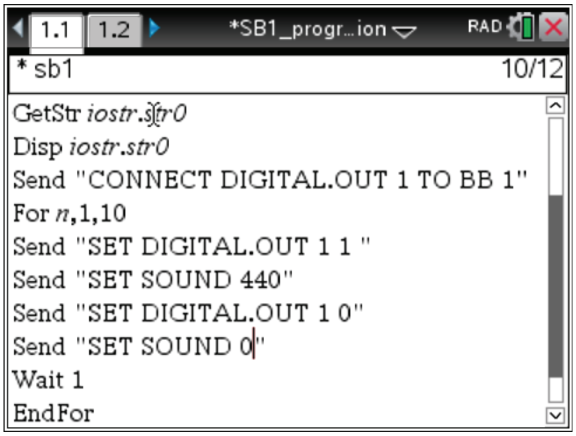
When light blinks, turn on a sound with it. Must turn on at same time

Turn off light and sound

End program

3. Try coding your modification. Does it work? If not, what errors were made?

   Possible answer:



```
GetStr iostr.str0
Disp iostr.str0
Send "CONNECT DIGITAL.OUT 1 TO BB 1"
For n,1,10
Send "SET DIGITAL.OUT 1 1 "
Send "SET SOUND 440"
Send "SET DIGITAL.OUT 1 0"
Send "SET SOUND 0"
Wait 1
EndFor
```

This page intentionally left blank

To create a program on the TI-84, press the program key (**prgm**), choose new, press enter, name the program using the alpha keys, and press enter. We will name this program BEACON.

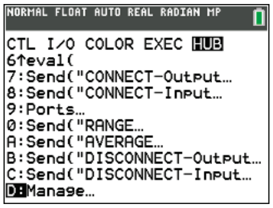
TECH TIP
Pressing the **clear** key erases the entire line of code. To delete text, use the arrow keys to arrow over to what you want to delete and press the delete (**del**) key.

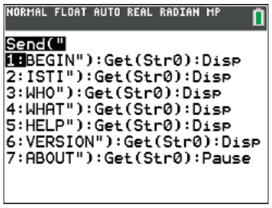
Block 1-

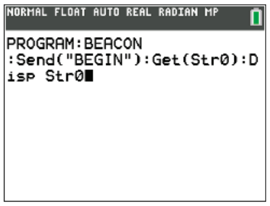
A. Press the **prgm** key again to gain access to programming menus. Right arrow over to **HUB**.

B. Arrow down (or up) until you come to **manage** and press **enter**, choose **Send("Begin"):Get(STr0):Disp**.

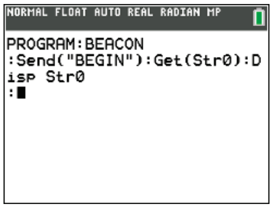
C. Press enter, then press enter again once the command is pasted to place the cursor on the next line.







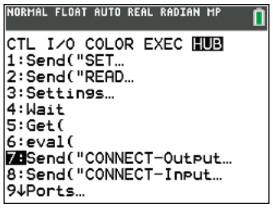
```
PROGRAM:BEACON
:Send("BEGIN"):Get(Str0):D
isp Str0
```

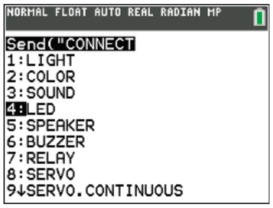


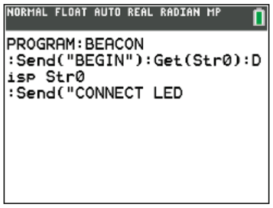
```
PROGRAM:BEACON
:Send("BEGIN"):Get(Str0):D
isp Str0
:
```

Block 2-

A. Press the **pgrm** key and arrow over to **Hub, Send: "CONNECT – Output"** and then select **LED**.







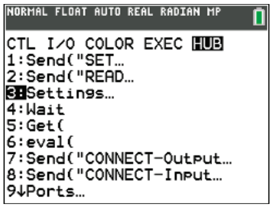
```
PROGRAM:BEACON
:Send("BEGIN"):Get(Str0):D
isp Str0
:Send("CONNECT LED
```

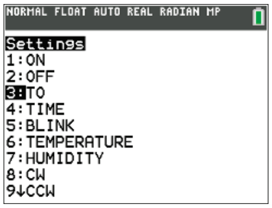
B. Enter the number 1.

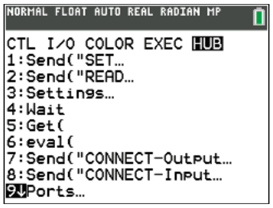
The number 1 will be used to make a connection between the number 1 and BB1. You can change this to a different integer value equal to or less than 8.

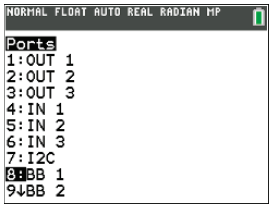
C. Return to the **Hub** menu by pressing the **prgm** key and choose **Settings** and then **TO**.

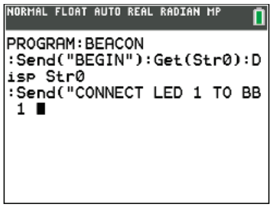
D. Return to the **Hub** menu and select **Ports** and choose **BB 1.** Complete the command by typing quotes (alpha and the + key) and closing the parentheses. Press enter to move the cursor to the next line ready for the next command.



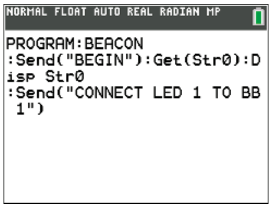




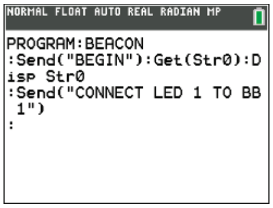




```
PROGRAM:BEACON
:Send("BEGIN"):Get(Str0):D
isp Str0
:Send("CONNECT LED 1 TO BB
 1 ∎
```



```
PROGRAM:BEACON
:Send("BEGIN"):Get(Str0):D
isp Str0
:Send("CONNECT LED 1 TO BB
 1")
```

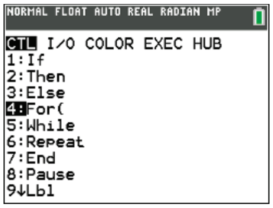


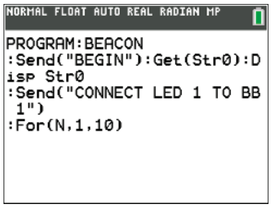
```
PROGRAM:BEACON
:Send("BEGIN"):Get(Str0):D
isp Str0
:Send("CONNECT LED 1 TO BB
 1")
:
```

Block 3 –

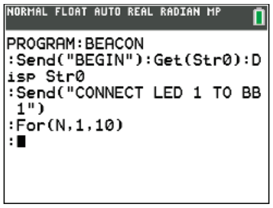
A. Press the **prgm** key and go to **Control, (CTL)**. Choose **For(.**

B. After the **For(** and within commas, type n,1,10, and then close the parentheses.





```
PROGRAM:BEACON
:Send("BEGIN"):Get(Str0):D
isp Str0
:Send("CONNECT LED 1 TO BB
 1")
:For(N,1,10)
```

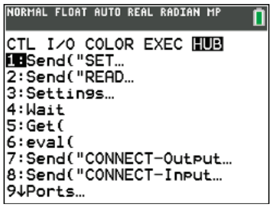


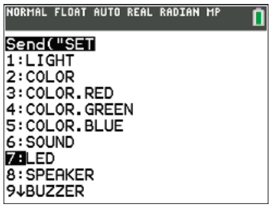
```
PROGRAM:BEACON
:Send("BEGIN"):Get(Str0):D
isp Str0
:Send("CONNECT LED 1 TO BB
 1")
:For(N,1,10)
:∎
```

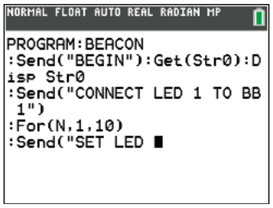
C. Return to the **Hub** menu and **Send (Set** and choose **LED**. Press **enter**.

D. Enter the number 1 to connect to BB1.

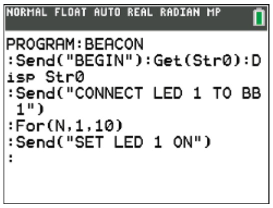
E. Enter the state of the LED after the number 1 to ON. Type the word ON using the alpha keys. Complete the command by typing quotes (alpha and the + key) and closing the parentheses. Press enter to move the cursor to the next line ready for the next command.







```
PROGRAM:BEACON
:Send("BEGIN"):Get(Str0):D
isp Str0
:Send("CONNECT LED 1 TO BB
 1")
:For(N,1,10)
:Send("SET LED ∎
```



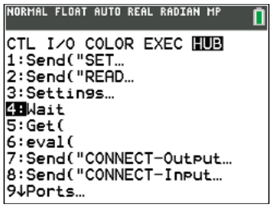
```
PROGRAM:BEACON
:Send("BEGIN"):Get(Str0):D
isp Str0
:Send("CONNECT LED 1 TO BB
 1")
:For(N,1,10)
:Send("SET LED 1 ON")
:
```

F. Enter the wait command by returning to the **Hub** command and choose **Wait.** Enter the number 1 for the time of one second.

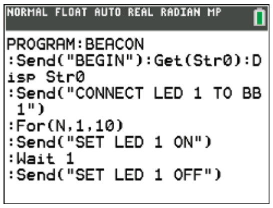
G. Repeat the **Send(SET LED** command but set the state to 0FF. (See steps C,D,E above.)

H. Enter another **Wait** command with a time of 1 second.

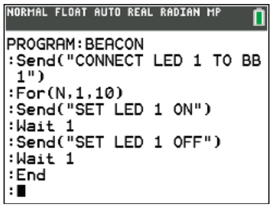
I. To end the program return to the **Control (CTL)** menu and select **End**.



```
NORMAL FLOAT AUTO REAL RADIAN MP
CTL I/O COLOR EXEC HUB
1:Send("SET…
2:Send("READ…
3:Settings…
4:Wait
5:Get(
6:eval(
7:Send("CONNECT-Output…
8:Send("CONNECT-Input…
9↓Ports…
```



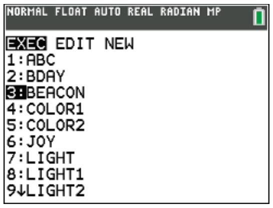
```
NORMAL FLOAT AUTO REAL RADIAN MP
PROGRAM:BEACON
:Send("BEGIN"):Get(Str0):D
isp Str0
:Send("CONNECT LED 1 TO BB
 1")
:For(N,1,10)
:Send("SET LED 1 ON")
:Wait 1
:Send("SET LED 1 OFF")
```



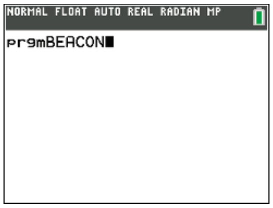
```
NORMAL FLOAT AUTO REAL RADIAN MP
PROGRAM:BEACON
:Send("CONNECT LED 1 TO BB
 1")
:For(N,1,10)
:Send("SET LED 1 ON")
:Wait 1
:Send("SET LED 1 OFF")
:Wait 1
:End
:■
```

Note: This screenshot does not include the entire program. Scroll if needed on your calculator to view.

J. To execute the program, press the 2nd then MODE keys (**quit**) to quit the program editor. Press the **pgrm** key and notice that the Beacon program now shows up under the **pgrm** menu (you may have more or fewer programs than the screenshot). Select BEACON, and press enter. This will paste your program to the home screen of the calculator. Press the enter key to execute the program. If you need to edit your program, press the program key, highlight the program, and select edit. Press enter to access your program for editing.



```
NORMAL FLOAT AUTO REAL RADIAN MP
EXEC EDIT NEW
1:ABC
2:BDAY
3:BEACON
4:COLOR1
5:COLOR2
6:JOY
7:LIGHT
8:LIGHT1
9↓LIGHT2
```



```
NORMAL FLOAT AUTO REAL RADIAN MP
prgmBEACON■
```

This page intentionally left blank

Skill Builder: Blink

Overview:

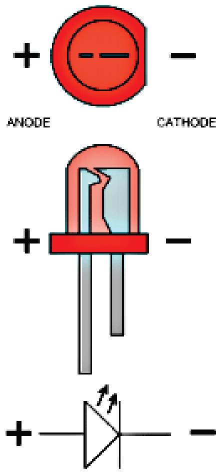
In this lesson, students will learn how to use a digital output pin on the breadboard connector of the TI-Innovator™ Hub. Students will learn what an electrical circuit is and how to represent that circuit with a schematic diagram. Students are also introduced to the resistor value color code table.

Goals:

1. Build an electrical circuit with a Red LED and 100 Ohm resistor on a breadboard and connect it to the TI-Innovator Hub.
2. Author a TI-Basic program on a TI-84 Plus CE calculator that will control the digital output on BB1 of the TI-Innovator Hub to repeatedly turn the Red LED on and off.
3. Interpret and trace the flow of current in a circuit.
4. Draw and interpret a schematic diagram of a circuit.
5. Read the resistance color code of a resistor.

Background:

An electrical circuit contains materials called conductors that allow electrons to flow through the circuit when the electrons are pushed by voltage. These conductive materials are built into parts such as wires, resistors and breadboards.



The electrons flow through a circuit, starting at negative (-) and ending at positive (+). A power source is needed to push the electrons through the circuit, usually; this push comes from a battery or a power source like the TI-Innovator Hub. A digital output pin on the TI-Innovator Hub is an electrical output connection that exists in one of two states: on or off. In the on state, which is represented by the number one (1), the pin is held at 3.3 volts. In the off state, which is represented as a zero (0), the pin is held at 0 volts.

The circuit you will build turns on a small light called an LED. An LED is a Light Emitting Diode, and it makes light as electrons are pushed through a special material called a semiconductor. The energy from the motion of the moving electron is converted into light. An LED has polarity. One lead is positive (+) and is referred to as the anode, and the other lead is negative (-) and is referred to as the cathode. Refer to the LED diagram on the right to distinguish between the anode and cathode leads.

It is a good practice to use a current-limiting resistor in an LED circuit. This resistor prevents too much current from flowing through the device and breaking (burning) the LED. Typically, the current-limiting resistor is placed in series with the cathode (-) lead.

Resistors vary in the amount of resistance to the flow of current. There are four colored bands marked on a resistor. The first and second band colors represent the first and second digits in the resistance value. The third colored band represents the factor the first two numbers are multiplied by to equal the resistor value. The fourth band is the manufactured tolerance and is not important in this lesson. The chart on the right can be used to determine the resistance value of resistors.

Color	Number	Multiplier
Black	0	1 or (10^0)
Brown	1	10 or (10^1)
Red	2	100 or (10^2)
Orange	3	1000 or (10^3)
Yellow	4	10000 or (10^4)
Green	5	100000 or (10^5)
Blue	6	1000000 (10^6)
Violet	7	10000000 or (10^7)
Gray	8	100000000 or (10^8)
White	9	1000000000 or (10^9)

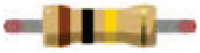
Practice:

Example:

For the 100 Ohm (Ω) resistor in this activity, the first and second colored bands are brown (1) and black (0), and the third band is brown 10^1(10). So the calculation would be:

10 x 10 = 100

Practice:



Find the value of the resistor on the left. The first color is brown; the second is black; and the third is yellow.

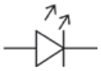
Answer: **10x10000 = 100,000 or 100k ohm**

Schematic Diagrams:

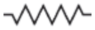
Below and to the right is the schematic diagram for this activity. Work with students to follow the flow of current through the diagram from the breadboard ground pin, through the current-limiting resistor, then through the LED, and finally back to the BB1 pin on the breadboard connector.

Some schematic symbols:

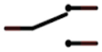
LED



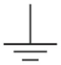
Resistor



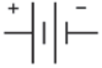
Switch

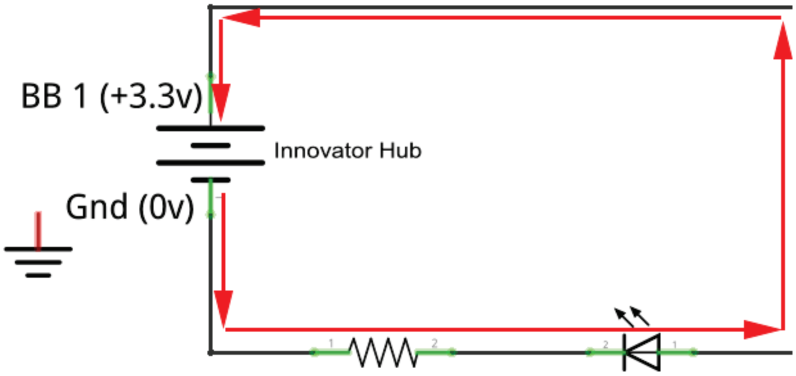


Ground



TI-Innovator Hub Battery

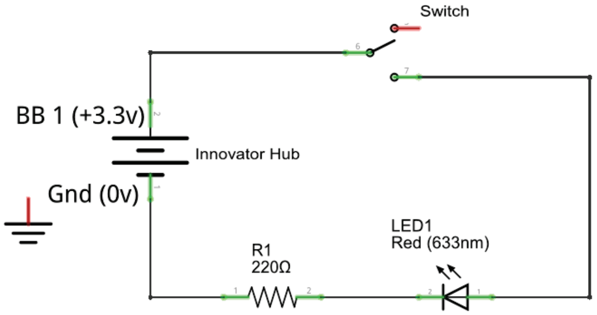




BB1 is a digital output pin on the TI-Innovator Hub and Gnd is the ground on the TI-Innovator Hub.

Practice:

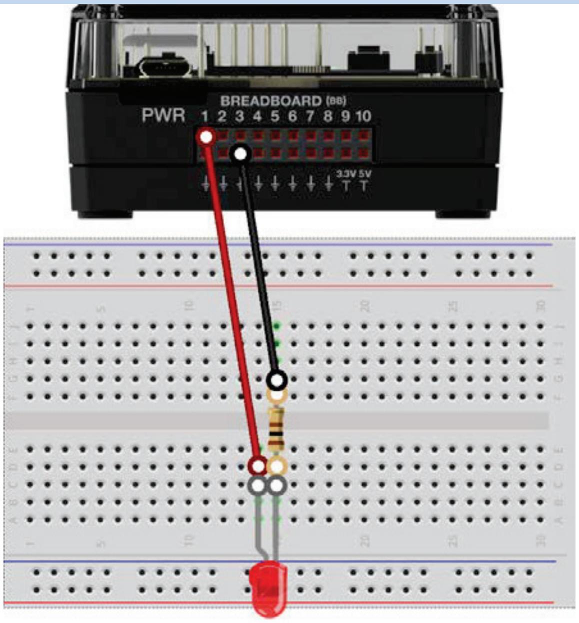
Using the example above, have students draw the same schematic but include a switch that will turn the LED on or off. Have the students trace the current flow through the schematic as well as label each component (resistor, battery, LED, and switch).



Materials and Tools:

- TI-84 Plus CE
- TI-Innovator™ Hub with USB Cable
- TI-Innovator Breadboard Pack:
  - Breadboard
  - Male to Male jumper Cables
  - Red LED
  - Resistor 100 Ohm (brown, black, brown)

Build the Hardware:



Assemble the circuit in the diagram by following these steps:

1. Insert a Red LED into the breadboard as shown on right.
   - Be sure each leg is in its own column so that the two legs are not electrically connected.
2. Use a red Male to Male Jumper Cable to connect the anode (+) leg of the Red LED to BB1 on the breadboard connector of the TI-Innovator Hub.
3. Place the 100 Ohm resistor into the same column as the cathode (-) of the Red LED.
4. Bend the 100 Ohm resistor so that it bridges the gap in the breadboard, and insert the opposite lead of the 100 Ohm resistor into any hole in the upper half of the breadboard.
   - Remember, the upper half and lower half of the breadboard are separated by the gap and are **not** electrically connected.
5. Insert a black Male to Male Jumper Cable into any hole in the same row as the lead of the 100 Ohm resistor. Connect the opposite end of the black Male to Male Jumper Cable into any ground pin on the breadboard connector of the TI-Innovator Hub.
6. Trace the flow of current through the circuit to ensure it is complete.
7. Plug the B end of the unit-to-unit USB cable into the TI-Innovator Hub and then the A end into the handheld device.

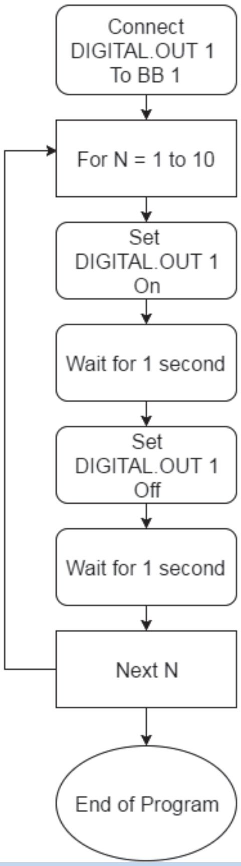
Teacher Tip: It is important to demonstrate to students how to be methodical when checking for hardware and software errors. Show them how to trace the current through the circuit and the requirement for a complete circuit from source to ground. A common problem is a reversed polarity on the Red LED or an incomplete circuit through the resistor.

Write the Software for the TI-84 Plus CE:

Task: Write a TI-BASIC program on the TI 84 Plus CE that will blink a red LED connected to pin 1 on the breadboard connector of the TI-Innovator Hub ten times with a 1 second pause between on and off.

Code for the TI-84 Plus CE:

```
Send("BEGIN"):Get(Str0):Disp Str0
Send("CONNECT LED 1 TO BB 1")
For(N,1,10)
Send("SET LED 1 ON")
Wait 1
Send("SET LED 1 OFF")
Wait 1
End
```



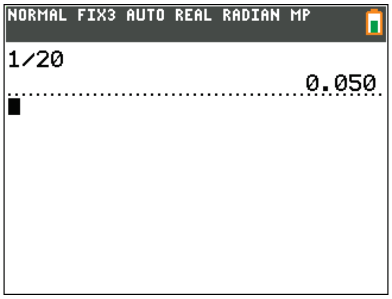
Extra for Experts:

Periodic phenomenon can be described by frequency; that is, how many times the event happens per second. Alternatively, the phenomenon can also be described by period; that is, how long the event lasts. The unit for frequency is in honor of Heinrich Hertz, the discoverer of electromagnetic waves. The unit Hertz (Hz) is the number of cycles per second of a periodic phenomenon such as a wave or a rotating wheel.

Frequency (f) is measured in unit of Hz.
Period (T) is measured in units of seconds.

$$f = \frac{1}{T}$$

Calculate the period (T) of a frequency (f) of 20 Hz on your calculator. Try a few more for practice.


```
NORMAL FIX3 AUTO REAL RADIAN MP
1/20
                              0.050
```

a) 5 Hz
Answer: **1/5 = .2 sec**

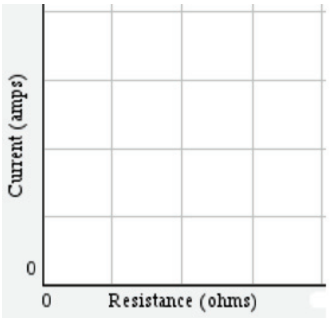
b) 10 Hz
Answer: **1/10 = .1 sec**

c) 75 Hz
Answer: **1/75 = .0133 sec**

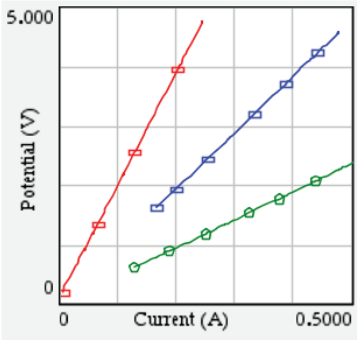
d) 4,000,000 Hz
Answer: **1 /4,000,000 = .00000025 sec**

1. How does changing the value of the resistance change the brightness of the bulb? Try to do this experimentally. Describe how you could change the resistance. Draw a schematic of your new circuit.

2. Sketch a graph of I (current) vs. R (resistance). Would this be a direct or inverse relationship? The product of current, I, and resistance, R, is the voltage, V, of a circuit. Write this as an equation and solve for the appropriate letter to explain why you think the relationship between current and resistance is direct or inverse.



3. The graph bellows shows another situation where the current, measured in amperes, and the potential difference or voltage, measured in volts, was measured for 3 different resistors. Since you know V=IR, approximate the resistance for each of the graphs below.



4. The frequency of the light turning on and off is originally 0.5 Hz (1 blink every two seconds). How can the program be modified so the frequency is 0.25 Hz? 2 Hz? Justify your answer.

5. If the original blinking in the program is set for 10 loops with a period of 2 s, how many seconds will the program run? How can the program be modified to make 5 loops?

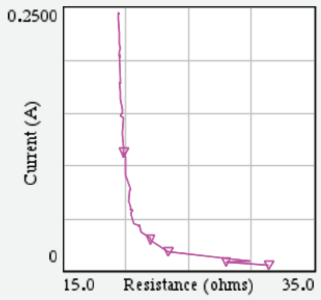
This page intentionally left blank

1. How does changing the value of the resistance change the brightness of the bulb? Try to do this experimentally. Describe how you could change the resistance. Draw a schematic of your new circuit.

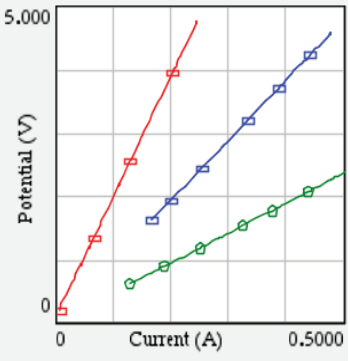
   When the current increases, the bulb gets brighter, and when the current decreases, the bulb is dimmer. Vary the resistance by either adding a second resistor in the same column as the first resistor and the anode of the LED, or changing the resistor to one of a different value and determine if the bulb got brighter or dimmer. Diagrams may vary depending on if they add more resistors or simply change the resistor.

2. Sketch a graph of I (current) vs. R (resistance). Would this be a direct or inverse relationship? The product of current, I, and resistance, R, is the voltage, V, of a circuit. Write this as an equation and solve for the appropriate letter to explain why you think the relationship between current and resistance is direct or inverse.

   Since V = IR, then I=V/R. So the relationship between current and resistance is INVERSE.
   The graph the right is an example of an inverse graph from actual data collected by a student experiment.



3. The graph bellows shows another situation where the current, measured in amperes, and the potential difference or voltage, measured in volts, was measured for 3 different resistors. Since you know V=IR, approximate the resistance for each of the graphs below.



The 1st graph (red) has a slope = (4-2)/(0.2-0.1)=2/.1= 20 ohms
The 2nd graph's (blue) slope is about = (3.8-2.8)/(0.4-0.3) = 10 ohms
The 3rd graph's (green) resistance is approx. (2-1)/(0.4-0.2) = 5 Ω

4. The frequency of the light turning on and off is originally 0.5 Hz (1 blink every two seconds). How can the program be modified so the frequency is 0.25 Hz? 2 Hz? Justify your answer.

   The Wait command is set for 1 second. Change this to 2 seconds to have a smaller frequency of 0.25 Hz. The period, T, and frequency, f, are inversely related. T=1/f. To make the frequency increase to 2 Hz, decrease the wait time to 0.5 seconds. So the program would be changed to Wait 2.

5. If the original blinking in the program is set for 10 loops with a period of 2 s, how many seconds will the program run? How can the program be modified to make 5 loops?

   20 seconds, change the for loop from For n,1,10 to n,1,5.

This page intentionally left blank

Skill Builder: Switch

Overview:

Students will learn how to read a digital input pin on the breadboard connector of the TI-Innovator™ Hub. The digital input pin will sense the on or off state of a toggle switch used to switch the power to an LED circuit on or off. A program will be written to read the state of the digital input pin and then, based on the state of the pin, display an appropriate message.

Goals:

1. Build an LED circuit with a toggle (power) switch on a breadboard.
2. Connect a digital input pin on the TI-Innovator Hub to the circuit switch to sense if the LED is on or off.
3. Create a TI-Basic program on a handheld device that displays an appropriate message based upon if the power switch is either on or off.

Background:

Connecting the BB 1 digital input pin to the power SPDT Slide Switch of an LED circuit sends a digital input signal to the TI-Innovator Hub.

The switch will allow the user to change between an open circuit and a closed circuit. A switch in the OFF state creates an open circuit and the LED will not turn on because electrons cannot move through the circuit. A switch in the ON state creates a closed circuit and allows electrons to flow, thus, turning on the LED.

Using TI-BASIC, the input pin can be read just like a variable. When 3.3 Volts is connected to the digital input pin, the variable value will read as a 1. When 0 Volts (ground) is connected to the pin, the value will read as 0. An “if -then” statement can be used to change the program action based upon the pin state.

A real-world application of a digital switch is a button on a game control pad that is programmed to make an animated character jump in a game when the button is pressed.

Flowcharts are often used to create or design computer programs. A flowchart illustrating a program algorithm was seen in the Unit 1 Skill Builder.

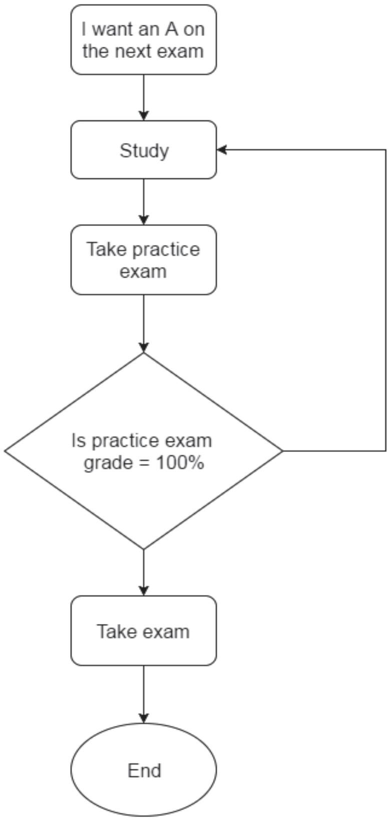
An algorithm is a series of steps that must be followed to complete a task and is used to organize the flow of a computer program. They are written in no specific computer language so anyone can understand them. Computer algorithms are often illustrated using flowcharts that contain rectangles, circles, and diamonds. Rectangles are program statements, while diamonds are “if -then” decision statements, and circles are the end of a program. Arrows are used in the flowchart to show how the program flows from start to finish.

Using these flowchart symbols, a programmer can describe a computer program in a way similar to how a football coach describes a play using Xs and Os to represent players on both sides and arrows showing individual player movements during the play.

Practice:

Task: Have students draw a flowchart to illustrate how to prepare well for a test.

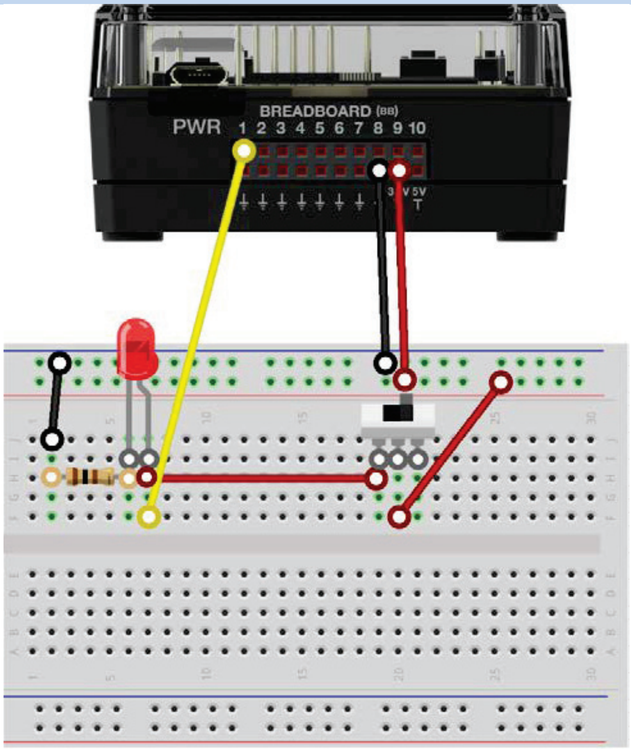
Sample Answer:



Materials and Tools:

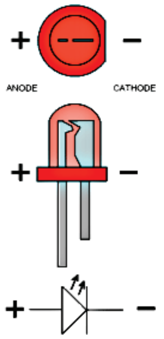
- TI-84 Plus CE
- TI-Innovator™ Hub with USB Cable
- Needle Nose pliers (optional)
- Wire nippers (optional)
- TI-Innovator Breadboard Pack:
  - Breadboard
  - Male to Male Jumper Cables
  - Resistor 100 Ohm (brown, black, brown)
  - SPDT Slide Switch
  - Red LED

Build the Hardware:



Assemble the circuit in the diagram on the right, following these steps:

1. Place an SPDT Slide Switch and a Red LED on the breadboard in the locations noted in the figure to the right.
2. Connect a red Male to Male Jumper Cable from the 3.3V on the TI-Innovator Hub to the red 3.3V bus on the breadboard.
3. Connect a black Male to Male Jumper Cable from any ground on the TI-Innovator Hub to the blue ground bus on the breadboard.
4. Connect a red Male to Male Jumper Cable from the center leg of the SPDT Slide Switch to the red 3.3V bus on the breadboard.
5. Use a red Male to Male Jumper Cable to connect an outer leg of the switch to the anode (+) of the Red LED.
6. Connect a 100 Ohm Resistor to the cathode(-) leg of the Red LED.
7. Use a black Male to Male Jumper Cable to connect the opposite end of the 100 Ohm Resistor to the blue ground bus on the breadboard.
8. Use a yellow Male to Male Jumper Cable to connect the anode (+) leg of the Red LED to BB1 on the TI-Innovator Hub. This point will be held at 3.3V when the SPDT Slide Switch is turned on and will be held at 0V when turned off.
   - The digital input on BB1 can detect this on or off state.
9. **9.** Plug the "B" end of the "unit to unit" USB cable into the TI-Innovator Hub and the "A" end into the handheld device.



Tech Tip: Vertical columns on the breadboard in the above picture are electrically connected. All rows as well as those separated by the gap in the board are not electrically connected.

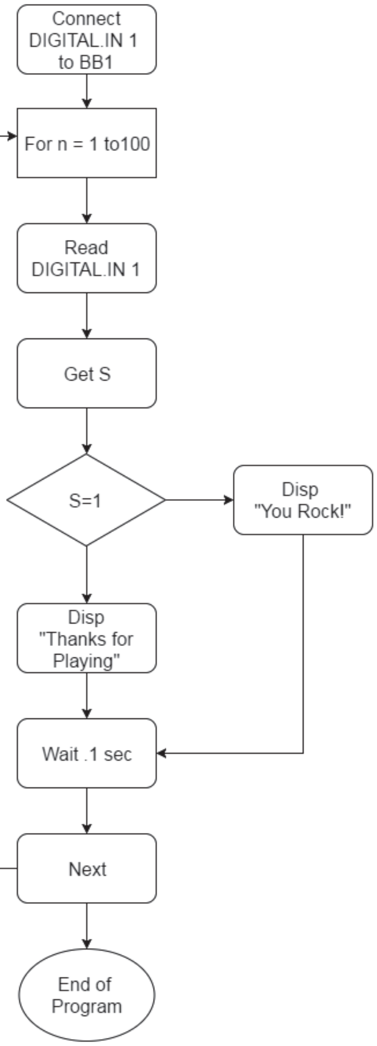
In Unit 1 Skill Builder, we followed the flow of current through a schematic with a switch. Ensure that the newly-created circuit has the appropriate location of the switch, battery, and ground by following the flow of current as in Unit 1.

Write the Software for the TI-84 Plus CE:

Task: Write a TI-BASIC program on the handheld that will read a red LED connected to a switch and pin 1 on the breadboard connector of the TI-Innovator Hub. If the switch is off, 0, display “Thanks for playing!” If the switch is on, 1, display “You rock!”

Code for the TI-84 Plus CE:

```
Send("BEGIN"):Get(Str0):Disp Str0
Send("CONNECT DIGITAL.IN 1 TO BB 1")
For(N,1,200)
Send("READ DIGITAL.IN 1")
Get(A)
If A=1
Then
Disp "YOU ROCK"
Else
Disp "THANKS FOR PLAYING"
End
End
```



Extra for Experts:

Background: Boolean algebra is the branch of mathematics dealing with logic. Variables are either TRUE or FALSE. Boolean operations are words like "**and**", "**or**" and "**not**". In digital electronics and computer science, TRUE and ON are represented with a 1, while FALSE and OFF are represented with a 0. Since computer circuits use transistors that are ON and OFF, that is digital, and software uses 1s and 0s, that is binary, Boolean algebra is useful to make logical decisions, such as the moves of a game piece in computer game.

Task: Add an additional SPDT Slide Switch to BB2 as a second digital input. Write a program that reads and displays the states of both switches and then based on those states, displays all the Boolean operations in TI BASIC, (**and**, **or**, **xor**) that are true and the **not** of each input.

Hint: the Boolean operations are under TEST then LOGIC.

Truth Table for Digital Inputs BB1 and BB2

BB 1	BB 2	BB1 **and** BB2	BB1 **or** BB2	BB1 **xor** BB2
ON (1)	ON (1)	TRUE	TRUE	FALSE
ON (1)	OFF (0)	FALSE	TRUE	TRUE
OFF (0)	ON (1)	FALSE	TRUE	TRUE
OFF (0)	OFF (0)	FALSE	FALSE	FALSE

Truth Table for Digital Input BB1

BB1	Not(BB1)
ON (1)	0
OFF(0)	1

Sample Extra for Experts Program:

```
Send("BEGIN"):Get(Str0):Disp Str0
Send("CONNECT DIGITAL.IN 1 TO BB 1 ")
Send("CONNECT DIGITAL.IN 2 TO BB 2 ")
For(N,1,200)
Send("READ DIGITAL.IN 1")
Get(A)
Send("READ DIGITAL.IN 2")
Get(B)
Disp "A is:"
Disp A
Disp "B is:"
Disp B
If A and B
Then
Disp "AND"
End
If A or B
Then
Disp "OR"
End
If A xor B
Then
Disp "XOR"
End
Disp "NOT(A)"
Disp not(A)
Disp "NOT(B)"
Disp not(B)

Wait 1
ClrHome
End
```

This page intentionally left blank

1. Is the signal continuous or discrete?

2. Is the signal digital or analog? What does it mean for it to be a digital signal and an analog signal?

3. Move the switches and jumper cables to different locations so the LED can still be controlled by the switch.

4. What happens to the LED if you increase the value of the resistor? Where could this be applied or necessary?

5. Provide examples of switches that are used every day.

6. Vary the code so that a sound will play when the LED is in the ON state. (The RGB LED will flicker in some cases when a sound is played.)

This page intentionally left blank

1. Is the signal continuous or discrete?

   Since we are using a digital output which can take on values of zero and 1, then this is a discrete signal.

2. Is the signal digital or analog? What does it mean for it to be a digital signal and an analog signal?

   The signal is digital since we can send only two values (zero for 0 voltage or one for 3.3 voltage).

3. Move the switches and jumper cables to different locations so the LED can still be controlled by the switch.

   Answers will vary. To help students, have them create a schematic and make sure the circuit is closed. The point is to be able to move the jumper cables and the switch and still make the switch turn on or turn off the LED.

4. What happens to the LED if you increase the value of the resistor? Where could this be applied or necessary?

   *Changing the resistor will impact the brightness of the LED. This idea deals with Ohm's Law (V=I*R). In terms for real-world application, a dimmer increases the light output by reducing the resistor (which in turn will increase the current flow).*

5. Provide examples of switches that are used every day.

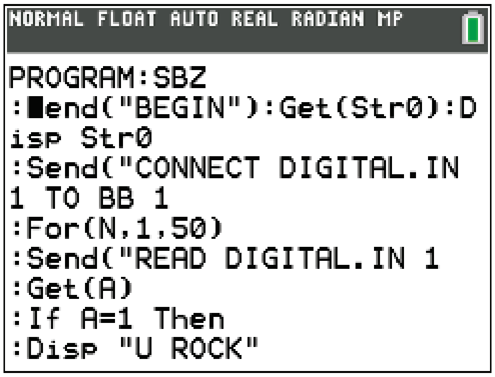
   Light switches, game controllers when you press A or B, keys on your keyboard and many other examples of switches in everyday lives that send some voltage when pressed/switched to zero voltage when off or not pressed.

6. Vary the code so that a sound will play when the LED is in the ON state. (The RGB LED will flicker in some cases when a sound is played.)

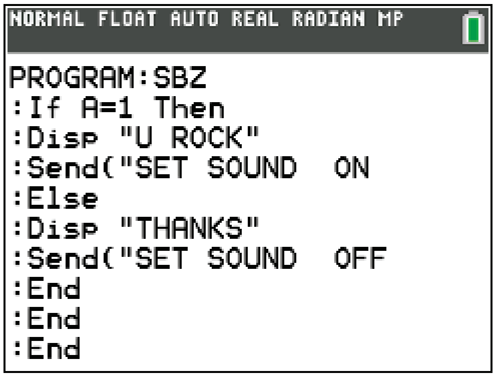
TI-Nspire™ CX Code

```
Define sb2()=
Prgm:Send "BEGIN":DelVar iostr.str0:GetStr
iostr.str0:Disp iostr.str0:Send "CONNECT
DIGITAL.IN 1 TO BB 1":For n,1,50:Send
"READ DIGITAL.IN 1":Get a:If a=1 Then:
Disp "YOU ROCK": Send "SET SOUND
ON":Else:Disp "THANKS FOR
PLAYING":Send "SET SOUND
OFF":EndIf:EndFor:EndPrgm
```

TI-84 CE Code



```
PROGRAM:SBZ
:█end("BEGIN"):Get(Str0):D
isp Str0
:Send("CONNECT DIGITAL.IN
1 TO BB 1
:For(N,1,50)
:Send("READ DIGITAL.IN 1
:Get(A)
:If A=1 Then
:Disp "U ROCK"
```



```
PROGRAM:SBZ
:If A=1 Then
:Disp "U ROCK"
:Send("SET SOUND  ON
:Else
:Disp "THANKS"
:Send("SET SOUND  OFF
:End
:End
:End
```

Overview:	Goals:
Students will apply their skill of digital input to detect when a banana is touched to trigger an alarm sound generated by the internal speaker on the TI-Innovator™ Hub. An electronic switch is built using a TTL Power MOSFET, commonly called an FET, with a fresh banana connected to the gate of the transistor. The electronic switch, once built on the breadboard, works just like the mechanical switch used in the related skill builder.	1. Assemble an electronic switch on a breadboard using a field effect transistor (TTL Power MOSFET). 2. Understand how a TTL Power MOSFET can be used in a circuit. 3. Apply the concept of getting digital input on the TI-Innovator Hub. 4. Author a TI-Basic program that polls the state of the digital input on BB1 and plays a tone on the internal speaker when the state of the switch is ON.

Background:

A Field Effect Transistor (FET) is a device that changes the conductivity of a circuit based on the voltage applied to the device. An FET has three legs named the gate, the source, and the drain. The two legs called the source and drain can be inserted into a circuit to act similarly to a switch controlling the current flow through the circuit. The gate controls the FET like a slide on a switch.

The big difference between a slide switch and the FET is that the FET is controlled by the application of a voltage to the gate, while a finger pushing the slide controls a mechanical switch. FETs are extremely sensitive to voltage changes and require virtually no current. These features make the FET a perfect choice for a banana switch!

In the circuit used for this activity, the banana is connected to the gate of the FET. The banana is charged from the 3.3V pin on the breadboard connector of the TI-Innovator Hub through the 10M Ohm resistor. The charge from the banana connected to the gate closes the source-drain circuit and keeps the FET on. When a well-grounded person touches the banana, the charge on the banana is conducted through the person, and the gate voltage drops to zero. Subsequently, the source and drain circuit is opened, and so the FET turns off. So, when the banana is touched and grounded, the FET is turned off. And when it is not being touched, the FET is turned on.

A feature of the FET is that the gate requires virtually no current, so there is no risk of shock to the person touching the banana and conducting the charge off the banana. In this circuit, a 1KΩ load resistor is placed in the source-drain circuit to produce a voltage drop to zero when the FET is on. When the FET is off, no current flows through the 1KΩ resistor, and there is no voltage drop from 3.3V.

The voltage drop across the load resistor can be monitored using the digital input on BB1. The digital input on the TI-Innovator Hub will report a 1 when it detects 3.3v while a 0 when it detects 0V. Thus, the digital input will return a 1 when the banana is touched because the FET is off and there is no voltage drop from 3.3V. The digital input will return a 0 when the banana is not being touched because the FET is on and there is a voltage drop to 0V.

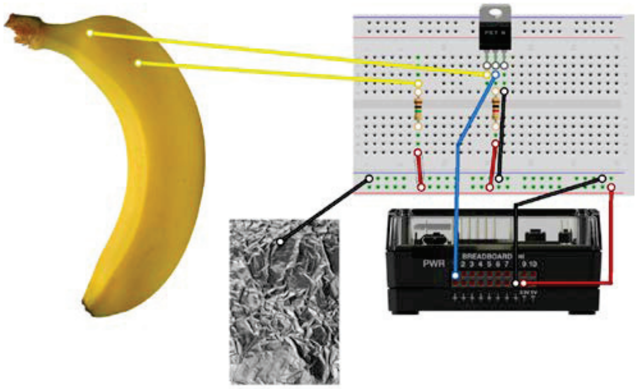
In this activity, to ground the person well that is touching the banana, they should touch or lean an arm on the square of foil attached to the ground on the TI-Innovator Hub. Again, the current and voltage are so small, there is no risk of shock.

Materials and Tools:

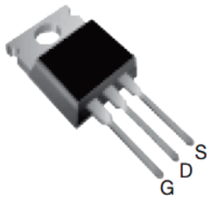
- TI-84 Plus CE
- TI-Innovator™ Hub with USB Cable
- Banana
- Aluminum foil
- Tape
- Scissors
- TI-Innovator Breadboard Pack:
  - ➢ Breadboard
  - ➢ Male to Male jumper cable
  - ➢ Potentiometer with Knob
  - ➢ Resistor 1K Ohm (brown, black, red)
  - ➢ Resistor 1M Ohm (brown, black, blue)
  - ➢ TTL Power MOSFET

Build the Hardware:

1. Use a red Male to Male Jumper Cable to connect the 3.3V of the TI-Innovator Hub to the red power bus on the breadboard.
2. Use a black Male to Male Jumper Cable to connect the Gnd of the TI-Innovator Hub to the blue ground bus on the breadboard.
3. Insert a TTL Power MOSFET into the middle of the breadboard.
4. Use a black Male to Male Jumper Cable to connect the far right leg (source) to the blue ground bus.
5. Insert a 1K Ohm Resistor into the center TTL Power MOSFET (S).
6. Use a red jumper wire to connect the other end of the 1 KΩ to the red 3.3V bus on the breadboard.
7. Use a blue Male to Male Jumper Cable to connect the center leg (D) to digital input BB1 on the TI-Innovator Hub.
8. Insert a yellow Male to Male Jumper Cable from the far left leg of the TTL Power MOSFET (G) to the banana. Push the jumper wire pin into the banana.
9. Insert a second yellow Male to Male Jumper Cable from the banana back into the breadboard.
10. Insert a 10 MΩ resistor into the same column as the yellow return Male to Male Jumper Cable from the banana.
11. Use a red Male to Male Jumper Cable to connect the 3.3V red power bus to the opposite side of a 10 MΩ resistor.
12. Fold a piece of foil into a 10 cm x 10 cm square and tape the pin of a black jumper wire to the one side of the foil square.
    - Ensure there is a good connection between the metal pin on the Male to Male Jumper Cable and the foil.
13. Insert the other end of the Male to Male Jumper Cable into the blue ground bus on the breadboard.



FET Transistor Led Diagram



G- gate, D- drain, S- sink

Write the Software for the TI-84 Plus CE:

Task: Write a program that uses digital input to check the state of the FET switch and plays a note on the internal speaker when a person touches the foil with one hand and the banana with the other hand.

When the **banana and foil are touched at the same time**, the FET is off, there is no voltage drop on the 1KΩ resistor, and the **digital input is equal to 1**.

When the banana and foil are not being touched, the FET is on, there is a voltage drop across the 1KΩ resistor, and the **digital input is equal to 0**.

Code for the TI-84 Plus CE:

```
Send("BEGIN"):Get(Str0):Disp Str0
Send("CONNECT DIGITAL.IN 1 TO BB1")
0→S
For(N,1,200)
Send("READ DIGITAL.IN 1")
Get(S)
If S=1
Then
Send("SET SOUND 500 TIME .1")
Disp "ON"
Else
Disp "OFF"
End
Wait (.1)
End
```

Extras for Experts:

Connect the speaker from the Unit 1 project to the TI-Innovator Hub and play the alarm tone through the hand-made speaker.

Sample Answer:

Hook up the speaker exactly as it was in Project 1. Be sure to connect the speaker to BB 2 on the TI-Innovator Hub with this command.

```
Send "CONNECT SPEAKER 1 TO BB 2"
```

Replace Send("SET SOUND 512 TIME .1")
with:
Send("SET SPEAKER 1 512 TIME .1")

This page intentionally left blank

Equipment Management

TI PROFESSIONAL DEVELOPMENT

Activity Overview

This is a time for teachers to brainstorm ideas for managing equipment in the classroom and what will work for them.

Use the provided space to brainstorm ideas on how equipment can be managed by students.

This page intentionally left blank

Creating and Implementing Projects in the Classroom

TI PROFESSIONAL DEVELOPMENT

Activity Overview

This is a time for teachers to brainstorm ideas for how students can create and manage their own projects in the classroom.

Use the provided space to brainstorm ideas on how to engage students in creating projects.

This page intentionally left blank

Coding Resources

TI-Innovator™ System eGuide
https://education.ti.com/go/eguide/hub/EN

TI-Basic Programming Guide for the TI-84 Plus CE Graphing Calculator
https://education.ti.com/html/webhelp/EG_TI84PlusCE/EN/content/eg_splash_page/ti-progguide_ce.HTML

TI Codes for the TI-84 Family
https://education.ti.com/en/us/solutions/ti-codes/84

TI-Nspire™ Programming eGuide
https://education.ti.com/html/webhelp/EG_TINspireCode/EN/content/eg_eguidehelp/egh_nspireprogrameguide.htm

TI Codes: TI-Nspire™ Technology
https://education.ti.com/en/us/solutions/ti-codes/nspire

Useful background articles on mapping of musical notes to frequencies:

http://www.sengpielaudio.com/calculator-notenames.htm

http://www.techlib.com/reference/musical_note_frequencies.htm

This page intentionally left blank

Coding for the TI-Innovator™ Hub Using Computer Software

TI PROFESSIONAL DEVELOPMENT

Activity Overview

This activity explores how to write programs, i.e., code, for the TI-Innovator™ Hub using computer software. Instructions are provided for using TI-Nspire™ Software, as well as for using TI-Connect™.

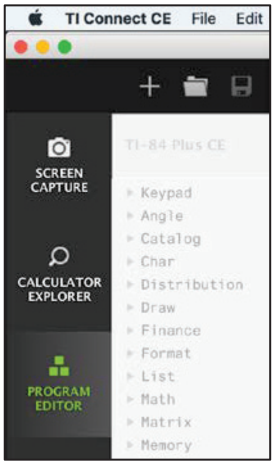
Introduction

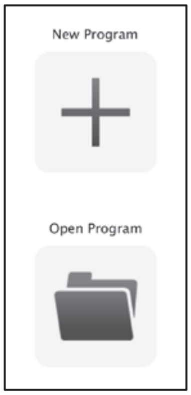
For students learning how to code, the handheld (especially the TI-84) offers the advantage of requiring that students use the **menus** available to them, thus making syntax errors less likely. When using TI-Nspire or TI-Connect software, one can simply type the code, which makes the programming go significantly faster. However, one must be cautious that syntax is typed in properly, especially with regards to syntax that is case-sensitive. The instructions below describe how to code for the TI-84 and TI-Nspire™ using computer software.

Coding for the TI-84 Using TI-Connect

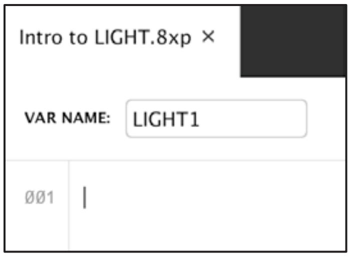
One can use TI-SmartView™ CE Emulator Software for the TI-84 Plus Family, but this behaves identically to how the handheld behaves, so it does not offer much advantage over using the handheld. Instead, we will use TI-Connect so that we can code directly by typing. *Note: You must use TI-Connect v5.2 or above.*

1. Launch the TI-Connect CE software.
2. Click on the PROGRAM EDITOR tab on the left.
3. In the main window, you can choose to create a New Program, or to Open an existing Program.





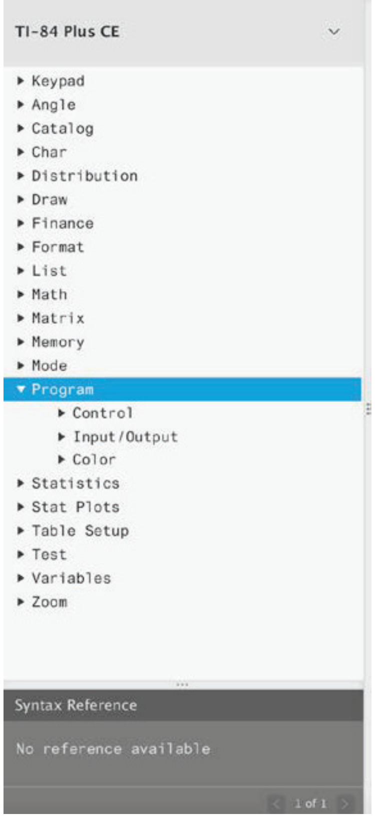
4. Note that there are **two** names for the program.
   a. The first is a **computer** file name that is saved to the computer (e.g., *Intro to LIGHT.8xp).* Programs for the TI-84 have an .8xp extension. To save a file, go to **File > Save**, or click on the *disk* icon.
   b. The second is a **device variable** name. This is the name of the program as it will appear on the handheld.



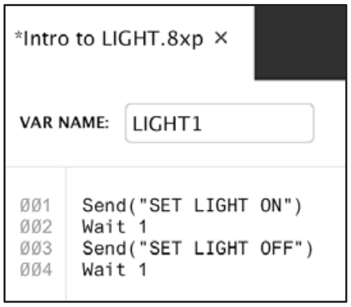
Intro to LIGHT.8xp ×

VAR NAME: LIGHT1

5. Now you can simply type in the code for the program, but be careful to type using correct syntax, especially case-sensitive code.
   A **catalog** is also available. To paste in an item from the catalog, double-click on it.



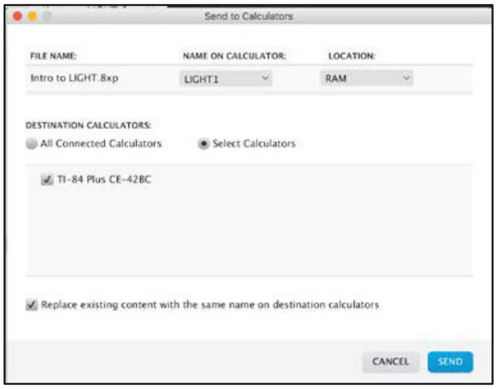
6. An example is shown to the right.
7. Save your computer file (**File > Save**).
8. In order to run the program to the TI-Innovator Hub, first connect a TI-84 to your computer using a USB cable.


*Intro to LIGHT.8xp ×

VAR NAME: LIGHT1

```
Send("SET LIGHT ON")
Wait 1
Send("SET LIGHT OFF")
Wait 1
```

9. Send the file to the TI-84 handheld using **Send to Calculators…** in the **Actions** menu, or click on the *Send program to connected calculator(s)* icon.
10. Connect the TI-Innovator Hub to the TI-84, and run your program.


Send to Calculators

FILE NAME:	NAME ON CALCULATOR:	LOCATION:
Intro to LIGHT.8xp	LIGHT1	RAM

DESTINATION CALCULATORS:
○ All Connected Calculators ● Select Calculators

☑ TI-84 Plus CE-42BC

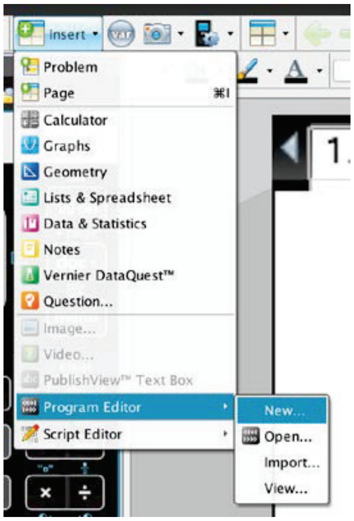
☑ Replace existing content with the same name on destination calculators

CANCEL SEND

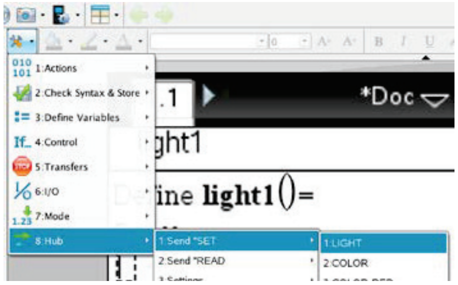
Coding for the TI-Nspire software

The TI-Nspire software is not only a convenient tool for writing code, it can also talk directly to the TI-Innovator Hub, so you do not need to send the file to a handheld. Generally speaking, though, when writing a program using the software it is always good practice to try it out on a handheld as well, just in case the program runs slower than intended.

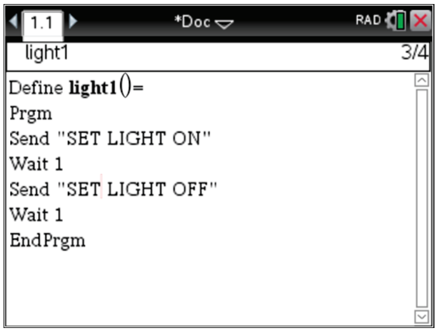
1. Launch the TI-Nspire software.
2. While you can use the handheld buttons to access menus:
    - It is faster to type in the coding directly, but be careful to match the case of the commands for the TI-Innovator.
    - The software has its own set of menus as well, and it is usually easier to access menu items using these rather than the handheld emulator buttons.
3. In a new document with a blank page (i.e., displaying "Press menu"), go to **Insert > Program Editor > New…**
4. Give your program a name.



5. You can access the **Hub** commands using the toolbox menu.



6. An example is shown to the right.
7. Use the menu to **Check Syntax & Store** to save your program.
8. Connect the TI-Innovator™ Hub to your computer.
9. Create a **Calculator** page and type in the name of your program followed by open and closed parentheses (or use var to select the program). Press enter to run the program.
10. If the program is trapped in an infinite loop or otherwise becomes unresponsive:
    - **Windows®:** Hold down the F12 key and press Enter repeatedly.
    - **Macintosh®:** Hold down the F5 key and press Enter repeatedly.



```
Define light1()=
Prgm
Send "SET LIGHT ON"
Wait 1
Send "SET LIGHT OFF"
Wait 1
EndPrgm
```

This page intentionally left blank

Updating the Operating System and Installing Add-On Apps

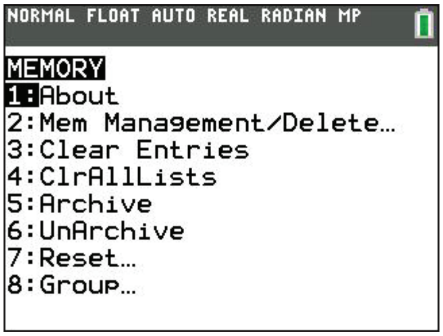
TI PROFESSIONAL DEVELOPMENT

Activity Overview

To update the operating system and install add-on applications to your calculator, you will need to connect your calculator to your computer using a connectivity cable and the TI Connect™ CE software.

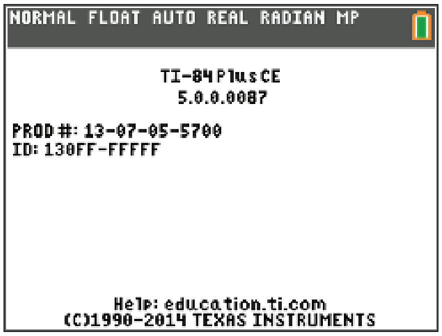
Checking the Calculator's Operating System

1. To check your calculator's current operating system, press [2nd] [+] to access the Memory menu.
2. **1:About** will be highlighted. Press [ENTER].



```
NORMAL FLOAT AUTO REAL RADIAN MP
MEMORY
1:About
2:Mem Management/Delete…
3:Clear Entries
4:ClrAllLists
5:Archive
6:UnArchive
7:Reset…
8:Group…
```

3. The OS version will be displayed on the line below the model name of the graphing calculator.
4. It is recommended that you periodically check TI's website to determine whether a new OS is available. If so, there is no charge to download and update a calculator's OS.



```
NORMAL FLOAT AUTO REAL RADIAN MP
                TI-84PlusCE
                 5.0.0.0087

PROD#: 13-07-05-5700
ID: 130FF-FFFFF




             Help: education.ti.com
     (C)1990-2014 TEXAS INSTRUMENTS
```

Checking the Calculator's Operating System

1. Go to TI's website, education.ti.com, and select **Downloads**.
   - In the Technology field, select your calculator model.
   - In the View field, select **Handheld Operating System**.
2. The latest operating system will be displayed. Click on its name to download the file.
3. Connect your graphing calculator to the computer with the standard A to mini-B USB cable.
4. Locate the OS file, right-click it, and select **Send to TI Device...** Once the Send to TI Device window opens, select **Send to Device.**

Note: For optimum performance when updating the OS (or installing any apps), be sure your calculator is fully charged (or has a new set of batteries).

Transferring the Operating System from Calculator to Calculator

You can also get the latest operating system from another TI-83 Plus or TI-84 Plus by linking the two calculators to each other. Although many apps can be run on both the TI-84 Plus and the TI-84 Plus, the operating system is unique to each calculator model. Do not try to download the TI-84 Plus OS to a TI-83 Plus or vice versa. Upgrading the OS might erase the memory, back up any files before updating an OS.

Downloading an App to Your Computer

1. Go to TI's website, education.ti.com, and select **Downloads**.
   - In the Technology field, select your calculator model.
   - In the View field, select either **Math Apps**, **Science Apps**, or **Utility Apps**, and then select **Find**.
2. Select the app you want to download, and then download it by clicking its name. The app will be saved to the default location on your computer.

Installing an App on Your Graphing Calculator

1. If necessary, download and install the TI Connect™ CE software and/or upgrade your calculator's operating system. Connect your calculator to your computer using a TI connectivity cable. Browse for the location of the app file(s) that you saved.
   - **For Windows® Users with the TI Connect software:**
     Drag and drop the app file(s) onto the TI Connect icon or open the TI Device Exporer™ window.
   - **For Mac® Users with the TI Connect software:**
     Launch the TI Device Explorer from within the TI Connect software. A device window will open listing the connected calculator. Drag and drop the app file(s) on the name of the calculator.
2. The number of apps that can be installed on your calculator depends on which model of the calculator you have and the specific apps you install.

   Your TI-83 Plus or TI-84 Plus graphing calculator has "app spaces" that are chunks of memory into which apps can be loaded. Some apps take up only one space, while others take up to four spaces. The number of app spaces available for each graphing calculator is listed below:

Model	Number of App Spaces
TI-84 Plus C Silver Edition	94
TI-84 Plus Silver Edition	94
TI-84 Plus	30
TI-83 Plus Silver Edition	94
TI-83 Plus	10

T^3 Ticket Outta Here

I have learned …

My next steps are …

T^3 Ticket Outta Here