



Challenge: Design a Digital Mood Ring

TI-84 PLUS CE

Overview:

In this project, students build and program an introductory example of a feedback and control loop. This engineering principle is central to many industrial systems and consumer products. Feedback and control principles are also used in many Texas Instruments TI-Innovator™ Hub projects. During the project students will engage with fundamental concepts in programming, such as variables, loops, conditional statements and Boolean operators.

The project is presented as a series of challenges that build the conceptual knowledge and skills needed for the final open-ended challenge.

Goals:

Students will:

- create and edit a TI-BASIC program on the calculator.
- connect the Hub with a sensor to the calculator.
- write programs that include many commonly used Hub and calculator commands.
- build a basic feedback and control system

Background:

Command	Example	Behavior
SET COLOR <r g b>	<code>Send("SET COLOR 255 0 0")</code>	Turns the color LED on with the color red.
Output(<row 1-10>, <column 1-26>,<expression> <variable name> or <string>) Note: It is recommended to use the ClrHome command before using Output commands	<code>Output(4,1,"TEMPERATURE= ")</code> <code>Output(4,14,T)</code>	When variable T has a value of 26, the following line is displayed on the calculator: TEMPERATURE = 26
CONNECT <type> <number> TO <port>	<code>Send("CONNECT TEMPERATURE 1 TO IN1")</code>	Associates the first TEMPERATURE object with a temperature module plugged into port IN1 on the Hub.
READ <type> <number>	<code>Send("READ TEMPERATURE 1")</code>	Reads a measurement from the first temperature object.
Get <variable>	<code>Get(T)</code>	Stores the temperature measurement into a variable named T. The value stored will contain the measurement from the immediately preceding READ command. Note: a Get command must immediately follow a READ command.
For <counter variable>,<start value>,<end value>,<step value>] <statements> End	<code>For(N,1,10)</code> <code>Output(3,1,N)</code> <code>End</code>	Runs For loop 10 times, starting at 1 and ending at 10. Executes the statement in the block each time, displays the value for the counter variable on row 3.
While <Boolean expression> <statements> End	<code>0→K</code> <code>While K≠45</code> <code>Send("READ TEMPERATURE 1")</code> <code>Get(T)</code> <code>Output(3,1,"TEMPERATURE= ")</code>	The statements in the While loop are executed until the clear key (positioned at row 4 and column 5 of the keyboard) is pressed. The While loop continues as long as the Boolean expression evaluates to "true". The variable K is set to an initial value of 0 using the store function, →. getKey is a function that returns a two-digit value (row and column) of the position on the keyboard of the last key pressed while a

	<pre>Output (3,14,T) getKey→K Wait 1 End</pre>	<p>programming is running. In this program the value of getKey is stored to the variable K.</p>
<pre>If <Boolean expression> Then <statements> EndIf</pre>	<pre>If T>26 Then Send("SET COLOR 255 128 0") End</pre>	<p>Checks to determine if the value of variable t is greater than 26. If the statement is true then the block of commands in the If..End block are executed. Otherwise the block is skipped. In this case, when the temperature is greater than 26 the calculator will send a command to the TI-Innovator to set the COLOR to be yellow.</p>
<pre><Boolean expression> and <Boolean expression></pre>	<pre>If T≥22 and T<24 Then Send("SET COLOR 0 255 0") End</pre>	<p>When both expressions are true the "and" function is "true" and the statements are executed. Otherwise, the function returns false and the statements are skipped.</p>

See TI-Innovator Technology eGuide for more background https://education.ti.com/html/webhelp/EG_Innovator/EN/index.html

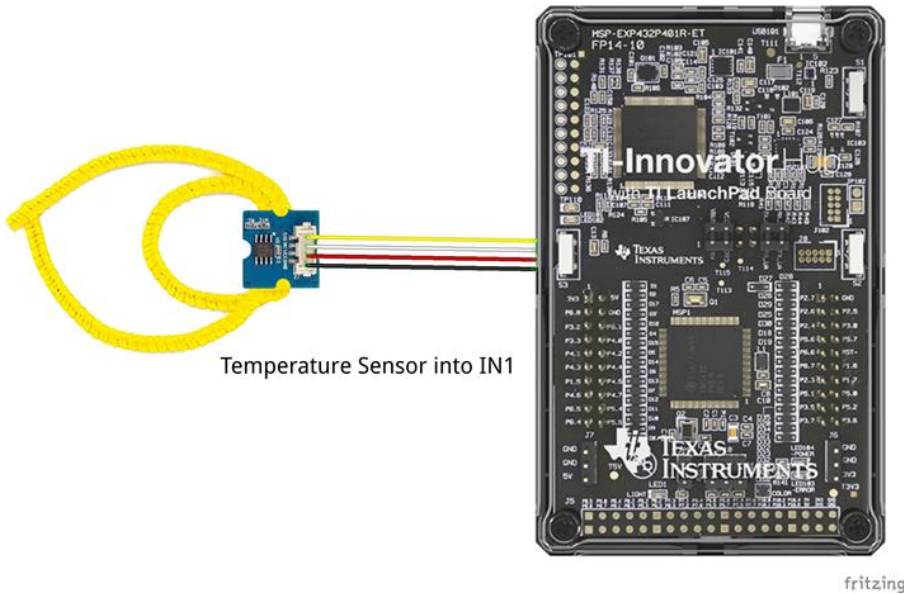
Setup Project:

Students may work in groups of two or three.

Supplies:

- Calculator
- Unit to Unit Cable
- Hub
- Temperature sensor
- Grove Cable
- Chenille Wire

Note: Students will need to push the wire through the holes on the temperature sensor board to create a ring. The sensing element is on the flat side of the sensor board. The student document shows a pipe cleaner, but it is not as pictured here. Expect (and encourage!) trial and error as students attempt to determine the most effective ring design.



Student Activity:

Sit in small groups with your calculator and supplies for this activity. Practice the guidance modeled by your teacher.

Teacher Notes:

Review and introduce the following calculator and Hub commands needed for this activity with students:

- Start a new document
- Insert a program
- Attach Hub (Plug B side of unit-to-unit cable into Hub Data port then plug A side into calculator.)

Note: If students are new to the technology and/or coding, refer students to step-by-step lessons in the 10 Minutes of Code with TI-Innovator lessons to get started, especially Unit 1 Skill Builder 1: Color and Unit 1 Skill Builder 2: Sound.

<https://education.ti.com/en/activities/ti-codes/84/10-minutes-innovator>

If some students are unfamiliar with Mood Rings or how they work, have students do a quick search on the internet for resources describing how a mood ring works. You may want to begin with this YouTube video describing how a mood ring works. <https://www.youtube.com/watch?v=gVGI51xOKTU&feature=youtu.be>

Optional: Distribute the associated Student Activity which contains background info and the Challenges listed here in this document. By doing so, student groups can work at their own pace through the Challenges. Alternatively, you may choose to display the Challenges one at a time.

Challenge 1: Use SET COLOR to explore using the color LED. Try to find the RGB values of all the colors in the mood chart.

e.g. Send("SET COLOR 255 128 0") will make yellow.

Teacher Guidance during Challenge 1:

- Here is one source for background information on the science of how we see color.
<http://www.cyberphysics.co.uk/topics/light/colorAddition.html>
- Note: You can find resources on the codes for colors with a web search of "RGB color codes".





- Example program

```
PROGRAM: CAA
Send("SET COLOR 255 128 0")
```

- Extension: Write a program that displays each of the colors that the mood ring will display.

```
PROGRAM: CAB
Send("SET COLOR 255 0 0")
Wait 2
Send("SET COLOR 255 0 255")
Wait 2
Send("SET COLOR 255 113 0")
Wait 2
Send("SET COLOR 0 0 0")
```

- Wait is useful to pause the program for a specified number of seconds.
- Try setting the color value to 0 0 0
- Try setting the color value to 255 255 255
- Try setting the color value to 5 5 5

Challenge 2: Use the Output(command to display your name at several locations on the screen.

Teacher Guidance during Challenge 2:

- The alpha key toggles between alpha and normal entry. The program name prompt starts in alpha entry mode. To enter the name C2 type C alpha (to toggle to normal entry mode) 2.
- Use the Output(command (Program I/O menu). Output(takes three inputs (also known as “arguments”). The first input is the row number to display on. The second number is the column number to begin on. The third input is a text string enclosed in quotes, a numeric value, a variable name or an expression to resolve to a result to display.
- Use the ClrHome command (Program I/O menu) to give you an empty home screen to display on.
- Example program

```
PROGRAM: C2
ClrHome
Output(4,1,"HELLO, MY NAME IS:")
Output(5,1,"HOLLINGSWORTH")
```



Challenge 3: Use a For loop to display the numbers 1 through 10.

Teacher Guidance during Challenge 3:

- Loops are used to repeat a set of command. A For loop repeats a specified number of times. In TI-Basic the programmer defines a For loop with four arguments: a counter variable, a beginning value for the counter variable, an ending value for the counter variable and an optional step value variable.
- Use the ClrHome command (Program I/O menu) to give you an empty home screen to display on.

- Example program

```
PROGRAM: C3
ClrHome
For(N,1,10)
Output(4,1,N)
Wait 0.5
End
```

- The value for step is assumed to be 1 if a fourth optional input for step is not included in the command.
- The set of commands to be repeated are defined within the For..End block.
- For and End are found on the Program CTL Menu
- Experiment with different values to display. For example, `Output(3,1,N*2)`
- Experiment with an optional input for the step of the For loop. For example, `For(N,1,10,2)`

Challenge 4: Connect a temperature sensor to the TI-Innovator Hub and display the temperature on the calculator.

Teacher Guidance during challenge 4:

- The Hub Connect commands provide the hub with information about the type of device plugged into external ports and how to communicate with that device.
- The Read command instructs the Hub to read a value from a device plugged into a specified port. The Hub places the value in a memory location on the Hub, waiting for the calculator to “get” the value.
- The Get command gets the value from the Hub memory location for values read and stores the value to a specified variable.
- Note that multiple Output(commands can be used to build a message combining text and values on a row.
- Use the ClrHome command (Program I/O menu) to give you an empty home screen to display on.



- Example program

```
PROGRAM: C4
ClrHome
Send("CONNECT TEMPERATURE 1 TO IN 1")
Send("READ TEMPERATURE 1")
Get(T)
Output(4,1,"TEMPERATURE= ")
Output(4,14,T)
```

Challenge 5: Use a For loop to read and display temperature.

Teacher Guidance during Challenge 5:

- Display the sample number (counter value) in addition to the temperature.
- Note that multiple Output commands can be used to build a message on a row. Make sure to count the number of characters to determine where messages should begin. Note: Find the degree symbol on the angle menu (2nd function of the apps key) and the equal symbol on the test menu (2nd Math).
- Example program

```
PROGRAM: C5
ClrHome
Send("CONNECT TEMPERATURE 1 TO IN 1")
For(N,1,10)
Send("READ TEMPERATURE 1")
Get(T)
Output(4,1,"SAMPLE NUMBER= ")
Output(4,16,N)
Output(5,1,"TEMPERATURE (°C)= ")
Output(5,18,T)
Wait 2
End
```



Challenge 5 Extension: Read, display and log into list variables for time and temperature. Graph the result.

Teacher Guidance during Challenge 5 Extension:

- **Note:** This challenge is optional. Logging sensor readings to list variables for later analysis is useful. However, logging the data is not required for a working digital mood ring.

- Example program

```
PROGRAM: C5E
ClrHome
Send("CONNECT TEMPERATURE 1 TO IN 1")
0→dim(L1)
0→dim(L2)
For(N,1,10)
Send("READ TEMPERATURE 1")
Get(T)
Output(4,1,"TIME (SEC)= ")
Output(4,12," ")
Output(4,12,N*0.5)
Output(5,1,"TEMPERATURE (°C)= ")
Output(5,18,T)
N*0.5→L1(N)
T→L2(N)
Wait 0.5
End
```

- The variables L₁ and L₂ are initialized as blank lists in the beginning section of the program by setting their dimension (length) to 0.
- The temperature value readings are stored to a variable T. During each loop cycle the value for T is stored to the position N of the list named L₂. The command is T→L₂(N)
- The value of time is calculated by multiplying the number of loop cycles, N, by the wait period, 0.5. The calculated time value is stored to position N of L₁ with this command N*0.5→L₁(N)
- The result of N*0.5 varies between one and three characters in length. To assure that there is no conflict with leftover characters from prior values we precede the Output command for the time calculation with an Output command of four blank spaces. This assures that we are displaying to a clear space.



Challenge 6: Use a While loop along with the getKey function to monitor temperature and make a decision to display a message, “Hot” or “Cold”.

Then modify your program to include a 3rd level between hot and cold, “Nice”.

Teacher Guidance during Challenge 6:

- While loops are useful when you would like a set of commands to be executed when a certain condition is true.
 - The While loop test checks to see if the variable named K is not equal to 45. If K is not equal to 45 then the commands in the loop are run. If K is equal to 45 then the program exits the While loop and moves to the next command after the End statement. In the example, the variable named K is initially set to be 0. This assures that the While loop will run at least once.
 - In the While loop a function called getKey is used to update the value stored in the variable named K. getKey is a TI-84 function that is useful for keyboard controlled While loops. getKey stores a 2-digit code value for the last key that you pressed while a program is running. The first digit is the row on the keyboard of the key, counting from the top. The second digit is the column of the key location on the keyboard. In this example, there is a check to see if the clear key, code value of 45, has been pressed.

- Example program:

```
PROGRAM: C6
ClrHome
Send("CONNECT TEMPERATURE 1 TO IN 1")
0→K
While K≠45
Send("READ TEMPERATURE 1")
Get(T)
Output(4,1,"TEMPERATURE (°C)= ")
Output(4,18,T)
If T<22
Then
Send("SET COLOR 0 0 255")
Output(5,1,"IT IS COLD")
End
If T≥22
Then
```




```
Send("SET COLOR 255 0 0")
Output(5,1,"IT IS HOT ")
End
getKey→K
Wait 0.5
End
```

Extension to include three levels of temperature:

- The Boolean operator “and” is used to define levels of temperature. The “and” function can be found on the LOGIC menu accessed by 2nd test on the keyboard. The “or” function is another commonly used Boolean operator. “or” can also be found on the LOGIC menu.

```
PROGRAM: C6E
ClrHome
Send("CONNECT TEMPERATURE 1 TO IN 1")
0→K
While K≠45
Send("READ TEMPERATURE 1")
Get T
Output(4,1,"TEMPERATURE (°C)= ")
Output(4,18,T)
If T<22
Then
Send("SET COLOR 0 0 255")
Output(5,1,"IT IS COLD")
End
If T≥22 and T<25
Then
Send("SET COLOR 0 255 0")
Output(5,1,"IT IS NICE")
End
If T≥25
```



Challenge: Design a Digital Mood Ring

TI-84 PLUS CE

```
Then
Send("SET COLOR 255 0 0")
Output(5,1,"IT IS HOT ")
End
getKey→K
Wait 0.5
End
```

Final Challenge: Build a mood ring to repeatedly read the temperature sensor, determine the mood of the person, display the temperature value and display the mood.

Teacher Guidance during Final Challenge:

- Have the students use the skills from the previous challenges.
- How many moods will the students have their programs check for?
- What values will the students use to define each mood level?
- Here is a simple version of the Digital Mood Ring program.

```
PROGRAM: MOOD
ClrHome
Send("CONNECT TEMPERATURE 1 TO IN 1")
Output(1,1,"THE MOOD RING")
Output(8,1,"HIT CLEAR TO QUIT")
0→K
While K≠45
getKey→K
Send("READ TEMPERATURE 1")
Get(T)
Output(4,1,"TEMPERATURE (°C)= ")
Output(4,18,T)
If T<24
Then
Send("SET COLOR 0 0 0")
Output(5,1,"YOU ARE STRESSED ")
End
If T≥24 and T<25
```



```
Then
Send("SET COLOR 255 0 0")
Output(5,1,"YOU ARE NERVOUS ")
End
If T≥25 and T<26
Then
Send("SET COLOR 255 150 50")
OUTPUT(5,1,"YOU ARE UNSETTLED")
End
If T≥26 and T<27
Then
Send("SET COLOR 0 255 0")
OUTPUT(5,1,"YOU ARE ACTIVE ")
End
If T≥27 and T<28
Then
Send("SET COLOR 0 255 150")
Output(5,1"YOU ARE RELAXED ")
End
If T≥28 and T<29
Then
Send("SET COLOR 0 255 255")
Output(5,1,"YOU ARE LOVEABLE ")
End
If T≥29 and T<30
Then
Send("SET COLOR 0 0 255")
Output(5,1,"YOU ARE ROMANTIC ")
End
If T≥30 and T<31
Then
Send("SET COLOR 255 100 75")
```



```
Output(5,1,"YOU ARE HAPPY  ")
End
If T≥31 and T<32
Then
Send("SET COLOR 255 255 0")
Output(5,1,"YOU ARE TIRED  ")
End
If T≥32
Then
Send("SET COLOR 255 0 255")
Output(5,1,"YOU ARE CALM  ")
End
Wait 0.5
End
Send("SET COLOR 0 0 0")
```

- The example program below includes many optional extensions. See explanations below.

```
PROGRAM: MOODEXT
1→A
0→T
0→S
0→R
0→K
1→N
0→dim(L₁)
0→dim(L₂)
Send("CONNECT TEMPERATURE 1 TO IN 1")
Wait 0.5
ClrHome
Output(1,1,"THE MOOD RING")
Output(8,1,"HIT CLEAR TO QUIT")
```



```
For(A,1,10)
Send("READ TEMPERATURE 1")
Get(T)
T+S→S
Wait 0.1
End
S/10→R
Output(2,1,"ROOM TEMPERATURE (°C)= ")
Output(2,22,round(R,1))
While K≠45
getKey→K
Send("READ TEMPERATURE 1")
Get(T)
Output(4,1,"Temperature (°C)= ")
Output(4,18,round(T,1))
If T<R+1
Then
Send "SET COLOR 0 0 0"
Output(5,1,"YOU ARE STRESSED ")
End
If T≥R+1 and T<R+2 Then
Send("SET COLOR 255 0 0")
Output(5,1,"YOU ARE NERVOUS ")
End
If T≥R+2 and T<R+3
Then
Send("SET COLOR 255 150 50")
Output(5,1,"YOU ARE UNSETTLED")
End
If T≥R+3 and T<R+4
Then
```



Challenge: Design a Digital Mood Ring

TI-84 PLUS CE

TI-INNOVATOR™ STEM PROJECT

TEACHER NOTES

```
Send("SET COLOR 0 255 0")
Output(5,1,"YOU ARE ACTIVE  ")
End
If T≥R+4 and T<R+5
Then
Send("SET COLOR 0 255 150")
Output(5,1,"YOU ARE RELAXED  ")
End
If T≥R+5 and T<R+6
Then
Send("SET COLOR 0 255 255")
Output(5,1,"YOU ARE LOVEABLE ")
End
If T≥R+6 and T<R+7
Then
Send("SET COLOR 0 0 255")
Output(5,1,"YOU ARE ROMANTIC ")
End
If T≥R+7 and T<R+8
Then
Send("SET COLOR 255 100 75")
Output(5,1,"YOU ARE HAPPY  ")
End
If T≥R+8 and T<R+9
Then
Send("SET COLOR 255 255 0")
Output(5,1,"YOU ARE TIRED  ")
End
If T≥R+9
Then
Send("SET COLOR 255 0 255")
Output(5,1,"YOU ARE CALM  ")
```



```
End  
Wait 0.5  
N*0.5→L1(N)  
T→L2(N)  
N+1→N  
End  
Send "SET COLOR 0 0 0"
```

Explanation of advanced program

- The program stores time and temperature values in lists L1 and L2 for later analysis.
- The program uses the ClrHome command from the Program I/O menu to assure that the home screen is clear of contents before beginning Output(commands).
- The program takes into account the temperature in the environment by using a For..End loop to calculate a mean temperature of 10 readings. The mean temperature is stored to the variable R (for room temperature), which is then used to shift the temperatures ranges of the mood categories.