STEMSEL Beginners Project 4: Smart Street Light

Problem

We use lots of street lights to light the roads at night. However, all of those lights use a lot of electricity which is entirely wasted if they turn on when it is not dark. We want to use our STEMSEL boards to make a smart street light that is only on when it is dark.

Background

Street lights should only come on when it is dark, either at night or during very cloudy days. Having the street lights only come on when they are needed saves electricity. This is clearly a good thing, since the fossil fuels like gas and coal that we burn to generate electricity create greenhouse gasses which contribute to global warming. This project allows you to create a smart street light.

Ideas

So how can we turn on the street light only when it is needed?

We could use a timer to turn the light on at say 6:00 in the evening and off again at 6:00 in the morning, but the problem with this is that the days are not always the same length. Because of this, sometimes the streetlight might be on when it is still light, or still off when it is already dark.

The best solution would be if the streetlight could see if it was dark or light, and turn itself on or off automatically. Can we do this using our kits?

Plan

We have light dependant resistors (LDR) in our kits which we can use to check if it is bright or dark. We can simulate the lightbulb in a streetlight using the smaller lightbulb in our kits. First, we will use the LDR to check he light level, then the microchip can turn on the lightbulb when the light level is low, and turn it off when the light level is high.
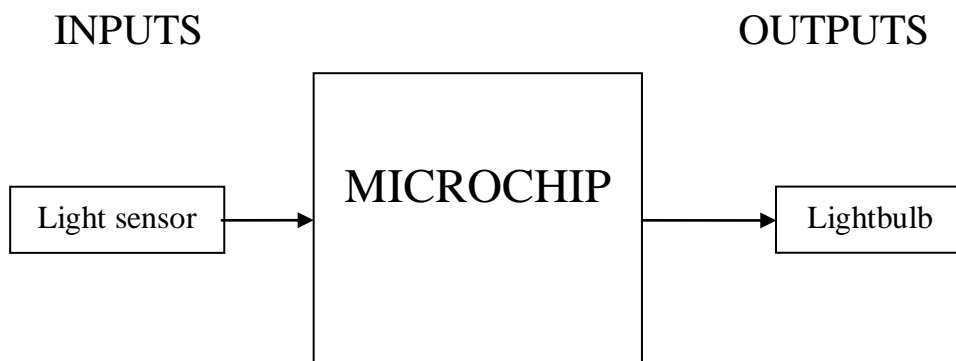
INPUTS   OUTPUTS

Light sensor → MICROCHIP → Lightbulb

Figure 1

Light level

Bright

● If the light level is above our threshold, it is too bright and the light should turn off.

Threshold

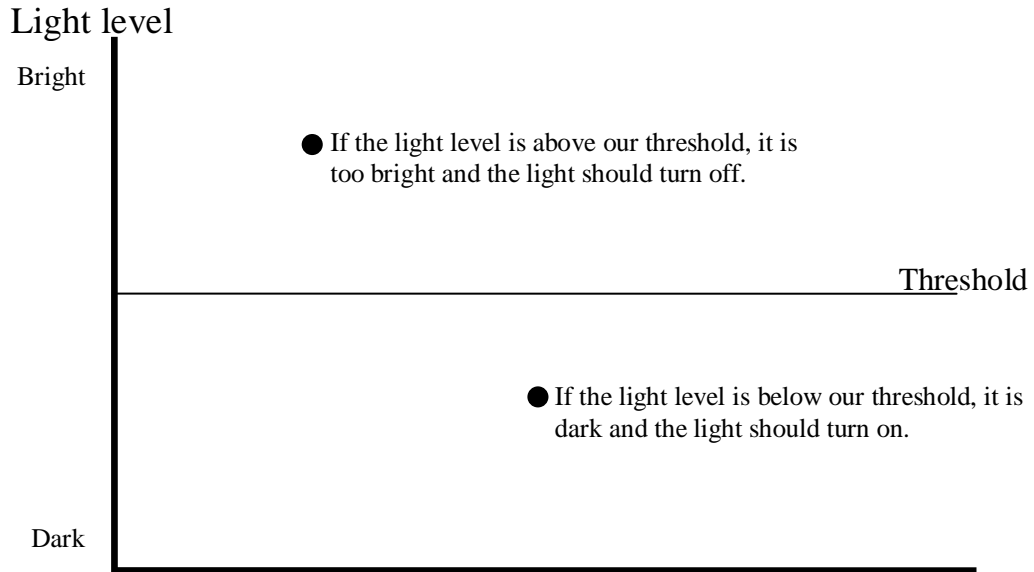● If the light level is below our threshold, it is dark and the light should turn on.

Dark

Figure 2

Design the Circuit

Use ezCircuit Designer to construct an input/output (I/O) diagram. Use a LDR (light dependant resistor) for the light sensor, and a driver icon to add the lightbulb. Remember to right click to rename the components.

Figure 3

Build the Circuit

Use the ezCircuit Designer diagram to connect the hardware. Remember that turning the screws clockwise will close the clamps, and turning the screws anticlockwise will open them. All black wires should go in the negative (-) terminal, red wires go in the positive (+) terminal, and white wires go in the terminal we designated in ezCircuit Designer.
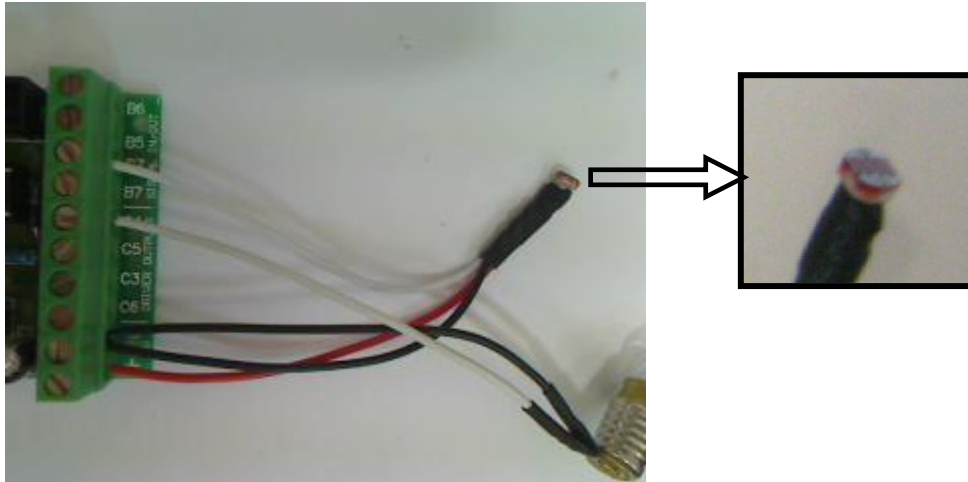

Figure 4

Programming

Click to "Send to CoreChart" button so we can start to program the functions of the smart street light.

1. First there will be a driver test routine to make sure the lightbulb is working correctly. Connect the STEMSEL board to the computer (don't forget to hold the button down!), then click the "Send Program To Chip" button.

2. After the chip has been successfully programmed, remove the programming cable and turn the STEMSEL on. The lightbulb should turn on briefly, then turn off again.

3. Delete the test routine so we can write our own program

4. According to the plan, we first need to check if it is bright or dark. Click the Inputs button, and select **Analog_In** from the Icon Properties list. Place the new icon just below START.

5. Double click the new icon, and use it to read the LightSensor, and save the result as LightLevel.
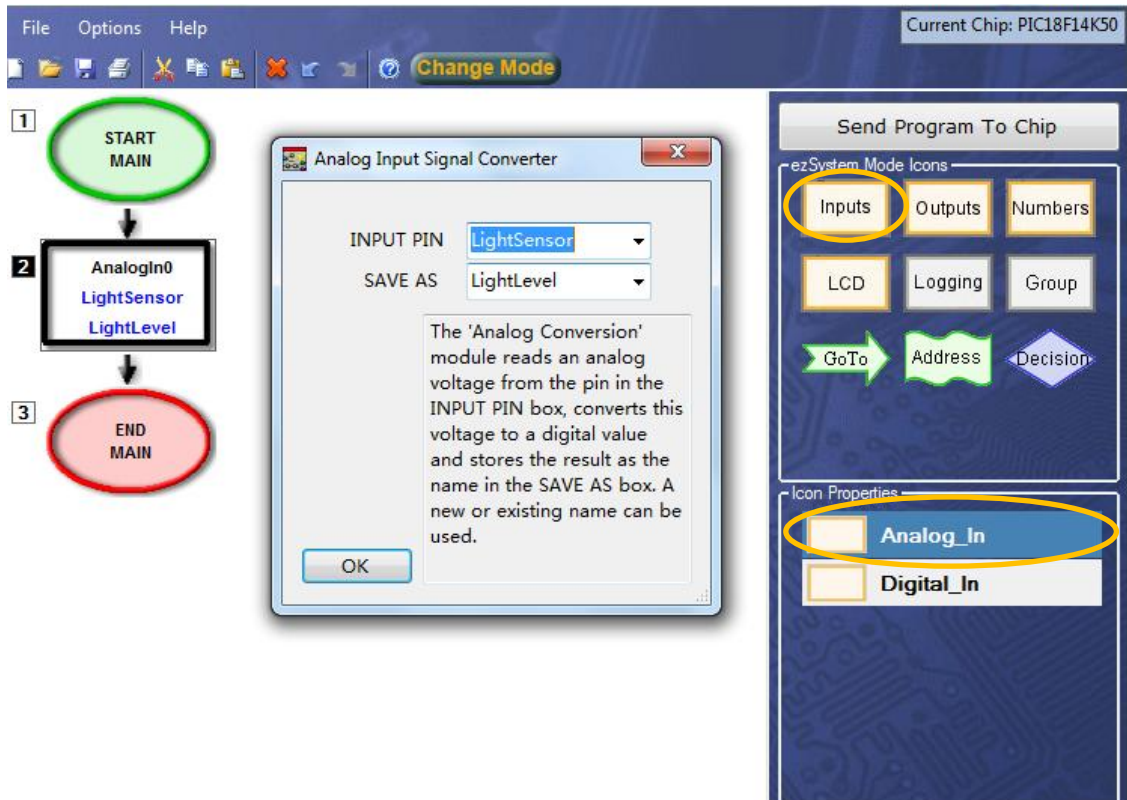
Figure 5

6. We will use a threshold value to decide if it's light or dark, just like the graph in our plan. Click the Numbers button, then select Compare from the Icon Properties list. Place the new icon below the AnalogIn.

7. Double click the new icon, and use it to compare LightLevel with 3V. Ensure you either type a capital 'V' after the 3, or tick the voltage box. Also tick the above and below checkboxes.
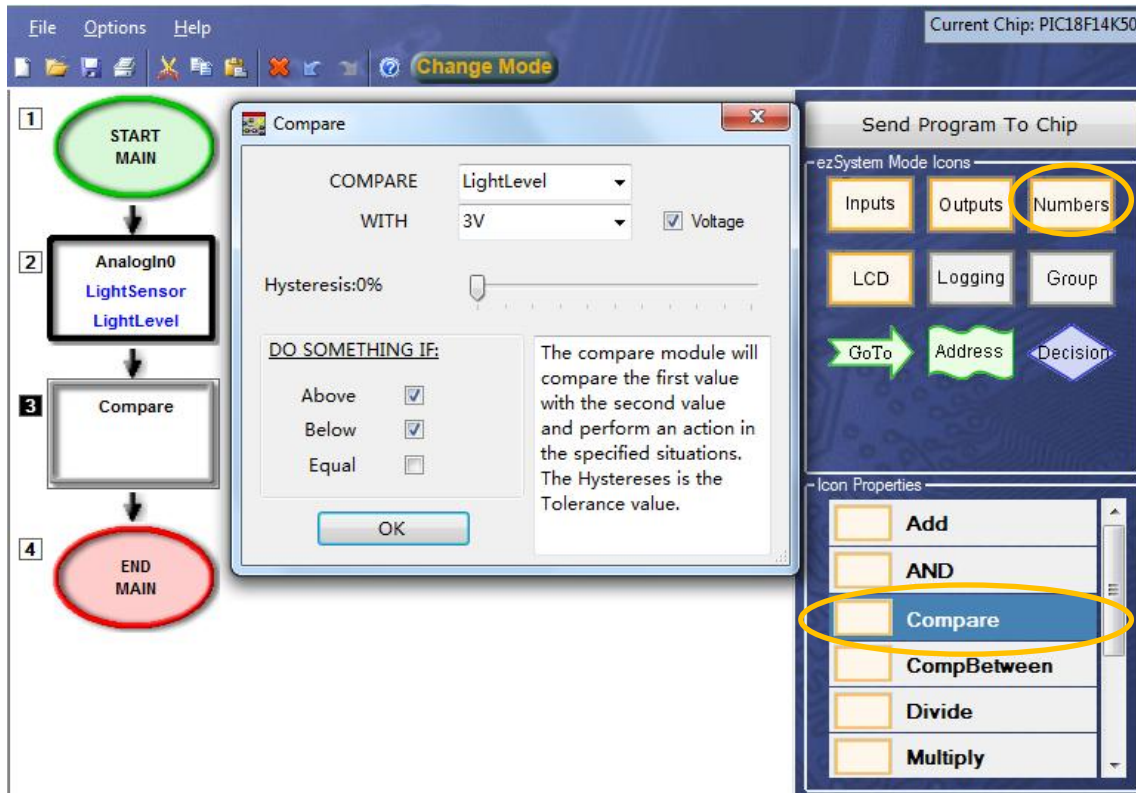
Figure 6

8. The compare icon should have generated two Decision icons, one for above and one for below. We will use these to turn on and off the streetlight.

9. First is the Below case. Again looking at the graph, we can see that if the light level is below our threshold of 3V, then it is dark and the streetlight should turn on. Click the Outputs button and use an OnOffPin icon to turn the lightbulb on. Add it just below the Below decision icon, it should overwrite the default GoTo Start icon that is already there.

10. Similarly, if the light level is above 3V, it means it is bright and the streetlight should turn off. Use another OnOffPin icon to turn the lightbulb off.

11. At the moment, the program will go through once and then stop. We want our streetlight to keep checking the light level and turn on or off if it changes. Add a GoTo START icon at the end of the program
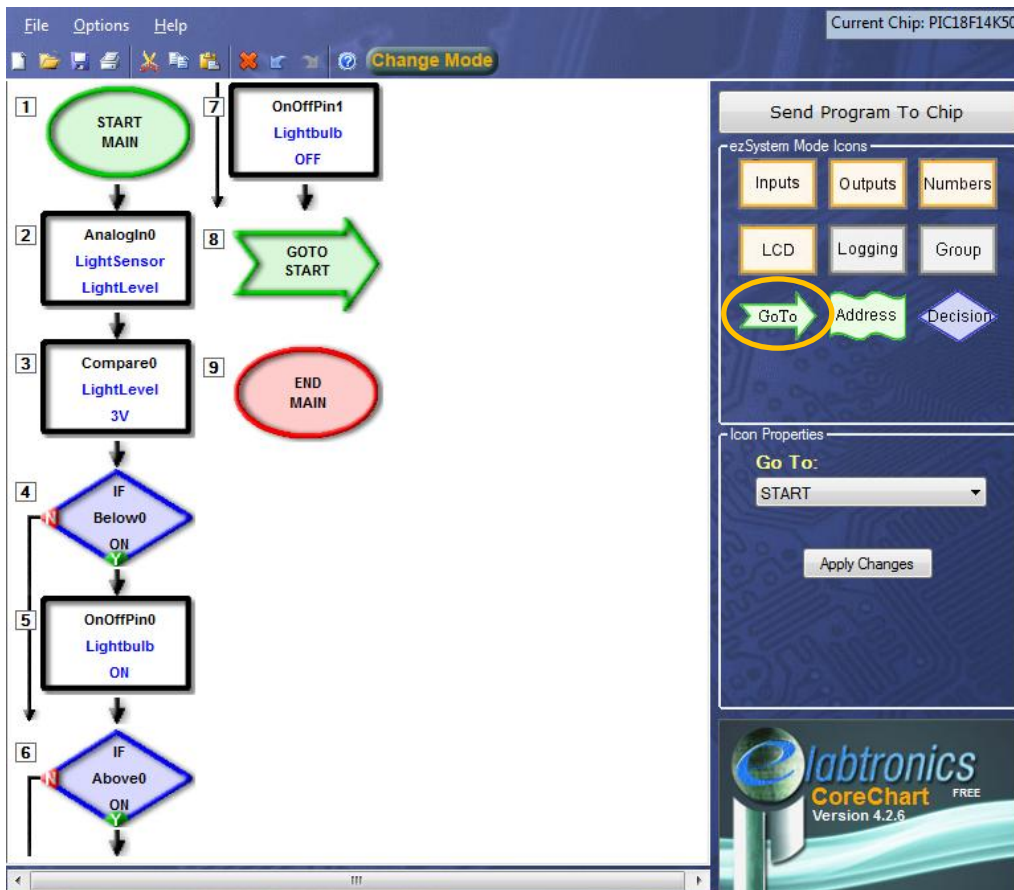
Figure 7

12. The program is now finished. Send the program to the chip and test to see if it works by either covering the LDR or pointing it at a light source. You may need to lower the threshold value if it can't turn on easily, or raise the threshold if it can't turn off easily.

Extension

Work through the following challenge and see if you can devise a solution to the problem. Make small changes to the program and test run the program regularly.

What will happen if the headlights of a car were to shine on the street light sensor at night? The program will turn the off the street light thinking that it is daylight.

1. Modify the smart street light program so that about five seconds of light on the light sensor is required to make the street light switch off.

Hint: Since the car head light is a spurious and momentary signal and will not expose the light sensor for a long period of time, a time delay can be used. Where can the time delay be inserted in the program?

Summary

It is important to light the streets at night so that people can see where they are going, but it is also important to save electricity. By ensuring our streetlight only comes on when it is dark, both of these considerations are addressed. During this project you should have learned how to use analog inputs to use a light sensor, and a compare icon to do different things depending on that input.