ezSystem elab16m Project 1F: Alarm System

(Full Project description)

ezSystem

The aim of ezSystem is to enable Creativity and Innovation at an early age in a Problem Based Learning (PBL) approach.

ezSystem integrates ezCircuit Designer, ezCoreChart and the eLabtronics PIC microcontroller boards into a user friendly training system. ezCircuit Designer speeds up the design of the hardware. ezCoreChart uses menu based FORMS to speed up the programming of the microcontrollers.

Learning Outcome

This workbook provides the user with hands on experience on how to use ezSystem to design, build, test and program a simple alarm system as well as give a brief introduction to PIC microcontrollers and electronics.

Equipment for workbook

Computer

PC running Windows 2000 SP4 or higher with USB connections

Software

- Microsoft .NET Framework V2 or higher (Required to run ezCircuit Designer)
- ezCircuit Designer Version 1.1.0.0 or higher
- CoreChart Version 3.0.0 or higher

Hardware

- elab16m (PIC16F819) Controller Board with battery holder and 3x AA batteries
- eLabtronics USBP Programmer with USB Cable
- Piezo buzzer
- Small flat head screwdriver

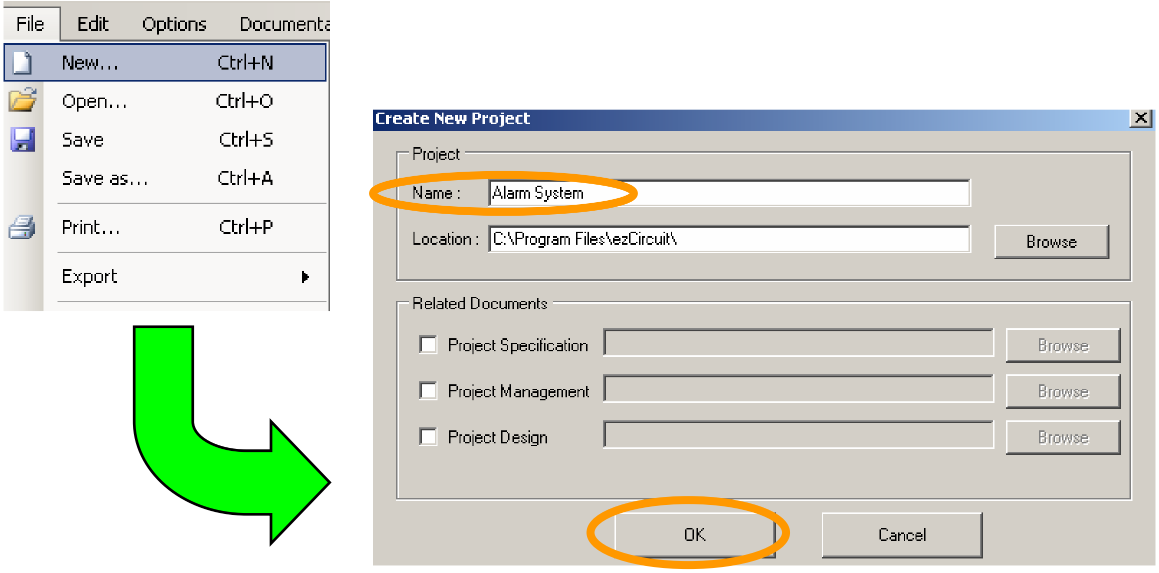
Part A: Design and Build the Alarm System

Alarm Systems are used to protect a wide variety of valuables, from a car to household belongings. Alarms utilise sensors which can be a simple switch or the more complex heat, light and motion sensors. A key component of an alarm system is a microcontroller to gather sensor information and decide when to activate the alarm. A microcontroller is a computer on a single chip that can be programmed to read inputs such as a switch and to control outputs like a buzzer. The alarm system in this project consists of an elab16m, LED, push button switch and a buzzer.

Step 1: Design the project

Start ezCircuit Designer from the desk top.

a. On the menu click “File” and select “New”.
b. In the Name field type “Alarm System” and click “OK”

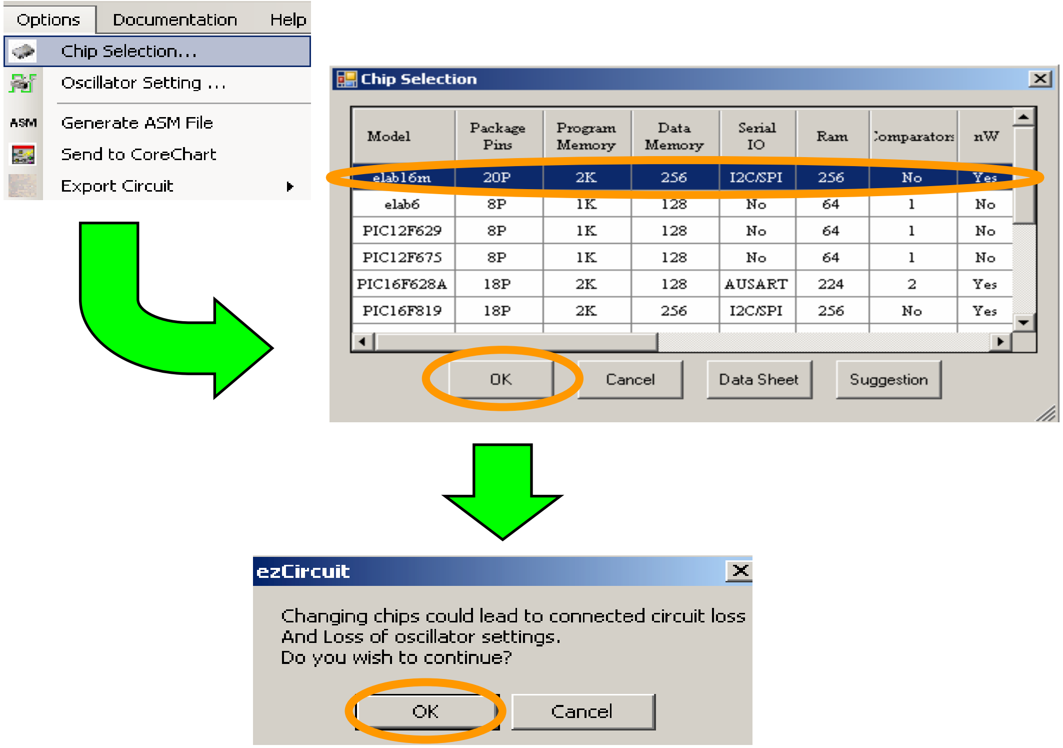


Name : Alarm System

Location : C:\Program Files\ezCircuit\

Figure 1.1

c. On the “Options” drop down menu click “Chip Selection…”
d. In the Chip Selection window click on “elab16m” controller board in the Model column and then click “OK”. When a confirmation window appears click “OK”



Model	Package Pins	Program Memory	Data Memory	Serial IO	Ram	Comparators	nW
elab16m	20P	2K	256	I2C/SPI	256	No	Yes
elab6	8P	1K	128	No	64	1	No
PIC12F629	8P	1K	128	No	64	1	No
PIC12F675	8P	1K	128	No	64	1	No
PIC16F628A	18P	2K	128	AUSART	224	2	Yes
PIC16F819	18P	2K	256	I2C/SPI	256	No	Yes

Changing chips could lead to connected circuit loss
And Loss of oscillator settings.
Do you wish to continue?

Figure 1.2

The elab16m controller board diagram which closely resembles the actual eLab16m board is displayed on the screen. This makes it easier to connect circuits from ezCircuit Designer to the eLab16m controller board.

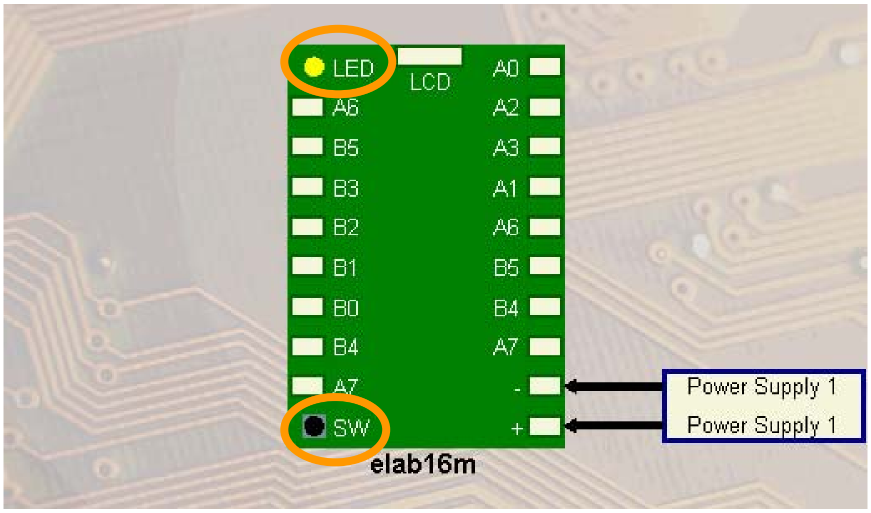


Figure 1.3

The push button and yellow LED required for the alarm system are already added to the eLab16m diagram. Only the buzzer needs to be added.

e. Add a buzzer to the design. Select the “Buzzer” icon in the “Circuit Groups” menu and click on pin “A0”.

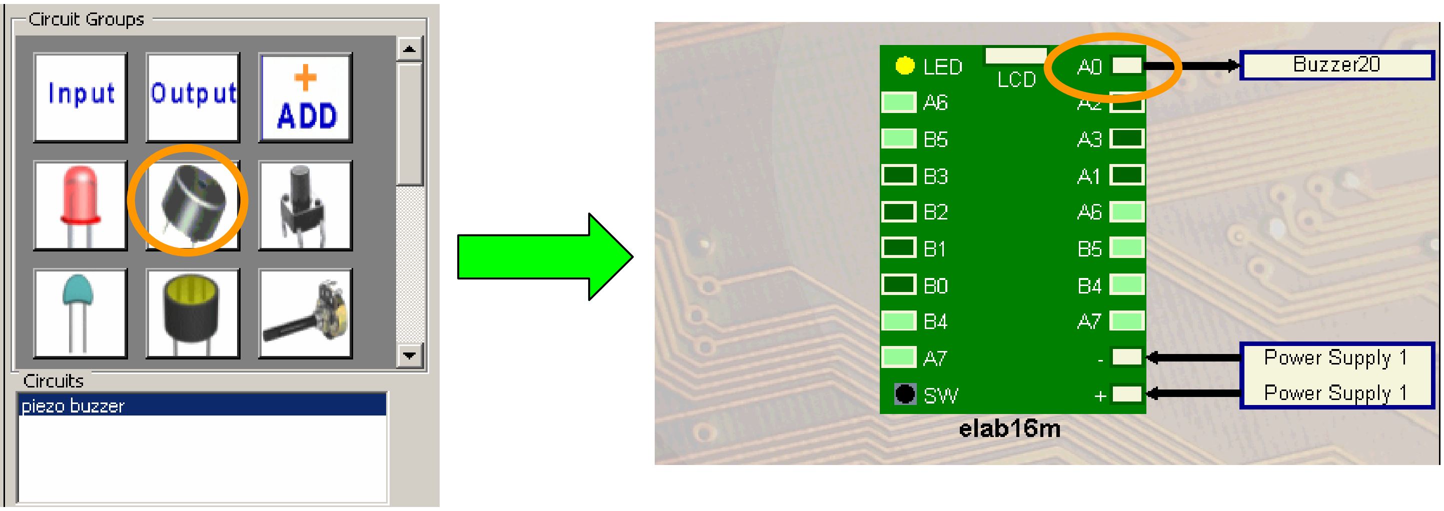


Figure 1.4

f. Rename the buzzer. Right click on the “Buzzer20” label and select “Rename”
g. Rename the buzzer to “Buzzer” and press enter on the keyboard.

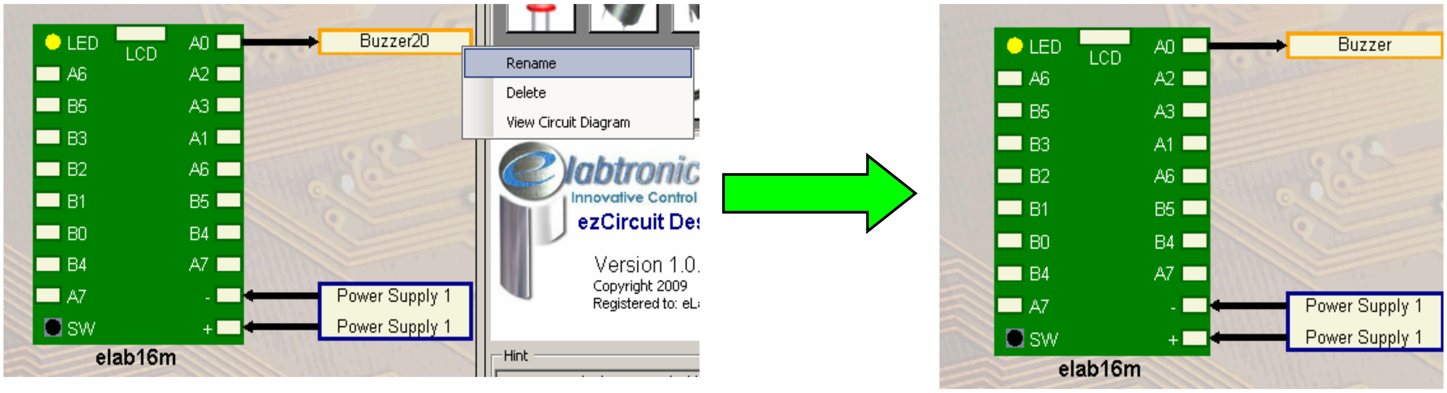


Figure 1.5

The buzzer, LED and push button switch have all been added to the design in ezCircuit Designer which completes the design stage of the alarm system

Step 2: Build the project

The push button switch and yellow LED are pre-built on the eLab16m controller board. Therefore only the buzzer and power supply need to be connected to the controller board.

a. Place the three AA batteries into the battery holder and connect the power pack to the white socket near to the ON/OFF switch on the controller board.

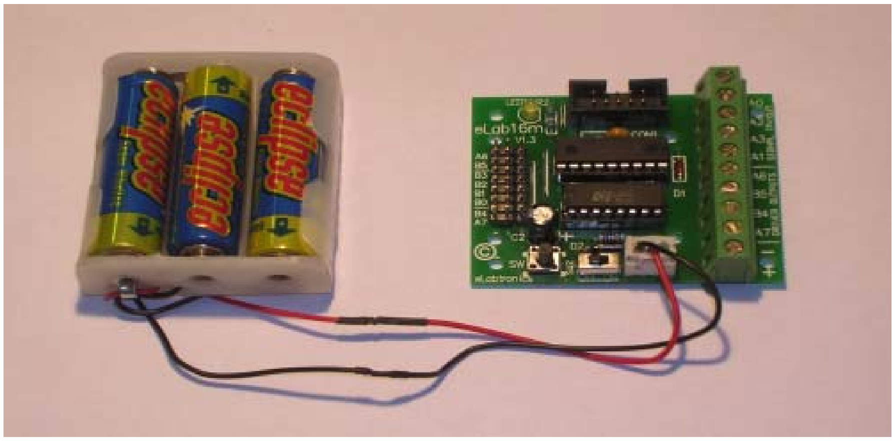


Figure 2.1

b. Connect the black wire of the buzzer to the negative terminal on the controller board and connect the white wire to terminal “A0”.

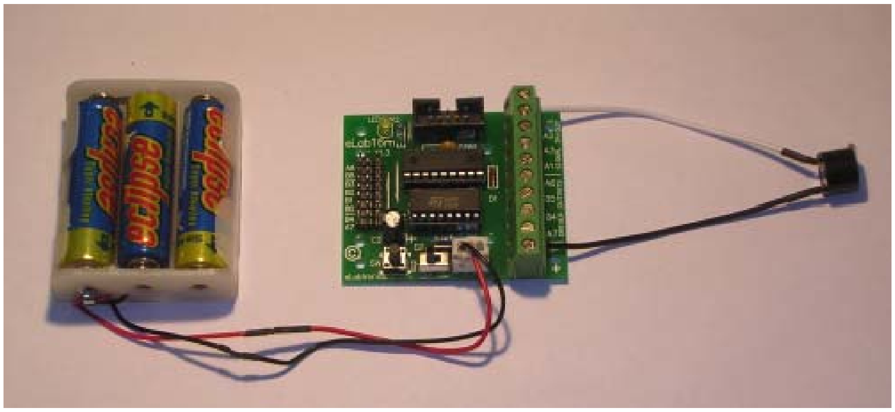


Figure 2.2

c. Connect the eLabtronics USBP Programmer to the 10 pin programming connection on the controller board.

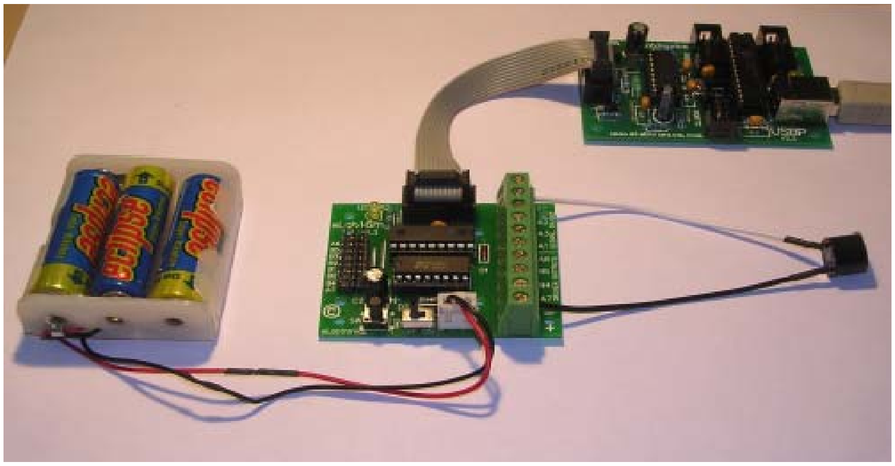


Figure 2.3

Part B: Test and Program the Alarm System

The alarm system is now designed and built. Tests need to be done to ensure that the output circuits connected to the controller board are working, the controller board is functioning correctly and the software is installed and is operating.

Step 3: Export design to ezCoreChart

ezSystem uses ezCircuit Designer to export circuit designs to ezCoreChart. Circuit names assigned in ezCircuit Designer are used in ezCoreChart to program the microcontroller.

a. Ensure ezCircuit Designer is opened. If not, open the “Alarm System” project by clicking on the “File” drop down menu & select “Open”. Navigate to the directory where the project was saved. Select the project & click the “Open” button.

b. To send the design to ezCoreChart, select the “Options” drop down menu and click on “Send to CoreChart”. Alternatively click the “Send to CoreChart” icon in the icon bar.

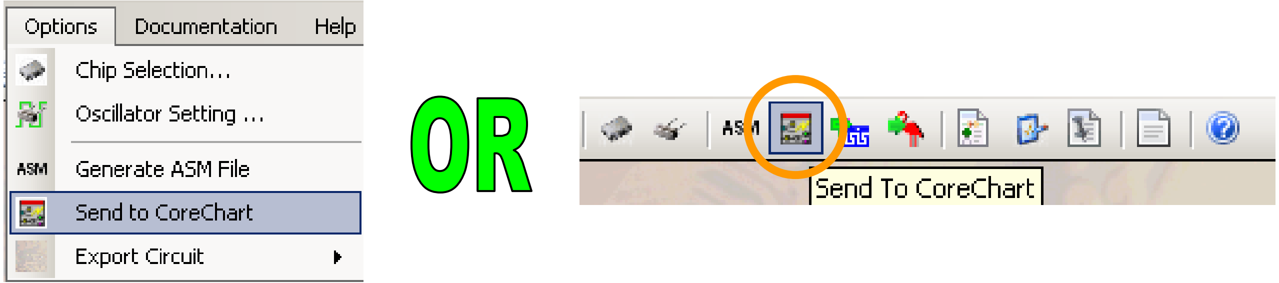


Figure 3.1

The main program looks like Figure 3.2. When a design is sent from ezCircuit Designer to ezCoreChart, a test subroutine is automatically generated for each of the output circuits in the design.

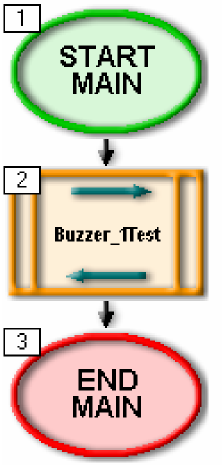
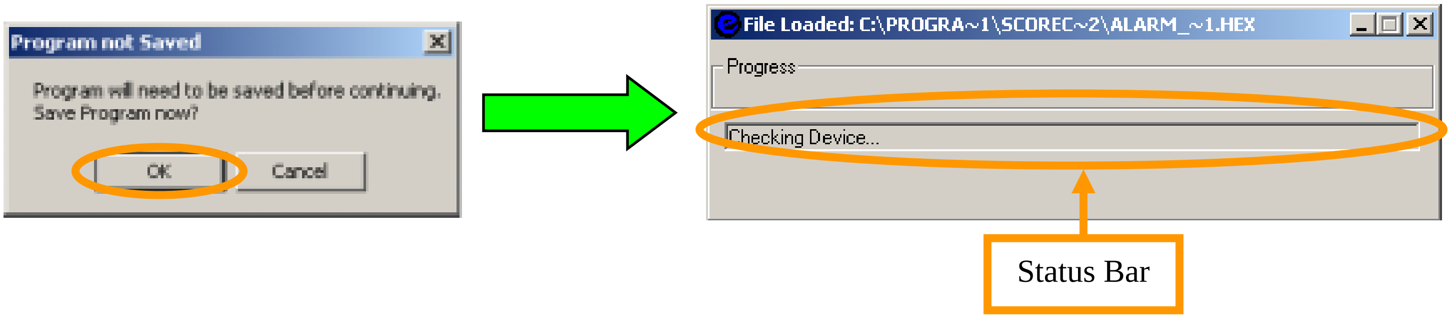


Figure 3.2

Step 4: Test the outputs

A test subroutine is automatically generated to check that the output buzzer circuit is working properly.

a. Save the ezCoreChart program. On the “File” drop down menu click “Save As…”. Type the program name “Alarm_System” and click “Save”.
b. Connect the USB cable of the eLabtronics USBP Programmer to the USB port on the computer.
c. Connect one end of the 10 pin programming cable to the USBP Programmer and the other end to the eLab16m controller board.
d. Check that the battery pack is connected and the power switch is switched to “ON”.
e. Click the “Send Program To Chip” button. When prompted to save the program click “OK”.
f. Wait for the program to download to the chip. If the download is successful the status bar will turn green and “Write Successful” will be displayed. The USB programming window will close automatically.



Program will need to be saved before continuing. Save Program now?

File Loaded: C:\PROGRA~1\SCOREC~2\ALARM_~1.HEX

Checking Device...

Figure 4.1

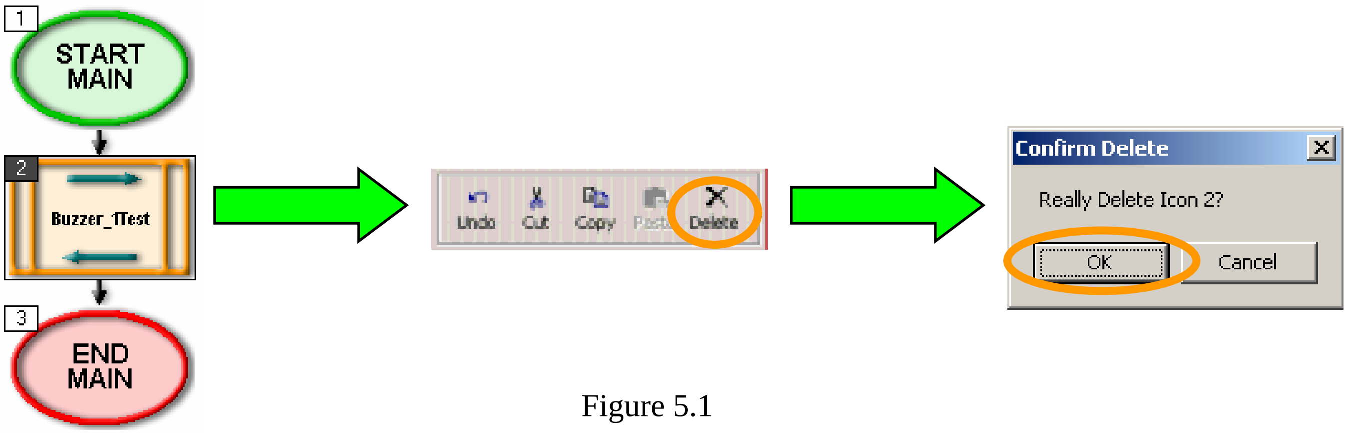
g. Switch the power on the eLab16m controller board to “OFF”.
h. Disconnect the controller board from the USBP Programmer and after a few seconds switch the power to “ON”
i. To run the test program push and release the push button switch on the eLab16m controller board.

This test subroutine will turn the buzzer on for two seconds and then turn the buzzer off. It is important that the buzzer is turned on and then off in the test procedure. If the test subroutine only turns the buzzer on and not turn it off, it will not check for short circuits.

Step 5: Program the Controller Board

The hardware testing is complete. We will proceed to program the microcontroller with ezCoreChart which enables users to quickly create programs for microcontrollers.
The alarm program will turn the buzzer and yellow LED on, delay for five seconds and turn them both off.

a. Delete the buzzer test subroutine by clicking on the icon and selecting "Delete" in the icon bar. Alternatively, use the Delete key on the keyboard.
b. When prompted to confirm the delete click "OK"



Really Delete Icon 2?

Figure 5.1

c. Select the "Outputs" button in the "ICONs" menu and click the "OnOffPin" item in the "ICON Properties" list.
d. Click on the black arrow below the "START MAIN" icon to place the icon.

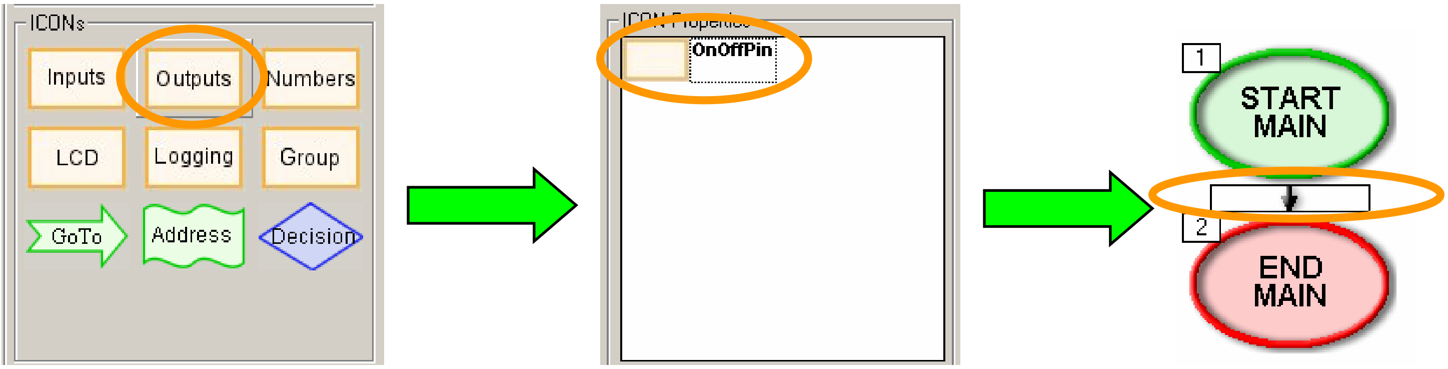
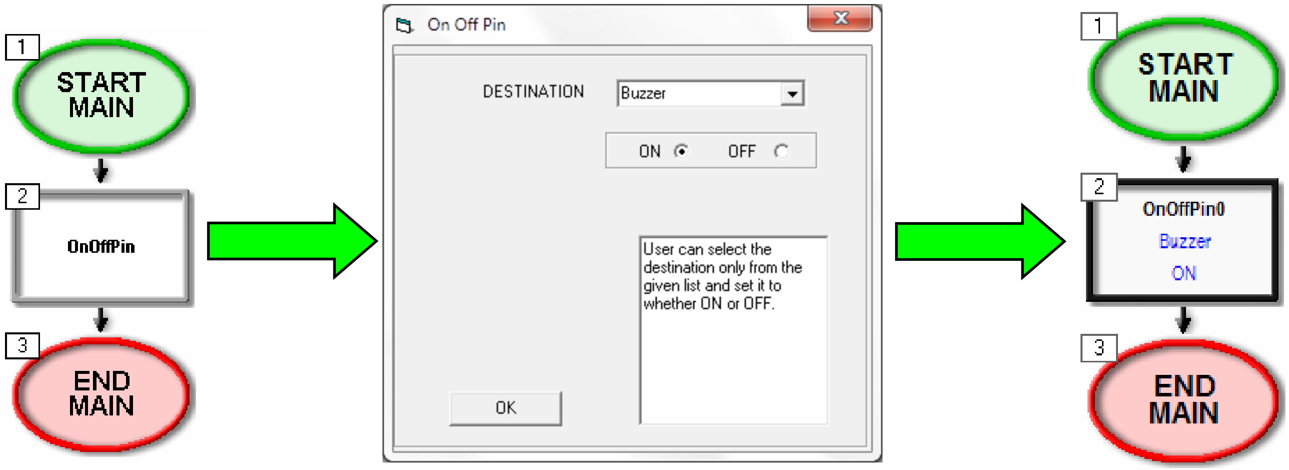


Figure 5.2

e. Double click on the "OnOffPin" icon that was placed to open a form with the icon options.
f. In the "DESTINATION" field select "Buzzer" and select "ON" to turn the buzzer on. Click "OK".



On Off Pin

DESTINATION Buzzer

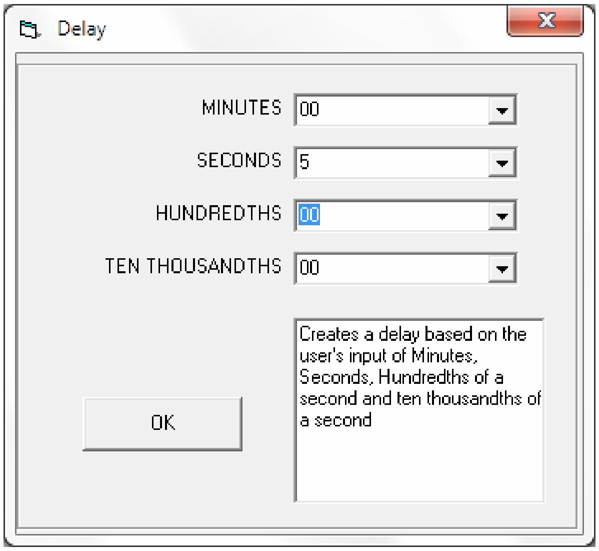
ON ● OFF ○

User can select the destination only from the given list and set it to whether ON or OFF.

OK

Figure 5.3

g. Click “Outputs” in the “ICONs” menu and select “OnOffPin” in the “ICON Properties” list.
h. Place the second “OnOffPin” icon below icon 2 by clicking the black arrow below the second icon.
i. Double click on the new grey icon and select “D7_YellowLED” in the “DESTINATION” field. Ensure that “ON” is selected before clicking “OK”.
j. Click the “Numbers” button in the “ICONs” menu and select “TimeDelay” from the “ICON Properties” list.
k. Place the “TimeDelay” icon after the first two “OnOffPin” icons by clicking the black arrow below icon 3.
l. Double click on the “TimeDelay” to open the options form. Type the number 5 into the “SECONDS” field.



Delay

MINUTES 00

SECONDS 5

HUNDREDTHS 00

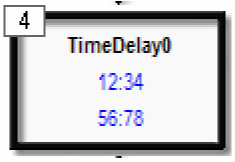
TEN THOUSANDTHS 00

Creates a delay based on the user's input of Minutes, Seconds, Hundredths of a second and ten thousandths of a second

OK

Figure 5.4

Note about time delay icons: TimeDelay icons have labels written as aa:bb:cc:dd, where aa = minutes, bb = seconds, cc = hundredths (of a second) and dd = ten thousandth (of a second). So for example:



This TimeDelay gives a time delay of 12 minutes 34 seconds 56 hundredths and 78 ten thousandth of a second.

m. In the “ICONs” menu select another “OnOffPin” icon from the “Outputs” group. Place it at the end of the program below the “TimeDelay” icon. Configure this “OnOffPin” icon to turn the “Buzzer” to “OFF”.

n. Place another “OnOffPin” icon below the icon 5 to turn the “D7_YellowLED” to “OFF”

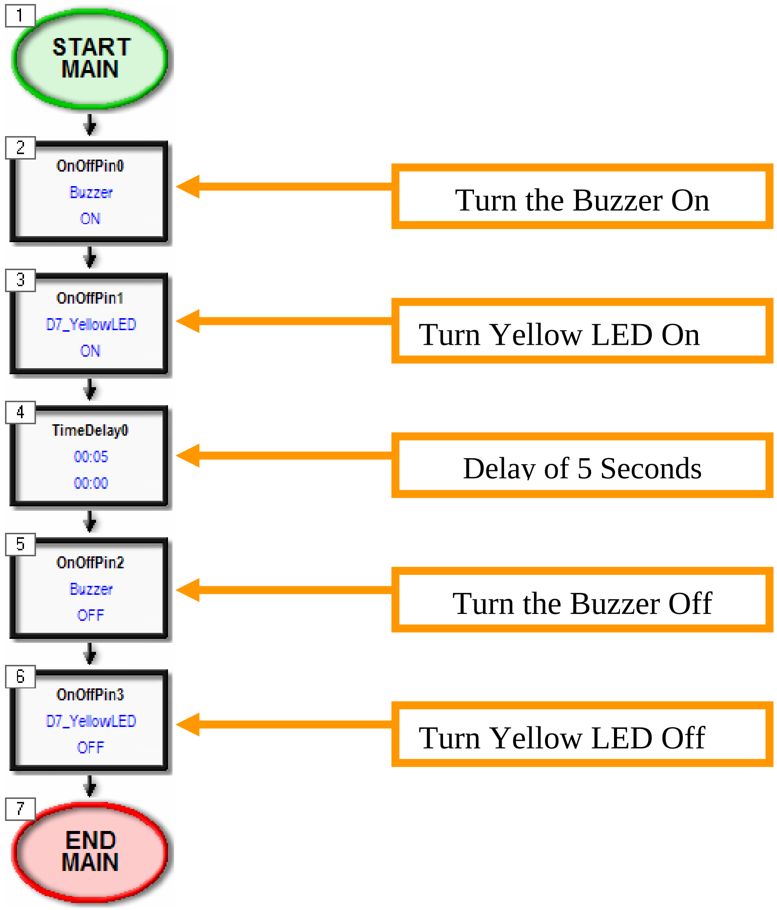


Figure 5.5

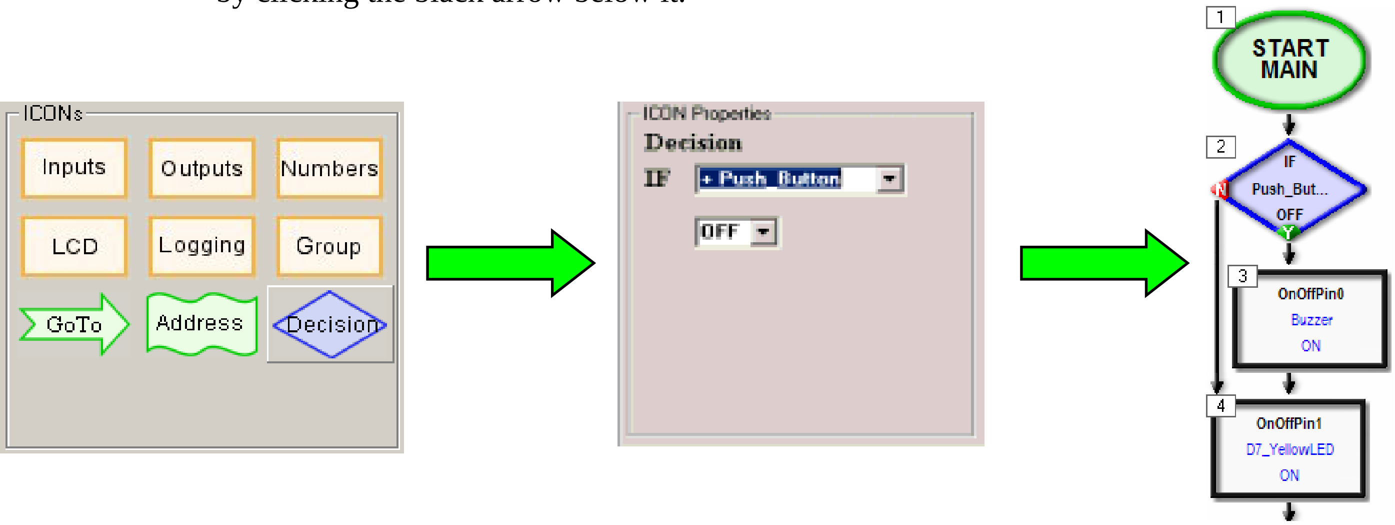
o. Check that the eLabtronics USBP Programmer is connected to the USB port of the computer and the programmer cable is connected to the eLab16m controller board.
p. Also check that the battery pack is connected and the power is switched ON.
q. Click the “Send Program To Chip” button and click “OK” when prompted to save the program.
r. Wait for the program to download to the controller. Once “Write Successful” is displayed, switch the eLab16m power to OFF.
s. Disconnect the USBP Programmer from the eLab16m. Wait a few seconds and turn the power back ON
t. Push and release the push button switch on the eLab16m to start the program.

The program will turn the buzzer and yellow LED on, delay for five seconds and turn them both off.

Step 6: Incorporate Input Sensor to the Program

A switch is used as an input sensor for the alarm system. When the push button switch is pressed the alarm will sound. A decision icon is used to check if the switch is pushed.

a. Click the “Decision” button in the “ICONs” menu.
b. In “ICON Properties” select “Push_Button” in the first box and select “OFF” in the second box. “OFF” in this case means the “Push_Button” switch is pressed and giving an output voltage of zero. See below for further explanation.
c. Place the decision at the beginning of program below the “START MAIN” icon by clicking the black arrow below it.



IF Push_Button
OFF

IF Push_But... OFF

OnOffPin0 Buzzer ON

OnOffPin1 D7_YellowLED ON

Figure 6.1

The decision icon has two branches. When the push button switch is pressed, the program will follow the path of the Green “Y”. If the push button is not pressed the program will follow the path of the Red “N” arrow.

The push button as shown in Figure 6.2 connects pin 7 to “Ground” or zero volts when it is pressed. This Active Low circuit sets the signal to zero volts or OFF when it is pressed. This is why the decision icon in the program is selected as “IF Push_Button OFF” in this case.

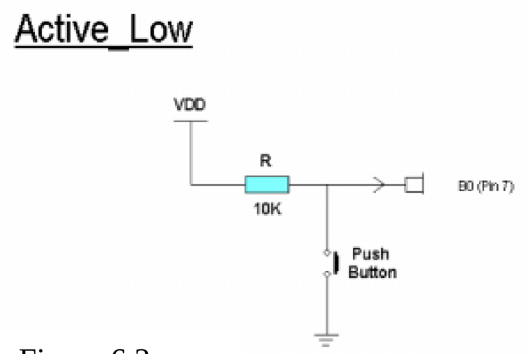


Figure 6.2

Most push button circuits are Active High. This means that the signal is ON when the push button is pressed. The push button on the eLab16m control board is Active Low because this type of circuit allows the push button on the control board to be used as a RESET button.

Active Low circuits are used as they require fewer wires than an Active High circuit. Active Low circuit only need to be connected to ground. For this reason most switches in cars are Active Low as the cars chassis is grounded. Meaning that one connection of the switch can be connected directly to the cars body and only require the other connection connect to a controller. On the other hand an Active High switch would require two wires, one connecting to a power supply and another connecting to a controller. Therefore Active Low switches require fewer wires compared to Active High switches. To see the effect of adding the decision icon run the program:

d. Save the program and check that the hardware is connected correctly.
e. Send the program to the chip.
f. Push the push button switch to run the program.

The decision icon placed in the program in Fig 6.1 to detect whether the push button is pressed. If it is pressed the program will only turn on the buzzer. In order for the decision icon to turn on the buzzer, the LED and the time delay, the group function can be used.

The group function is used to group multiple icons into a subroutine. Follow these steps to use the group function:

g. Click on icon 3 to select it and continue to hold the shift key down on the keyboard. Click on icon 5 to select icons 3 through to 5.
h. Click on the "Group" button in the "ICONs" menu. Select "Group_Name" and type "Alarm_ON" as the name of this group. Click the "Create New Group" button

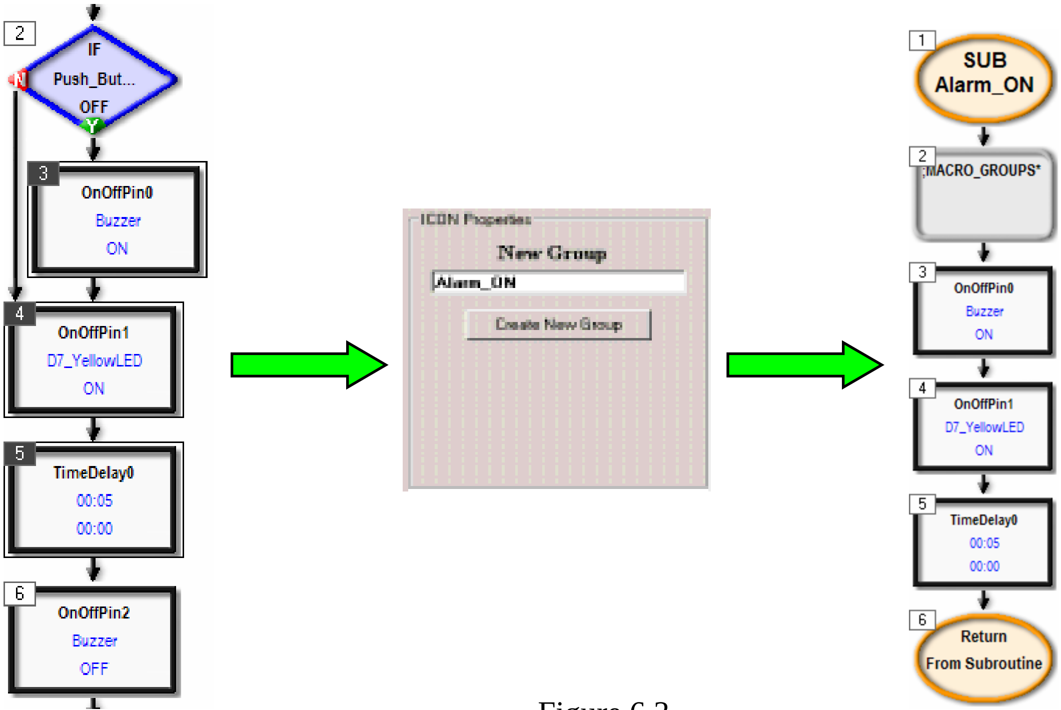


Figure 6.3

When a new group is created the contents of the new group are displayed as the current program. To return to the main program, double click on ether the first icon, "SUB Alarm_ON" or the last icon, "Return from Subroutine".

The alarm system program is still unsatisfactory because the program will only run once. When the program starts it will check the push button switch. If the push button switch is pressed then the buzzer and the LED will turn on. After five seconds they will both turn off and the program will end. On the other hand if the push button switch is not pressed the program will end immediately. This problem can be fixed by using a GoTo icon to create a loop to run the program continuously.

i. Return to the main program by double clicking ether the first or last icon in the group.

j. Click the “GoTo” button in the “ICONs” menu. In the “ICON Properties” the box next to the label “GoTo” should have “START” selected.

k. Place the icon at the end of the program by clicking the last black arrow.

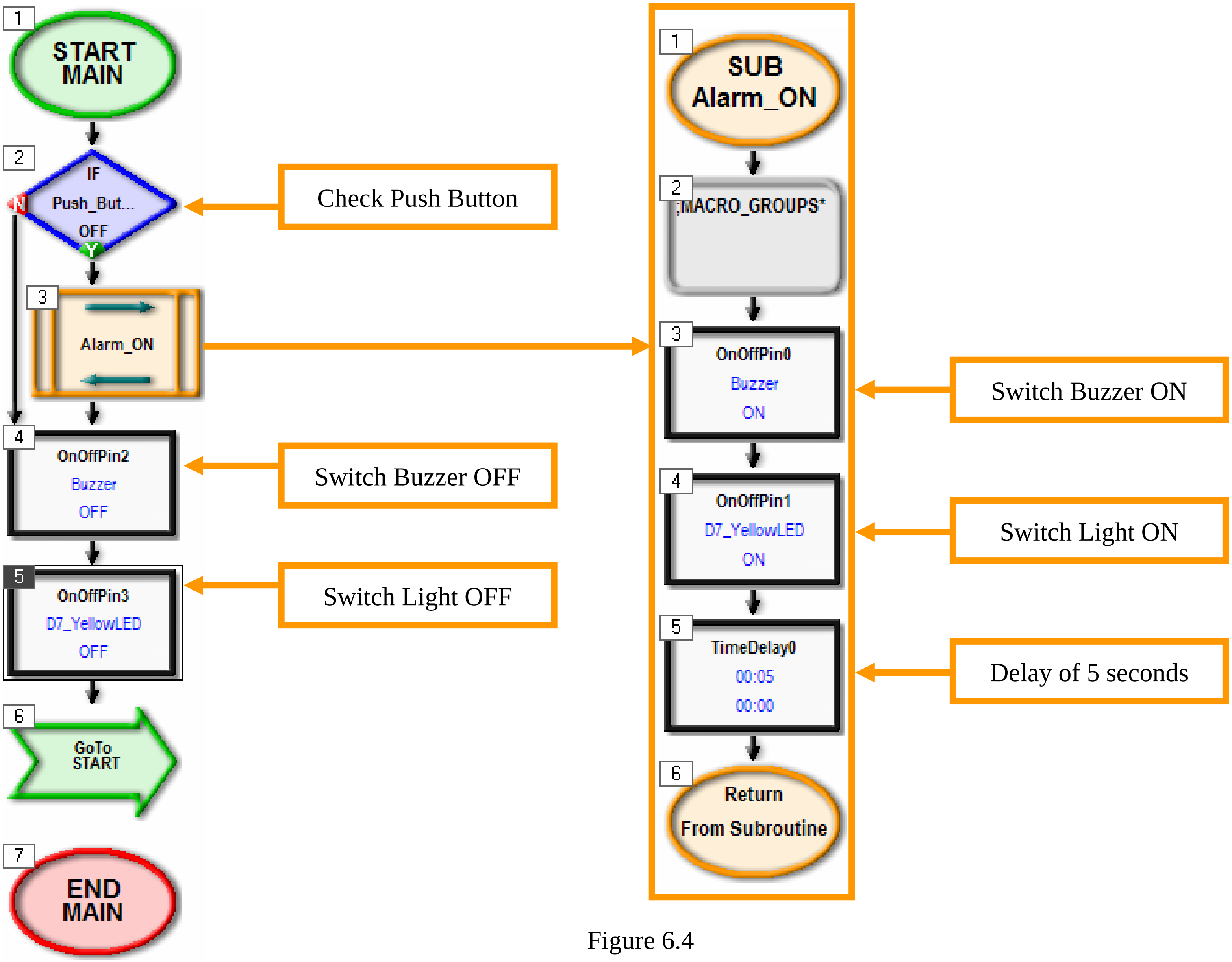


Figure 6.4

l. Save the program, make sure the hardware is connected properly as described earlier in this workbook and send the program to the chip.

m. Press and release the push button switch to start the program

The program will turn on the buzzer and yellow LED for five seconds when the push button switch is pushed.

Part 7: Extend the Alarm System Program

The alarm system now works as a basic alarm that will activate when the push button switch is pressed. There are still some areas where the alarm system can be improved. Work through the following challenges and devise a solution to the problem. Make small changes to the program and run the program regularly as you progress.

Extensions:

As it is, the alarm will only sound while the push button switch is pressed i.e. the sensor is activated continuously. The alarm will stop when the switch is no longer pressed or released. This is not very satisfactory.

1. Modify the Alarm System so that once the alarm is activated it will continue to sound until it is reset by turning the power off and then back on.

The alarm system is still not satisfactory because if the alarm is activated while the owner is away on holidays, the alarm will sound continuously disturbing the neighbours. The alarm system may not be complying with noise pollution laws if it sounds for a long period of time continuously after it has been tripped.

2. Modify the Alarm System so that when the alarm is activated the buzzer and yellow LED will be on for five seconds and off for ten seconds and then continue to loop until is reset.

Further Extension:

The alarm system has been improved. However new features and functions can still be added by changing the program in ezCoreChart.

3. Modify the Alarm System so that when the alarm is tripped the yellow LED will flash quickly. The buzzer will continue to be on for 5 seconds and off for 10 seconds.

Hint: Use a group of icons that will flash the yellow LED a number of times. This new group of icons can then be used to replace the 5 second time delay.

ezSystem elab16m Project 1F: Alarm System Completed

(Full Project Description)