A Project Based Lab Report

On

SMART BLIND STICK USING AURDINO

SUBMITTED BY:

I.D NUMBER NAME

180060017 S. SUBBARAO

180060018 K.NAGARJUNA

180060019 D.SAI VENKAT

Sec -01 , Batch-05

II/IV BACHELOR OF TECHNOLOGY

IN

Electrical and Electronics engineering

(SEMESTER-IV)

April 2020

ACKNOWLEDGEMENTS

it is great pleasure for us to express our gratitude to our honorable president **sri. koneru satyanarayana,** for giving the opportunity and platform with facilities in accomplishing the project based laboratory report.

we express the sincere gratitude to our principal **PROF DR.K.SUBBARAO** for his administration towards our academic growth.

we express sincere gratitude to HOD-EEE **DR.S.V.N.LALITHA** for his leadership and constant motivation provided in successful completion of our academic semester. we record it as my privilege to deeply thank for providing us the efficient faculty and facilities to make our ideas into reality.

we express our sincere thanks to our project supervisor **MR.K.UDAY KIRAN**

for his novel association of ideas, encouragement, appreciation and intellectual zeal which motivated us to venture this project successfully.

finally, it is pleased to acknowledge the indebtedness to all those who devoted themselves directly or indirectly to make this project report success.

180060017 S. SUBBARAO

180060018 K.NAGARJUNA

180060019 D.SAI VENKAT

ABSTRACT

This project describes ultrasonic blind walking stick with the use of arduino. according to who, 30 million peoples are permanently blind and 285 billion peoples with vision impairment . if you notice them , you can very well know about it they can't walk without the help of other. one has to ask guidance to reach their destination. they have to face more struggles in their life daily life. using this blind stick , a person can walk more confidently. this stick detects the object in front of the person and give response to the user either by vibrating or through command. so, the person can walk without any fear. this device will be best solution to overcome their difficulties

INDEX

INTRODUCTION

visually impaired persons have difficulty to interact and feel their environment. they have little contact with surroundings. physical movement is a challenge for visually impaired persons, because it can become tricky to distinguish obstacles appearing in front of them, and they are not able to move from one place to another.

they depend on their families for mobility and financial support. their mobility opposes them from interacting with people and social activities. in the past, different systems are designed with limitations without a solid understanding of the nonvisual perception. researchers have spent the decades to develop an intelligent and smart stick to assist and alert visually impaired persons from obstacles and give information about their location. over the last decades, research has been conducted for new devices to design a good and reliable system for visually impaired persons to detect obstacles and warn them at danger places.

smart walking stick is specially designed to detect obstacles which may help the blind to navigate care-free. the audio messages will keep the user alert and considerably reduce accidents. a voice enabled automatic switching is also incorporated to help them in private space as well. this system presents a concept to provide a smart electronic aid for blind people, both in public and private space the proposed system contains the ultrasonic sensor, water sensor, voice play back board, raspberry pi and speaker. the proposed system detects the obstacle images which are present in outdoor and indoor with the help of a camera. the stick measures the distance between the objects and smart walking stick by using an ultrasonic sensor. when any objects or obstacles come in range of an ultrasonic sensor and it make buzzer sound.

AIM

The main objective of this is to provide an application for blind people to detect the obstacles in various directions, detecting pits and manholes on the ground to make free to walk. in an innovative stick is designed for the visually disabled people for their easy navigation.

APPARATUS

1.aurdino board uno

2.ultra sonic sensor

3.b10 buzzer

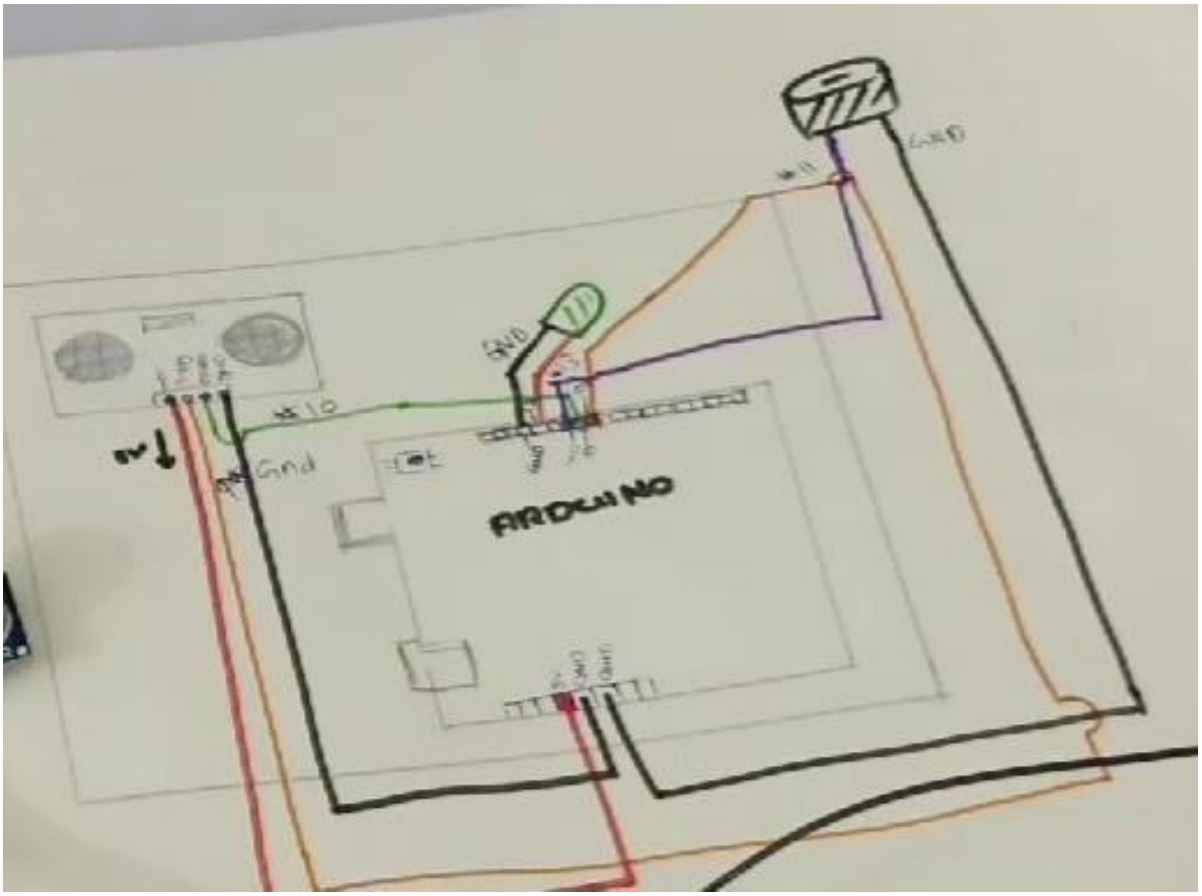
4.jumper wires

5.led

6. aurdino software

7.stick

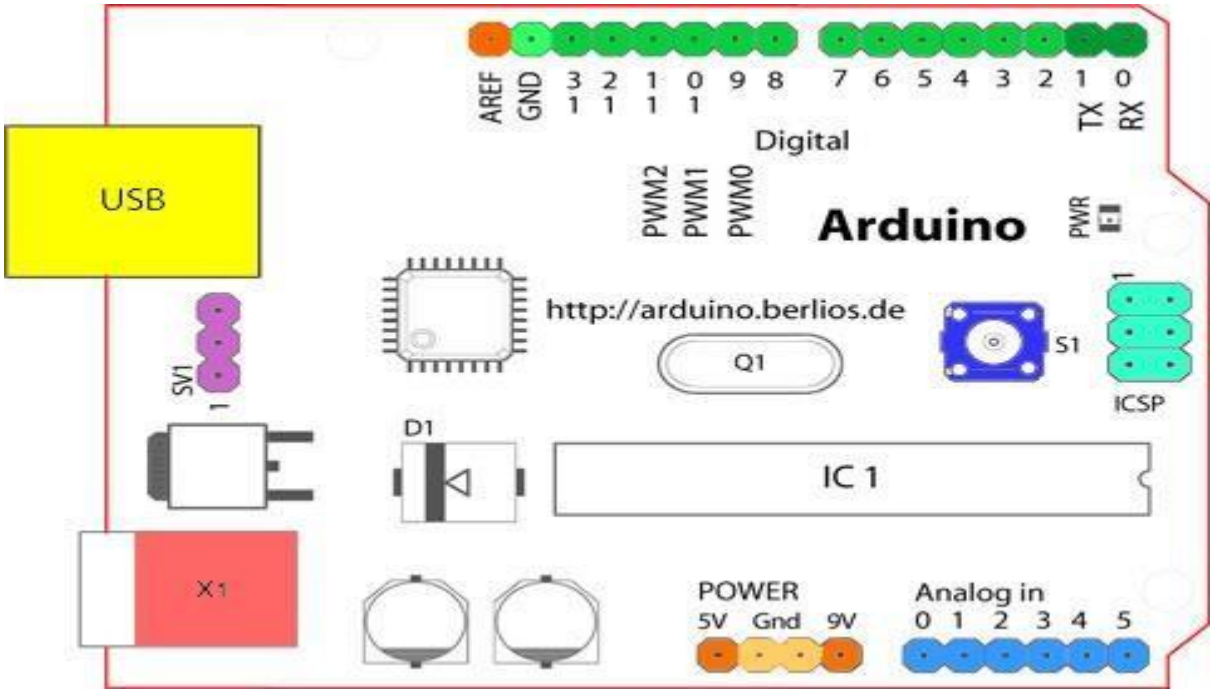
BLOCK DIAGRAMS



PROCEDURE

INTRODUCTION TO THE ARDUINO BOARD

looking at the board from the top down, this is an outline of what you will see (parts of the board you might interact with in the course of normal use are highlighted):



STARTING CLOCKWISE FROM THE TOP CENTER:-

- analog reference pin (orange)
- digital ground (light green)
- digital pins 2-13 (green)
- digital pins 0-1/serial in/out - tx/rx (dark green) - these pins cannot be used for digital *i/o* (digitalread and digitalwrite) if you are also using serial communication (e.g. serial.begin).
- reset button - s1 (dark blue)

- ❖ in-circuit serial programmer (blue-green)
- ❖ analog in pins 0-5 (light blue)
- ❖ power and ground pins (power: orange, grounds: light orange)
- ❖ external power supply in (9-12vdc) - x1 (pink)
- ❖ toggles external power and usb power (place jumper on two pins closest to desired supply) - sv1 (purple)
- ❖ usb (used for uploading sketches to the board and for serial communication between the board and the computer; can be used to power the board) (yellow)

Digital Pins:-

in addition to the specific functions listed below, the digital pins on an arduino board can be usedforgeneralpurposeinputandoutputviathe pinmode(), digitalread(),and digitalwrite() comm ands. each pin has an internal pull-up resistor which can be turned on and off using digitalwrite() (w/ a value of high or low, respectively) when the pin is configured as an input. the maximum current per pin is 40 ma.

.serial: 0 (rx) and 1 (tx). used to receive (rx) and transmit (tx) ttl serial data. on the arduino diecimila, these pins are connected to the corresponding pins of the ftdi usb-to-ttl serial chip. on the arduino bt, they are connected to the corresponding pins of the wt11 bluetooth module. on the arduino mini and lilypad arduino, they are intended for use with an external ttl serial module (e.g. the mini-usb adapter).

- external interrupts: 2 and 3. these pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. see the attachinterrupt() function for details.
- pwm: 3, 5, 6, 9, 10, and 11. provide 8-bit pwm output with the analogwrite() function. on boards with an atmega8, pwm output is available only on pins 9, 10, and 11.
- bt reset: 7. (arduino bt-only) connected to the reset line of the bluetooth module.
- spi: 10 (ss), 11 (mosi), 12 (miso), 13 (sck). these pins support spi communication, which, although provided by the underlying hardware, is not currently included in the arduino language.
- led: 13. on the diecimila and lilypad, there is a built-in led connected to digital pin 13. when the pin is high value, the led is on, when the pin is low, it's off.

ANALOG PINS:-

in addition to the specific functions listed below, the analog input pins support 10-bit analog-to-digital conversion (adc) using the analogread() function. most of the analog inputs can also be used as digital pins: analog input 0 as digital pin 14 through analog input 5 as digital pin 19. analog inputs 6 and 7 (present on the mini and bt) cannot be used as digital pins.

- i^2c: 4 (sda) and 5 (scl). support i^2c (twi) communication using the wire library (documentation on the wiring website).

POWER PINS:-

- vin (sometimes labelled "9v"). the input voltage to the arduino board when it's using an external power source (as opposed to 5 volts from the usb connection or other regulated power source). you can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin. note that different boards accept different input voltages ranges, please see the documentation for your board. also note that the lilypad has no vin pin and accepts only a regulated input.
- 5v. the regulated power supply used to power the microcontroller and other components on the board. this can come either from vin via an on-board regulator, or be supplied by usb or another regulated 5v supply.
- 3v3. (diecimila-only) a 3.3 volt supply generated by the on-board ftdi chip.
- gnd. ground pins.

OTHER PINS:-

- aref. reference voltage for the analog inputs. not currently supported by the arduino software.
- reset. (diecimila-only) bring this line low to reset the microcontroller. typically used to add a reset button to shields which block the one on the board.

the text of the arduino getting started guide is licensed under a creative commons attribution-sharealike 3.0 license. code samples in the guide are released into the public domain.

arduino isan open-sourcehardware and software company, projectandusercommunity that designs and manufactures single-board microcontrollers and microcontroller kits for building digital devices. its products are licensed under the gnu lesser general public license (lgpl) or the gnu general public license (gpl), permitting the manufacture of arduino boards and software distribution by anyone. arduino boards are available commercially in preassembled form or as do-it-yourself (diy) kits.

arduino board designs use a variety of microprocessors and controllers. the boards are equipped with sets of digital and analog input/output (i/o) pins that may be interfaced to various expansion boards ('shields') or breadboards (for prototyping) and other circuits. the boards feature serial communications interfaces, including universal serial bus (usb) on some models, which are also used for loading programs from personal computers. the microcontrollers can be programmed using c and c++ programming languages. in addition to using traditional compiler toolchains, the arduino project provides an integrated development environment (ide) based on the processing language project.

the arduino project started in 2005 as a program for students at the interaction design institute ivrea in ivrea, italy, aiming to provide a low-cost and easy way for novices and professionals to create devices that interact with their environment using sensors and actuators. common examples of such devices intended for beginner hobbyists include simple robots, thermostats and motion detectors.

the name arduino comes from a bar in ivrea, italy, where some of the founders of the project used to meet. the bar was named after arduin of ivrea, who was the margrave of the march of ivrea and king of italy from 1002 to 1014

APPLICATIONS:-

- arduboy, a handheld game console based on arduino
- arduinome, a midi controller device that mimics the monome
- ardupilot, drone software and hardware
- ardusat, a cubesat based on arduino.

- c-stem studio, a platform for hands-on integrated learning of computing, science, technology, engineering, and mathematics (c-stem) with robotics.
- data loggers for scientific research.
- obduino, a trip computer that uses the on-board diagnostics interface found in most modern cars
- openevse an open-source electric vehicle charger
- xod, a visual programming language for arduino

SOFTWARE:-

a program for arduino hardware may be written in any programming language with compilers that produce binary machine code for the target processor. atmel provides a development environment for their 8-bit avr and 32-bit arm cortex-m based microcontrollers: avr studio (older) and atmel studio (newer)

IDE:-

the arduino integrated development environment (ide) is a cross-platform application (for windows, macos, and linux) that is written in the programming language java. it originated from the ide for the languages processing and wiring. it includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, and syntax highlighting, and provides simple one-click mechanisms to compile and upload programs to an arduino board. it also contains a message area, a text console, a toolbar with buttons for common functions and a hierarchy of operation menus. the source code for the ide is released under the gnu general public license, version 2.

the arduino ide supports the languages c and c++ using special rules of code structuring. the arduino ide supplies a software library from the wiring project, which provides many common input and output procedures. user-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main() into an executable cyclic executive program with the gnu toolchain, also included with the ide distribution. the arduino ide employs the program avrdude to convert the executable code into a text file in hexadecimal encoding that is loaded into the arduino board by a loader program in the board's firmware.

pro ide:-

on october 18th, 2019, arduino pro ide (alpha preview) was released. the system still uses arduino cli (command line interface), but improvements include a more professional development environment, autocompletion support, and git integration. the application frontend is based on the eclipse theia open source ide. the main features available in the alpha release are:

- modern, fully featured development environment
- dual mode, classic mode (identical to the classic arduino ide) and pro mode (file system view)
- new board manager
- new library manager
- board list
- basic auto-completion (arm targets only)
- git integration
- serial monitor
- dark mode

SKETCH[EDIT]:-

a *sketch* is a program written with the arduino ide sketches are saved on the development computer as text files with the file extension .ino. arduino software (ide) pre-1.0 saved sketches with the extension .pde.

a minimal arduino c/c++ program consists of only two functions

- setup(): this function is called once when a sketch starts after power-up or reset. it is used to initialize variables, input and output pin modes, and other libraries needed in the sketch. it is analogous to the function main()
- loop(): after setup() function exits (ends), the loop() function is executed repeatedly in the main program. it controls the board until the board is powered off or is reset. it is analogous to the function while(1)

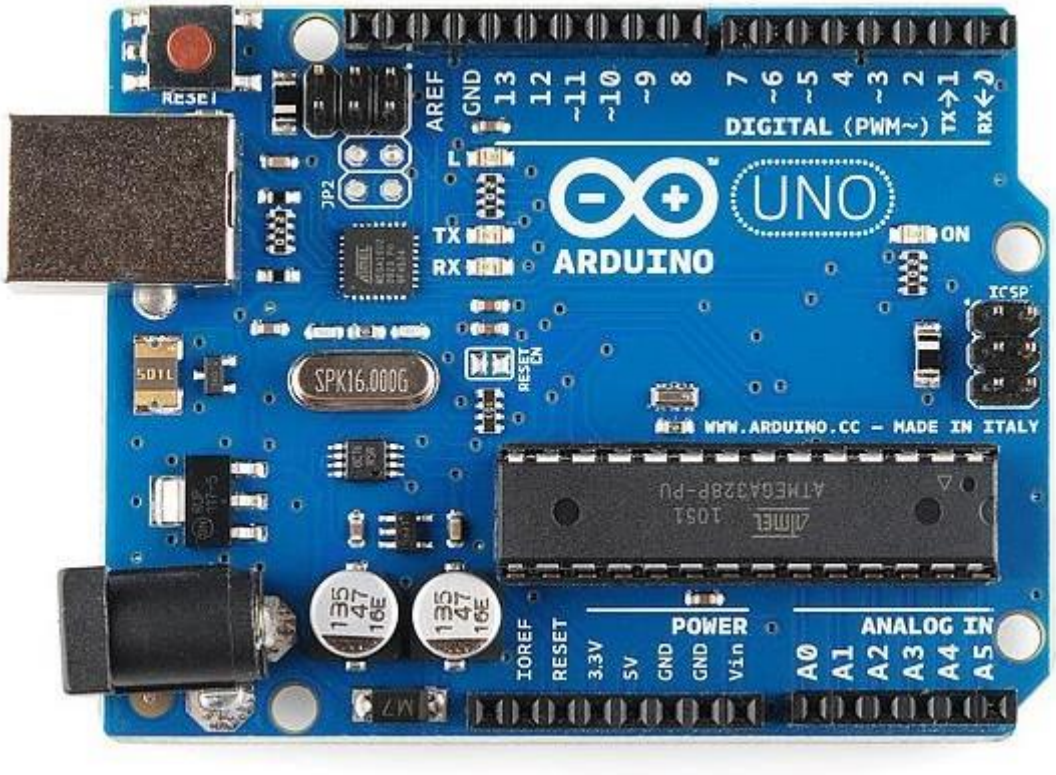
blink example

power led (red) and user led (green) attached to pin 13 on an arduino compatible board

most arduino boards contain a light-emitting diode (led) and a current limiting resistor connected between pin 13 and ground, which is a convenient feature for many tests and program functions.[63] a typical program used by beginners, akin to hello, world!, is "blink", which repeatedly blinks the on-board led integrated into the arduino board. this program uses the functions pinmode(), digitalwrite(), and delay(), which are provided by the internal libraries included in the ide environment. this program is usually loaded into a new arduino board by the manufacturer.

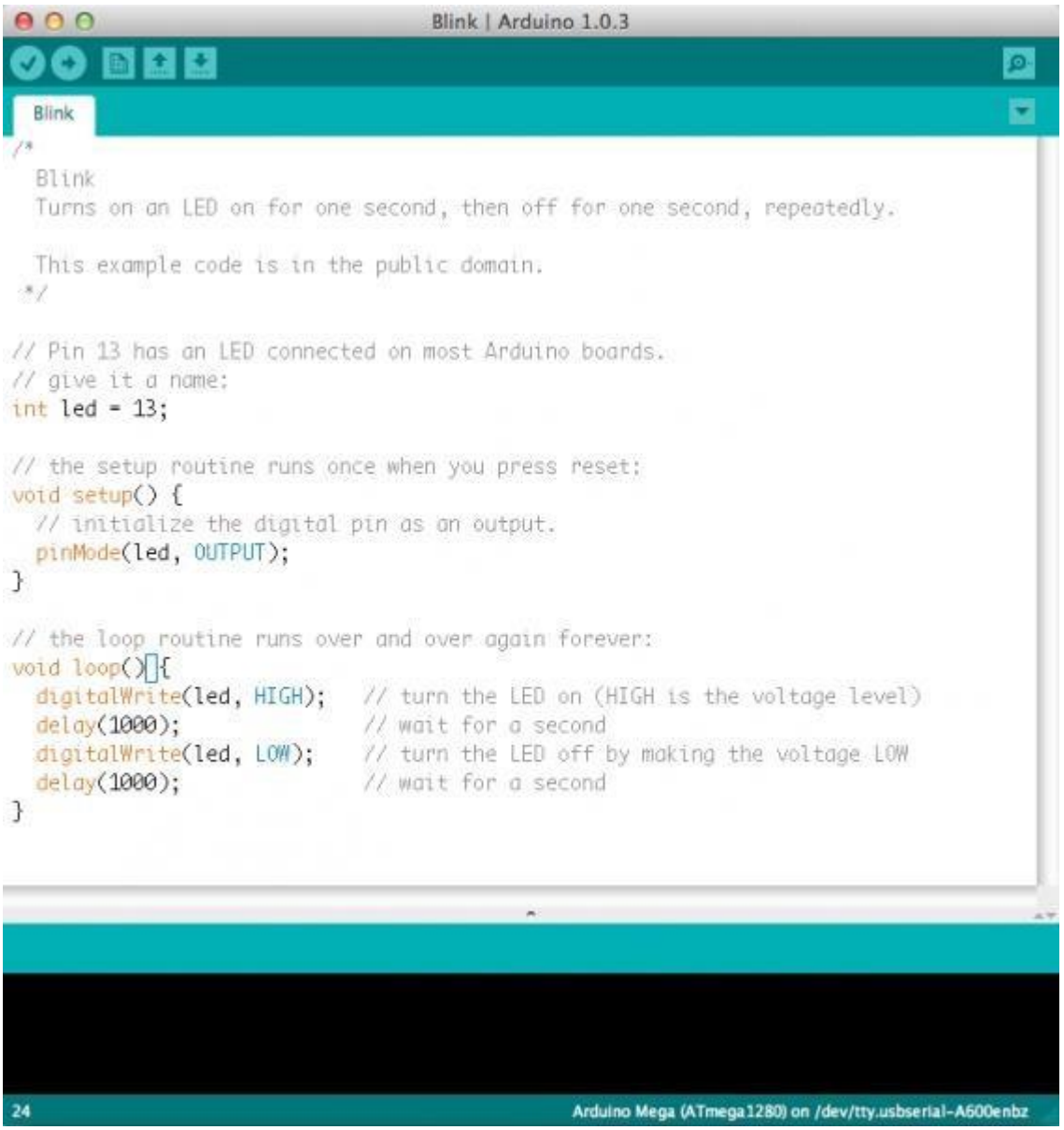
LIBRARIES[EDIT]

the open-source nature of the arduino project has facilitated the publication of many free software libraries that other developers use to augment their projects.



This is an Arduino Uno

the uno is one of the more popular boards in the arduino family and a great choice for beginners. we'll talk about what's on it and what it can do later in the tutorial.



This is a screenshot of the Arduino IDE.

believe it or not, those 10 lines of code are all you need to blink the on-board led on your arduino. the code might not make perfect sense right now, but, after reading this tutorial and the many more arduino tutorials waiting for you on our site, we'll get you up to speed in no time!

you will learn

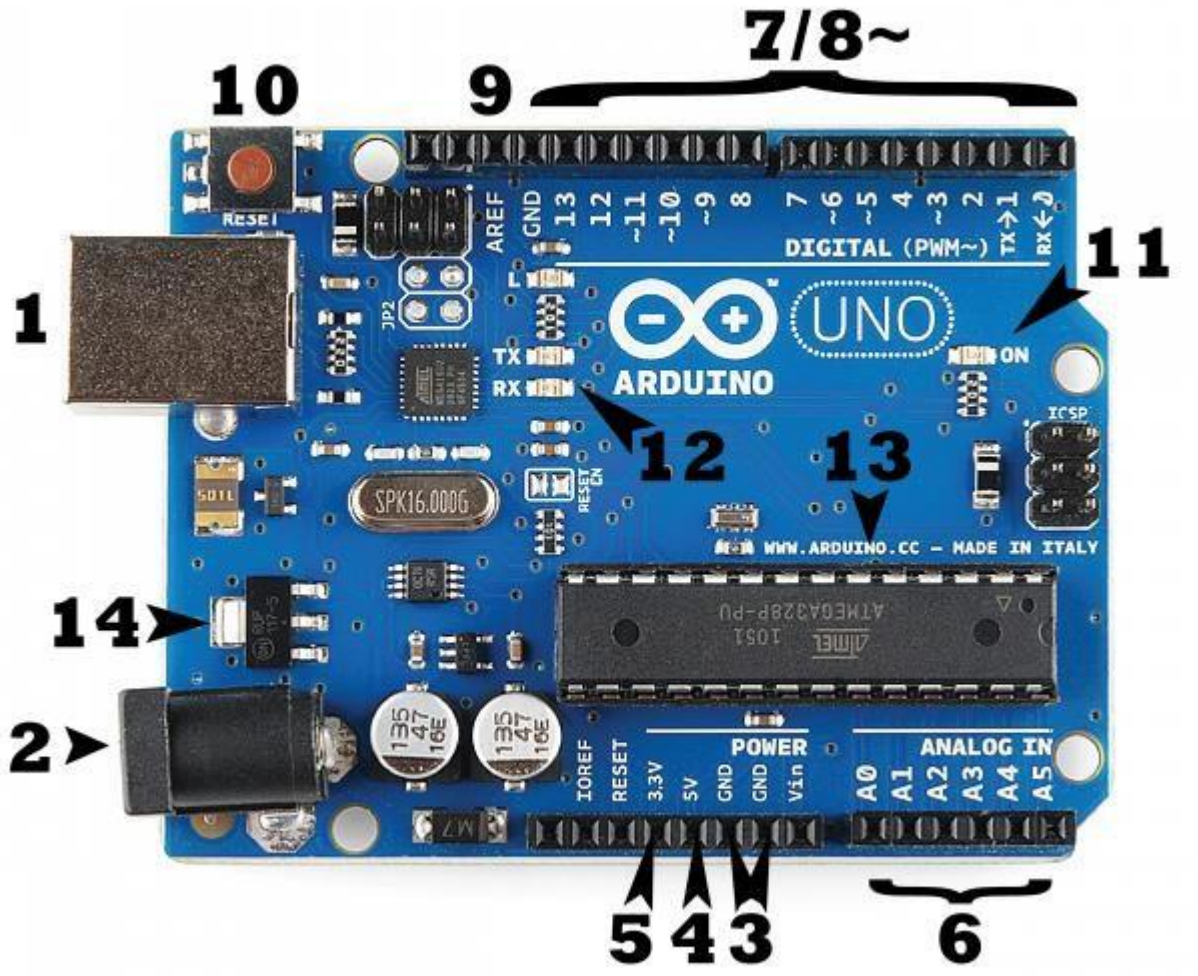
in this tutorial, we'll go over the following:

- what projects can be accomplished using an arduino
- what is on the typical arduino board and why

- the different varieties of arduino boards
- some useful widgets to use with your arduino

suggested reading

arduino is a great tool for people of all skill levels. however, you will have a much better time learning along side your arduino if you understand some basic fundamental electronics beforehand. we recommend that you have at least a decent understanding of these concepts before you dive in to the wonderful world of arduino.



Power (USB / Barrel Jack)

every arduino board needs a way to be connected to a power source. the arduino uno can be powered from a usb cable coming from your computer or a wall power supply (like this) that is terminated in a barrel jack. in the picture above the usb connection is labeled **(1)** and the barrel jack is labeled **(2)**.

the usb connection is also how you will load code onto your arduino board. more on how to program with arduino can be found in our installing and programming arduino tutorial.

note: do not use a power supply greater than 20 volts as you will overpower (and thereby destroy) your arduino. the recommended voltage for most arduino models is between 6 and 12 volts.

Pins (5V, 3.3V, GND, Analog, Digital, PWM, AREF)

the pins on your arduino are the places where you connect wires to construct a circuit (probably in conjuction with a breadboard and some wire. they usually have black plastic 'headers' that allow you to just plug a wire right into the board. the arduino has several different kinds of pins, each of which is labeled on the board and used for different functions.

- **gnd (3)**: short for 'ground'. there are several gnd pins on the arduino, any of which can be used to ground your circuit.
- **5v (4) & 3.3v (5)**: as you might guess, the 5v pin supplies 5 volts of power, and the 3.3v pin supplies 3.3 volts of power. most of the simple components used with the arduino run happily off of 5 or 3.3 volts.
- **analog (6)**: the area of pins under the 'analog in' label (a0 through a5 on the uno) are analog in pins. these pins can read the signal from an analog sensor (like a temperature sensor) and convert it into a digital value that we can read.
- **digital (7)**: across from the analog pins are the digital pins (0 through 13 on the uno). these pins can be used for both digital input (like telling if a button is pushed) and digital output (like powering an led).
- **pwm (8)**: you may have noticed the tilde (~) next to some of the digital pins (3, 5, 6, 9, 10, and 11 on the uno). these pins act as normal digital pins, but can also be used for something called pulse-width modulation (pwm). we have a tutorial on pwm, but for now, think of these pins as being able to simulate analog output (like fading an led in and out).

- **aref (9)**: stands for analog reference. most of the time you can leave this pin alone. it is sometimes used to set an external reference voltage (between 0 and 5 volts) as the upper limit for the analog input pins.

RESET BUTTON

just like the original nintendo, the arduino has a reset button (10). pushing it will temporarily connect the reset pin to ground and restart any code that is loaded on the arduino. this can be very useful if your code doesn't repeat, but you want to test it multiple times. unlike the original nintendo however, blowing on the arduino doesn't usually fix any problems.

POWER LED INDICATOR

just beneath and to the right of the word "uno" on your circuit board, there's a tiny led next to the word 'on' (11). this led should light up whenever you plug your arduino into a power source. if this light doesn't turn on, there's a good chance something is wrong. time to re-check your circuit!

TX RX LEDS

tx is short for transmit, rx is short for receive. these markings appear quite a bit in electronics to indicate the pins responsible for serial communication. in our case, there are two places on the arduino uno where tx and rx appear -- once by digital pins 0 and 1, and a second time next to the tx and rx indicator leds (12). these leds will give us some nice visual indications whenever our arduino is receiving or transmitting data (like when we're loading a new program onto the board).

MAIN IC

the black thing with all the metal legs is an ic, or integrated circuit (13). think of it as the brains of our arduino. the main ic on the arduino is slightly different from board type to board type, but is usually from the atmega line of ic's from the atmel company. this can be important, as you may need to know the ic type (along with your board type) before loading up a new program from the arduino software. this information can usually be found in writing on the top side of the ic. if you want to know more about the difference between various ic's, reading the datasheets is often a good idea.

VOLTAGE REGULATOR

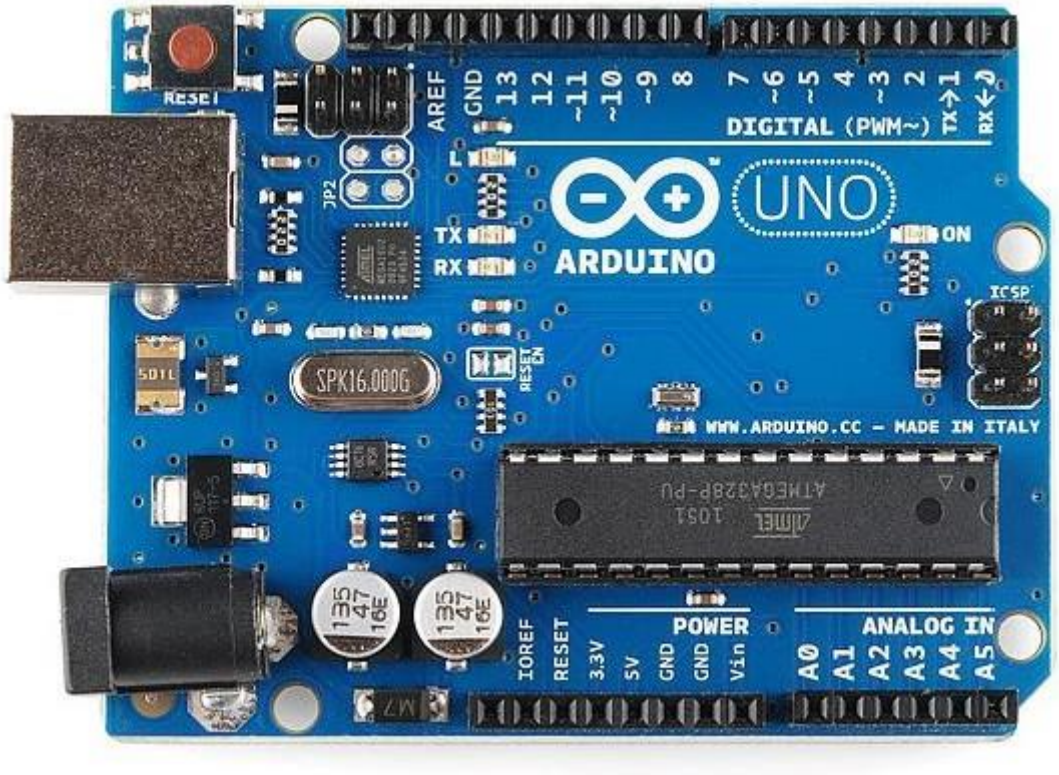
the voltage regulator (14) is not actually something you can (or should) interact with on the arduino. but it is potentially useful to know that it is there and what it's for. the voltage regulator does exactly what it says -- it controls the amount of voltage that is let into the arduino board. think of it as a kind of gatekeeper; it will turn away an extra voltage that might harm the circuit. of course, it has its limits, so don't hook up your arduino to anything greater than 20 volts.

THE ARDUINO FAMILY

arduino makes several different boards, each with different capabilities. in addition, part of being open source hardware means that others can modify and produce derivatives of arduino boards that provide even more form factors and functionality. if you're not sure which one is right for your project, check this guide for some helpful hints. here are a few options that are well-suited to someone new to the world of arduino:

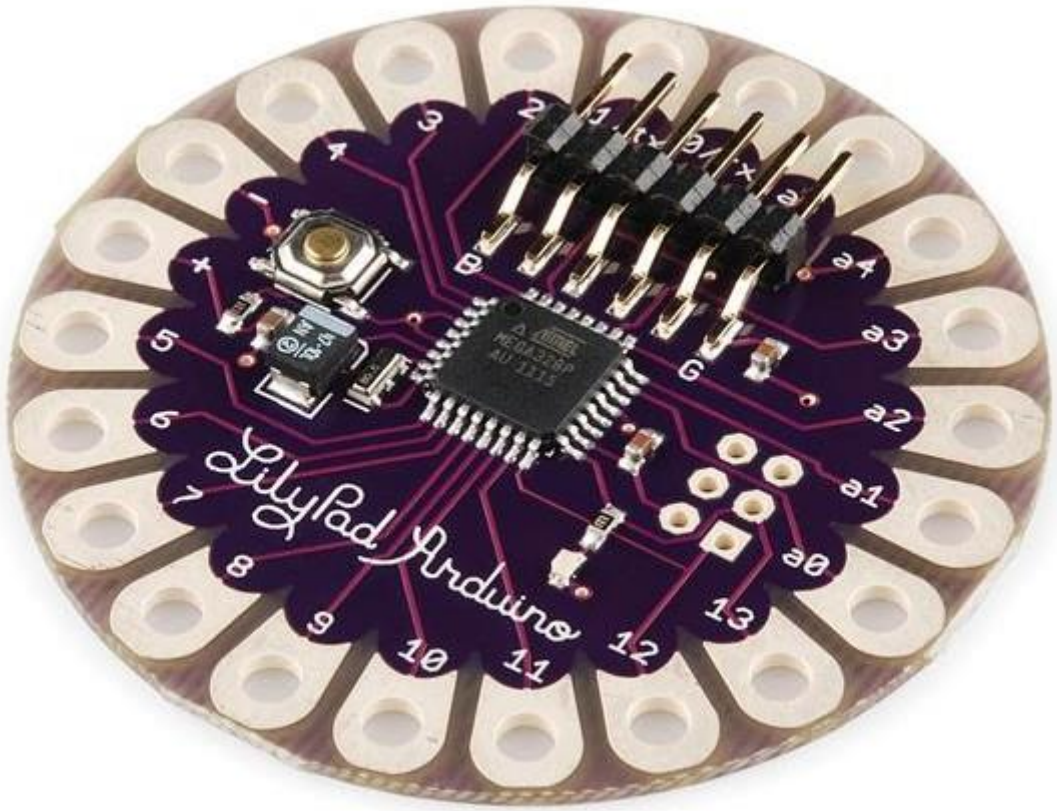
ARDUINO UNO (R3)

the uno is a great choice for your first arduino. it's got everything you need to get started, and nothing you don't. it has 14 digital input/output pins (of which 6 can be used as pwm outputs), 6 analog inputs, a usb connection, a power jack, a reset button and more. it contains everything needed to support the microcontroller; simply connect it to a computer with a usb cable or power it with a ac-to-dc adapter or battery to get started.



LILYPAD ARDUINO

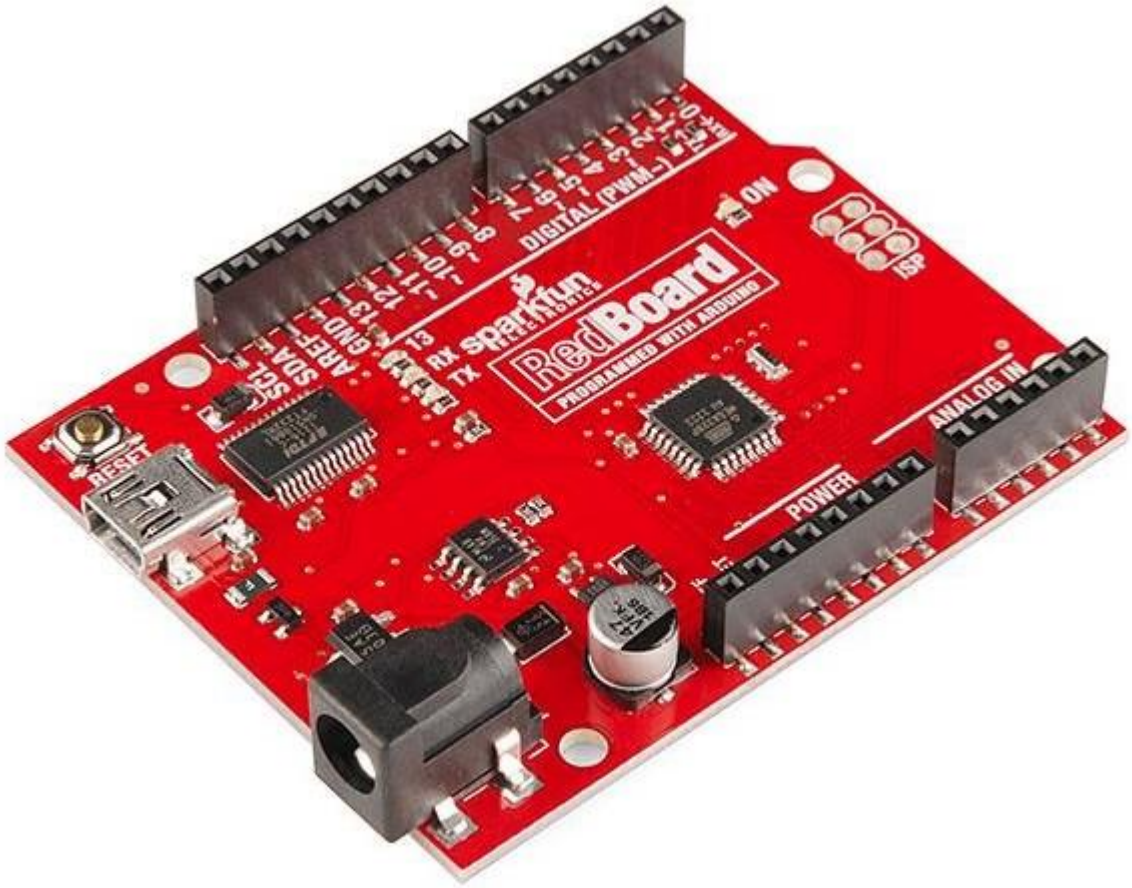
this is lilypad arduino main board! lilypad is a wearable e-textile technology developed by leah buechley and cooperatively designed by leah and sparkfun. each lilypad was creatively designed with large connecting pads and a flat back to allow them to be sewn into clothing with conductive thread. the lilypad also has its own family of input, output, power, and sensor boards that are also built specifically for e-textiles. they're even washable!



REDBOARD

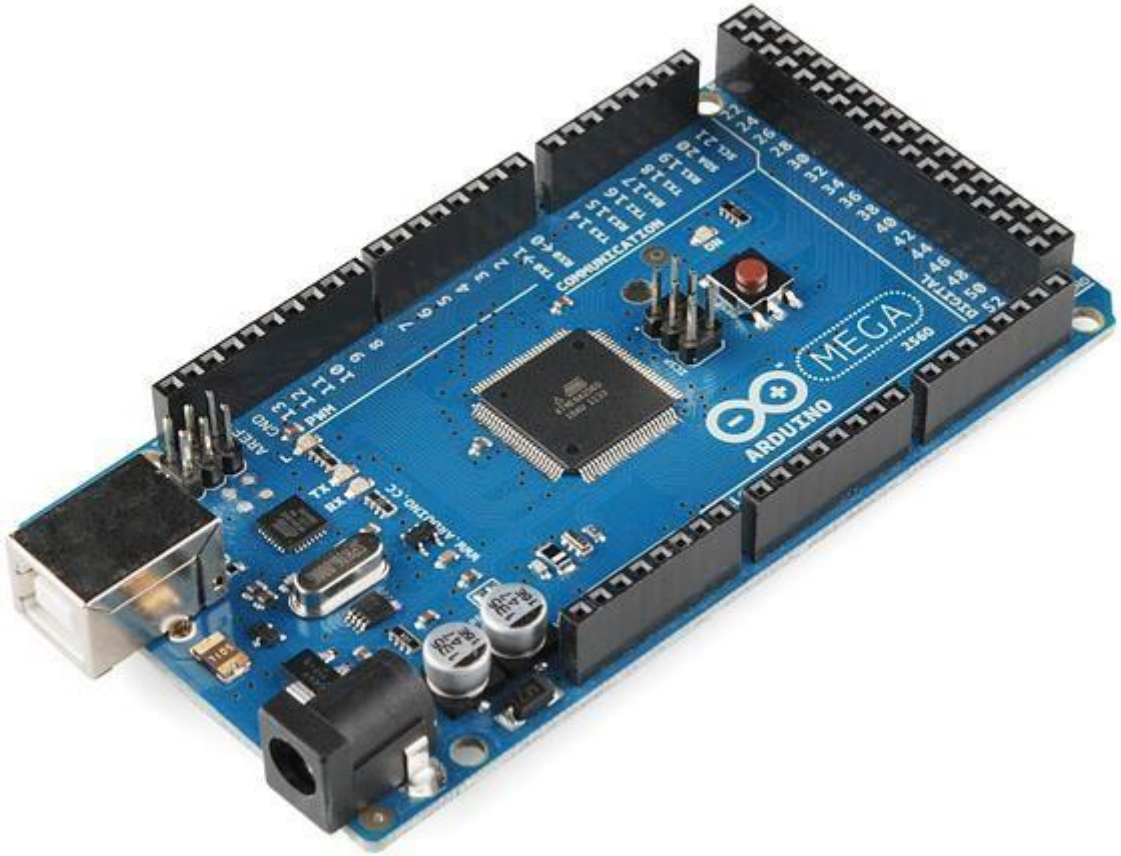
at sparkfun we use many arduinos and we're always looking for the simplest, most stable one. each board is a bit different and no one board has everything we want -- so we decided to make our own version that combines all our favorite features.

the redboard can be programmed over a usb mini-b cable using the arduino ide. it'll work on windows 8 without having to change your security settings (we used signed drivers, unlike the uno). it's more stable due to the usb/ftdi chip we used, plus it's completely flat on the back, making it easier to embed in your projects. just plug in the board, select "arduino uno" from the board menu and you're ready to upload code. you can power the redboard over usb or through the barrel jack. the on-board power regulator can handle anything from 7 to 15vdc.



ARDUINO MEGA (R3)

the arduino mega is like the uno's big brother. it has lots (*54!*) of digital input/output pins (14 can be used as pwm outputs), 16 analog inputs, a usb connection, a power jack, and a reset button. it contains everything needed to support the microcontroller; simply connect it to a computer with a usb cable or power it with a ac-to-dc adapter or battery to get started. the large number of pins make this board very handy for projects that require a bunch of digital inputs or outputs (like lots of leds or buttons).



ARDUINO LEONARDO

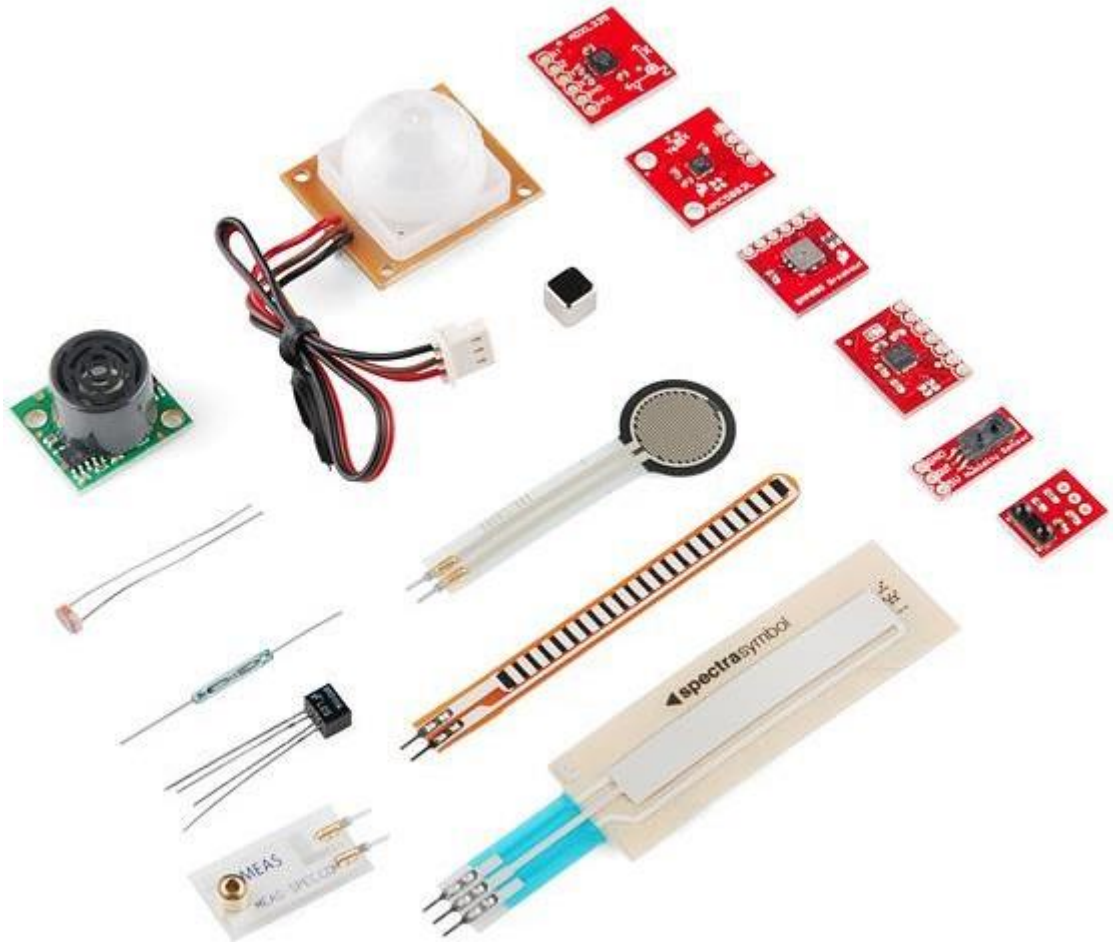
the leonardo is arduino's first development board to use one microcontroller with built-in usb. this means that it can be cheaper and simpler. also, because the board is handling usb directly, code libraries are available which allow the board to emulate a computer keyboard, mouse, and more!

the extended family

while your arduino board sure is pretty, it can't do a whole lot on its own -- you've got to hook it up to something. there are lots of tutorials here on learn as well as the links back in the 'what does it do' section, but rarely do we talk about the general *kinds* of things you can easily hook into. in this section we'll introduce basic **sensors** as well as arduino **shields**, two of the most handy tools to use in bringing your projects to life.

sensors

with some simple code, the arduino can control and interact with awidevarietyof sensors thingsthatcanmeasure light, temperature, degreeofflex, pressure, proxi mity, acceleration, carbon monoxide, radioactivity, humidity, barometric pressure, you name it, you can sense it!

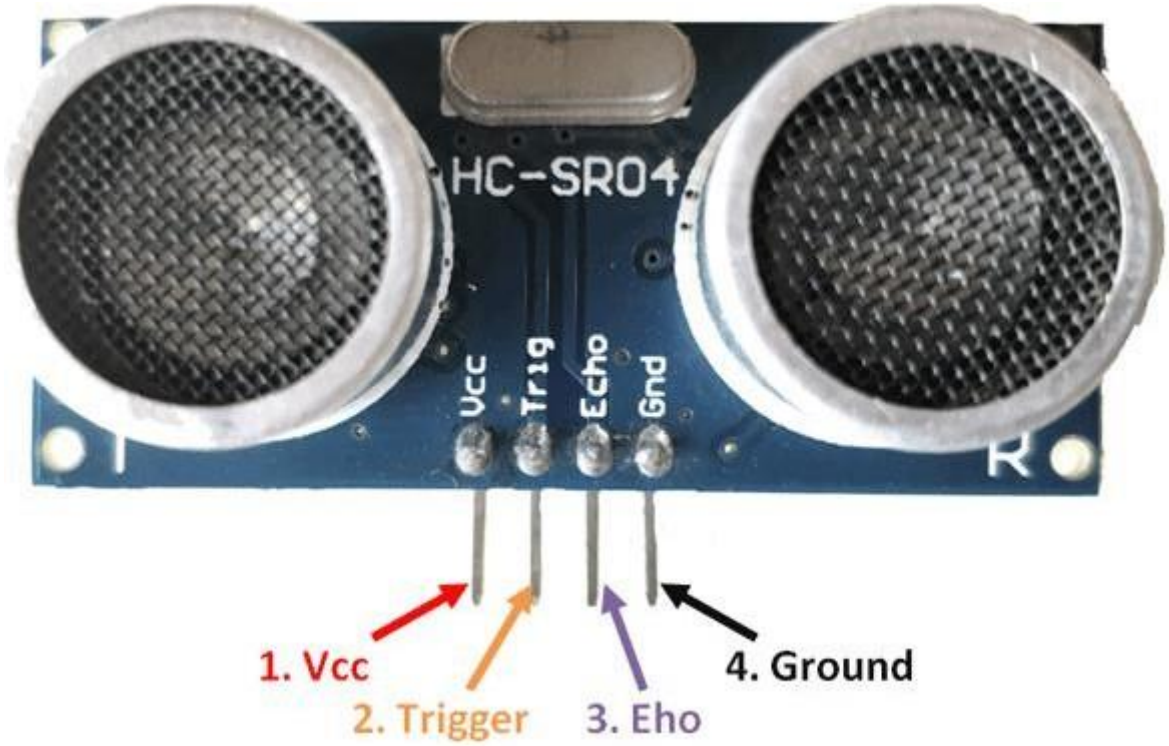


just a few of the sensors that are easily compatible with arduinoshields

additionally, there are these things called shields -- basically they are pre-built circuit boards that fit on top of your arduino and provide additional capabilities -- controlling motors, connecting to the internet, providing cellular or other wireless communication, controlling an lcd screen, and much more.

a partial selection of available shields to extend the power of your arduino

HC-SR04 Ultrasonic Sensor



ULTRASONIC SENSOR HC SR04

ULTRASONIC SENSOR HC SR04 PIN DIAGRAM

ULTRASONIC SENSOR PIN CONFIGURATION

PIN NUMBER	PIN NAME	DESCRIPTION
1	VCC	THE VCC PIN POWERS THE SENSOR, TYPICALLY WITH +5V
2	TRIGGER	TRIGGER PIN IS AN INPUT PIN. THIS PIN HAS TO BE KEPT HIGH FOR 10US TO INITIALIZE MEASUREMENT BY SENDING US WAVE.
3	ECHO	ECHO PIN IS AN OUTPUT PIN. THIS PIN GOES HIGH FOR A PERIOD OF TIME WHICH WILL BE EQUAL TO THE TIME TAKEN FOR THE US WAVE TO RETURN BACK TO THE SENSOR.
4	GROUND	THIS PIN IS CONNECTED TO THE GROUND OF THE SYSTEM.

HC-SR04 SENSOR FEATURES

- operating voltage: +5v
- theoretical measuring distance: 2cm to 450cm
- practical measuring distance: 2cm to 80cm
- accuracy: 3mm
- measuring angle covered: <15°
- operating current: <15ma
- operating frequency: 40hz

you can buy hc-sr04 ultrasonic sensor from here.

equivalent distance measuring sensors

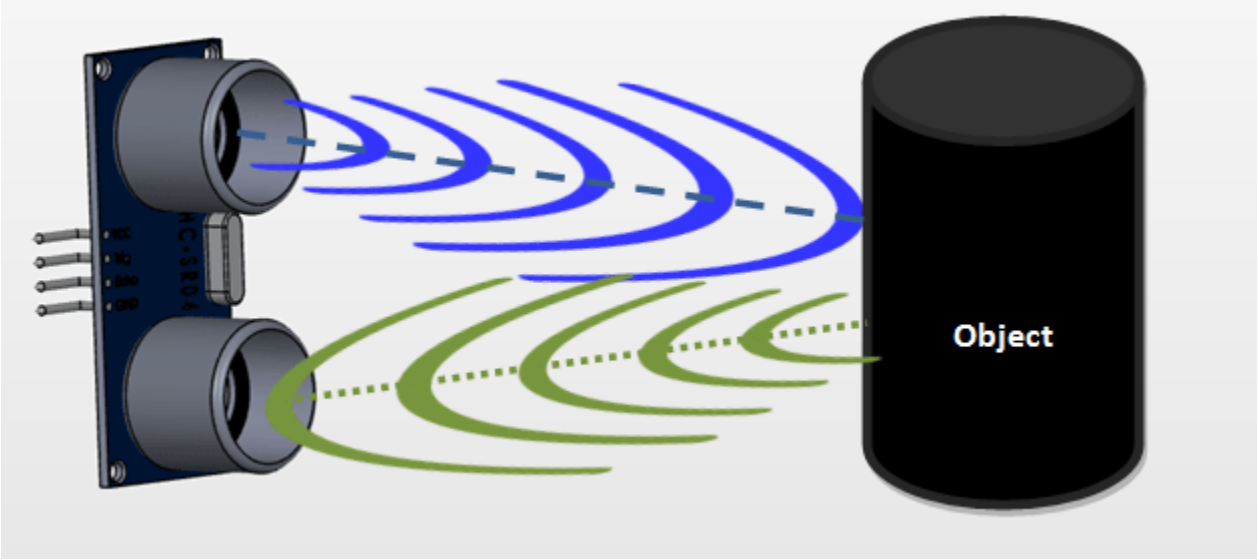
us transmitter receiver pair, ir sensor module, ir sensor pair, ir analog distance sensor,

hc-sr04 ultrasonic sensor - working

as shown above the **hc-sr04 ultrasonic (us) sensor** is a 4 pin module, whose pin names are vcc, trigger, echo and ground respectively. this sensor is a very popular sensor used in many applications where measuring distance or sensing objects are required. the module has two eyes like projects in the front which forms the ultrasonic transmitter and receiver. the sensor works with the simple high school formula that

distance = speed × time

the ultrasonic transmitter transmits an ultrasonic wave, this wave travels in air and when it gets objected by any material it gets reflected back toward the sensor this reflected wave is observed by the ultrasonic receiver module as shown in the picture below



now, to calculate the distance using the above formulae, we should know the speed and time. since we are using the ultrasonic wave we know the universal speed of us wave at room conditions which is 330m/s. the circuitry inbuilt on the module will calculate the time taken for the us wave to come back and turns on the echo pin high for that same particular amount of time, this way we can also know the time taken. now simply calculate the distance using a microcontroller or microprocessor.

how to use the hc-sr04 ultrasonic sensor

hc-sr04 distance sensor is commonly used with both microcontroller and microprocessor platforms like arduino, arm, pic, raspberry pie etc. the following guide is universally since it has to be followed irrespective of the type of computational device used.

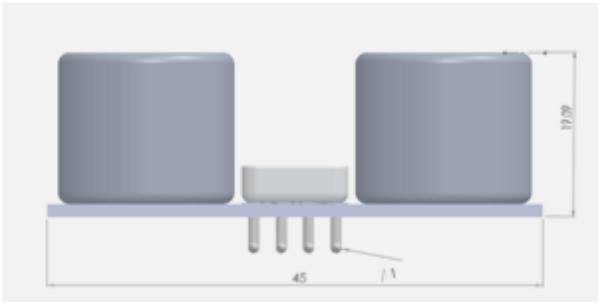
power the sensor using a regulated +5v through the vcc ad ground pins of the sensor. the current consumed by the sensor is less than 15ma and hence can be directly powered by the on board 5v pins (if available). the trigger and the echo pins are both i/o pins and hence they can be connected to i/o pins of the microcontroller. to start the measurement, the trigger pin has to be made high for 10us and then turned off. this action will trigger an ultrasonic wave at frequency of 40hz from the transmitter and the receiver will wait for the wave to return. once the wave is returned after it getting reflected by any object the echo pin goes high for a particular amount of time which will be equal to the time taken for the wave to return back to the sensor.

the amount of time during which the echo pin stays high is measured by the mcu/mpu as it gives the information about the time taken for the wave to return back to the sensor. using this information the distance is measured as explained in the above heading.

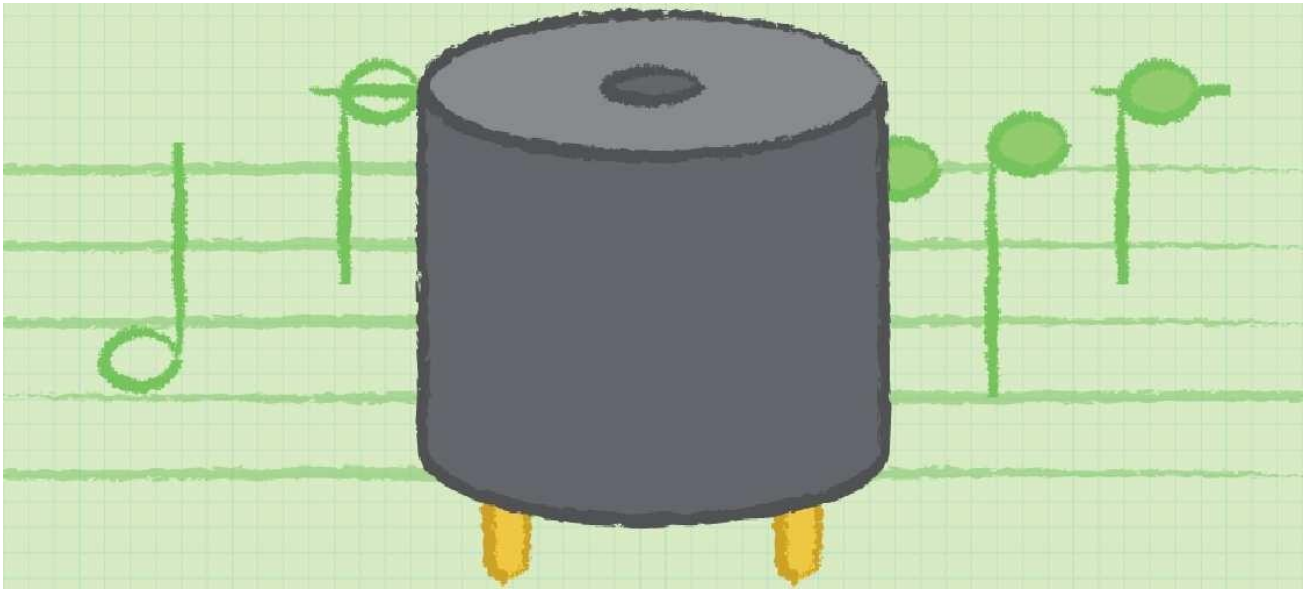
APPLICATIONS

- used to avoid and detect obstacles with robots like biped robot, obstacle avoider robot, path finding robot etc.
- used to measure the distance within a wide range of 2cm to 400cm
- can be used to map the objects surrounding the sensor by rotating it
- depth of certain places like wells, pits etc can be measured since the waves can penetrate through water

2D MODEL OF THE COMPONENT



B 10 BUZZER



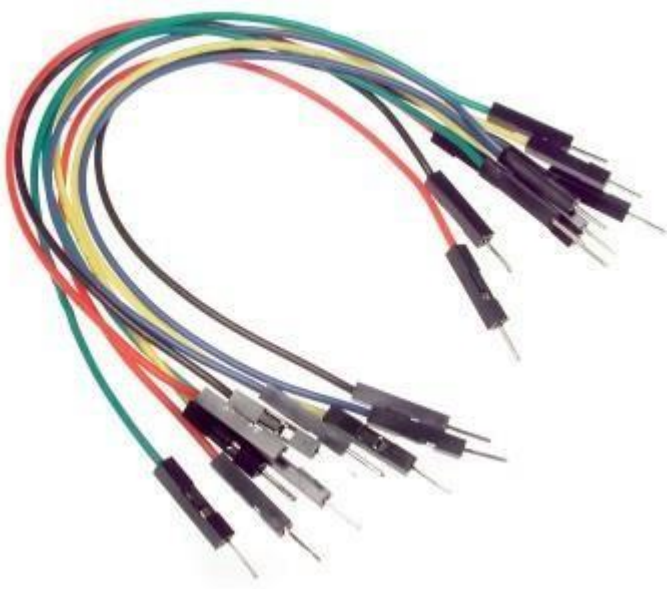
a small buzzer is a common feature in electronic products and can provide an effective way of interacting with users or raising an alarm.

depending on the type and strength of the signals available to drive the buzzer, the physical space available, and the required audio sound pressure level (spl), a magnetic or piezoelectric type will be the most common options for your application.

one also has the choice between an indicator or transducer design. indicators have built-in drive circuitry and are easy to design-in but can produce only a single, continuous tone or pulsed output. on the other hand, transducers can produce more complex sounds but need you to provide an excitation waveform and external components for switching and amplification.

this cui insights blog guides you through “buzzer basics”, including the different types of buzzers, their features, and associated design challenges, to help you choose the most suitable type for your application.

JUMPER WIRES



jumper wires are used for making connections between items on your breadboard and your arduino's header pins. use them to wire up all your circuits!

buy jumper wires from amazon, sparkfun, adafruit, or newark.

USB CABLE

IMAGE COURTESY OF SPARKFUN ELECTRONICS

this is a standard a-b usb cable. it can be used to connect your computer to arduinos that use a full-sized b-type usb connection, such as the uno and mega2560.

buy a usb cable from amazon, sparkfun, or adafruit.

9V BATTERY

IMAGE COURTESY OF SPARKFUN ELECTRONICS

use a 9v battery with your arduino projects to provide a higher-current power supply for things like motors. you can also feed 9v power into your arduino's dc barrel jack using 9v battery clip with a jack to get a regulated 5v from the internal regulator.

buy a 9v battery from amazon, sparkfun, adafruit, or newark.

9V BATTERY CLIP

IMAGE COURTESY OF SPARKFUN ELECTRONICS

use a 9v battery clip to easily connect your 9v battery to your arduino. use one with leads to connect your battery directly to a breadboard or the vin pin on the arduino. you can also buy a clip with a barrel jack for plugging directly into the arduino's barrel jack (shown above)

buy a 9v battery clip with a barrel jack connector from amazon, adafruit or sparkfun. buy a 9v battery clip with leads from newark.

RED LED

IMAGE COURTESY OF SPARKFUN ELECTRONICS

an average 5mm red led has a 2v forward voltage drop, and a forward current of 20ma. don't forget to use a current-limiting resistor when you connect an led to your arduino!

buy red leds from amazon, sparkfun, adafruit, or newark.

CODE

```
// DEFINES PINS NUMBERS
CONST INT TRIGPIN = 9;
CONST INT ECHOPIN = 10;
CONST INT BUZZER = 11;
CONST INT LEDPIN = 13;

// DEFINES VARIABLES
LONG DURATION;
INT DISTANCE;
INT SAFETYDISTANCE;


VOID SETUP() {
PINMODE(TRIGPIN, OUTPUT); // SETS THE TRIGPIN AS AN OUTPUT
PINMODE(ECHOPIN, INPUT); // SETS THE ECHOPIN AS AN INPUT
PINMODE(BUZZER, OUTPUT);
PINMODE(LEDPIN, OUTPUT);
SERIAL.BEGIN(9600); // STARTS THE SERIAL COMMUNICATION
}


VOID LOOP() {
// CLEARS THE TRIGPIN
DIGITALWRITE(TRIGPIN, LOW);
```

```
DELAYMICROSECONDS(2);


// SETS THE TRIGPIN ON HIGH STATE FOR 10 MICRO SECONDS

DIGITALWRITE(TRIGPIN, HIGH);

DELAYMICROSECONDS(10);

DIGITALWRITE(TRIGPIN, LOW);


// READS THE ECHOPIN, RETURNS THE SOUND WAVE TRAVEL TIME IN
MICROSECONDS

DURATION = PULSEIN(ECHOPIN, HIGH);


// CALCULATING THE DISTANCE

DISTANCE= DURATION*0.034/2;


SAFETYDISTANCE = DISTANCE;

IF (SAFETYDISTANCE <= 20){

DIGITALWRITE(BUZZER, HIGH);

DIGITALWRITE(LEDPIN, HIGH);

}

ELSE{

  DIGITALWRITE(BUZZER, LOW);

  DIGITALWRITE(LEDPIN, LOW);

}

}
```

CONNECTIONS

ULTRA SONIC SENSOR CONNECTION WITH AURDINO

1. Vcc.....5v pin
2. Trig.....9^{th} pin
3. ECHO-.....10^{th} pin
4. Gnd.....-gnd pin

BUZZER CONNECTION WITH AURDINO

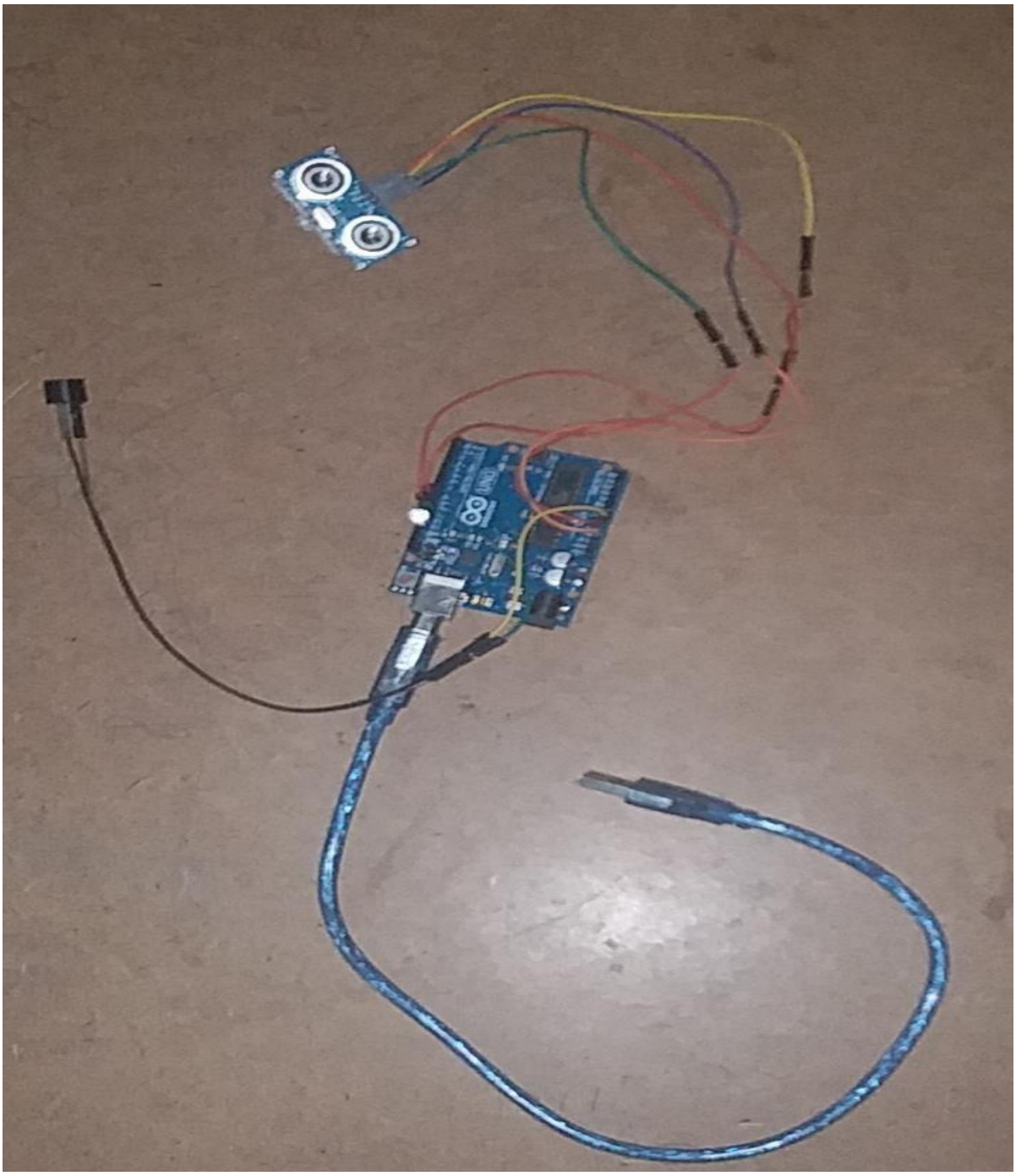
1. +ve leg(big leg)---------- 11^{th} pin
2. -ve leg (small leg)---------- gnd pin

LED CONNECTION WITH AURDINO

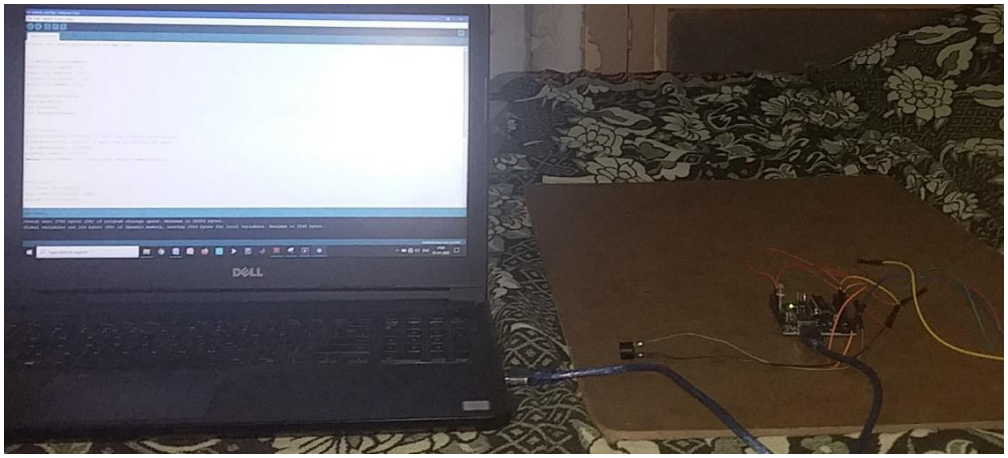
1. +ve leg(big leg)---------- 13^{th} pin
2. -ve leg (small leg)---------- gnd pin

OUTPUT

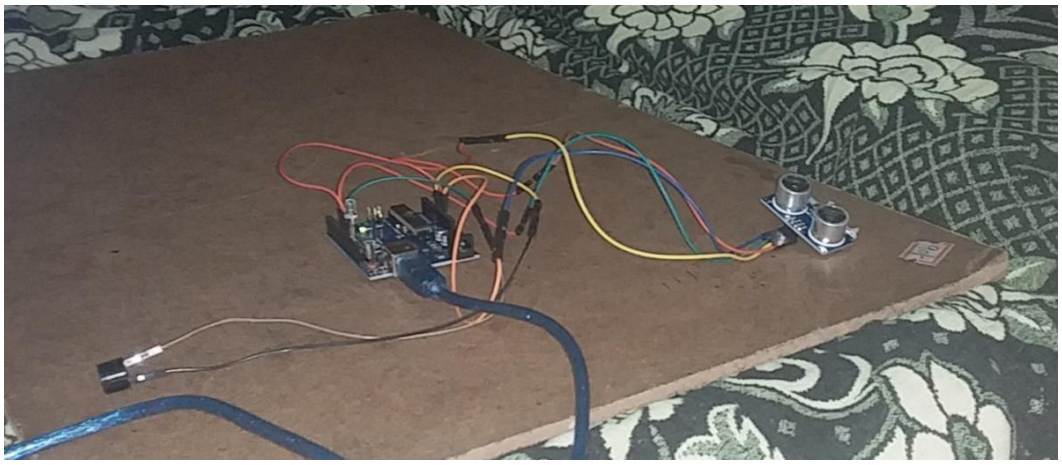
1. AFTER CONNECTIONS ARE COMPLETED



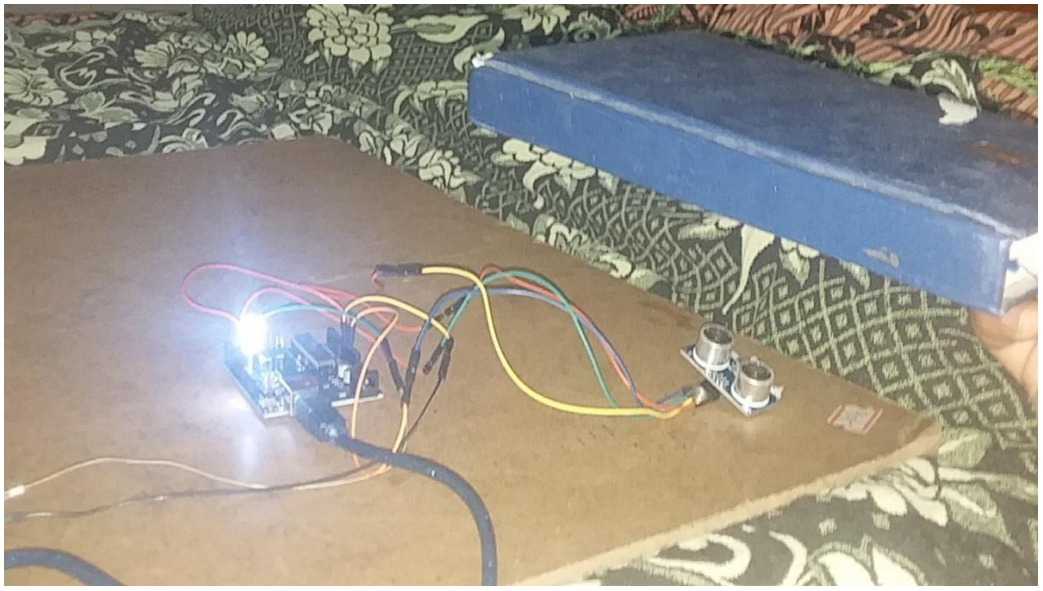
2. AFTER THE COMPLETED CONNECTIONS AND UPLOAD THE PROGRAME IN AURDINO BOARD USING AURDINO SOFTWARE



3. TEST-1:- WITH OUT ANY OBSTACLES IN FRONT OF ULTRA SONIC SENSOR



4. TEST-2:-WITH A OBSTACLES IN FRONT OF ULTRA SONIC SENSOR



AFTER CONNECTION TO THE PIPE .HENCE IT IS A SMART BLIND STICK USING AURDINO

CONCLUSION

the smart walking stick, constructed with at most accuracy, will help the blind people to move from one place to another without others help. this could also be considered a crude way of giving the blind a sense of vision. this stick reduces the dependency of visually impaired people on other family members, friends and guide dogs while walking around. the proposed combination of various working units makes a real-time system that monitors position of the user and provides dual. the smart stick detects objects or obstacles in front of users and feeds warning back, in the form of voice making rather than vibration. also the incorporation of automatic room equipment switching in the stick will be useful while they are indoor. the advantage of the system lies in the fact that it can prove to be a low cost solution to millions of blind person worldwide.