Unit 3: BRIGHTNESS, IF, and WHILE | **Skill Builder 3: BRIGHTNESS and COLOR**

In this third lesson of Unit 3, we'll make use of the BRIGHTNESS value to control the COLOR LED.

Objectives:

- Read the light sensor and control either the COLOR LED brightness or the sound coming out of the speaker
- Use conversion formulas to change from BRIGHTNESS values to COLOR values

---

We will build a product that reacts to the brightness in the room. The brighter the room lighting, the brighter the COLOR LED. The tricky part here is *converting* the BRIGHTNESS value into an appropriate COLOR value. Think of this product as an auto changing (dimming or brightening) light bulb.

BRIGHTNESS ***B*** *ranges from 0 to 100.*
COLOR ***C*** (all three channels) can vary from 0 to 255.
How do we convert from B to C?

Setting up the Program

1. Start a new program, and name it BRIGHT3.
2. Add the commands **ClrHome** and **Disp** to display the title as shown to the right.
3. Set the variable ***B*** as shown.
4. Add a **While** loop to read the brightness using **Send ("READ BRIGHTNESS")** and get the brightness variable with **Get(B)**.

```
NORMAL FLOAT AUTO REAL RADIAN MP
PROGRAM:BRIGHT3
:ClrHome
:Disp "LIGHT 2 COLOR"
:2→B
:While B>1
:Send("READ BRIGHTNESS")
:Get(B)
:
:
:
```

5. Use the variable ***C*** to represent the COLOR value that we'll send to *all three channels* of the COLOR LED. The conversion factor is 2.55. That is, **2.55*B→C**.
   Check this formula with the two pairs of values given.
   When B=0 then C=2.55*0 ==> 0; when B=100, then C=2.55*100 ==> 255.
6. Add a **Send("SET COLOR** statement before the **End** of the loop. This statement will control the brightness of the color LED.

```
NORMAL FLOAT AUTO REAL RADIAN MP
PROGRAM:BRIGHT3
:ClrHome
:Disp "LIGHT 2 COLOR"
:While 1
:Send("READ BRIGHTNESS")
:Get(B)
:2.55*B→C
:Send("SET COLOR
:End
```

```
NORMAL FLOAT AUTO REAL RADIAN MP
PROGRAM:BRIGHT3
:ClrHome
:Disp "LIGHT 2 COLOR"
:2→B
:While B>1
:Send("READ BRIGHTNESS")
:Get(B)
:2.55*B→C
:Send("SET COLOR eval(C) e
val(C) eval(C)")
```

7. Finally, complete the **SET COLOR** command by using **eval(C)** three times (once for each of the three color channels).
   - When all three channels have the same value, then the color of the LED is white and the brightness of the LED changes depending on that value.
8. Connect the TI-Innovator™ Hub, and run the program.
9. Change the brightness by pointing the light sensor at different things. Watch the intensity of the COLOR LED on the TI-Innovator Hub.
10. To end the while loop and hence the program, cover the sensor completely so that the value of **B** goes below 1.

You might want to add some **Output(** statements to your program to see the values of ***B*** and ***C***.

But wait! The effect is wrong! The *darker* the room lighting, the *brighter* the LED should be! How can you reverse the effect?

Another Challenge: Consider changing the program so that different brightness values produce different colors.