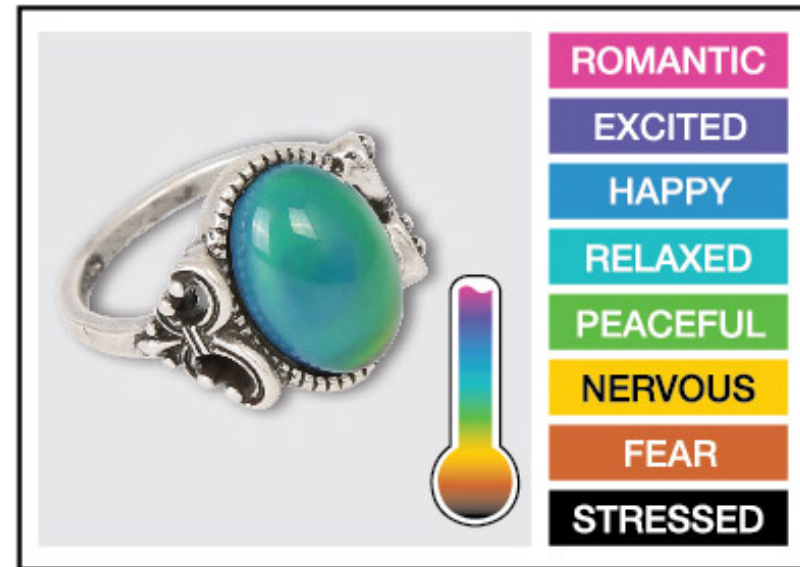


PROJECT OVERVIEW

Mood Rings: How do they work?

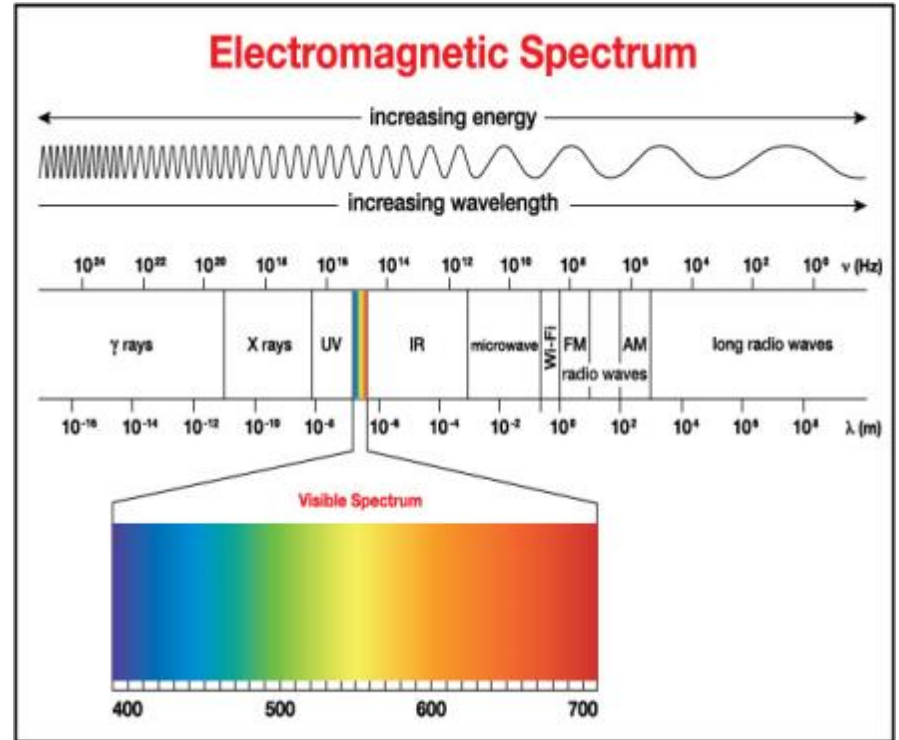
Background:

The mood ring was invented by Joshua Reynolds. Mood rings enjoyed fad popularity in the 1970s and are still around today. The stone of the ring changes color, supposedly according to the mood of the wearer. The 'stone' of a mood ring is really a hollow glass shell containing *thermotropic* liquid crystals. Modern mood rings are made from a flat plastic strip containing liquid crystals that is inserted into the hollow glass and mounted within the bezel of the mood ring.



Mood Rings: How do they work?

- The crystals respond to changes in temperature by twisting in a regular way. The twisting changes their molecular geometry, which alters the wavelengths of light that are reflected from the crystals.
- **Wavelengths** of light are another way of saying color, so when the temperature of the liquid crystals change, so does the color reflected from the stone. Thus, the ring changes color with the hand temperature of the wearer.



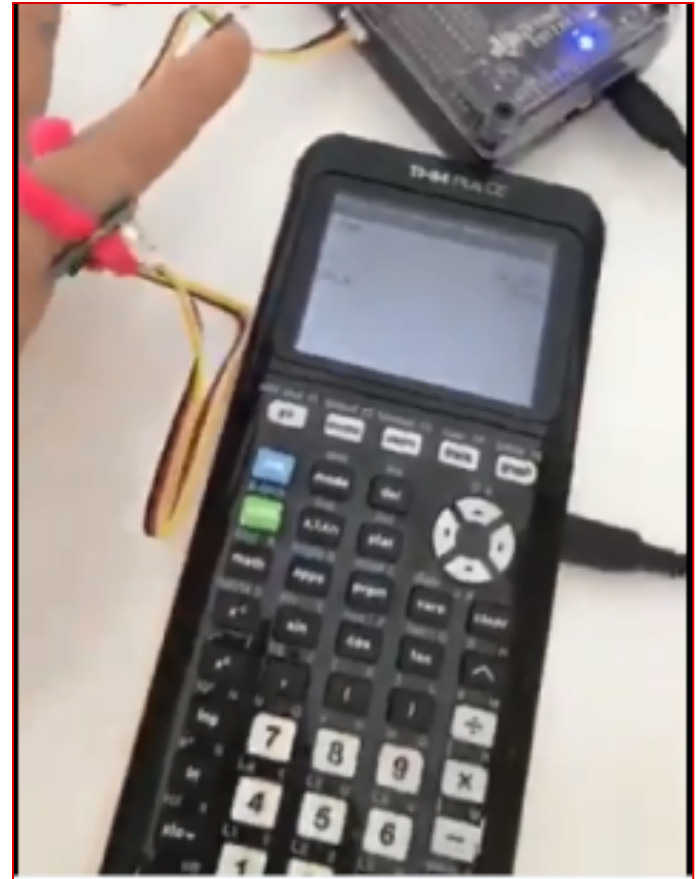
Time to DIY: Digital Mood Ring!

The Design Challenge:

Design and build a digital mood ring using the pipe cleaner that fits snugly on your finger, placement should be similar to any ring worn on the finger. Be sure the flat part of the temperature sensor is touching your skin.

In this project, you will investigate:

- temperature measurement using a temperature sensor and then build a digital mood ring that changes the color of the Hub's Red, Green, and Blue (RGB) light emitting diode (LED) as the temperature your finger changes.
- You can even display what mood you are in on your calculator!

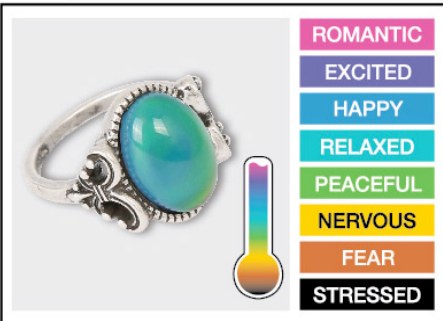


Mood Ring Project Tasks:

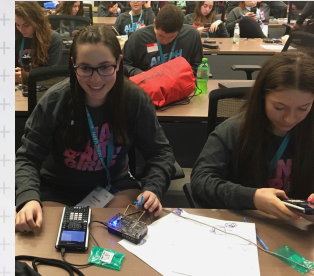
- ✓ Students are acquainted with the Mood Ring project and the graphing calculator platform. Create and execute a program to display “Hello World”.
- ✓ Students are acquainted with the TI-Innovator Hub. Create and execute a program using COLOR command to display different colors using the Red-Green-Blue LED.
- ✓ Students create a program that connects an external temperature sensor, reads values from the sensor and then displays values from the sensor. Students use the pipe cleaner to connect the TI-Innovator and temperature sensor to form a “ring.”
- ✓ Students build on the program from above to use if-then logic to control COLOR outputs based on temperature readings to create a digital mood ring.

PROJECT CONCEPTS

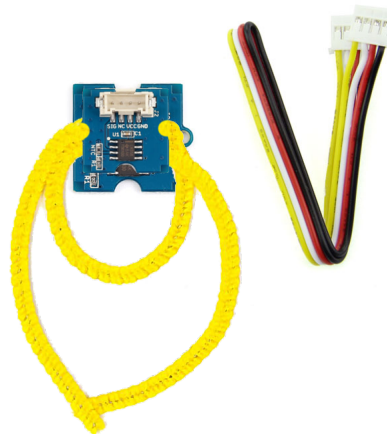
Digital Mood Ring (introduction to STEM and coding Projects for all levels)



Bring science and coding together (no coding experience necessary) while developing a mood ring! Explore the science of color mixing while determining the right body temperature thresholds.



This is an excellent first TI-Innovator STEM and coding project. The project includes variables, loops, conditional statements, Boolean operators and other fundamental programming concepts.



Digital Mood Ring Challenges

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.

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

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

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

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

Challenge 5 Extension (Optional): Use a loop to read, display and log into list arrays for time and temperature. Graph the result.

Challenge 6: Use a While..End loop along with 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”.

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.

Summary of programming concepts in the project

- Control of a red-green-blue LED output
 - Send("SET COLOR 255 0 0")
- Display of text strings, variable values and expressions
 - Output(4,1,"TIME (SEC) =")
 - Output(4,13,N*0.5)
- For loops to repeat a set of commands a specified number of times
 - For(N,1,10)
 commands within loop
End
- Connecting, reading and getting to a variable temperature sensor inputs
 - Send("CONNECT TEMPERATURE 1 TO IN 1")
 - Send("READ TEMPERATURE 1")
 - Get(T)
- Storing variables to variables using the store operator.
 - $9/5 \times T + 32 \rightarrow F$
- List variables of a set of values and accessing individual elements of the list using the position number.
 - $\{0.5, 1, 1.5, 2\} \rightarrow L_1$
 - $2.5 \rightarrow L_1(5)$
- True/False comparison operators used in conditional statements for program control: =, ≠, <, ≤, >, ≥
- Assignment of initial values to variables to give the program known starting points.
 - $0 \rightarrow K$
 - $0 \rightarrow \dim(L_1)$
- While loops to repeat a set of commands while a condition is true
 - While $K \neq 45$
 commands within loop
End
- Keyboard inputs using the TI-84 Plus CE getKey command that returns a two-digit number for the row and column (top left key, y=, is 11) of the last key pressed.
 - getKey $\rightarrow K$
- If-Then conditional statements that do a set of commands if a condition is true and are skipped if the condition is false
 - If $T < 22$
 Then
 commands within If-Then block
End
- Boolean operator *and* to require two conditional statements to be true for the entire statement to be true
 - If $T \geq 22$ and $T < 25$
 Then

Summary of TI-84 Plus CE concepts and skills

- ON starts at the home screen. The home screen is used for calculations, running programs and for program display commands.
- 2nd Quit will return you to the home screen from any situation on the calculator.
- Select from menus using arrow keys and enter or by the shortcut of the number or letter preceding the menu item
- clear key to back out of menus and dialogues
- The keyboard is divided into three sections:
 - Scientific calculator and math functions (bottom)
 - Entry, navigation and setup (middle)
 - Graphing and function keys (top)
- Press and release the 2nd key to access function labeled above the keys. The input cursor changes to an up arrow to indicate 2nd entry mode. Press the 2nd key again to toggle back to normal entry mode.
- Press and release the Alpha key to access Alpha characters labeled above the keys. The input cursor changes to an Alpha to indicate Alpha entry mode. Press the Alpha key again to toggle back to normal entry mode.
- The delete key deletes the character at the current cursor location.
- 2nd Insert sets insert mode to enter characters before the current cursor location. Insert mode is indicated by an underline cursor. 2nd Insert or arrow key toggles back to normal entry mode.
- Pressing the prgm key while on the home screen gives you options to select a program to run on the home screen, to edit an existing program or to create a new program.
- Editing a program or creating a new program puts you in the program editor. Pressing the prgm key while in the program editor brings up menus with programming functions.
- Alpha f5 while in the program editor brings up a menu of editor tools, including copy, paste and insert line and execute program.
- 2nd Quit while in the program editor exits the editor and returns you to the home screen.
- The Sto→ key stores values to variables.
- The calculator has single variable values of A-Z and Θ and list variables of L1 and L6.
- 2nd catalog brings up a list of the functions of the calculator.

Standards

- NGSS 3-D Standards

- MS-PS1-1 Matter and its Interactions

- PS1-A Structure and Properties of Matter- solids may be formed from molecules, or they may be extended structures with repeating subunits (e.g., crystals)

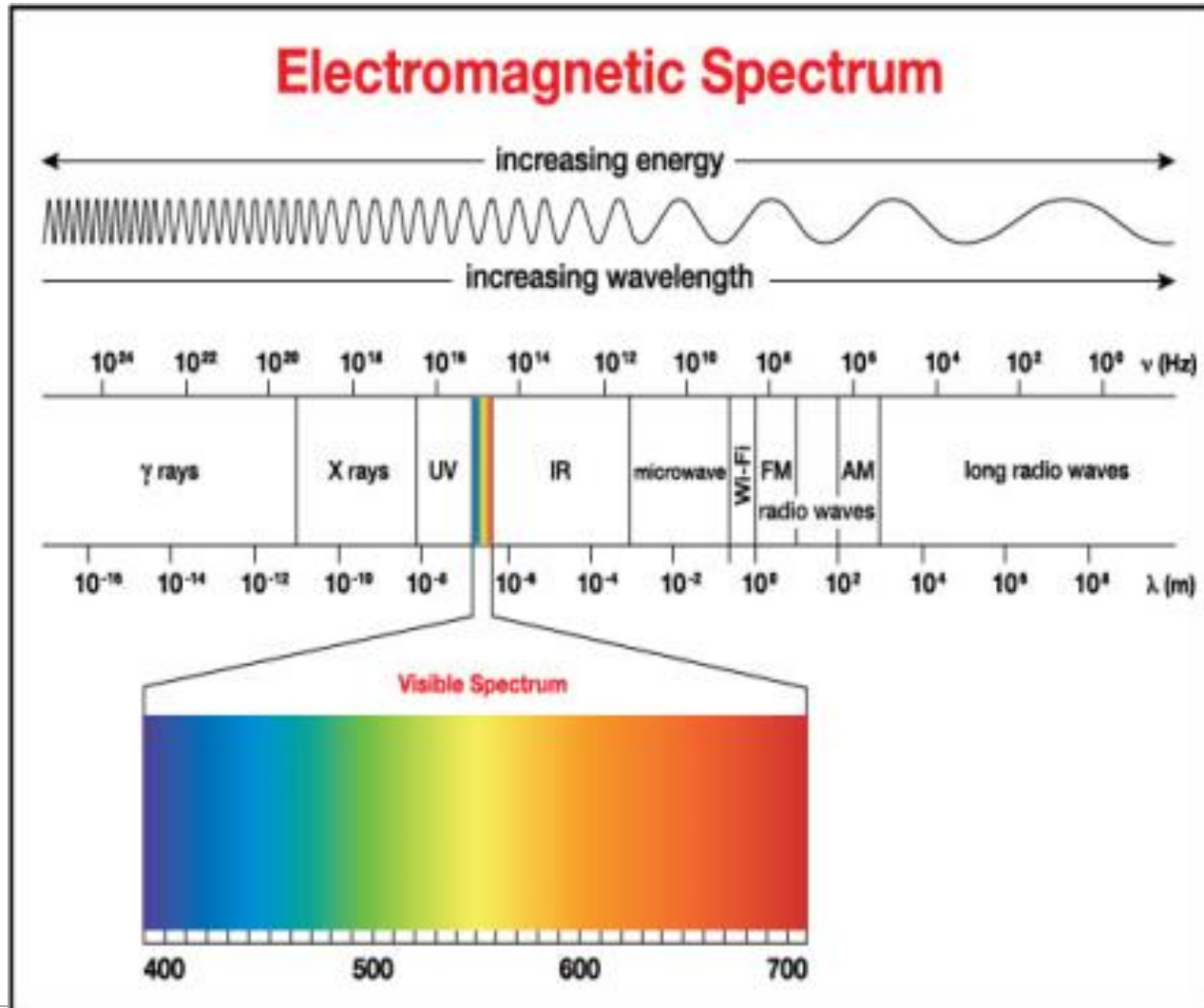
- Science and Engineering Practice:

- Developing and using Models
 - Develop a model to predict and/or describe a phenomenon

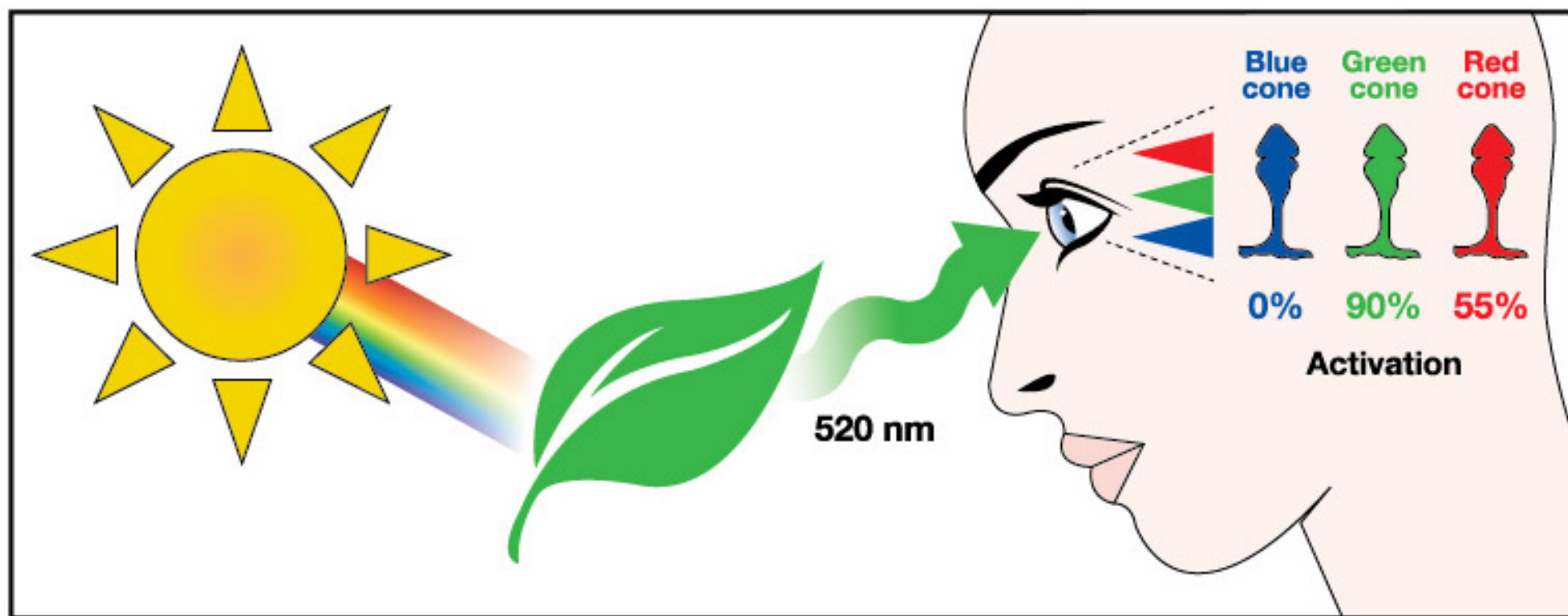
- Crosscutting Concept:

- Within a natural system, the transfer of energy drives the motion of matter

Background: The Electromagnetic Spectrum



How do we see color?



TI-84 Plus CE Resources:

- Step-by-Step YouTube videos designed for the teacher with a detailed, demonstration of each challenge.

The videos are intended to guide those new programming in general, programming on the TI-84 Plus CE and programming with the TI-Innovator Hub.

<https://www.youtube.com/playlist?list=PL17Fe0ZmhCR-PAuv5pw85vxrtLu5YGeWI>

- 10 Minutes of Code with TI-Innovator:

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

- 10 Minutes of Code Teacher's Lounge (TI-84 Plus CE): Select link to download zip files with PDF's of all of the 10 Minutes of code lessons, teacher and student documents.

<https://education.ti.com/en/activities/ti-codes/84/teacher-lounge>

- TI-Innovator Technology eGuide

https://education.ti.com/html/webhelp/EG_Innovator/EN/index.html

- TI-Innovator Hub Commands

https://education.ti.com/html/webhelp/EG_Innovator/EN/content/eg_innovsys/resources/pdf/ti-innovator_hub_commands_en.pdf

- TI-84 Plus CE TI-Basic Programming Guide

https://education.ti.com/html/webhelp/EG_TI84PlusCECode/EN/index.html

- TI-84 Plus CE Reference Guide

https://education.ti.com/html/webhelp/EG_TI84PlusCECode/EN/index.html

PROJECT FINAL CHALLENGE

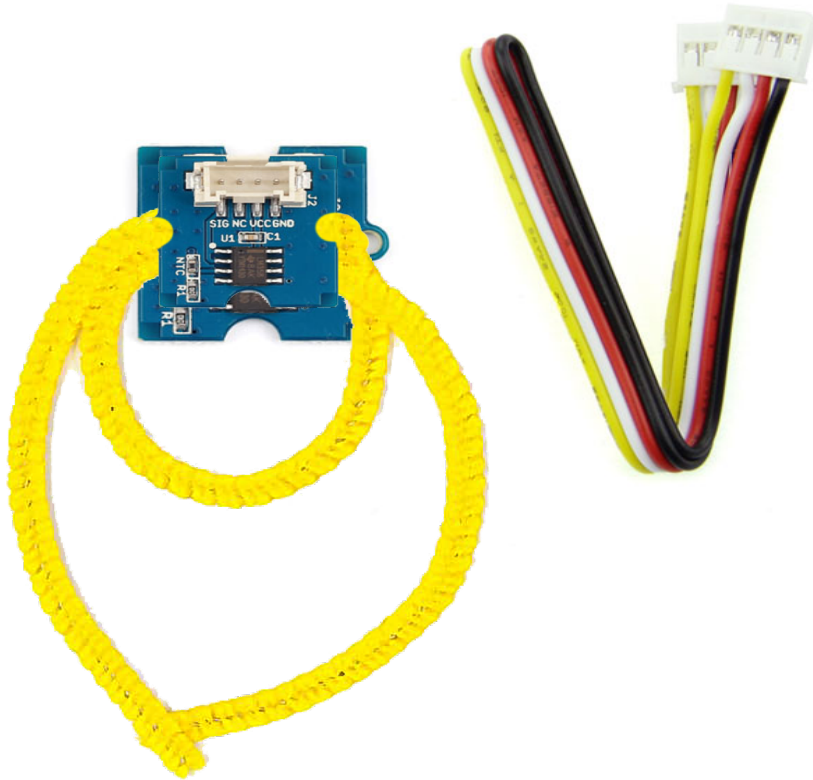


Final Challenge

Digital Mood Ring with TI-Nspire and TI-Innovator™ Hub student STEM project

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

Chenille Wire (AKA “Pipe Cleaner”)



Defining your digital mood ring

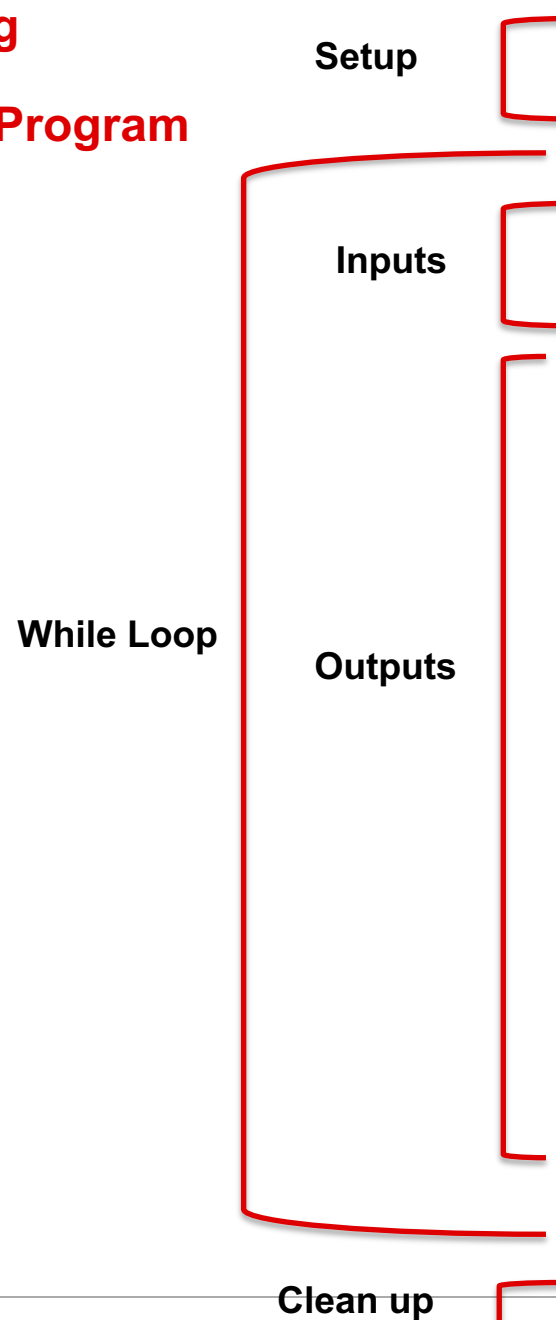
Temperature level definitions	Mood Message	Color Name or Description	Color RGB Code	Other Output (Optional) Sound? Vibration?...

Temperature level definitions	Mood Message	Color Name or Description	Color RGB Code	Other Output (Optional) Sound? Vibration?...
$t < 24$	"You are Stressed"	off	0 0 0	
$t \geq 24$ and $t < 26$	"You are Nervous"	red	255 0 0	
$t \geq 26$	"You are Calm"	magenta	255 0 255	

Digital Mood Ring for TI-84 Plus CE Simple Example Program

```
Send("CONNECT TEMPERATURE 1 TO IN 1")
0→K
Output(8,1,"PRESS CLEAR TO QUIT")
While K≠45
getKey→K
Send("READ TEMPERATURE 1")
Get(T)
Output(4,1,"TEMP(°C)= ")
Output(4,11,T)
If T<24
Then
Send("SET COLOR 0 0 0")
Output(5,1,"YOU ARE STRESSED")
End
If T≥24 and T<26
Then
Send("SET COLOR 255 0 0")
Output(5,1,"YOU ARE NERVOUS ")
End
If T≥26
Then
Send "SET COLOR 255 0 255"
Output(5,1,"YOU ARE CALM  ")
End
Wait 0.5
EndWhile
Send "SET COLOR 0 0 0"
```

Digital Mood Ring for TI-84 Plus CE Simple Example Program



```
Send("CONNECT TEMPERATURE 1 TO IN 1")
0→K
Output(8,1,"PRESS CLEAR TO QUIT")
While K≠45
getKey→K
Send("READ TEMPERATURE 1")
Get(T)
Output(4,1,"TEMP(°C)= ")
Output(4,11,T)
If T<24
Then
Send("SET COLOR 0 0 0")
Output(5,1,"YOU ARE STRESSED")
End
If T≥24 and T<26
Then
Send("SET COLOR 255 0 0")
Output(5,1,"YOU ARE NERVOUS ")
End
If T≥26
Then
Send "SET COLOR 255 0 255"
Output(5,1,"YOU ARE CALM  ")
End
Wait 0.5
EndWhile
Send "SET COLOR 0 0 0"
```

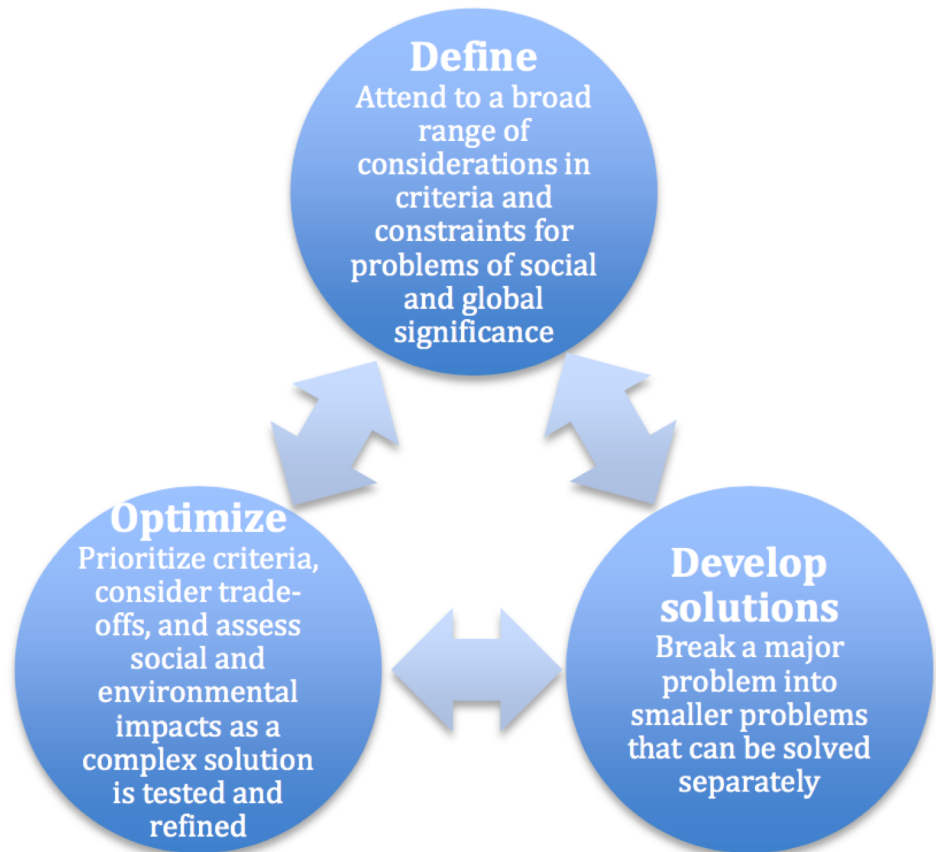

PROCESS OF ENGINEERING DESIGN AND COMPUTATIONAL THINKING

Engineering Design Process

- Define Problem
- Design a proposed solution
- Present group's design to class.
- Listen to feedback and make appropriate changes.
- Build and test the design.
- Use test results to make appropriate and informed changes to design.
- Present your solution to class.

Engineering Design Process

- **Defining** and delimiting engineering problems involves stating the problem to be solved as clearly as possible in terms of criteria for success, and constraints or limits.
- **Designing** solutions to engineering problems begins with generating a number of different possible solutions, then evaluating potential solutions to see which ones best meet the criteria and constraints of the problem.
- **Optimizing** the design solution involves a process in which solutions are systematically tested and refined and the final design is improved by trading off less important features for those that are more important.
- It is important to point out that these component ideas do not always follow in order, any more than do the “steps” of scientific inquiry. At any stage, a problem-solver can redefine the problem or generate new solutions to replace an idea that just isn’t working out.



Authors: NGSS Lead States

Title: Next Generation Science Standards: For States, By States – Engineering Design

Copyright Date: 2013

Source: Next Generation Science Standards

http://www.nextgenscience.org/sites/default/files/Appendix%20I%20-%20Engineering%20Design%20in%20NGSS%20-%20FINAL_V2.pdf



TEXAS INSTRUMENTS

Computational Thinking

NGSS Science and Engineering Practices

Using Mathematics and Computational Thinking

<http://ngss.nsta.org/Practices.aspx?id=5>

Value of Pseudocode

•What is Pseudo code?

- Pseudo code is basically written instructions of what a program should do to complete an operation. The pseudo code is written in your speaking language. You simply write down the steps of an algorithm or procedure to make it easier for you or someone else to read it and convert it to a real programming language.

•Why was it made/is it used?

- It was made to help you specify the steps of a procedure, process or an algorithm without using a programming language. One possible alternative is to use flowcharts.

Planning your program

Important steps in your program	TI-Basic commands to implement steps in your program
1.	
2.	
3.	
4.	
5...	

Using a flowchart to plan:

Sample, simple flowchart:

