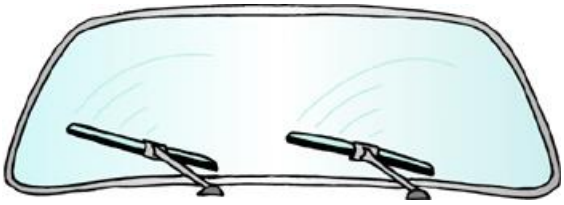
STEMSEL Automotive Project 1 – Windscreen Wiper Controller

Problem

We want to create a dial that controls a car windscreen wiper when turned using our STEMSEL board.



Background

Windscreen wipers are not the only component on a car controlled electronically. Most modern cars have functionality controlled by a buttons, levers, or dials. A few examples of this are moving windows up and down, locking doors, turning on the air conditioner and adjusting the air conditioner fan level. Some modern cars even have automated functionality for the convenience of drivers and safety. Automated features include automatic rain sensing windscreen wipers, headlights that turn on if it is getting dark and automatically setting the car's speed when using cruise control.

Ideas

What are the modes of a windscreen wiper? How do we detect water on the windscreen? Since the windscreen scenario cannot be reproduced directly using the STEMSEL kit, what other sensors can we use as a substitute? What speed should the wipers be set to in auto mode?

Plan

The windscreen wiper system will be using a potentiometer as the dial, a light sensor to detect "rain" and a motor simulate the wipers. To indicate which mode the wipers are in we will have the LED, to turn on for the auto mode.

The potentiometer has a range from 0V to 5V. The most anticlockwise position corresponds to 0V and the most clockwise position corresponds to 5V. There is no number scale on the potentiometer so the modes will be set between a range, rather than a specific number.

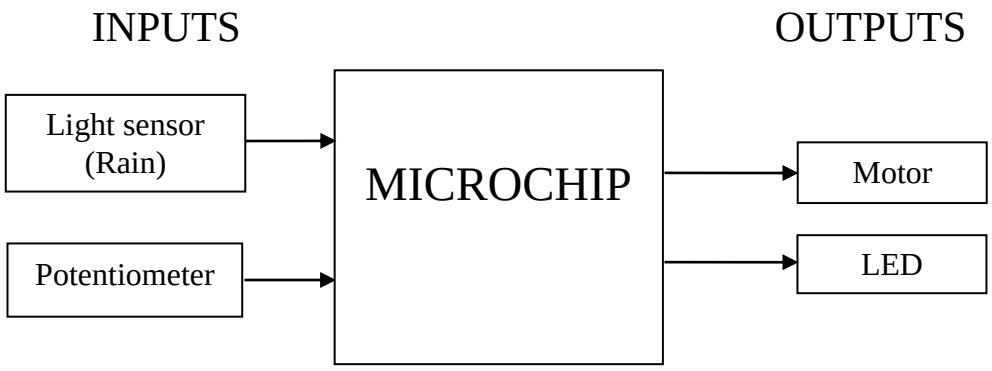


Figure 1: circuit plan

Copyright © eLabtronics 2015. All rights reserved

Design

According to the plan, we will need a potentiometer, light sensor, motor, a LED and the STEMSEL board to interpret the potentiometer value, then executes the correct operation.

Open ezCircuit Designer and start a new project, name it “WindscreenWiper”.

Now add all the components to the project as shown in the image below. Remember to right click to rename all components.

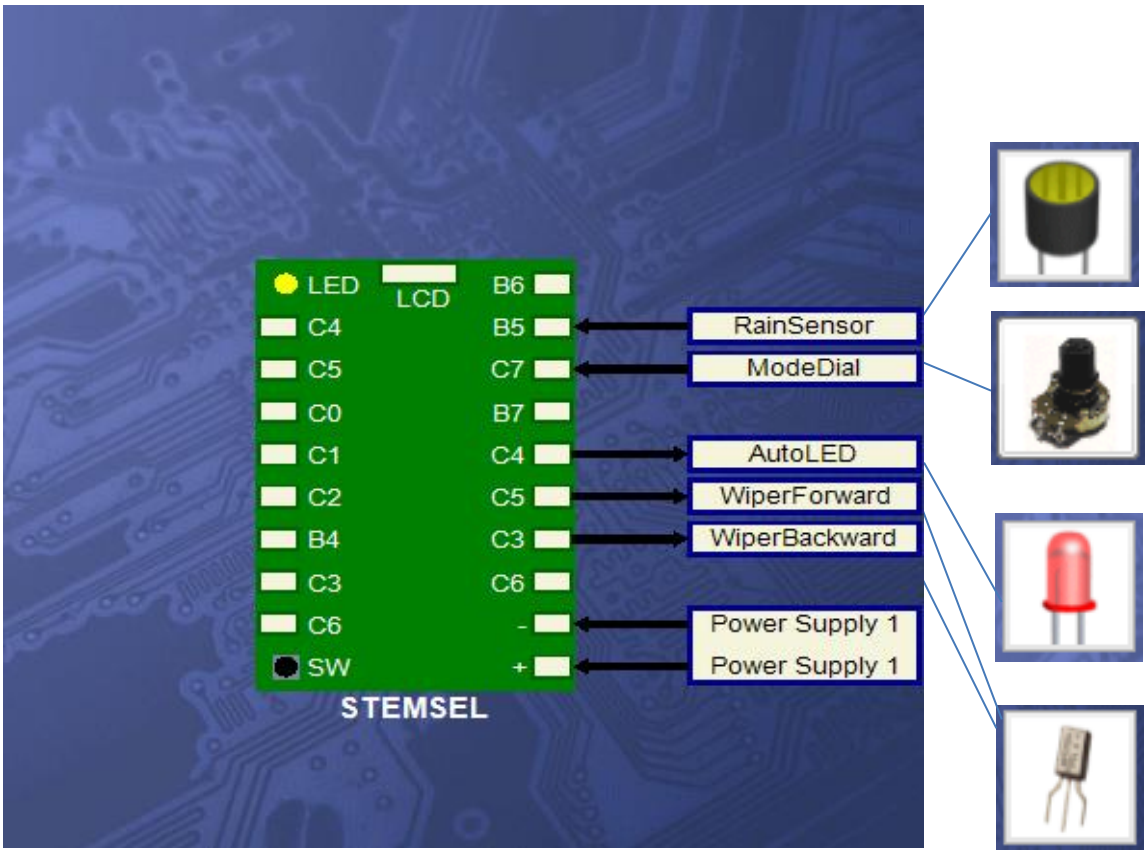


Figure 3: Circuit Design

Copyright © eLabtronics 2015. All rights reserved

Build the circuit

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

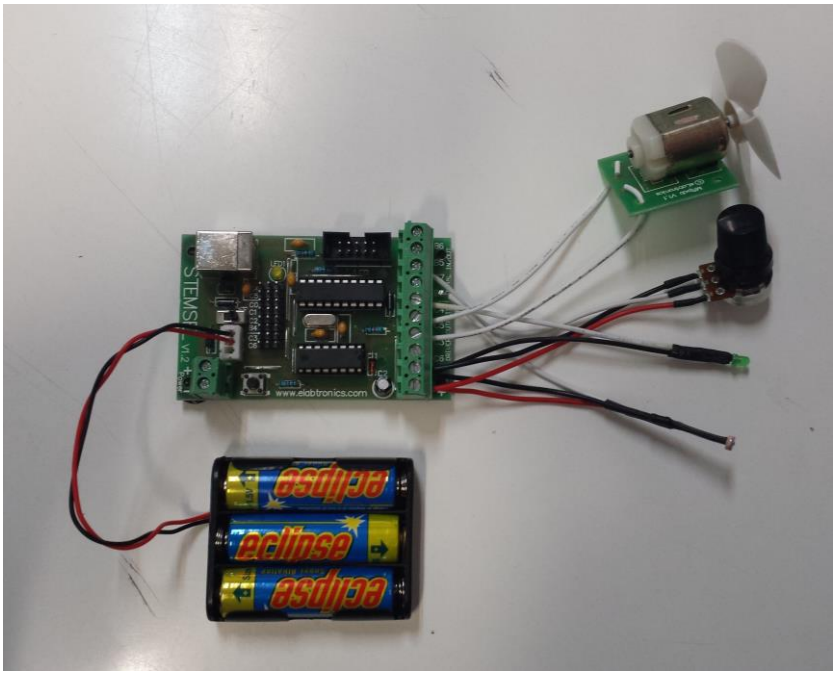


Figure 4: Circuit

Programming

Once you have assembled the circuit, send the design to CoreChart by clicking on "Send to CoreChart" button. After testing the circuit by sending the test routines to the chip, delete the test routines so we can write the windscreen wiper program.

1. First read the voltage from the potentiometer using the *Analog_in* icon from the *Input* button. Double click the *Analog_in* icon to edit, for *INPUT PIN* select *ModeDial* and set *SAVE AS* to *dialvolts*.

2. To determine which wiper mode we're in we need to check where the dial has turned to. We will program the *OFF* mode first. Click on *Numbers* button and select *Compare* and place it after *Analog_In*. Double click on the *Compare* icon to edit its properties, compare dialvolts with 1V. The *Voltage* checkbox should be ticked. For tick the *Below* checkbox.

3. The *Compare* icon will generate a *Decision* icon. When the dial is set below *1V* the mode will be *OFF*. Click on *Outputs* button and select *OnOffPin* and place just after the *Decision* icon. Edit the properties, for *OUTPUT PIN* selecting *AutoLED* and setting to *OFF*. Repeat this step for *WiperForward* and *WiperBackward*.

4. Now that the *OFF* routine is complete, group icons together, name the group ***OFF***.

Copyright © eLabtronics 2015. All rights reserved

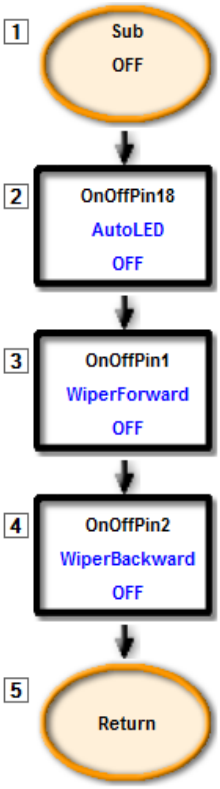


Figure 5: OFF Routine

5. Now work on the *FAST* mode. Select the *Compare* icon and place after the *OFF* group. Double click on the *Compare* icon and edit to compare *dialvolts* with *3V*, make sure the voltage and above checkbox is ticked.

6. For the *FAST* mode we will need to alternate from *WiperForward* and *WiperBackward* turning *ON* every half second (50 hundredths). To do this we select the *OnOffPin* and place it after the new *Decision* icon and edit it to turn *WiperForward ON*. Click on the *Numbers* button and select *TimeDelay* and place it after the *WiperForward* icon. Edit the *TimeDelay* icon for 50 hundredths. Select the *OnOffPin* and place it after this *TimeDelay* icon and edit it to turn *WiperForward OFF*. Now add another *OnOffPin* to turn *WiperBackward ON*. Add a *TimeDelay* of 50 hundredths after turning *WiperBackward ON*. Add an *OnOffPin* after the new *TimeDelay* to turn *WiperBackward OFF*. Group everything from this step and name it as ***FASTER***.

Copyright © eLabtronics 2015. All rights reserved

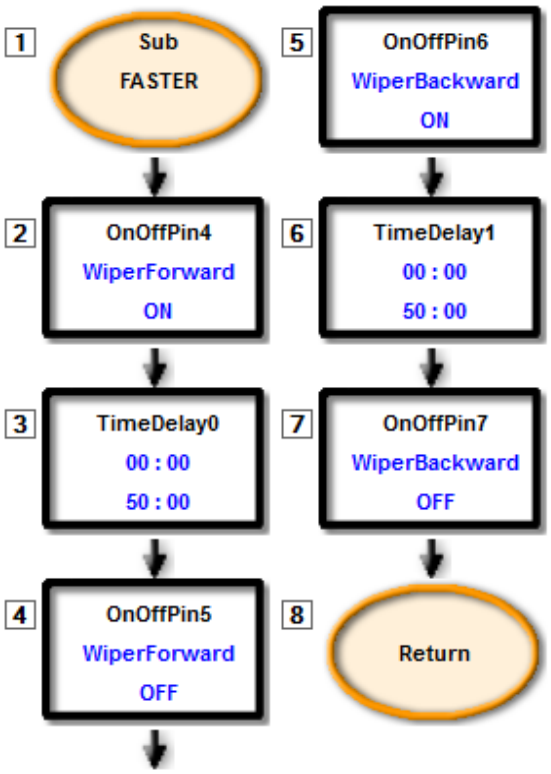


Figure 6: FAST Routine

7. If the dial remains in the same position for *any* mode it should remain in the same routine. Add a *GOTO START* at the end of the program. This allows the program to loop and execute endlessly.

8. For the *SLOW* mode place a compare between icon after the *OFF* routine. Edit it to compare *dialvolts* with a range between *2V* and *3V*. Repeat everything from steps 6 but change the time delays to 1 second. Repeat step 7 but compare with a range between 2V and 3V. Group everything together and name the group ***SLOW***.

Copyright © eLabtronics 2015. All rights reserved

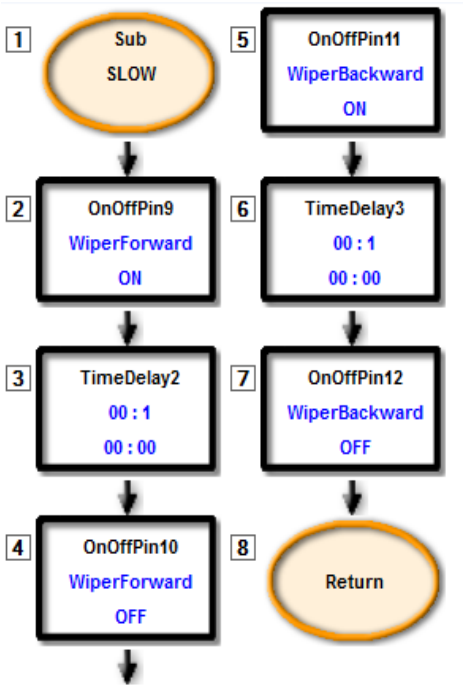


Figure 7: SLOW Routine

9. For the *AUTO* mode place a *CompareBetween* icon after the *OFF* routine. Edit to compare *dialvolts* with a range between *1V* and *2V*. After the *Decision* icon turn on the *AutoLED*, *WiperForward OFF* and *WiperBackward OFF*. Group these icons under the name ***AUTO*** and double click the newly made group to work inside it.

10. The *AUTO* mode will be determined by the *RainSensor* which will detect how much rain (in our case we're using a light sensor, so how much light) is detected. Add an *Analog_in* for the *RainSensor* and save it as *rainvolts*. Add a *CompareBetween* for *rainvolts* between *2V* and *3V* to set the wipers to *SLOW* mode. Also add a *Compare* icon for *rainvolts* above *3V* to set the wipers to *FAST* mode. Since we have already created routines for *SLOW* and *FAST* mode we can click on the *Group* button and select *SLOW* icon to place it under the *Decision* icon when comparing between *2V* and *3V*, and the *FASTER* icon to place under the *Decision* icon when comparing above *3V*. Remember at the end of the AUTO mode to turn off the *AutoLED*.

Copyright © eLabtronics 2015. All rights reserved

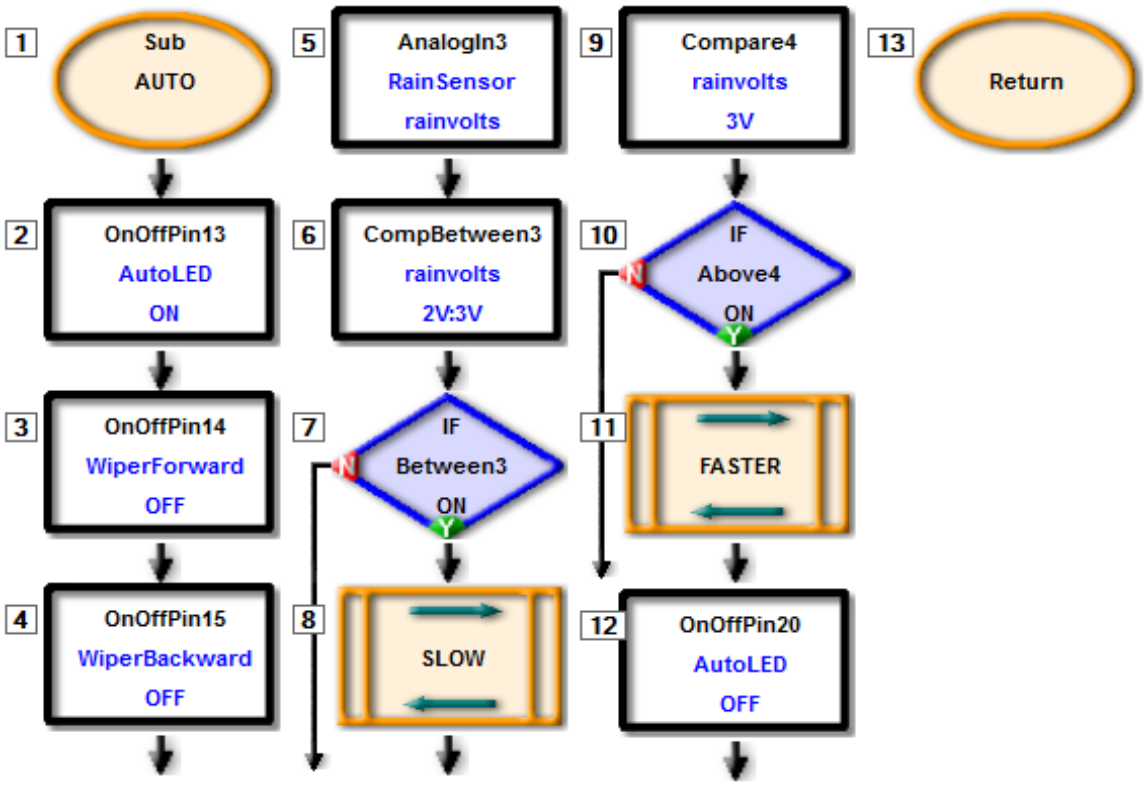


Figure 8: AUTO Routine

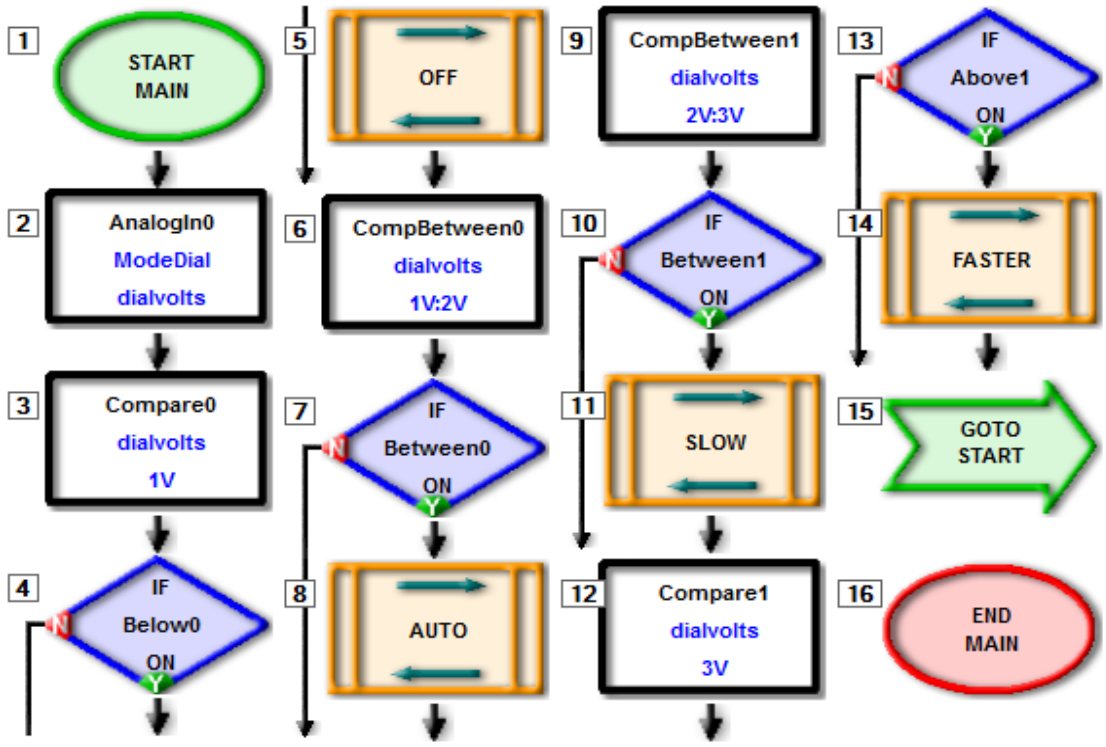


Figure 9: Everything put together

Copyright © eLabtronics 2015. All rights reserved

Activity/extension

What if we wanted to include an intermittent mode? Since the dial has a voltage range from *0V* to *5V* we can include 5 modes if each mode has a *1V* range. For example, *OFF* is between *0V* to *1V*, *INT* is between *1V* to *2V*, *AUTO* is between *2V* to *3V*, *SLOW* is between *3V* to *4V*, and *FAST* is between *4V* to *5V*.

How will the intermittent mode work? Can have the wipers turning forward for 1 second, turning backward for 1 second, and then waiting 3 seconds before repeating?

Summary

From this project you can see that the windscreen wiper system may seem easy at first, but when you really think about how each mode needs to work then it becomes quite complex. By designing each mode separately we can see a clearer picture of what is needed for a single mode rather than trying to do everything at once.

The *AUTO* mode demonstrates how tasks can be automated which is both convenient and safe. We also saw that the *SLOW* and *FAST* routines could be reused in the *AUTO* mode so it saves time to use them again rather than writing the same code again.

Copyright © eLabtronics 2015. All rights reserved