AIM OF THE PROJECT

To measure the speed of a moving object using ARDUINO UNO and IR Sensors.

REQUIREMENTS

- ARDUINO UNO board
- ARDUINO IDE
- Two IR Sensors
- Breadboard
- Object (Car)
- USB Cable
- Jump Wires

DESCRIPTION

- **ARDUINO UNO board**

Arduino is a microcontroller on a circuit board which makes it easy to receive inputs and drive outputs. Inputs may be temperature sensor, motion sensor, distance sensor and so forth.
Its uses are :

- Traffic Light Count Down Timer
- Parking Lot Counter
- Weighing Machines
- Emergency Light for Railway
- Security System
- CCTV Switchers

Jump Wires :

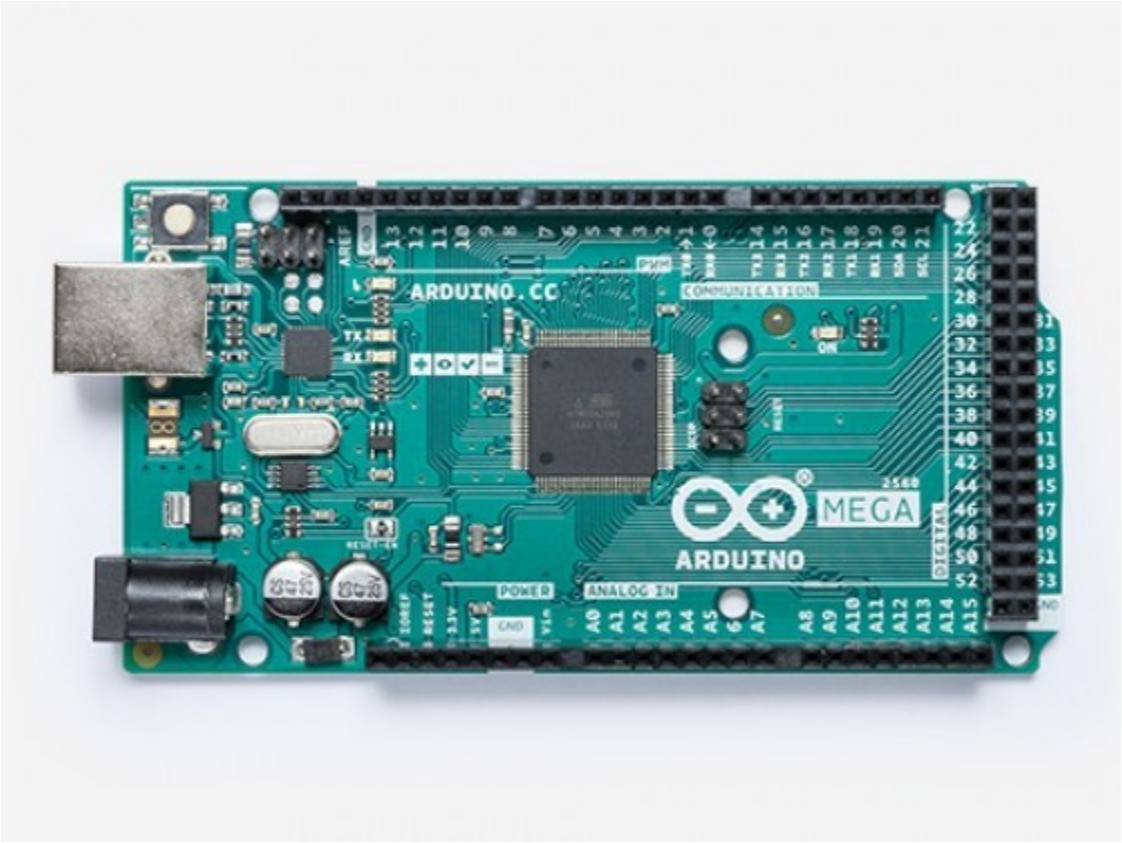
there are three types of Jump Wires used in this project. Those are :

- Male to Male
- Male to Female
- Female to Female

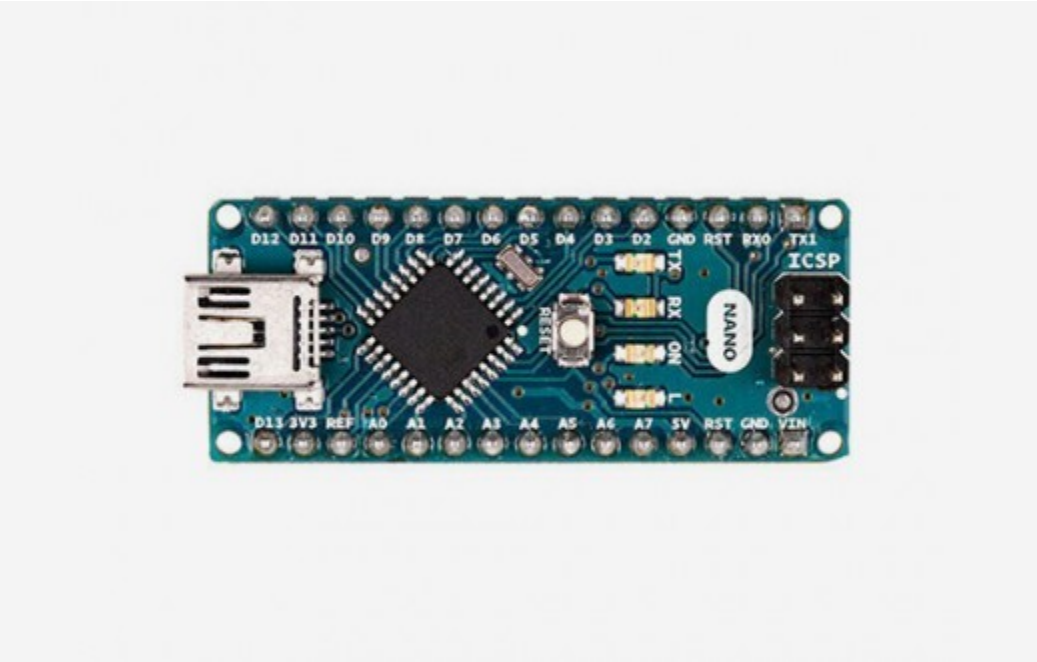
Type of Arduino

- Arduino Mega
- Arduino Uno
- Arduino Nano
- Arduino Pro
- Arduino 101

Arduino Mega



Arduino Nano

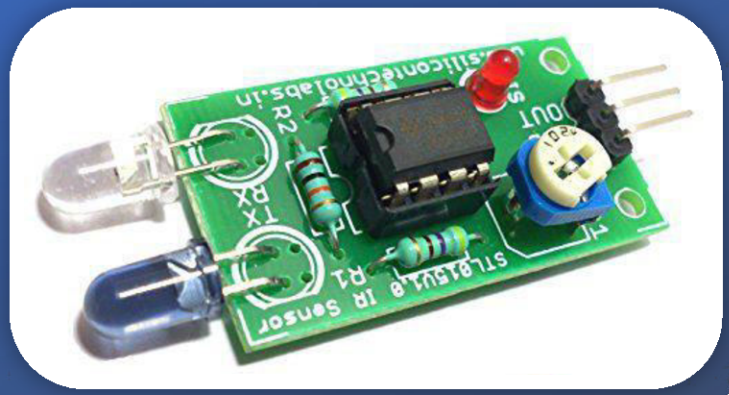


Why Arduino Uno ?

- Arduino uno is best for beginners
- It has enough features
- Easy to control compare to others

Some features of Arduino Uno

- It can also be powered by a 9V-12V AC to DC adapter.
- The Uno features 14 Digital I/O pins and 6 Analog I/O pins.
- The Uno features 14 Digital I/O pins and 6 Analog I/O pins.



- **ARDUINO IDE**

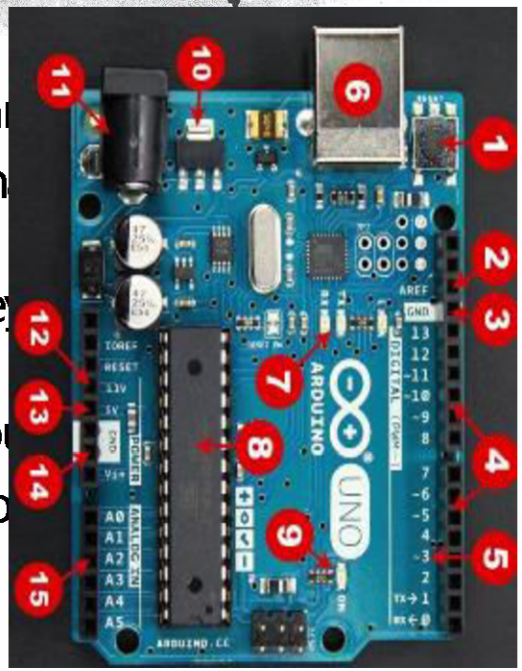
The open source Arduino Software(IDE 1.8.8) makes it easy to write code and upload it to the board

- **IR Sensors:**

The IR sensor includes an IR LED and a phototransistor. When an object passes between the sensors, light reflects from the object and falls on the phototransistor. An operational amplifier IC (LM358) is used and the phototransistor is connected to it. When object come in front of sensor, it sends a logical HIGH signal to Arduino

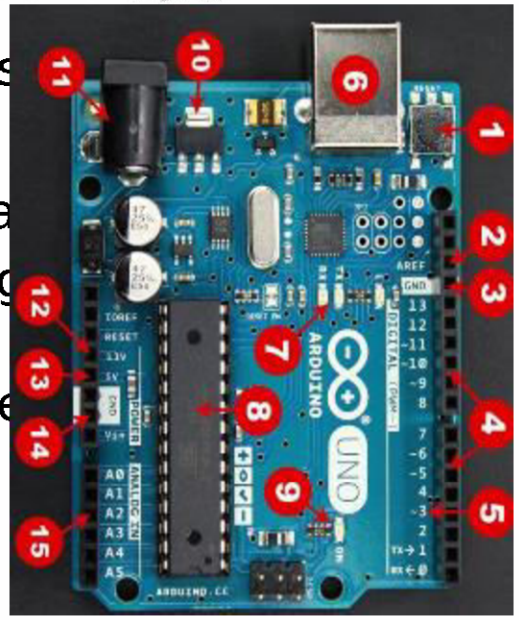
PIN DESCRIPTION OF ARDUINO UNO

- Reset Button: restart any code that is loaded to the Arduino boa
- AREF: stands for "Analog Reference" and is used to set an extern reference voltage
- Ground Pin: There are a few ground pins on the Arduino and the the same
- Digital Input/Output: Pins 0-13 can be used for digital input or o
- PWM: the pins marked with (~)symbol can simulate analog outp



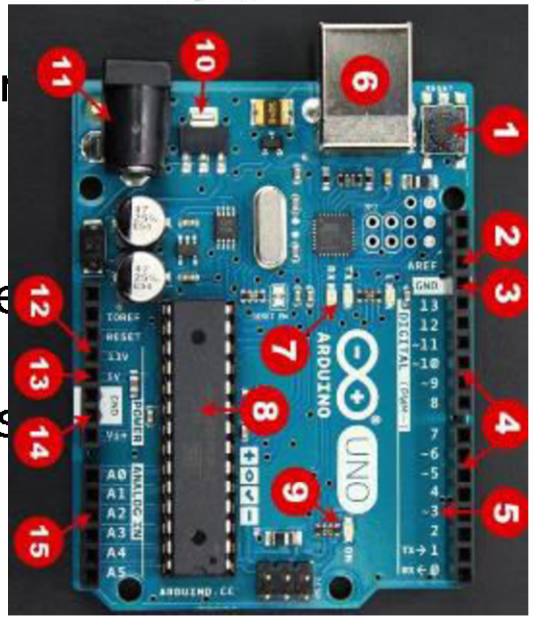
PIN DESCRIPTION OF ARDUINO UNO

- USB Connection: used for powering up Arduino and uploading s
- TX/RX: transmit and receive data indication LEDs
- ATmega Microcontroller: the brains and is where the programs a
- Power LED Indicator: this LED lights up anytime the board is plug power source
- Voltage Regulator: controls the amount of voltage going into the board

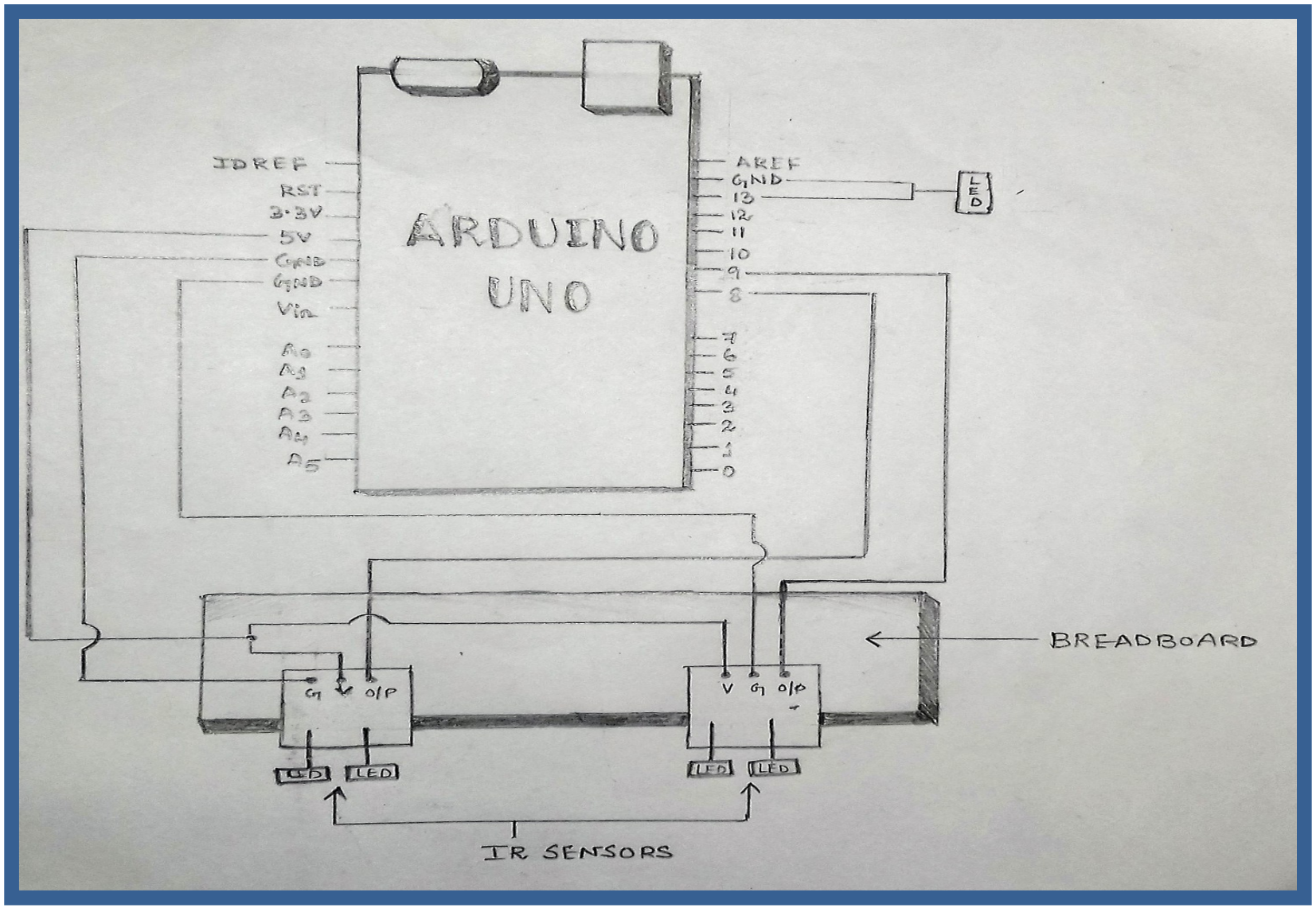


PIN DESCRIPTION OF ARDUINO UNO

- DC Power Barrel Jack: used for powering Arduino with a power
- 3.3V Pin: supplies 3.3 volts of power to projects
- 5V Pin: supplies 5 volts of power to projects
- Ground Pins: there are few ground pins on the Arduino and the
the same
- Analog Pins: these pins can read the signal from an analog sens
and convert it to digital



CIRCUIT DIAGRAM



CIRCUIT ANALYSIS

The physical implementation of the proposed technique requires an Arduino UNO board , two IR sensors , a breadboard and wires.

The two IR sensors are attached to the breadboard. The voltage terminals of the two IR sensors are connected together to the 5V voltage pin of the Arduino UNO board. The Arduino UNO board has two Ground pins. Each Ground pin is connected to the Ground terminal of each IR sensor. The output pin of one IR sensor is connected to the 8^{th} Digital Input/Output pin of the Arduino UNO board and the output pin of the other sensor is connected to the 9^{th} pin.

When a object moves in front of the first sensor, it gives the output HIGH signal to Arduino, Arduino detects the falling wave, now internal timer of Arduino is started and when car moves in front of second sensor timer is stopped.

Now Arduino measures the speed of car which is measured by distance time relationship

Speed = Distance / Time

Speed : Car's speed

Distance : Distance between sensors

Time : measured by Arduino

Programming in Arduino

Arduino is based on C/C++ language.

- Arduino language is merely a set of C/C++
- Arduino libraries files written in C/C++

Programming paradigm in Arduino

- Divided into three parts:
    1. Defining the variables

        #define led 13

        #define led1 9

    2. setup()

        pinMode(pin, mode)

        Eg: pinMode(13,output)

        pinMode(8,input)

Programming paradigm in Arduino

- **loop()**
  - Defining main program
  - Example:

```
void loop()
{
    digitalWrite(13,HIGH)
   delay(1000)
   digitalWrite(13,LOW)
  delay(1000)
}
```

Some important function for arduino

- digitalWrite()

  Used to write a high or low value to a digital pin

- digitalRead()

  Used to read a high or low value of digital pin

- millis()

  Returns the amount of millisecond that has elapsed since the program has started.

Source code of project

```
double calibrate=0;
float time_for_speed=0, velocity=0 ;
boolean active=0;


void setup()
{
  Serial.begin(9600);           // 9600 bits data information passing in a second
  pinMode(8, INPUT);          //    pin no. 8 is used as input pin
  pinMode(9, INPUT);          //pin no. 8  is used as input pin
  pinMode(13, OUTPUT);    // pin no. 13 is used as output pin for LED
}
```

```
void loop()                              // main part of program
{
   calibrate= millis();       /* function returns the time
   elapsed when
                                                  program starts*/
    if( digitalRead(8)==1)    //   chek when pin no. 8 is
   High
    {
        while(digitalRead(9)!=1);              // chek when
   pin no. 9 is high
        time_for_speed millis()-calibrate;  // Actual time
   b/w sensors
        while(digitalRead(9)!=0);              //   chek when
   pin no. 9 is low
        velocity= 756/time_for_speed;   //  velocity in
```

```
  if(active==1)                          // chek active condition
   {
      Serial.print("speed ");
      Serial.print(velocity);        // print the speed
      Serial.println(" km/h");
      active=0;                           // assign active again zero
      if(velocity > 1)                   // if velocity execed 1 km/h
      {
        digitalWrite(13,HIGH);    // Led goes to high on pin no.  13
        delay(2000);                      //wait for two second
        digitalWrite(13,LOW);    //  Led goes to off
    }
    }
 }
```

Calculation of speed in km/h

- We know time=time_for_speed;
  distance=21cm =0.21 m // fixed before

  speed=(distance)*1000/(time_for_speed) m/s
  =0.21*(1000)/(time_for_speed) m/s
  =210/(time_for_spped) m/s
  =756/(time_for_speed) km/h

OUTPUT

```
OUTPUT OF SPEED OF OBJECT::

speed   0.03 km/h
speed   1.81 km/h
speed   4.13 km/h
speed   0.42 km/h
speed   3.04 km/h
speed   3.28 km/h
speed   5.28 km/h
speed   1.47 km/h
speed   4.00 km/h
speed   0.74 km/h
speed   0.80 km/h
speed   2.16 km/h
speed   2.73 km/h
```

APPLICATIONS

- Traffic police applies this speed measurement technique to detect the vehicles that violate the speed limit regulation.

LIMITATION OF THE PROPOSED TECHNIQUE

- IR sensors are not perfect, so sometimes it may consider the reflection of the object as the object itself.
- The two sensors used here can detect only one moving object at a time.
- Performance degrades if the moving object is at a longer distance from the system.