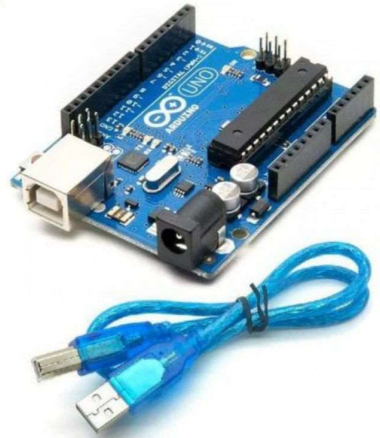
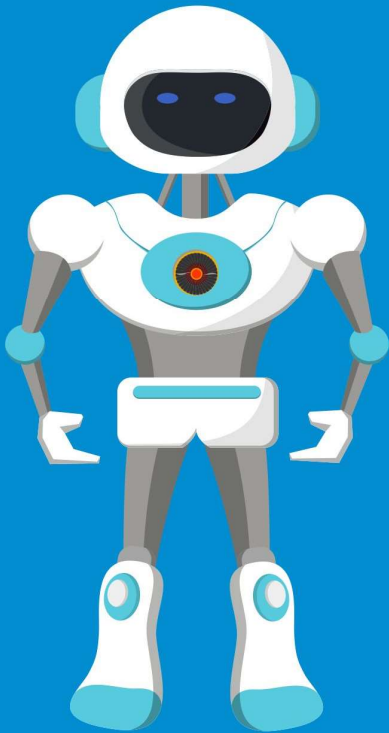


Fundamentals of Robotics Using Arduino



USER MANUAL



www.texoham.com

About this Document

Title:

Fundamentals of Robotics Using Arduino

Document Type:

User Manual

Applicable Product:

Robotics Kit – Fundamentals of Robotics Using Arduino

Author

Komaragiri Sahadev

Director, Texoham Centre for Advanced Computing

Document Last Update Date:

November 2, 2019

© *Texoham Centre for Advanced Computing*

Important First Steps

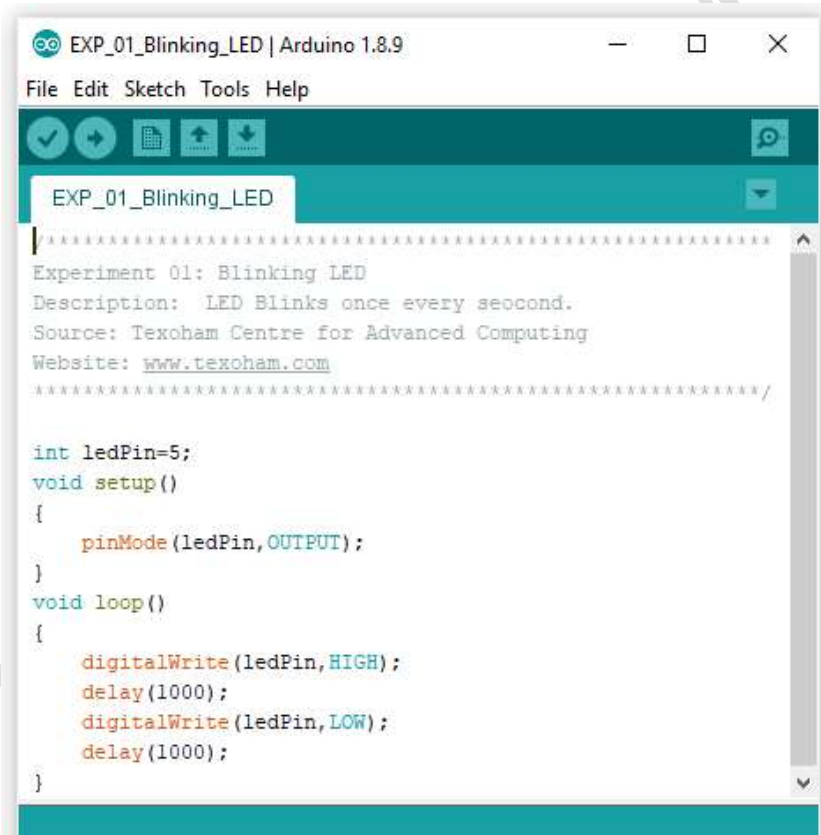
Before you start using the kit, please complete the following steps:

1. Install Arduino IDE
 - a. You can get Arduino IDE software from <https://www.arduino.cc/en/main/software>
 - b. Download and Install the IDE that is appropriate for your operating system.
2. To get the source and code libraries needed for this kit, please visit <http://www.texoham.com/support>
3. Download the source code that is appropriate for
 - i. *Product Name: Fundamentals of Robotics Using Arduino.*
 - ii. *Product Type: Robotics Kit.*
4. Unzip the downloaded source code files into a suitable folder (Example: C:\Texoham\FRUA\Code).
5. Download the Libraries that are appropriate for
 - i. *Product Name: Fundamentals of Robotics Using Arduino.*
 - ii. *Product Type: Robotics Kit.*
6. Unzip the downloaded libraries to a suitable folder. Example: C:\Texoham\FRUA\Library.
7. Copy the all the contents from the extracted folder to <ARDUINO_IDE>\libraries. <ARDUINO_IDE> is the folder where Arduino is installed. Example: C:\Program Files (x86)\Arduino.

Steps to Upload a Program to Arduino

Here are the steps to upload the program to Arduino.

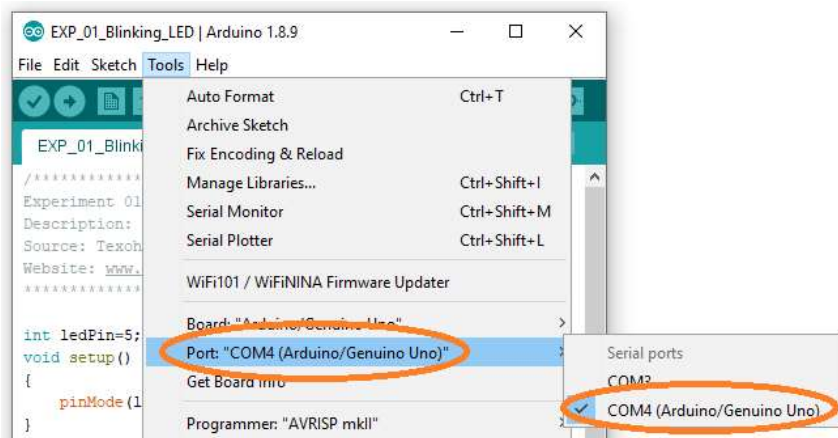
1. Open the relevant file using Arduino IDE. Double click on the file to open. All the files that you need for your kit are already provided to you. (See Important First Steps section).
2. The program looks like this once it is opened in Arduino IDE.

A screenshot of the Arduino IDE window titled "EXP_01_Blinking_LED | Arduino 1.8.9". The window has a menu bar with "File", "Edit", "Sketch", "Tools", and "Help". Below the menu bar is a toolbar with icons for saving, undo, redo, and uploading. The main editor area shows the following code:

```
*****  
Experiment 01: Blinking LED  
Description: LED Blinks once every second.  
Source: Texoham Centre for Advanced Computing  
Website: www.texoham.com  
*****/  
  
int ledPin=5;  
void setup()  
{  
  pinMode(ledPin,OUTPUT);  
}  
void loop()  
{  
  digitalWrite(ledPin,HIGH);  
  delay(1000);  
  digitalWrite(ledPin,LOW);  
  delay(1000);  
}
```

3. Connect Arduino to your computer. It should be visible in the connected ports list as shown below. In this case it is COM4. *There should be a ✓ mark next to the connected Arduino as shown.* The port number can vary

for each Arduino connection.



4. Below picture identifies the buttons that can be used.



1. Compile the program
 2. Upload the program
 3. Name of the file
5. Click on the compile program button to compile the code. This step is optional.
6. Click on the upload button to upload the program

Chapter 1 – Introduction

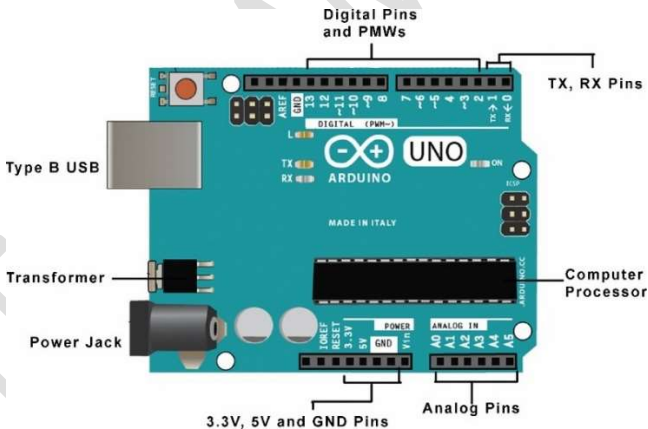
This chapter introduces you to this book, explains certain key concepts and provides you with the basic information that you need to get started.

About the Manual

This entire hands-on manual is spread over 6 chapters. Chapter 1 is this Introduction chapter. In chapters 2 to 5 you will learn how to conduct some very basic experiments in robotics. In chapter 6 you will learn how to complete some robotics projects.

Arduino & Its Components

Think of Arduino as a real tiny computer that has no keyboard, mouse, display or hard disk. It can be used to do things that are too small for a computer to do. Also, computers are too big to fit into small objects such as toys and robots that can move around.



a. External Power Connections

You Arduino works only when it is connected to a power source. Power can be supplied to Arduino using a Type B USB cable connected to a computer or a wall outlet. It can also be powered using an external power supply of 6 to 12 volts through power jack.

b. Digital Pins & PWMs

You can connect LEDs and various other sensors, motors and displays to an Arduino board. They are typically connected to the Digital and PWM pins. The PWM pins can be identified by the “~” symbol that is placed next to the pin number. Thus pins 3, 5, 6, 9, 10 and 11 are PWM pins.

c. Analog Pins

Analog pins are used for connecting analog devices.

Note that all the digital pins, PWM and analog pins are also called as data pins as they exchange data between the connected item and the Arduino.

d. TX, RX pins

TX pins are transmission pins and RX pins are receiving pins.

e. 3.3V, 5V and GND Pins

Some of the devices connected to the Arduino need additional power. This power is drawn through connections to 3.3V and 5V pins. GND pins are for connecting ground pins.

f. Transformer

A typical Arduino cannot handle more than 5 volts of power supply. If the external power sources supply higher voltage, the transformer reduces the voltage supplied to the Arduino to just 5 volts.

g. Computer Processor

The most important part of the Arduino is its processor. It is an AT-Mega328 processor. It is the brain of the Arduino that can follow and execute the program given to it.

Arduino Computer Program

a. An Arduino program is a computer file that has all the instructions to run

your robot. This is the file that executes the commands necessary to run your experiments.

- b. This file is usually written in Arduino programming language.
- c. For every experiment or project that you undertake there will be a separate file.
- d. This file must be uploaded to Arduino using a computer.
- e. Without uploading the correct file, the corresponding experiment or the project will not work as expected.

Uploading a program to an Arduino

You can upload your program to an Arduino device using its USB input that is connected to a computer. To upload a program to Arduino, please read *Steps to Upload a Program to Arduino* section in this document.

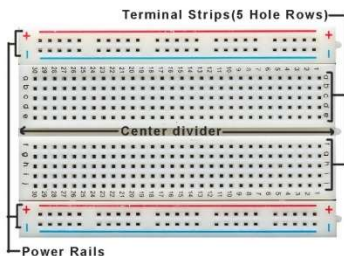
Connecting components to Arduino?

It is not possible to connect all your components directly to your Arduino. To connect to an Arduino, you must first connect your components to a breadboard. From the breadboard the components are connected to the Arduino using a series of wires forming a complete circuit.

This process of creating circuits using components connected to a breadboard and Arduino is called prototyping.

Breadboard

Here is a picture of a typical breadboard that comes with your kit.



Think of the breadboard as an experimental circuit board to which several components and wires can be connected. This is used to test all your connections to make sure that your experiment or project can be completed successfully. Notice that there are 5 parts in the Breadboard. Two power rails at the top and bottom, two sets of Terminal strips (5-hole rows A to E and F to J) and a centre divider.

Tips on using breadboard

- a. A breadboard must always be held in such a way that the power rails are always at the top.
- b. The power rail has a red line below which you have a row for +ve pin connections. +ve connections are used for connecting 5V power connections.
- c. Similarly, it has a blue line where you will find the -ve pin connections. -ve pins are used for earthing/grounding (GND) purposes.
- d. Pins on the power rails are used from left to right or right to left i.e. pins are used in rows. For example, if we connect a jumper wire from 5V power supply pin on Arduino, that entire line can supply 5V. Any of the items on the terminal strips that need 5V supply should be connected to that line.
- e. The terminal strips are used for terminating connections from top to bottom or bottom to top i.e. pins are used in columns.
- f. Each of the terminal strips function independently.

Sensors, Displays and Motors

Throughout this book you will use different types of sensors, motors, displays and other devices. When using these devices, you must be able to understand their purpose and how they are used. You must keep the following in mind:

- a. You must know the name of the device. The name of the device usually tells us the purpose of the device. For example, a distance sensor is a sensor that can tell us the distance of the objects that it can sense.
- b. The devices have certain prefix or suffix that tells us a little more about

the device. For example, A 7-Segment display is a display type device that has 7 segments in it.

- c. Sometimes the device also has a certain number attached to it. This number indicates technical details that may not be easy to understand. It can also be a version number. You must know which type and version of the device to use.
- d. Let us now look at all the items that we will use throughout this book.

TEXOHAM.COM

Reference

List of Experiments

#	Name of the Experiment	Code to Upload	Remarks
1	Blinking LED	EXP_01_Blinking_LED	
2	Button Controlled LED	EXP_02_Button_Controlled_LED	
3	Using a Serial Input	EXP_03_Using_a_Serial_Input	
4	Flowing LEDs	EXP_04_Flowing_LEDs	
5	Fading/Breathing LED	EXP_05_Breathing_LED	
6	Tilt Switch	EXP_06_Tilt_Switch	
7	Using a Servo Motor	EXP_07_Using_a_Servo_Motor	Libraries
8	Using a Stepper Motor	EXP_08_Using_a_Stepper_Motor	
9	Seven Segment Display	EXP_09_Seven_Segment_Display	
10	Simple Counter	EXP_10_Simple_Counter	Libraries
11	Using LCD (1602)	EXP_11_Using_LCD_1602	Libraries
12	IR Receiver and Controller	EXP_12_IR_Receiver_and_Controller	Libraries
13	Ultrasonic Distance Sensor	EXP_13_Ultrasonic_Distance_Sensor	
14	Using a Photoresistor	EXP_14_Using_a_Photo_Resistor	
15	Using a Temperature Sensor	EXP_15_Using_a_Temperature_Sensor	
16	Active Buzzer	EXP_16_Active_Buzzer	
17	Flame Sensor with Buzzer	EXP_17_Flame_Sensor_with_Buzzer	
18	Fun Game with LCD	EXP_18_Fun_Game_With_LCD	

Chapter 2 – Working with LEDs

This chapter deals with LEDs and multiple ways of working with them using Arduino.

Experiment 1 – Blinking LED

I. Goal

This is one of the most fundamental experiments using an Arduino. The goal of the experiment is to connect an LED to an Arduino board. The LED should stay on for 1 second, stay switched off for 1 second and switch on again for 1 second in a loop.

II. Material Required

Arduino	USB Cable	Breadboard
LED	M2M Wires	220 Ω Resistors

III. Procedure

Complete all the connections using the following circuit diagram. Connect the Arduino to a computer and load the program.

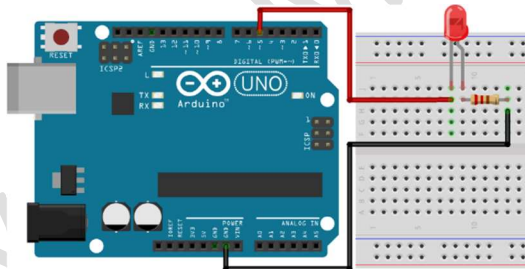


Figure 1 – Blinking LED

IV. Desired Outcome

As soon as the program is loaded you should observe that the LED starts blinking. The LED starts blinking at an interval that is identified in the program that is uploaded to the Arduino.

Experiment 2 – Button Controlled LED

I. Goal

The goal of the experiment is to connect an LED and a button to an Arduino board. The LED must be switched on or switched off, using the button.

II. Material Required

Arduino	USB Cable	Breadboard
LED	Push Button	M2M Wires
220 Ω Resistors	10K Ω Resistors	

III. Procedure

Complete all the connections using the following circuit diagram. Connect the Arduino to a computer and load the program.

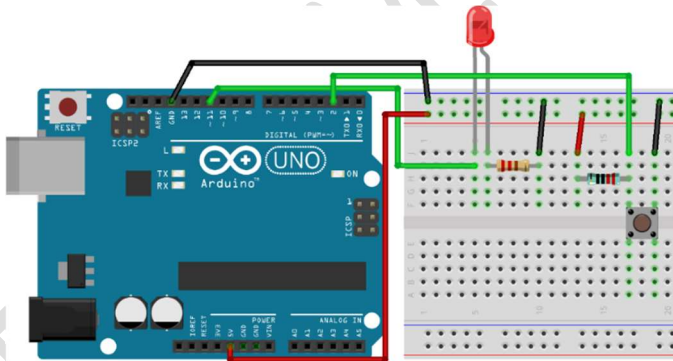


Figure 2 – Button Controlled LED

IV. Desired Outcome

As soon as the program is loaded into the Arduino it should be ready for you to operate. Press the button, the LED must switch on. Press the button again and the LED must get switched off.

Experiment 3 – Using a Serial Input

I. Goal

The goal of the experiment is to connect an LED to an Arduino board. The LED must be operated i.e. switched on or switched off, using the serial port of a computer.

II. Material Required

Arduino	USB Cable	Breadboard
LED	M2M Wires	220 Ω Resistors

III. Procedure

Complete all the connections using the following circuit diagram. Connect the Arduino to a computer and load the program.

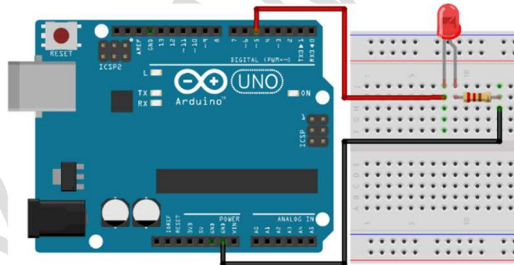
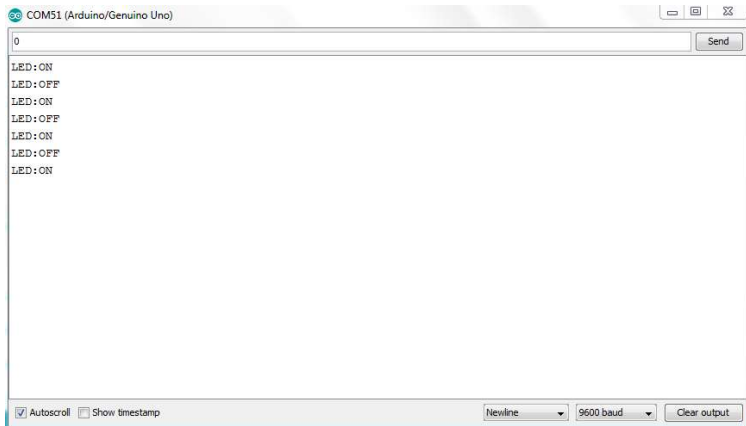


Figure 3 – Serial Input Controlled LED

Take a close look at the connections show above.

IV. Desired Outcome

As soon as the program is loaded into the Arduino it should be ready for you to operate. The LED is switched on when it receives instructions from the serial port. If the serial input is not working, you must check if the serial port is properly enabled. Reconnect the Arduino to the PC and try again.



Experiment 4 – Flowing LEDs

I. Goal

The goal of this experiment is to connect multiple LEDs and make them all switch on and off in a certain pattern.

II. Material Required

Arduino	USB Cable	Breadboard
LEDs – 7	M2M Wires	220 Ω Resistors – 7

III. Procedure & Circuit Diagram

Complete all the connections using the following circuit diagram. Connect the Arduino to a computer and load the program.

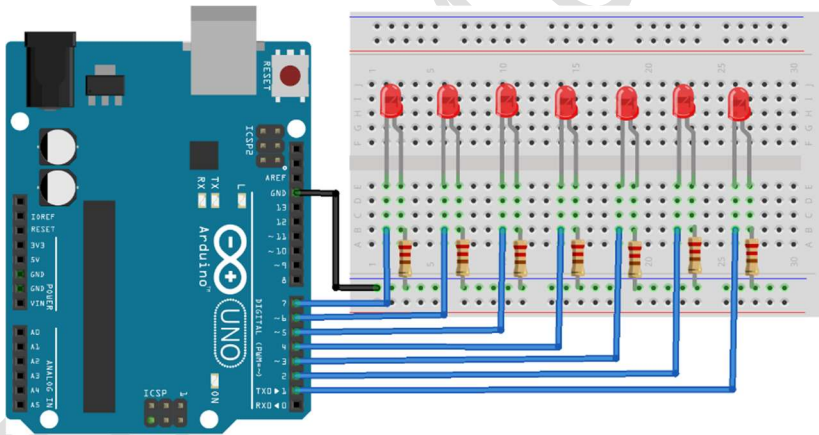


Figure 4 – Flowing LEDs

IV. Desired Outcome

As soon as the program is loaded into the Arduino the LEDs will swing into action. The seven LEDs will be switched on and off in a series from left to right. This will continue as long as the Arduino is connected to a power supply or to the computer.

Experiment 5 – Fading/Breathing LED

I. Goal

The goal of this experiment is to demonstrate the use of PWM signal of an Arduino board. This PWM signal allows gradual increase or decrease of power supply to the LED.

II. Material Required

Arduino	USB Cable	Breadboard
LED	M2M Jumper Wires	220 Ω Resistors

III. Procedure & Circuit Diagram

Complete all the connections using the following circuit diagram. Connect the Arduino to a computer and load the program.

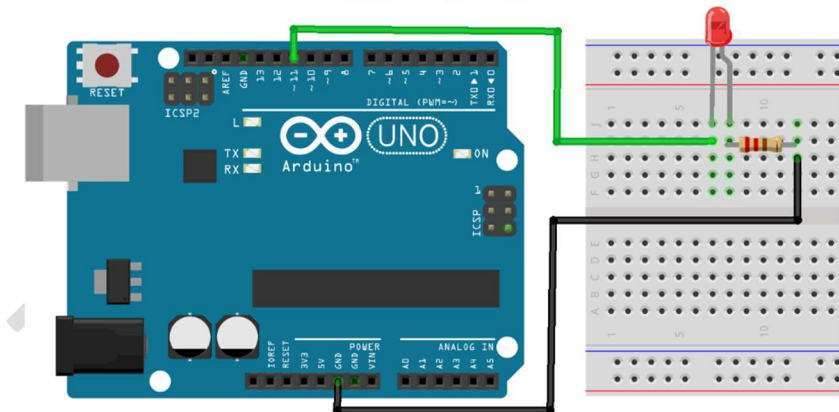


Figure 5 – Fading/Breathing LED

IV. Desired Outcome

As soon as the program is loaded into the Arduino the LED will switch on. It will gradually brighten to a high and slowly fade to a low.

Chapter 3 – Working with Motors

A motor works on a very simple principle of converting electric energy into mechanical energy. All electrical appliances like fans and printers use the same principle. From cars to aeroplanes every moving object uses some kind of a motor.

In this chapter, we will learn how to operate various types of motors using Arduino.

Experiment 6 – Tilt Switch

I. Goal

The goal of this experiment is to demonstrate the use of a simple mechanical device in a digital environment.

II. Material Required

Arduino	USB Cable	Breadboard	220 Ω Resistors
LED	Tilt Switch	M2M Wires	

III. Procedure

Complete all the connections using the following circuit diagram. Connect the Arduino to a computer and load the program.

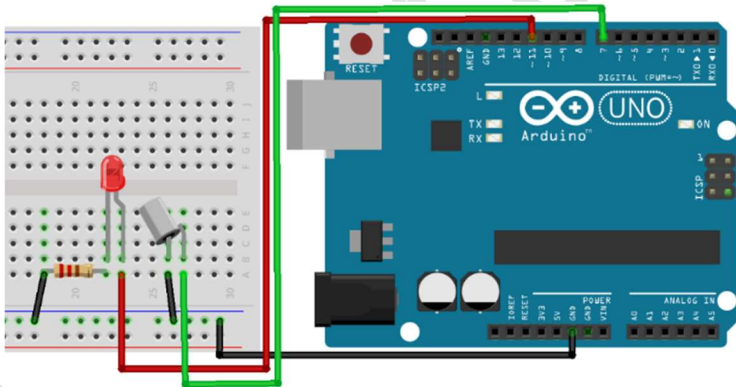


Figure 6 – Tilt Switch

IV. Desired Outcome

When the bread board is held in horizontal position, the tilt switch is in vertical position, and therefore the LED is switched off. When the bread-board is held in vertical position, the tilt switch is in horizontal position, and therefore the LED is switched on. This demonstrates the need for a mechanical movement to complete a circuit.

Experiment 7 – Using a Servo Motor

I. Goal

In this experiment you will learn to connect and run a simple servo motor to an Arduino board.

II. Material Required

Arduino	USB Cable	Servo Motor SG90
M2M Wires		

III. Procedure & Circuit Diagram

Complete all the connections using the following circuit diagram. Connect the Arduino to a computer and load the program.

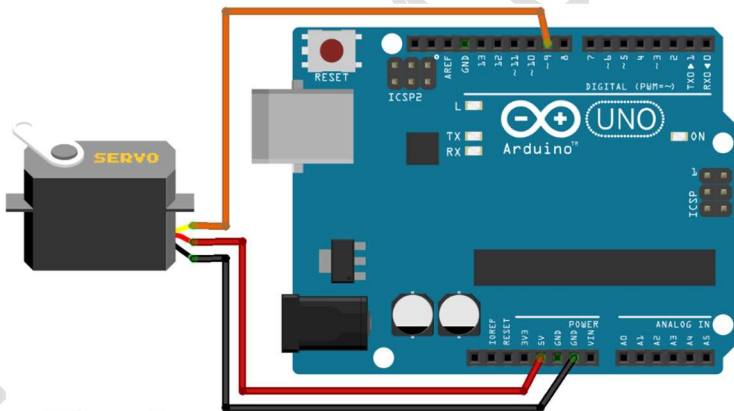


Figure 7 – Using a Servo Motor

IV. Desired Outcome

Once the program is loaded you will notice that the shaft of the servo motor starts rotating slowly. From its neutral position, the blade will turn 90° in one direction and returns to its neutral position and later turn another 90° in the opposite direction. Thus, it will complete a 180° degree movement and return to its original position.

Experiment 8 – Using a Stepper Motor

I. Goal

The goal of this experiment is to demonstrate the use of a stepper motor with an Arduino.

II. Material Required

Arduino	USB Cable	5V Stepper Motor
ULN2003 Board	M2M Wires	

III. Procedure & Circuit Diagram

Complete all the connections using the following circuit diagram. Connect the Arduino to a computer and load the program.

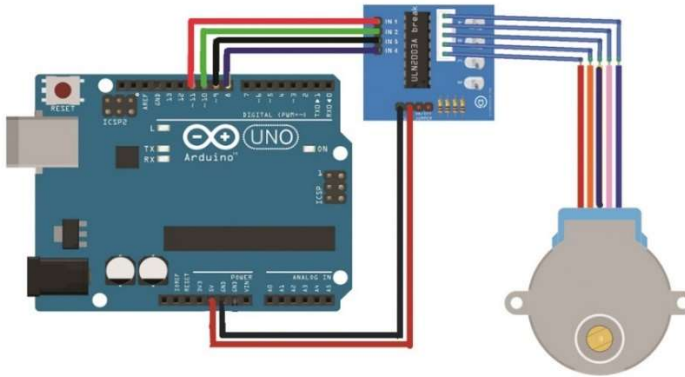


Figure 8 – Using a Stepper Motor

IV. Desired Outcome

As soon as the program is loaded, the shaft on the stepper motor starts rotating in slow jerks. Since the shaft is small, its movement may not be visible clearly. To see the movement of the shaft, attach a small tape to it.

Chapter 4 – Working with Displays

There must be a way to show the outcome or result of an action. In electronic items and in robots this is done using display modules. There are several different types of display – starting from a simple one-digit display to a complex LCD screen that can display information on a wider screen.

This chapter deals with the introduction to the various types displays. We will learn how to connect these displays to Arduino.

Experiment 9 – Seven Segment Display

I. Goal

The goal of this experiment is to demonstrate how a single digit display works. We will understand the circuitry and the connections that make a display work.

II. Material Required

Arduino	USB Cable	Breadboard
7-segment display	M2M Wires	220Ω Resistors

III. Procedure & Circuit Diagram

Complete all the connections using the following circuit diagram. Connect the Arduino to a computer and load the program.

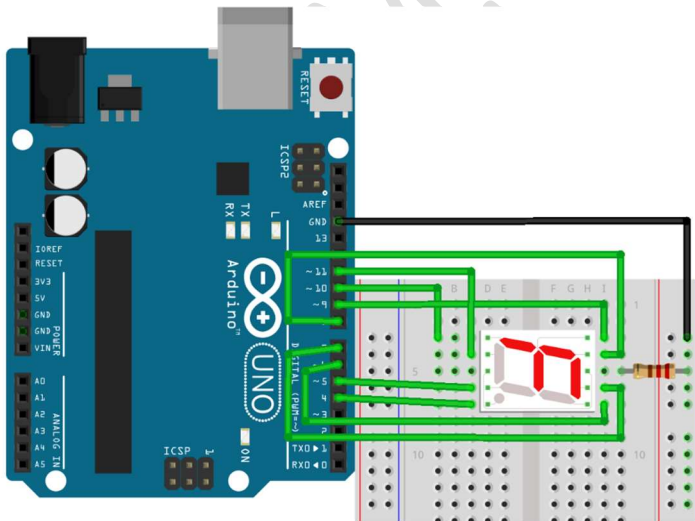


Figure 9 – Seven Segment Display

IV. Desired Outcome

As soon as the program is loaded, the display unit will display numbers from 1 to 9 in a sequence followed by letters A to F.

Experiment 10 – Simple Counter

I. Goal

The goal of this experiment is to demonstrate the display of 4-digit numbers using a 4-digit 7-segment display unit.

II. Material Required

Arduino	USB Cable	Breadboard
M2M Wires	4-digit display	

III. Procedure & Circuit Diagrams

Complete all the connections using the following circuit diagram. Connect the Arduino to a computer and load the program.

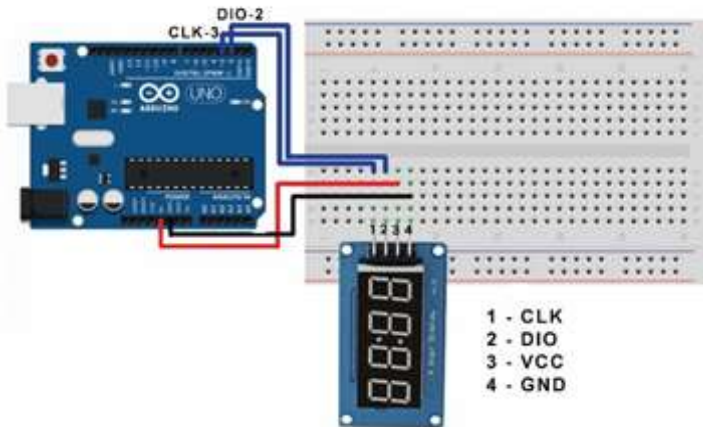


Figure 10 – Simple Counter

IV. Desired Outcome

Once all the connections are in place and the program is uploaded, the display will start displaying a series of numbers from 1 to 9999 in a quick increasing order. All the four 7-segment display digits are visible one after the other.

Experiment 11 – Using LCD (1602)

I. Goal

The goal of this experiment is to demonstrate the display of a combination of numbers and text within a single display unit.

II. Material Required

Arduino	USB Cable	Breadboard
LCD1602 Display	M2M Wires	

III. Procedure & Circuit Diagram

Complete all the connections using the following circuit diagram. Connect the Arduino to a computer and load the program.

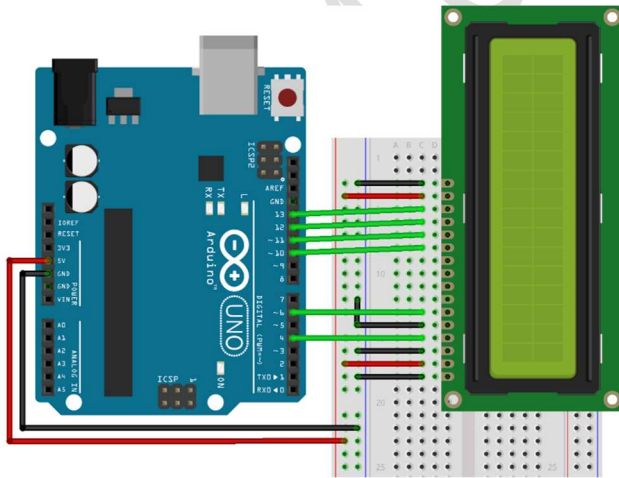


Figure 11 – Using LCD (1602)

IV. Desired Outcome

As soon as the program is loaded, and if the connections are made correctly, the LCD panel will display a message. If the message is broken or not visible properly, you have to check the wires and ensure that they are not loose or incorrectly connected. Use reset button if needed.

Chapter 5 – Working with Sensors

Sensors are those electronic units that can sense or detect. They are capable of detecting and/or measuring a change in their environment such as temperature, presence and intensity of sunlight, humidity and even flames. Certain sensors, like the IR sensors, can receive signals and interpret what they mean. There are also sensors, like the ultrasonic sensor that can send signals out in search of an object and detect how far that object is at.

In this chapter we will work with various sensors and learn how to connect them to Arduino and visually see what a sensor is able to detect. The role of sensors in robotics and artificial intelligence is huge. The topic of AI and robotics is of no relevance to us without the sensors.

Experiment 12 – IR Receiver and Controller

I. Goal

The goal of this experiment is to demonstrate how to integrate an Infra-Red Receiver (IR) with Arduino and capture the signals sent from a remote controller (RC). The captured signals will be displayed using LCD1602 display unit.

II. Material Required

Arduino	USB Cable	Breadboard
IR Receiver	IR Remote	LCD1602 Display
Jumper Wires		

III. Procedure & Circuit Diagrams

Complete all the connections using the following circuit diagram. Connect the Arduino to a computer and load the program.

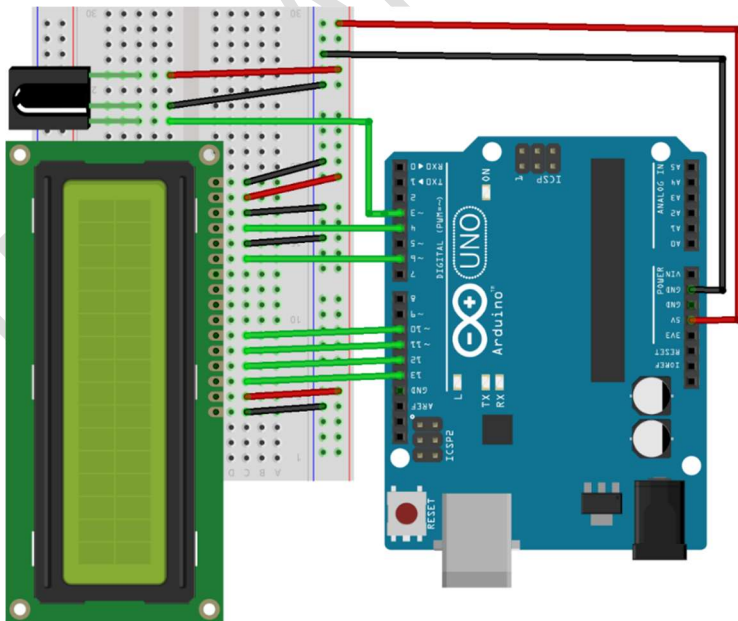


Figure 12 – IR Receiver and Controller

IV. Desired Outcome

Once the program is loaded, the IR receiver is ready to receive the signals. Take the remote controller provided in your kit and press any button on it. You will notice that a text message gets displayed in the display unit. This is a unique message that pertains to that button. Press any other button and you will notice that the display changes to a new message. You can continue to play with all the buttons and notice how the message changes each time.

You are free to test any other remote controller that you may have. The IR receiver will be able to capture signals from all the remotes and display the corresponding message.

No two buttons on the same remote will ever show the same message.

Experiment 13 – Ultrasonic Distance Sensor

I. Goal

The goal of this experiment is to integrate a distance sensor with an Arduino board and measure the distances of the objects which are in its path and range.

II. Material Required

Arduino	USB Cable	Breadboard
Ultrasonic Sensor	Jumper Wires	LCD16O2 Display

III. Procedure & Circuit Diagram

Complete all the connections using the following circuit diagram. Connect the Arduino to a computer and load the program.

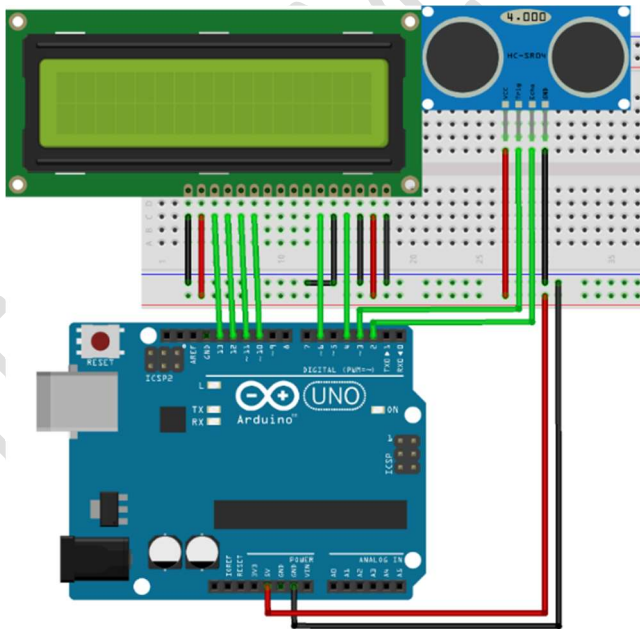


Figure 13 – Ultrasonic Distance Sensor

IV. Desired Outcome

As soon as the program is loaded, the ultrasonic device is ready for business. Place any object, say your hand, in front of the sensor. You will notice that the LCD1602 display will immediately display the distance of your hand from the sensor. You can now place various objects in front of the ultrasonic sensor to know how far they are from it.

TEXOHAM.COM

Experiment 14 – Using a Photoresistor

I. Goal

The goal of this experiment is to demonstrate the integration of a photo resistor with Arduino and display the measurements.

II. Material Required

Arduino	USB Cable	Breadboard
Photo Resistor	LCD1602 Display	10K Ω Resistor
Jumper Wires		

III. Procedure & Circuit Diagrams

Complete all the connections using the following circuit diagram. Connect the Arduino to a computer and load the program.

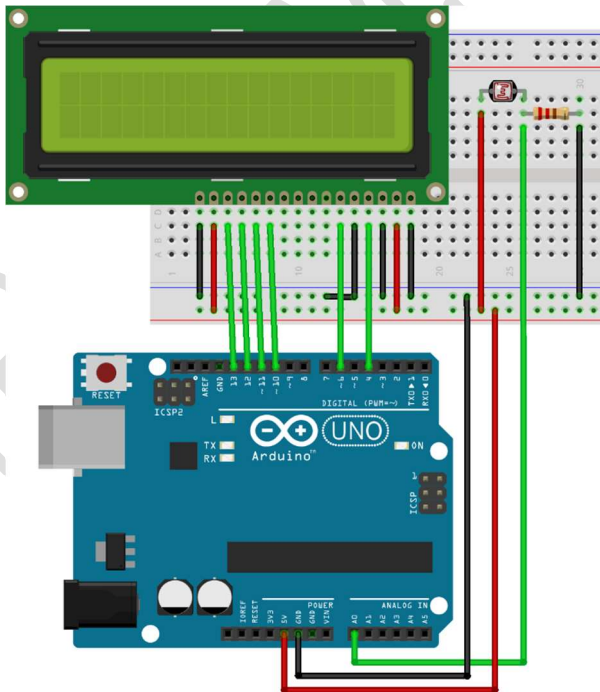


Figure 14 – Using a Photoresistor

IV. Desired Outcome

Once the program is uploaded, the photoresistor is ready for operations. It can detect the presence of light in its path. Use a torch light (torch on your smartphone) and direct it on to the surface of the photoresistor. Note the value on the display. You will notice that the value on the display, which is a measure of intensity of light, increases with increased intensity of light and decreases with decreased intensity of light.

Experiment 16 – Active Buzzer

I. Goal

The goal of this experiment is to integrate a buzzer with an Arduino unit.

II. Material Required

Arduino	USB Cable	Active Buzzer
Jumper Wires		

III. Procedure

Complete all the connections using the following circuit diagram. Connect the Arduino to a computer and load the program.

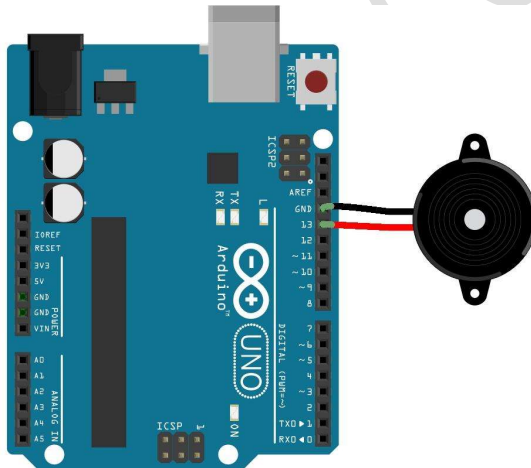


Figure 16 – Active Buzzer

IV. Desired Outcome

As soon as the program is loaded, you will notice the active buzzer producing a continuous sharp shrill sound.

Experiment 17 – Flame Sensor with Buzzer

I. Goal

The goal of this experiment is to demonstrate the use a buzzer and a flame sensor with Arduino.

II. Material Required

Arduino	USB Cable	Breadboard
LED	Flame Sensor Module	Active Buzzer
220 Ω Resistor	Jumper Wires	

III. Procedure

Complete all the connections using the following circuit diagram. Connect the Arduino to a computer and load the program.

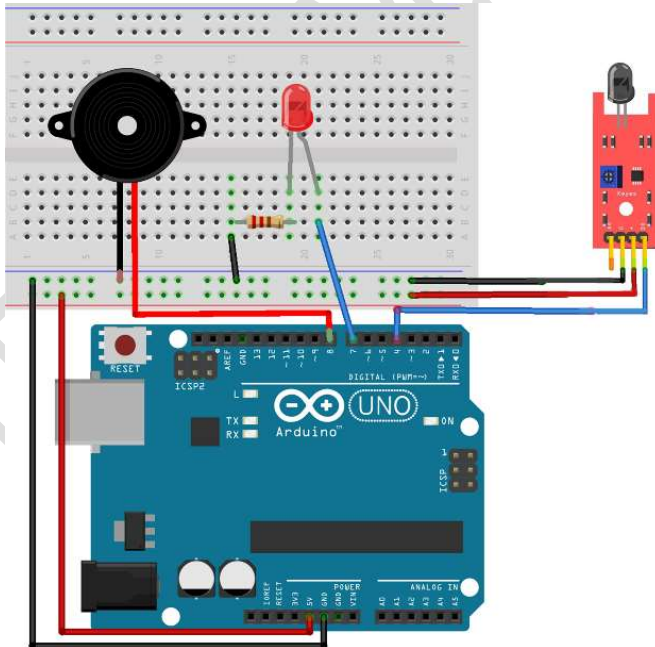


Figure 17 – Flame Sensor with Buzzer

IV. Desired Outcome

As soon as the program is loaded, the flame sensor module will be ready to detect flames. Light a match stick and place it in the vicinity of the sensor module. When it detects a flame, the LED light will start blinking and the buzzer will start buzzing. This will continue as long as the match stick keeps burning. Once the flame is extinguished, the LED light stops blinking and the buzzer will stop buzzing.

You can keep testing with various levels of flame. But be careful when working with the match sticks. Do not get too close to the flames. Do not take the burning match sticks too close to Arduino or sensor or the breadboard.

Experiment 18 – Fun Game with LCD

I. Goal

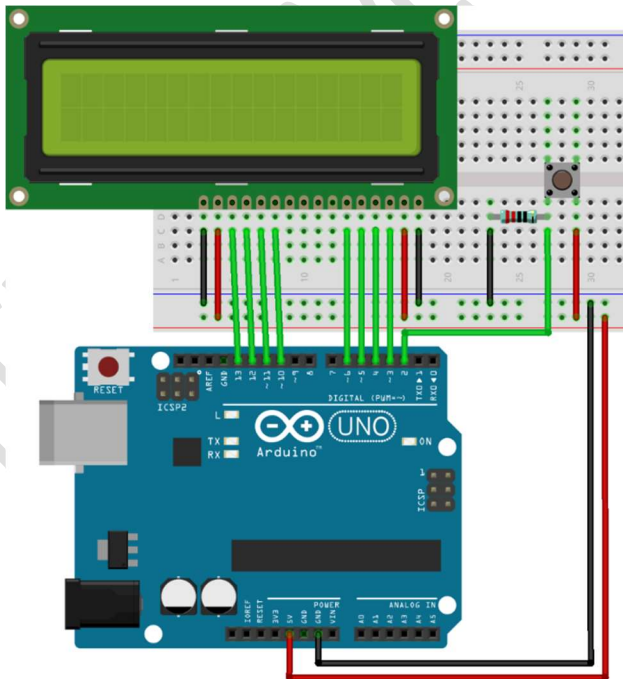
To create a simple fun game using LCD1602 display and a button. The flow of the game is controlled using a button.

II. Material Required

Arduino	USB Cable	Breadboard
LCD1602 Display	Push Buttons	Jumper Wires

III. Procedure & Circuit Diagram

Complete all the connections using the following circuit diagram. Connect the Arduino to a computer and load the program.



IV. Desired Outcome

As soon as the program is loaded the game is ready for you to play.

V. How to play the game?



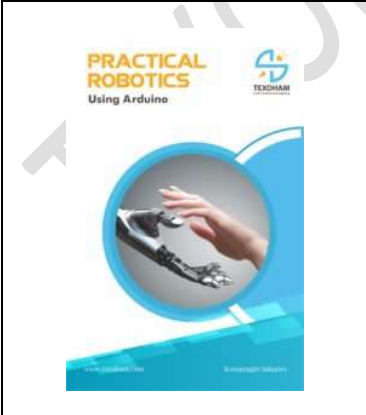
To play the game follow these instructions:

1. Press the push button to start the game.
2. You will notice a small human figure in the lower row of the display unit on the left side. This human figure can be made to jump up by pressing the push button.
3. A series of the blocks will move from the right side of the screen to the left. Each of these blocks will take up one-character space on the bottom row. These blocks keep moving in one, two, three or even four blocks at a time.
4. You will also notice that similar blocks will keep moving from left to right on the top row of the display.
5. Your skill in playing this game is determined by your ability to dodge the blocks from hitting the human figure. Remember that you can control the position of the human figure by making it jump from the bottom row to the top row, thus avoid being hit by the blocks.
6. Do not forget that if you jump in the path of the blocks in the top row, you will lose the game.
7. The game is over as soon as you get hit by a block on the top or the bottom row.
8. You will notice the final score as soon as the game is over.
9. You can restart the game by pressing the push button.

This game was developed by Joshua Brooks. The game in its original format may be downloaded from: <https://www.instructables.com/id/Arduino-LCD-Game/>

Our Products

Please visit www.texoham.com to know more about our products

	<ul style="list-style-type: none">• Use this kit to master Fundamentals of Robotics.• Kit comes with several types of displays, sensors, motors and accessories.• Online support include manual, source code and libraries.
	<ul style="list-style-type: none">• Use this kit to complete several exciting practical robotics projects.• Kit comes with several types of displays, sensors, modules, DC Motors, motor drivers, wheels and other accessories.• Online support include manual, source code and libraries.
	<ul style="list-style-type: none">• Use this book to master the fundamentals of robotics and extend your knowledge to complete some practical projects.• Using this book, you will develop a good understanding of sensors, modules, drivers and prototyping approaches?• Online support include manual, source code and libraries.