

STEMSEL LCD Project 8 : Vending Machine

Problem

How does a vending machine work? How can we use a POT to adjust the amount of money that a customer wants to insert? How can we use a LCD to display more than one source of information at once?

Background

Vending machines are commonly seen at schools, bus stations and parks etc. Products in the vending machine can be purchased by simply inserting the correct amount of money. Then the machine will calculate how much change should be returned and in the meantime the product will be ejected for the customer to collect.

In this project, a potentiometer will be used to adjust the amount of money inserted, a LCD will display the amount of money inserted each time and the microchip will keep a count of how much money has been input so far. Due to the complexity of this project change will not be returned instead put towards the purchase of additional items.

Ideas

How can we calculate the remaining amount of money? When should the LCD display the amount of money? How can the vending machine know if the amount of money inserted reaches the required amount? How does the vending machine indicate this?

Plan

For the vending machine workshop we can use a POT to adjust the amount of money that we intend to insert. A push-button can be added to inform the program that some money has been inserted so that the LCD will display the amount of money remaining to buy the item. Then a LED will be activated to indicate that the amount of money inserted is enough and the product will be sent out. The classifications of the components in terms of in/outputs are shown below.

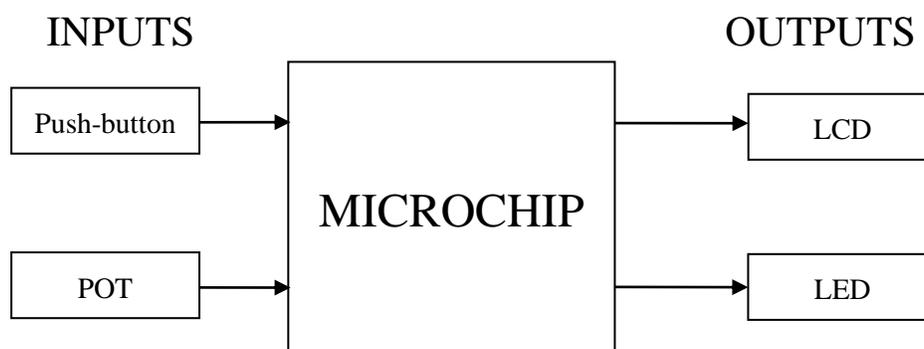


Figure 1: Inputs and Outputs

The vending machine should only contain three items, which cost 200c, 160c and 100c respectively. Imagine a customer will come and buy all the three items.

Design

Firstly, begin a new STEMSEL project and then add a LCD, a POT (potentiometer), a LED and a push-button on the STEMSEL controller board. All the components can be found in the circuit group. Make sure each component is inserted into the correct port.

The completed design should appear as shown below, where “InsertCoin” is the push-button, “Cent” is the POT, “ProductOut” is the LED and “Screen” is the LCD.

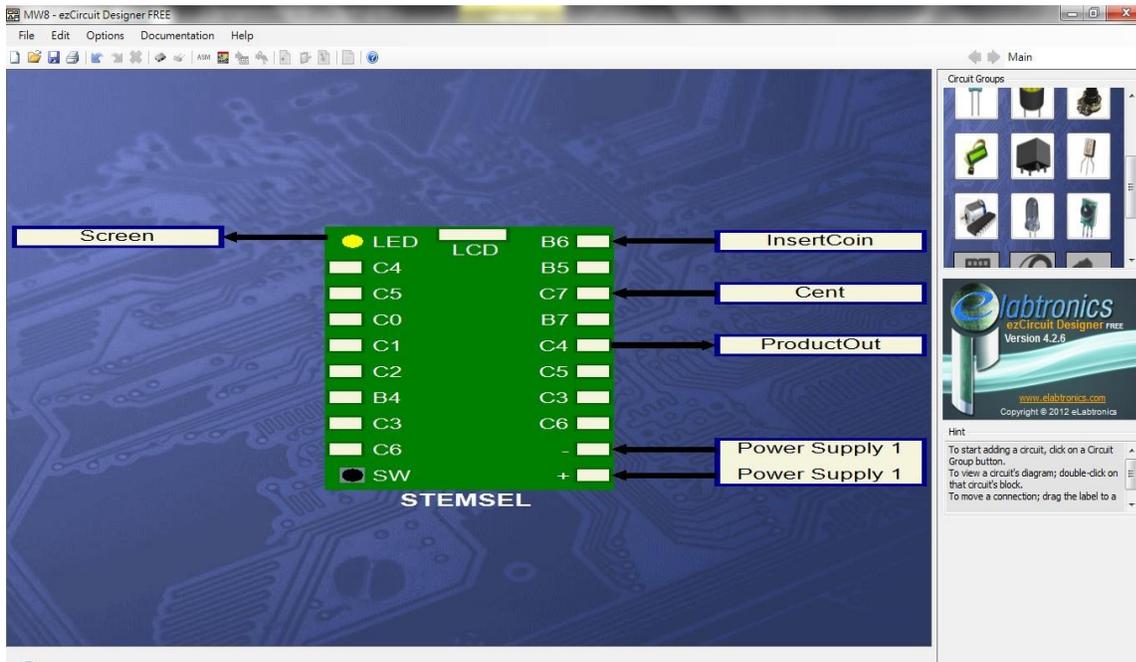


Figure 2: Circuit Design

Build

Once the circuit design is done, we can start building the circuit. Make sure that the black wires and the white signal wires are inserted into the negative (-) slot and the in/output slots respectively. Plug the LCD into the LCD port on the top of the board.

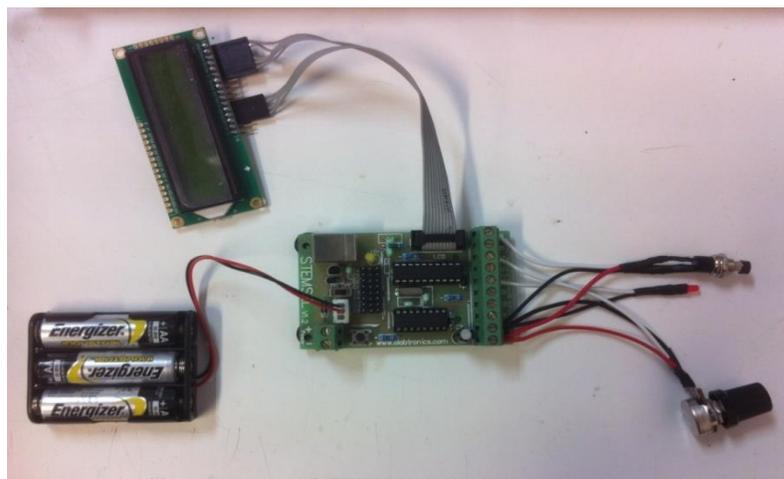


Figure 3: Circuit

Programming

With the design and assembly completed, send the design to CoreChart to program the microchip. Make sure the button is pushed when connecting the STEMSEL controller board to your computer. Before starting to program, make sure the STEMSEL controller board and the components are tested using the testing program. The testing program will be created once the design is sent to CoreChart.

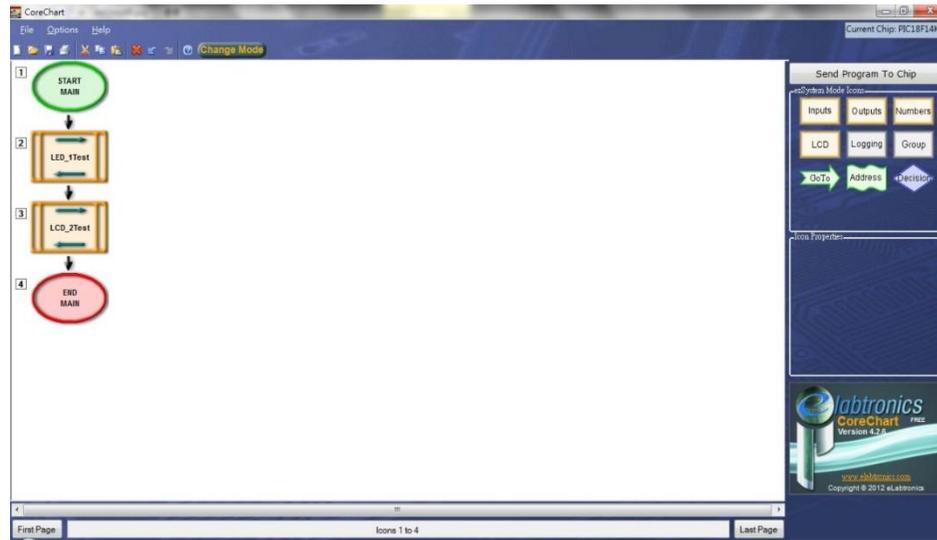


Figure 4: Testing Program

There are two main groups in the program as shown in Figure 5, which are “Money_Display” and “Trading”. In the “Money_Display” group, the value of the POT is shown as “Money”; when the push-button is pressed, the number of “Money” will be added to “Total”. In contrast, “Trading” will only be activated when “A3_PushButton” is pressed. The “Trading” group will subtract 100c, 160c or 200c from “Total” to simulate the payment.

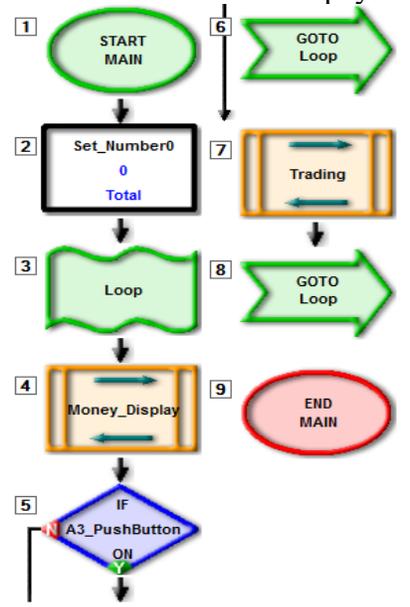


Figure 5: Main program

First let's work on displaying how much money is being deposited into the vending machine and how much money is already inside it, we will call this group "Money_Display".

1. Insert LCDFormat button and clear the screen at the beginning with no delay. use the "Set_Number" icon to create a "Total" Variable with the value of zero.
2. Show the message "Total" by using LCDMessage and the number "Total" by using DisplayNumberLCD.
3. Add an "AnalogIn" to record the value of the POT which we called "Cent", save as "Money".
4. Switch the LCD to the second row by using LCDFormat with no delay on the LCD and display the message "Money" and number "Money" by using LCDMessage and DispNumberLCD respectively.
5. Place a "TimeDelay" of 40 hundredth of a second to decrease the frequency of flashing.
6. Insert a "Decision" at the end. If the push-button is pressed i.e. "ON", add "Money" to "Total" and save the result as "Total".

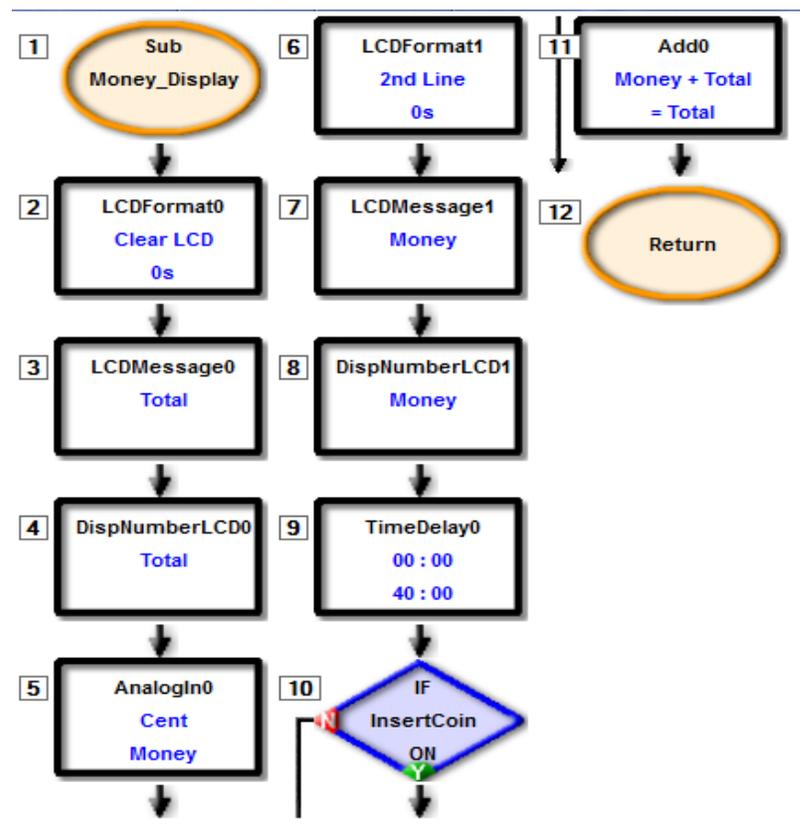


Figure 6: Money Display

Now we want to create a light flash group.

1. Insert an "OnOffPin" which turns the LED on, "TimeDelay" for 50 hundredth of a second, "OnOffPin" turning the LED off and another "TimeDelay" for 50 hundredth of a second. Call this group "Light_Flash".
2. Now we want to create a range of items for our vending machine to dispense, we will use sub groups to name each different product.

3. For the item that costs 200c, subtract “200” from “Total” and make the LED flash three times.
4. For the item that costs 160c, subtract “160” from “Total” and make the LED flash twice.
5. For the item that costs 100c, subtract “100” from “Total” and make the LED flash once.

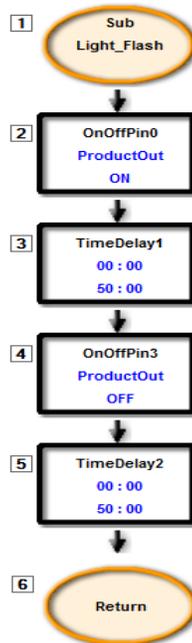


Figure 7: Light_Flash group

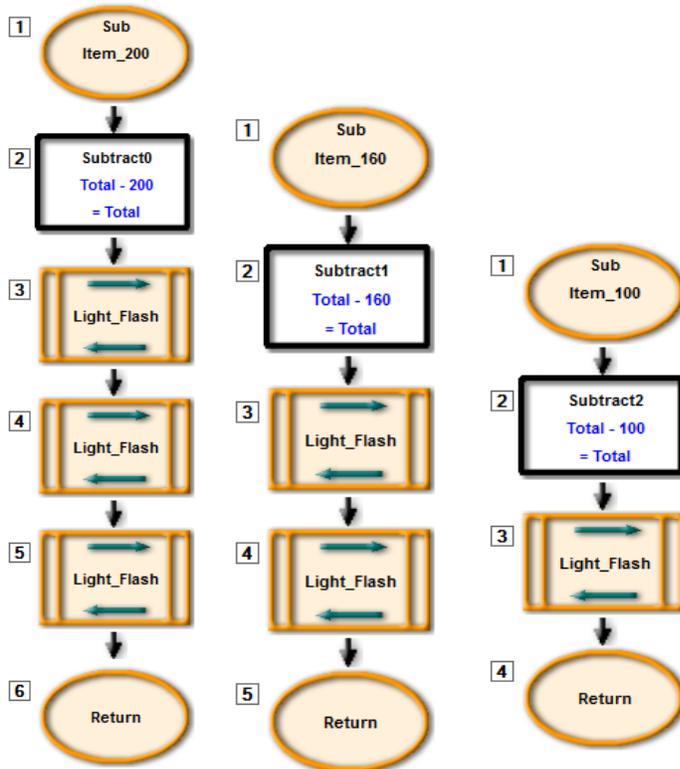


Figure 8: Different item prices

Now we will create a subroutine for “Trading” so that it checks how much money is inserted into the vendor machine and determines what can be bought and will also show us how much money the vending machine is in credit.

1. Add a compare icon and compare the total to see if it is above 199 which means you have enough money to buy the item which costs 200 cents and carry over any additional money.
2. Add another compare icon and compare the “Total” amount and see if it is greater than “159”
3. Add one more compare icon and check to see if it is above “99” which checks to see if you can purchase the cheapest item.

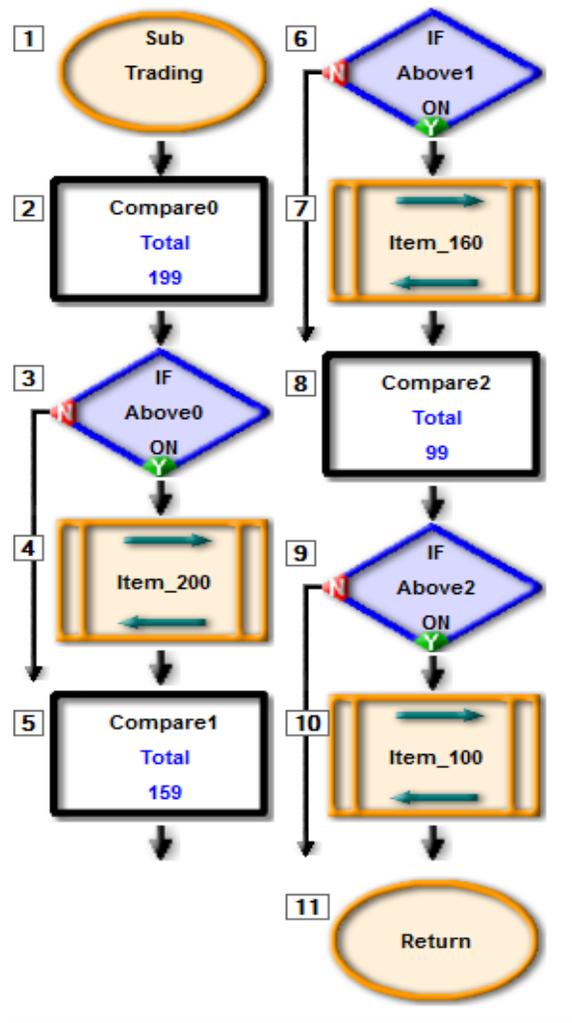


Figure 9: Trading

1. Now that we have created every subroutine, start the program by setting “0” as “Total”, which indicates the total amount of money that has been inserted into the vending machine.
2. Create an address called “Loop” right after the Set_Number that was created just before, this address will continually count how much change is left in the vending machine.
3. After the “Loop” there will be a group called “Money_Display” which was created at the start
4. In order to simulate that we are accepting the transaction we will use the “A3_PushButton”. Therefore we will create a decision, if the button is “ON” then it will go back to the “Loop”. If it is not going back to the loop then it will go to the group we made called “Trading”.
5. For the program to display how much money is inside the vending machine, create a GOTO to the “Loop”.
6. Compare your program to Figure 5 and send your program to chip.

Extension

How can we change the program so that the vending machine can return the change? You can also try to modify the program to allow customers to buy only one or two items. Imagine the real life scenario when we only have 5c, 10c, 20c and 50c etc, what changes should we make to the program so that the machine can only take in the coins with these values.

Summary

Microchips are basically small simple computers that we can use to do many different things. In this project the STEMSEL microchip was programmed to use a POT as an input, and would activate a buzzer, LED and a LCD to operate a vending machine. This project provides good practise in programming the STEMSEL controller board using ezSystem, which can be used to make lots of other exciting and useful things.

At the end of the exercise, you should be able to:

1. Manipulate the ‘add’ and ‘subtract’ functions in CoreChart.
2. Use a LCD to display information in different formats.
3. Use a push-button to control the program.