Binary Output

Answers

7 8 **9** **10** 11 12

TI-Nspire | Innovator | Investigation | Student | 60 min

Binary Output

In this activity you will learn how to generate a text message using light pulses, in much the same way as data is transmitted through optic fibre cable or read from a DVD / Blue-Ray disc. The TI-Innovator™ has an in-built diode which will be used to provide a visual means of seeing the data transfer. The data transfer speeds used here will be much, much slower than those used in current digital equipment. The human eye is much too slow to detect the billions of On/Off signals per second, in this activity data transmissions are slowed to around 10 bits per second.

Coding the Algorithm

Instructions:

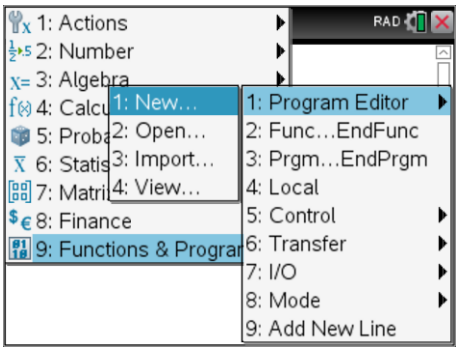
Connect the TI-Innovator hub to your calculator using the Calculator to Calculator cable. The "B" end connects to the Innovator and the "A" end to the calculator.

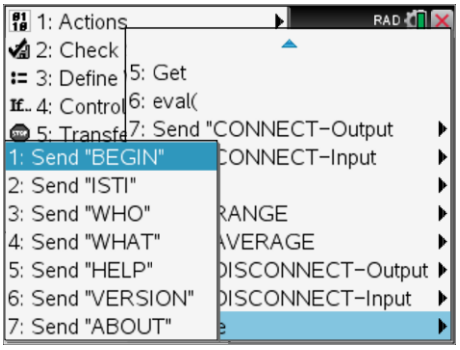
Start a new document; insert a calculator application followed by a program.

Call the program: Blink

Before programming the microprocessor on-board the TI-Innovator all previous content should be cleared. From the menu select **Hub** > **Manage** > **Send "BEGIN"**.

A series of commands will automatically be placed at the start of the program. Each time the program is run the TI-Innovator's microprocessor will be cleared ready to accept commands.





The next step is to switch the LED (Light) ON in the TI-Innovator.

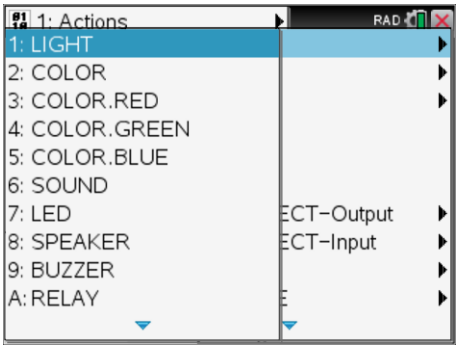
From the menu select: **HUB** > **Send "SET** > **LIGHT**

Send "SET LIGHT "

Notice that the command being sent to the Innovator is contained within quotation marks; it requires further information to communicate whether it is to be switched ON or OFF.

With the cursor between LIGHT and second quotation mark, insert the ON command from the menu: **HUB** > **Settings** > **ON**.

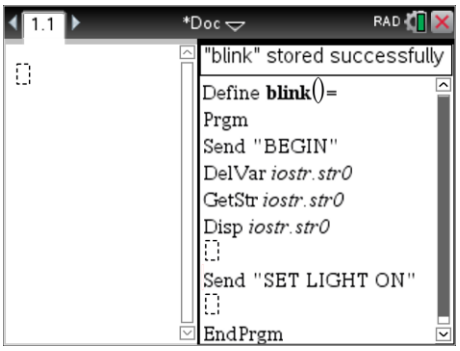
The command should now read: Send "SET LIGHT ON".



Author: P. Fox

Save the program by pressing CTRL + B, shift the focus to the calculator application (CTRL + TAB) and run the program.

The program can be accessed using the VAR key.



Question: 1.

What colour is the light when the program is run? The default setting for the RGB LED is red.

The problem with the program at this stage is that the light remains on even when the program is finished. Explore the hub program commands and locate a command that will send an OFF instruction. Place this OFF instruction after the Send "SET LIGHT ON" command then save and run the program.
You may need to run the program a couple of times to see what's happening.

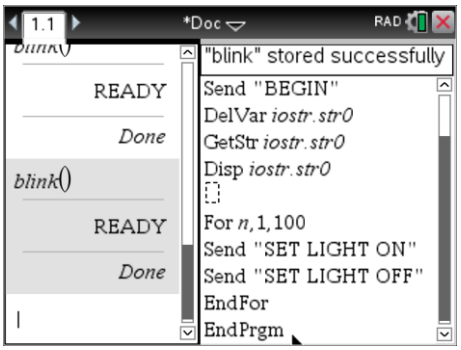
Question: 2.

Approximately how long does the light remain on?
Answers will vary. The LED stays on for only fraction of a second, approximate 1/10th.

A **For** loop can be placed around the Send commands so that they can be repeated. The command can be entered directly from the keyboard, the syntax is shown opposite. Alternatively, highlight the commands that are to be repeated, then from the **Control** menu, select **For** … **EndFor**.

Set the loop to repeat 100 times.

Once the **For** loop has been included, save and run the program.



Question: 3.

Use a stopwatch to record how long the program takes to execute. How long does it take to execute the ON and OFF commands? (Use this to check your estimate in Question 2)
Answers will vary but should be close to 10seconds for 100 commands, meaning that each pair of instructions are executed in approximately 1/10th of a second.

Sending a Message

The LED on the TI-Innovator will now be used to send a message. The ASCII character set will be used as the code to help write the message.

ASCII Table – (Letters only)

A = 65	B = 66	C = 67	D = 68	E = 69	F = 70	G = 71	H = 72	I = 73	J = 74
K = 75	L = 76	M = 77	N = 78	O = 79	P = 80	Q = 81	R = 82	S = 83	T = 84
U = 85	V = 86	W = 87	X = 88	Y = 89	Z = 90				
a = 97	b = 98	c = 99	d =100	e = 101	f = 102	g = 103	h = 104	i = 105	j = 106
k = 107	l = 108	m = 109	n =110	o = 111	p = 112	q = 113	r = 114	s = 115	t = 116

Author: P.Fox

u = 117	v = 118	w = 119	x = 120	y = 121	z = 122				

Question: 4.

Write a *very* short message, it could be your name. Use the ASCII code to change each letter in your message to a number.

Answers will vary depending on the message students choose to write.

Teacher Notes:

Encourage students to keep their message VERY short. The next task is to convert the message into binary. Each letter in their original message will be replaced by a 8 bit binary number. Each bit from this set of binary numbers will then be entered into a list. So a message with 20 characters will become a list of 1's and 0's consisting of 160 bits!

Question: 5.

Convert all the decimal numbers from Question 4 into binary. (Leave a small space between nibbles). The table below provides a sample message.

Letter	**ASCII**	$2^7 = 128$	$2^6 = 64$	$2^5 = 32$	$2^4 = 16$	$2^3 = 8$	$2^2 = 4$	$2^1 = 2$	$2^0 = 1$	Result
H	72	0	1	0	0	1	0	0	0	0100 1000
E	69	0	1	0	0	0	1	0	1	0100 0101
L	76	0	1	0	0	1	1	0	0	0100 1100
L	76	0	1	0	0	1	1	0	0	0100 1100
O	79	0	1	0	0	1	1	1	1	0100 1111

Starting with the most significant bit for the first letter, the digital transmission of the word "HELLO" or "Hello" would appear as follows:

HELLO = 0100 1000 0100 0101 0100 1100 0100 1100 0100 1111

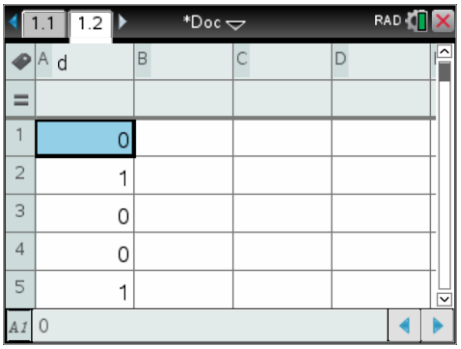
Hello = 0110 1000 0110 0101 0110 1100 0110 1100 0110 1111

Store all the 1's and 0's in a single list. This can be done using the spreadsheet or directly from the calculator application.

The example shown here uses the spreadsheet so it is easier to track your input by using the row numbers.

To insert a Spreadsheet application press CTRL + I.

Label the list "D". Enter the data 'bit' by 'bit' so that each cell contains a single bit.

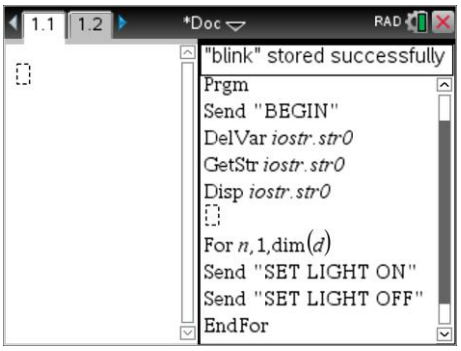


Author: P.Fox

The program can now be edited to use this binary code to turn the LED light ON / OFF accordingly.

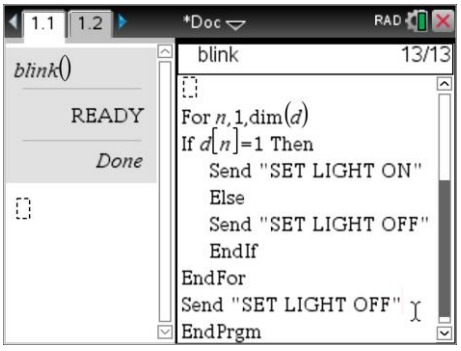
The first thing that needs to be done is to automatically determine the size of the list (d). The DIM command returns the dimensions of a list.

Edit the FOR loop so that it is repeated according to the dimensions of list d. (Shown opposite)



The next step is to only turn the light on IF the data value (bit) is equal to 1.

Within the FOR loop, this can be written as:




```
IF d[n] = 1 THEN
  Send “SET LIGHT ON”        {If data bit = 1, LED is on}
  ELSE
  Send “SET LIGHT OFF”       {If data bit = 0, LED is off}
  EndIf
```

Note that the text above {If data bit = …} is not required in the program, it is included here as a comment only.
At the end of the program, turn the light off.

Question: 6.

Run the blink() program and describe what happens with the LED on the TI-Innovator.
The LED on the TI-Innovator flashes on / off rapidly, for students this should appear almost random.

Teacher Notes:

A 120 bit code will be sent in around 12 seconds, despite the ‘relatively’ slow speed, the message is much too fast for students to interpret. For real data transmissions students would not be able to distinguish of the LED was on or off with transfer rates of the order of Megabits per second!

Question: 7.

The data speed can be slowed down by including a Wait command immediately after the EndIf command. Use a 1 second wait time, “Wait 1”. Save then run the program.
Explain why it is still difficult to accurately read the data being displayed by the LED.
The code is still difficult to read because it is hard to tell when the LED is on for two or more bits rather than a single bit, similarly with the off states.

Teacher Notes:

To provide a ‘clue’ for students, ask them if it would help if they had a stop watch. The idea of the stop watch is to help them know ‘when’ to check if the LED is on / off.

Question: 8.

Explain how it might be made easier to visually read the On / Off signals that refer to each bit.
(ie: Using the human eye, not another electronic device.)
Answers will vary. The idea is to consider the requirement for a ‘clock’ pulse to regulate the transfer of data and reading on/off signals.

Author: P.Fox