

Design

According to the plan, a light sensor will be needed as an input, and a LCD board as a display output.

Open ezCircuit Designer and add the components that were chosen in the plan.

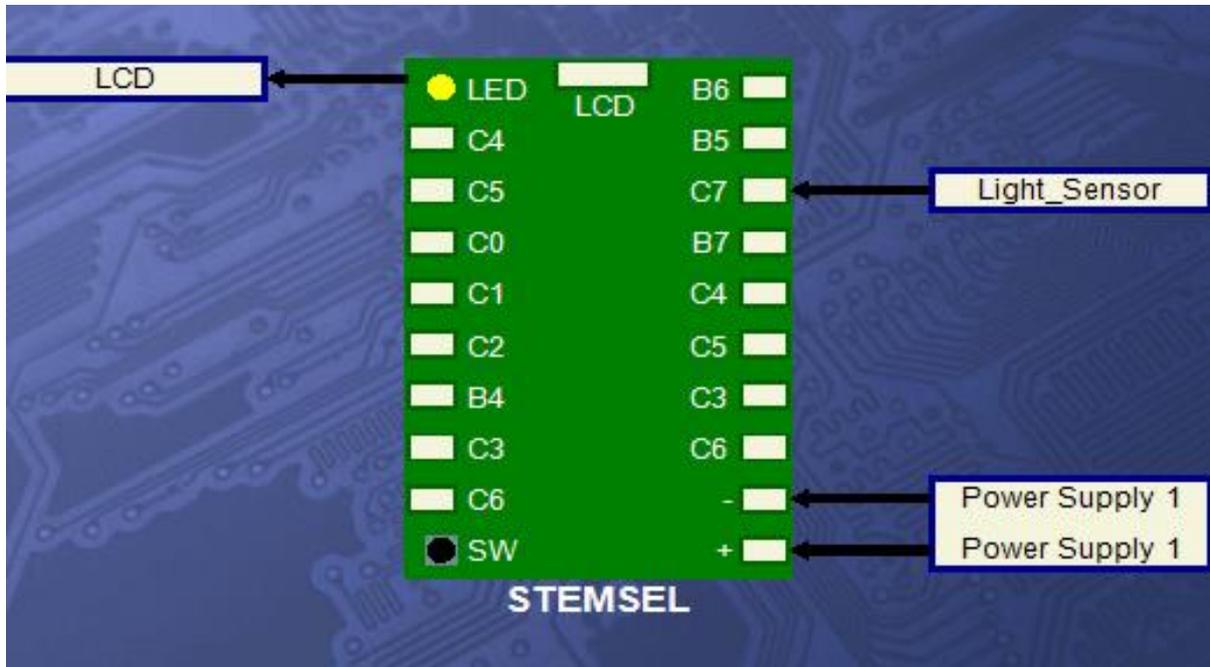


Figure 2: Circuit Design

Build

Use the ezCircuit Designer I/O diagram to connect the hardware. Remember that black wires go into the negative port, red wires go into the positive port, and the white wires go into the ports specified in the design. For the LCD, there is a port allocated to it, plug the head of the LCD into it.

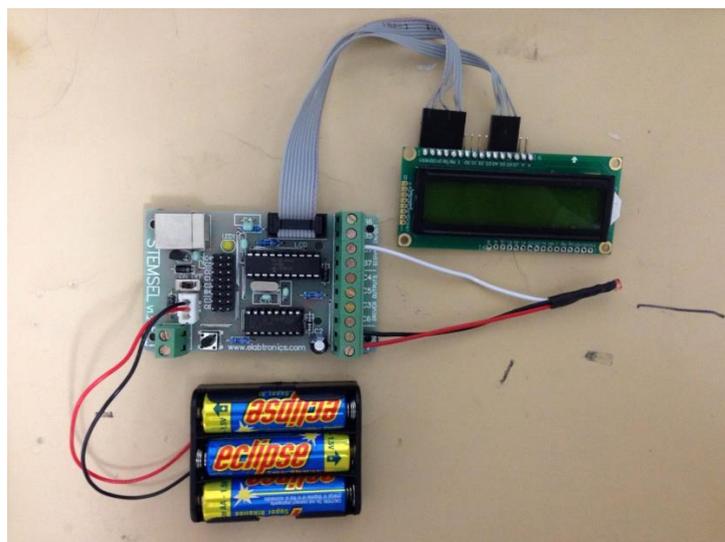


Figure 3: Circuit

Programming

Use CoreChart to program the functions of the Light Level Display. After the output has been checked using the test routines, delete them so that we can start the new program.

Here are some steps for reference:

1. According to the plan, the first thing to do is to get inputs from the LDR to detect the initial light level in the room. Use an AnalogIn icon to read the value from the LDR (name these “light_level_max” and “light_level_min”, respectively).
2. Use LCDFormat to clear the LCD and set it so that it has “No Delay”, this prepares it to display information.
3. Now, the initial setup for the program is finished. The next step is to read in the new input and compare. Use the “AnalogIn” icon to read the light level in the room again (name this “light_new”).
4. Use a “Compare” icon to compare the ‘light_new’ value with the ‘light_level_max’ value. Tick both the ‘Above’ and ‘Below’ checkboxes.
5. As in the design, if the ‘light_new’ is below ‘light_level_max’, ‘light_level_max’ keeps its current value. If ‘light_new’ is above ‘light_level_max’, ‘light_new’ will be set to new ‘light_level_max’.
6. After the comparison for the maximum light level, the next step is to compare for minimum light level. Similar to the maximum value part, if the ‘light_new’ is above ‘light_level_min’, ‘light_level_min’ keeps its current value. Otherwise, set ‘light_new’ to be the new ‘light_level_min’.
7. After the comparisons, the next step will be display those values on the LCD. Use “LCDMessage” and DispNumberLCD to display ‘MaxLight’ and the value of ‘light_level_max’.
8. Use “LCDFormat” to move the cursor to the second line of the LCD. Repeat the “LCDMessage” and “DispNumberLCD” icons for the minimum value and call the minimum value “MinLight”.
9. Use the “TimeDelay” icon to hold the display for 1 second
10. Use the “LCDFormat” to clear the LCD with no delay and prepare it for the next display.
11. Use the “Group” icon to group the steps from 7 to 9, name this “Display”.
12. Since the program only needs to repeat the parts that read new input, compare and display the results, an Address icon (named “Complight”) needs to be added in front of the step that reads in the new input from the LDR. Add a “GoTo Complight” at the end, and then the program will only repeat the section from “Complight” to the “Goto”.

Now, your code should look like this:

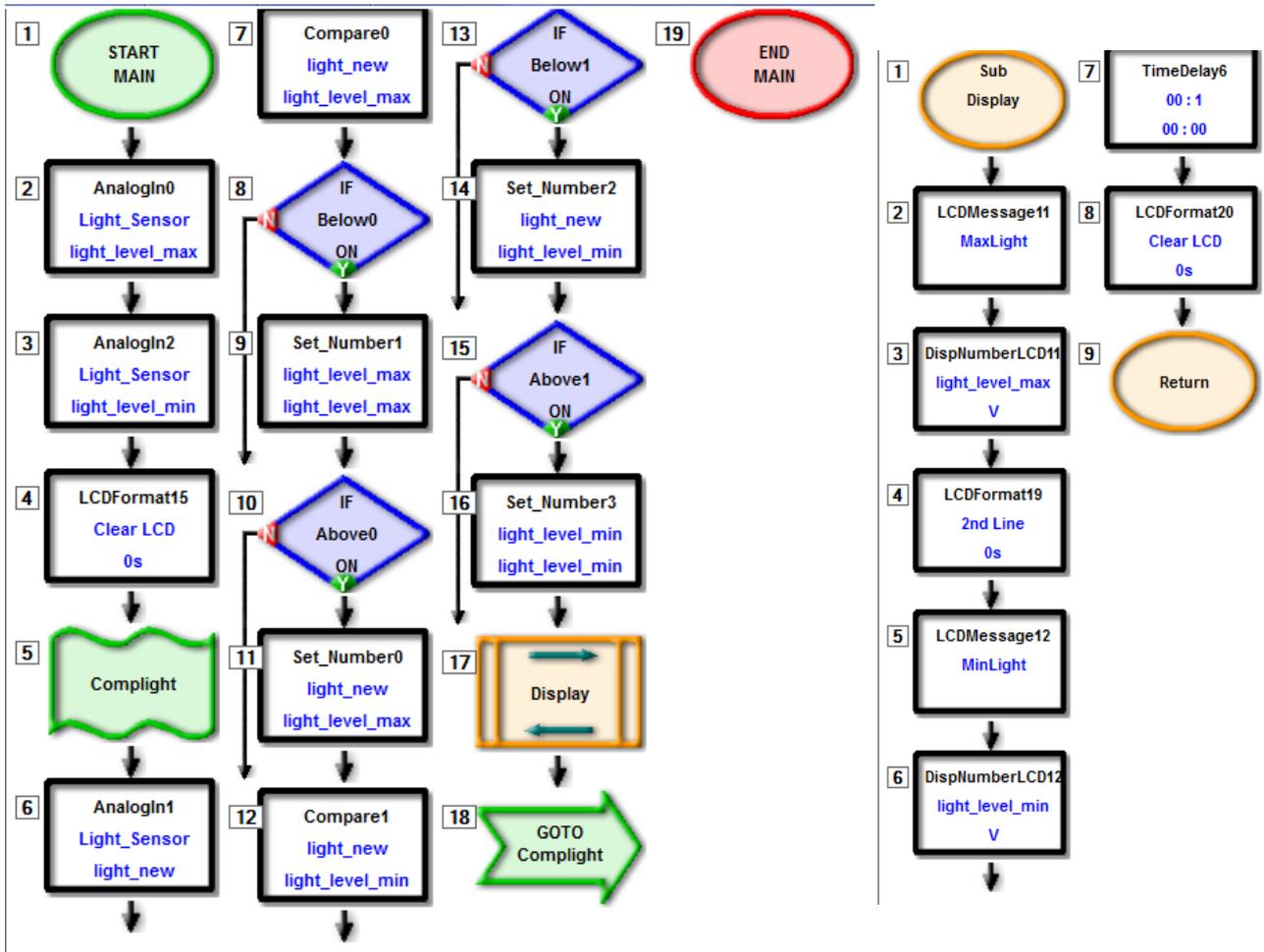


Figure 4: CoreChart Code & Sub Group Display

Extension

If the light level detector works properly, take it with you and walk around the room, notice how the light level changes in the room by reading the display. After examining the whole room, compare your values with your friends to see if you end up with same values. If not, why is it different?

Summary

In this project, we made a light level detector that can store the brightest and dimmest light levels in the environment and display it back to the user. After this project, we should understand how to use a LCD to display a message and how to move the cursor between lines on the LCD. Also, after this project, you should be more familiar with how to use the Compare and Address icon. This project is not very complex. It can be further developed to detect the outside light level to control a blind or to close a window.