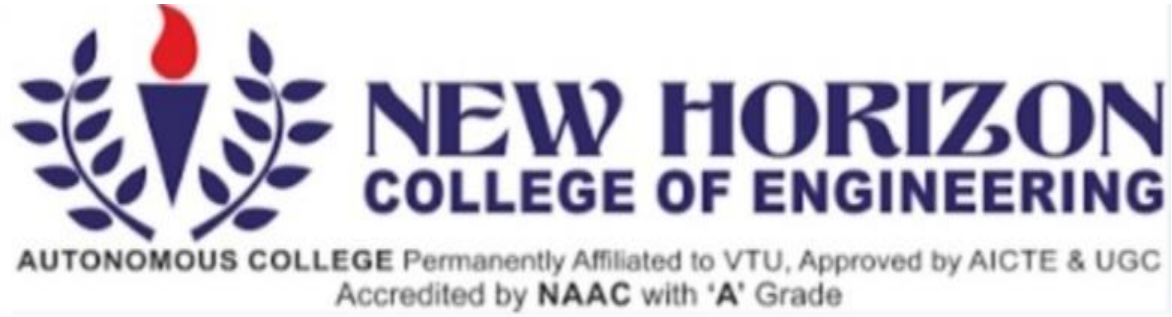


BOOLOEAN ALGEBRA CALCULATOR

A MINI PROJECT REPORT

Submitted by

M.JAHNAVI (1NH18EC729)

SAHANA.Y (1NH18EE051)

SUBHASHINI.S (1NH18EC750)

SWATHI.A (1NH18EC752)

In partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION

BOOLOEAN ALGEBRA CALCULATOR

A MINI PROJECT REPORT

Submitted by

M.JAHNAVI (1NH18EC729)

In partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION

BOOLOEAN ALGEBRA CALCULATOR

A MINI PROJECT REPORT

Submitted by

SAHANA .Y (1NH18EE051)

In partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION

BOOLOEAN ALGEBRA CALCULATOR

A MINI PROJECT REPORT

Submitted by

SUBHASHINI.S (1NH18EC750)

In partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION

BOOLOEAN ALGEBRA CALCULATOR

A MINI PROJECT REPORT

Submitted by

SWATHI.A (1NH18EC752)

In partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION

NEW HORIZON COLLEGE OF ENGINEERING

DEPARTMENT OF COMMUNICATION ELECTRONICS AND ENGINEERING

CERTIFICATE

Certified that the mini project work entitled "BOOLEAN ALGEBRA CALCULATOR'' carried out by M.JAHNAVI(1NH18EC729), SAHANA.Y(1NH18EE051), SUBHASHINI.S(1NH18EC750), SWATHI.A(1NH18EC752) bonafide students of Electronics and Communication Department , New Horizon College of Engineering, Bangalore.

The mini project report has been approved as it satisfies the academic requirements in report of mini project work prescribed for the said degree.

Project guide HOD ECE

MS.RAJANI Dr. SANJEEV SHARMA

EXTERNAL VIVA

Name of the Examiner Signature with Date

1.

2.

ACKNOWLEDGEMENT

The satisfaction that accompany that successful completion of any task would be, but impossible without the mention of the people who made it possible, whose constant guidance and encouragement helps us to succeed.

We thank **Dr. Mohan Manghnani,** chairman of **New Horizon college of Institution,** providing necessary infrastructure and creating good environment

We also record here the constant encouragement and facilities extended to us by **Dr.Manjunatha,** Principal, NHCE and **Dr. Sanjeev Sharma,** head of the department of Electronics and Communication Engineering. We extend sincere gratitude to them. We sincerely acknowledge the encouragement, timely help and guidance to us by our beloved guide **Ms.Rajani** to complete the project within stipulated time successfully.

Finally, a note of thanks to the teaching and non-teaching staff of Electronics and Communication Department for their co-operation extend to us, who helped us directly or indirectly in this successful completion of mini project

M.JAHNAVI - 1NH18EC729

SAHANA.Y - 1NH18EE051

SUBHASHINI.S - 1NH18EC750

SWATHI.A - 1NH18EC752

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Logic gates are the building blocks in the logic design. The fundamental gates like AND, OR, NAND, NOR, NOT are used in the sequence and appropriate manner in developing sequential and combinational circuits for complex problems. It is a strong learning aid that the understanding of functionality of the basic gates through simulation using LED and switches controlling them by the Arduino microcontroller. The first step is to propose the concept of demonstration of functionality of the basic logic gates. It is making use of the latest version of art technologies to present in a easy and lucid manner to the learner. Arduino controller with its control of inputs/outputs through program and provides strong capability and also expansion.

It is a challenge to a designer to be provide with the extensive electronic gadgets in this digital world to find the suitable substitute of the failed electronic integrated circuit (Logical IC). It is a challenge and threat that the components are not available. To reproduce the functionality of failed components(IC) using circuit diagrams and experience, then replacement of such component can also be worked out using flexible components like Arduino microcontroller. The similar activity of replacing such basic logic gate integrated circuits(IC's) by programmed arduino in the place with suitable retrofit is focused by the present work.

CHAPTER 2

LITERATURE SURVEY

ARDUINO HISTORY:

ARDUINO is the open source software company, project and also user community that designs and manufactures the single-board microcontroller kits for building digital devices. Its products are license under the GNU Lesser General Public License (LGPL) or the General Public License (GPL).

In electronics, logic gate is the idealized or a physical device implementing a Boolean function it also performs a logical operation on one or two binary inputs and produces a single binary output. Logic gates are primarily implemented using transistors or diodes acting as electronic switches that it can also be constructed using the vacuum tubes.

The binary number system was refined by the **Gottfried Wilhelm Leibniz** (published in 1705), influenced by ancient I Ching's binary system. Leibniz established that, and by using the binary system, the principle of the arithmetic and logic could be combined.

In 1886, **Charles sanders peirce** described how the logic operations can be carried out by the electrical switching circuits.

It was in the year 2005, that the first ever Arduino board was born in classrooms of interactive Design institute in lvrea, Italy. Ardinuo is the Open source microcontroller based on the development board that has to be opened the doors of the electronics to a number of designers and also creative engineers.

An Arduino can be also connected to all kinds of lights, motors, sensors and other devices Today, there are many Arduino-based LED cubes Twitter displays, DNA analysis kits, Breathalyser.

Keil was found in 1982 by the **GUNTER** and **REIN HARD KEIL** initially as a german gbr. In April 1985 the company converted to the keil Elektronik GmbH to the market add on products for the Tools and their development provided by the silicon vendors.

CHAPTER 3

PROPOSED METHODOLOGY

PRINCIPLE:

Simulation of the digital logic provides a viable technique for development and diagnosis of the digital systems. These models are discussed with a summary of structure and timing techniques. A methodology for functional stimulation is a conjunction with gate level stimulation has been discussed. Errors in design detected at the functional level are categorized.

IMPLEMENTATION OF LOGIC GATES USING ARDUINO

Arduino Uno is based on the ATMEGA328 microcontroller and it is also a micro controller. This microcontroller is intelligent and low cost. It is flexible and can develop small circuits using electronic components. The arduino via the digital and analog input/output pins should be connected to hard ware components and controlled through programs known as sketches. These sketches should be transferred to the microcontroller and it will be enable by

integrated development environment to the programmer to transfer.

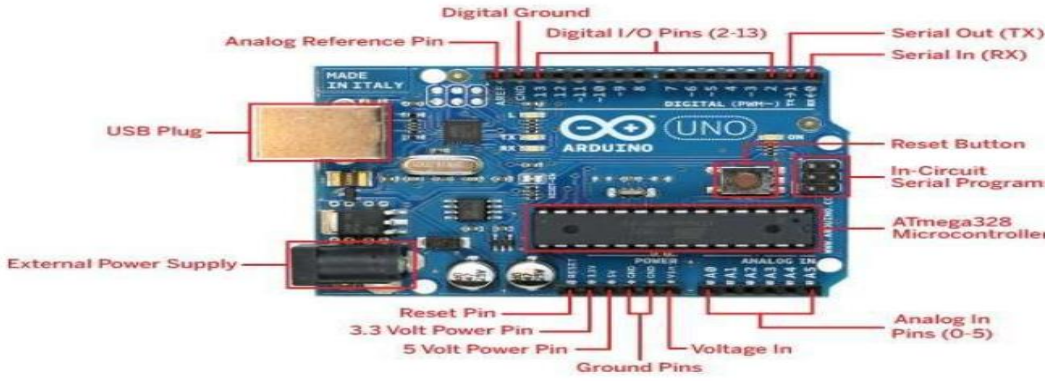


FIG3.1

HARD WARE SETUP FOR LOGIC GATE SIMULATION USING ARDUINO

The basic logic gates like OR, NAND, AND, XOR, NOR, NOT are stimulated using suitable interface and modification as required. Proper configuration set up by arduino programming can be referred by using basic truth tables and is done as required.

LOGIC GATES AND TRUTH TABLES

The logic gates and Boolean algebra are very important for logic design and development of various digital circuits. Table 1 given below shows the common logic gates and their Boolean expression governing the output of gate. Similarly, table 2 represents the logic truth table for each of such other logic gates. Truth tables are made to verify the circuit. Logic gates are the building blocks of every digital system.

Symbols of the logic gates are:

NOT GATE

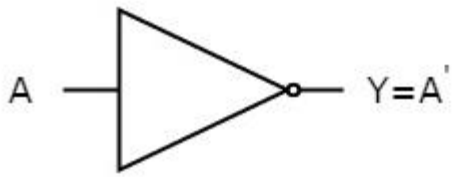


FIG3.2

It is a digital gate that it is inversion of its input. It is also called as inverter.

Its pin number is 7404

AND GATE

Fig3.3

It is a basic logic gate it implements the function of conjunction.

Its pin number is 7408

OR GATE.

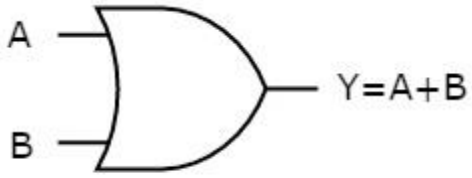


Fig3.4

It is a digital logic gate that implements the function of disjunction.

Its pin number is 7432

NAND GATE

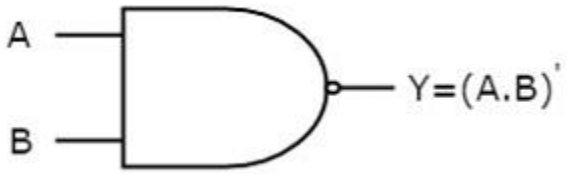


Fig 3.5

It is a digital circuit which is a inversion of logical AND. It is also universal gate.

Its pin number is 7400

NOR GATE

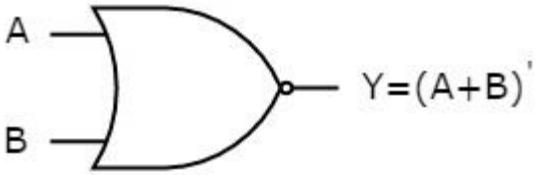


Fig3.6

It is a digital circuit which is inversion of logical OR. It is universal gate.

Its pin number is 7402.

EX-OR GATE

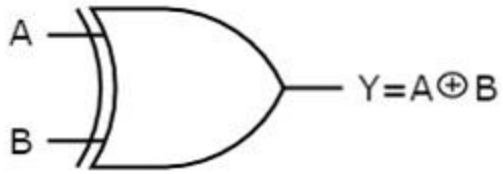


Fig 3.7

It is a special gate. Ex-OR gate operation is similar to that of OR gate, except for few combinations of inputs. Ex-OR gate is also called as an odd function.

Its pin number is 7486

EX-NOR GATE

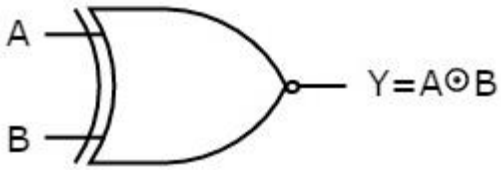


Fig 3.8

It is a special gate. EX_NOR gate operation is similar to that of NOR gate, except for few combinations of inputs. EX_NOR is also called even function.

Table3.1: Boolean expression for basic gates

LOGIC FUNCTION	BOOLEAN NOTATION
OR	A+B
AND	A.B
NOT	$\bar{A}$
NAND	$\overline{A.B}$
NOR	$\overline{A+B}$
EX-OR	(A.$\bar{B}$)+(B.$\bar{A}$)
EX-NOR	($\bar{A}$)$\bar{B}$

TABLE 3.2: Comparative truth table of Basic gates

INPUTS		Truth table outputs for each gate					
A	B	OR	AND	NAND	NOR	EX-OR	EX-NOR
0	0	0	0	1	1	0	1
0	1	1	0	1	0	1	0
1	0	1	0	1	0	1	0
1	1	1	1	0	0	0	1

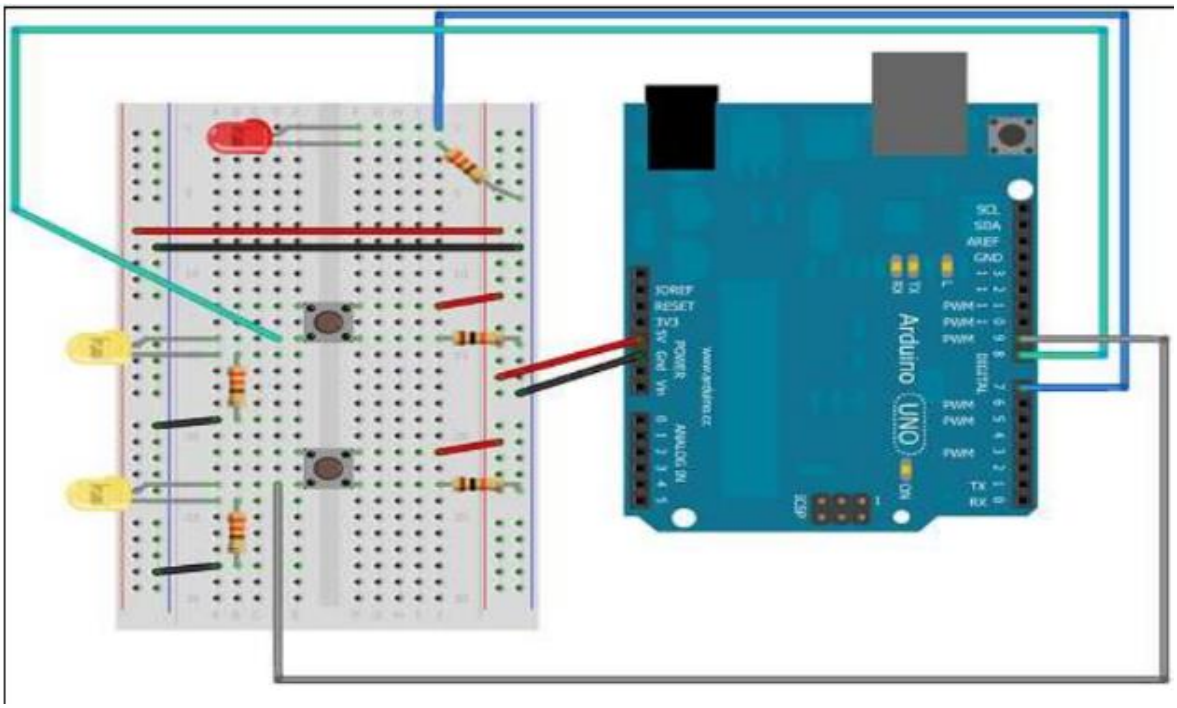
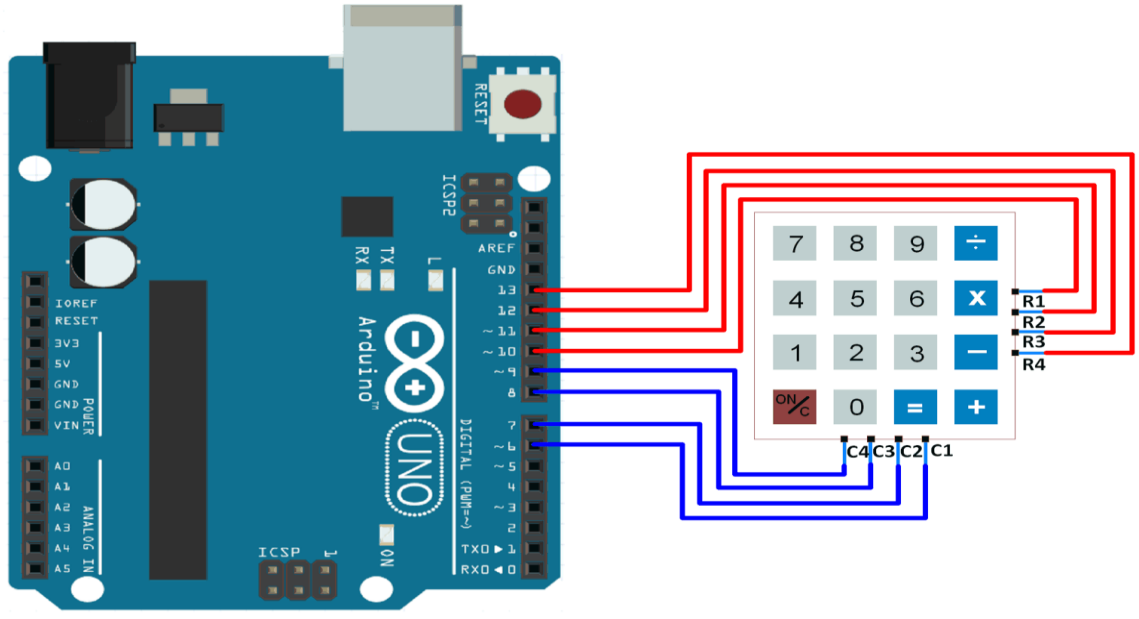


FIG3.9

FIG 3.10



CHAPTER 4

PROJECT DESCRIPTION

HARD WARE COMPONENTS:

- LED
- BREADBOARD
- JUMPER WIRES
- 330ΩRESISTOR
- AURDUINO UNO MICROCONTROLLER
- SWITCHES
- KEYPAD

LED-

Light emitting diode (LED) is a two-lead semiconductor light source and it resembles a basic p-n junction diode, except that an led and also emits lights when an LED's anode leads the voltage that has more positive than cathode lead by at least the LED's forward voltage drop, current flows. Electrons are able to recombine with holes within the device, releases energy in the form of photons. This effect is called electroluminescence, and the color of the light is determined by the energy band gap of the semiconductor.

Light emitting diode have many types but we are using 2 Led's for input through switches and one output led using ardinuo microcontroller. We use binary code to process the board.

Fig4.1

BREADBOARD-

Breadboard is a construction base for prototyping of electrons. Breadboard will be having many tiny sockets arranged on a 0.1 grid. Most electronic components in electronic circuits can be interconnected by inserting their leads or terminals into the holes and then making connections through wires with respect to circuit.

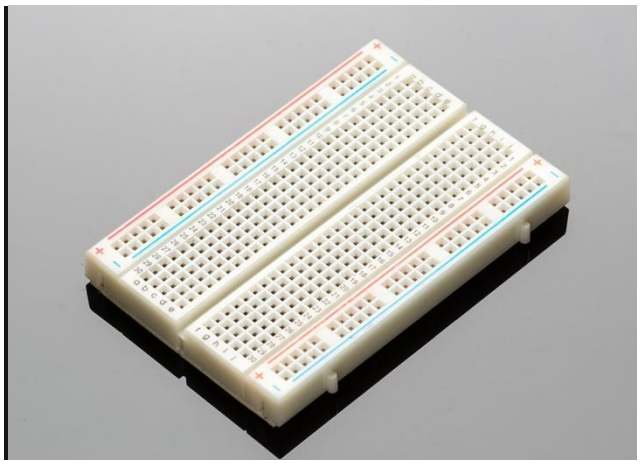


Fig4.2

JUMPER WIRES-

Jumper wires are wires that have connector pins at each end, that allows connect two points to each other so that there is no need of soldering. Jumper wires are typically used with breadboards and other prototyping tools in order to make it easy to change the circuit we need.

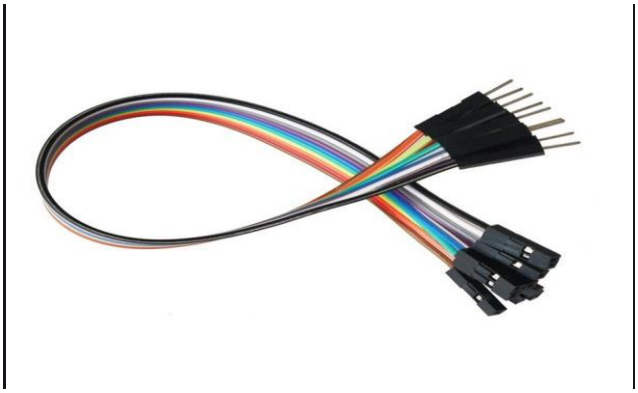


Fig4.3

RESISTOR:

Resistor is an electrical component that reduces the electric current. The ability to reduce current by resistors is called resistance. It is measured in units of ohms.

Fig4.4

ARDUINO UNO MICROCONTROLLER:

The Arduino UNO is an open source microcontroller and it is based on the microchip ATmega328p microcontroller and developed by Arduino.cc. The board is equipped with sets of digital and analog input/output pins that may be inferred to various expansion boards and other circuits. It has 14 digital input/output pins (of which 6 can used as PWM outputs), 6 analog inputs, 16MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a rest button. Arduino boards are read inputs and an open source electronics platform based on hardware and software.

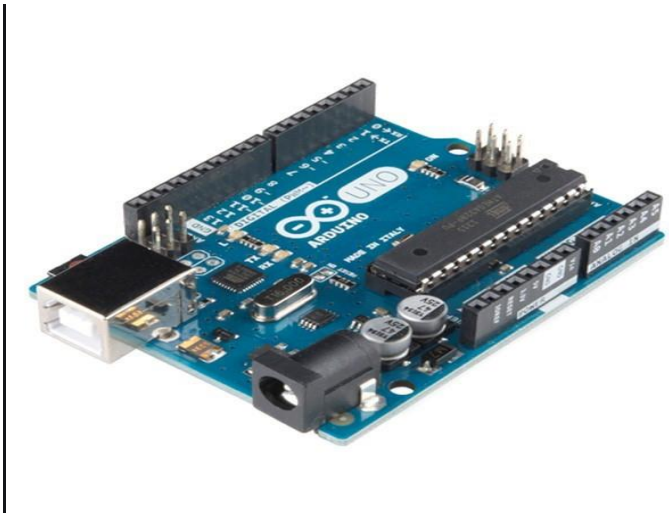


Fig4.5

SWITCHES:

A SWITCH is networking hardware component and is used in wired network to connect to other devices using Ethernet cables. The SWITCH allows each connected device to talk to others. Devices such as wireless routers and adapters communicate directly with one another so that switches are not used in this case.

Fig4.6

Fig 4.7

SOFTWARE DESCRIPTION:

This software is an integrated development environment (IDE), The embedded 'C' is most popular programming language in software field for developing electronic gadgets.

The keil 8051 development tools were designed for facing embedded software developers and for solving complex problems. The arduino language is merely c/c++ functions that can form a code. As it is special software running on the system that allows to program arduino. It is based on a very simple hardware language called processing and it is similar to 'c'language.

Arduino is a open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs – light on a sensor, a finger on a button, or a twitter message- turn it into an output – activating motor, turning on a LED, publishing

something online .

The industry standard Keil C compilers, macro assemblers, Debuggers, Real time kernels, single board computers and emulators support all 8051 derivatives.

CHAPTER 5

RESULTS AND DISCUSSION

VERIFICATION OF GATES USING ARDUINO:

Realization of 2 input OR and AND gate is done by configuring pins of the Arduino properly. The digital pin 8 and 9 are configured as input and pin7 as output pin. The digital pin 8 and 9 are configured as input and pin 7 as output pin. The general purpose of breadboard having requisite resistors, tactile switches and LED's is interfaced for the work. The sketch /program is written and transferred from the laptop or a computer to the arduino controller after understanding the functionality of OR and AND gates. Once the sketch is successfully transferred the user can be able to verify the working of arduino controller as OR and AND gate respectively according to the truth table. The configuration of switch 1 is input A and the configuration of switch 2 is input B. the output Q is configured as pin7 on the arduino. the clip of sketch is given below. The code can be easily put in the header by suitable modification.

Arduino code for OR gate:

```
//Simple sketch to verify OR gate

//functionality

//define input and output pins

int Out=7;

int inA=8;

int inB=9;
```

```
void setup()
{
pinMode(inA,INPUT);
pinMode(inB,INPUT);
pinMode(Out,OUTPUT);
}
Void loop()
{
boolean inAState=digitalRead(inA);
boolean inBState=digitalRead(inB);
boolean OutState;
pinOutState=inAstate|inBstate;
digitalWrite(Out,OutState);
}
```

Arduino code for AND gate

```
//Simple sketch to verify AND Gate
//functionality
```

```
//Define input and output pins
int Out=7;
int inA=8;
int inB=9;
void setup()
{
pinMode(inA,INPUT);
pinMode(inB,INPUT);
pinMode(Out,OUTPUT);
}
void loop()
{
boolean inAState=digitalRead(inA);
boolean inBstate=digitalRead(inB);
boolean Outstate;
pinOutSate=inAState&inBState;
digitalWrite(Out,OutState);
}
```

MODIFICATION FOR OTHER GATES

```
// Simple modification required in sketch to

//verify other gates//modification for NOT gate

OutState=!inAState;

//modification for XOR gate

OutState=inAState^inBState;

//modification for NAND gate

OutState=!(inAstate&inBState);

//modification for NOR gate

OutState=!(inAState|inBState);
```

The hardware verification is done by using switches and LED's on the breadboard. The setting of each switch based on each combination of the truth table is worked out and respective output is viewed on the output LED. The states of switches together with LED justify the gate under test giving the practical based approach for learning and implementation using programmable arduino. This kind of work out enhances the ability of the user to design and develop innovatively. The visualized outputs with logic combinations for OR gate and AND gate is shown below.

VERIFICATION OF OR GATE

Table 5.1

Hardware switch and LED STATUS			**Truth table Boolean Expression Q=A+B**		
SW1(pin8)	SW2(pin9)	LED(pin7)	A	B	Q
OFF	OFF	OFF	0	0	1
OFF	ON	ON	0	1	1
ON	OFF	ON	1	0	1
ON	ON	ON	1	1	1
ON=Logic high, OFF=Logic low			Read as A OR B gives Q		

VERIFICATION OF AND GATE

Table5.2

Hardware switch and LED STATUS			**Truth table Boolean Expression Q=A.B**		
SW1(pin8)	SW2(pin9)	LED(pin7)	A	B	Q
OFF	OFF	OFF	0	0	0
OFF	ON	OFF	0	1	0
ON	OFF	OFF	1	0	0
ON	ON	ON	1	1	1
ON=Logic high, OFF=Logic low			Read as A AND B gives Q		

VERIFICATION OF NAND GATE Table5.3

Hardware switch and LED status			**Truth table BooleanExpression Q=** $\overline{A.B}$		
Switch(pin8)	Switch(pin9)	Switch(pin7)	A	B	Q
OFF	**OFF**	**OFF**	**0**	**0**	1
OFF	**ON**	**OFF**	**0**	1	1
ON	**OFF**	**OFF**	1	**0**	1
ON	**O N**	**ON**	1	1	0
ON=LOGIC HIGH,OFF=LOGIC LOW			**Read as A NAND B gives Q**		

VERIFICATION OF NOR GATE

Table5.4

Hardware switch and LED STATUS			**Truth table Boolean Expression Q=** $\overline{A+B}$		
SW1(pin8)	SW2(pin9)	LED(pin7)	A	B	Q
OFF	OFF	OFF	0	0	0
OFF	ON	OFF	0	1	0
ON	OFF	OFF	1	0	0
ON	ON	ON	1	1	1
ON=Logic high, OFF=Logic low			Read as A NOR B gives Q		

VERIFICATION OF EX-OR GATE

Table5.5

Hardware switch and LED STATUS			**Truth table Boolean Expression Q=(A.$\overline{B}$)+(B.$\overline{A}$)**		
SW1(pin8)	SW2(pin9)	LED(pin7)	A	B	Q
OFF	OFF	OFF	0	0	0
OFF	ON	OFF	0	1	1
ON	OFF	OFF	1	0	1
ON	ON	ON	1	1	0
ON=Logic high, OFF=Logic low			Read as A EX-OR B gives Q		

VERIFICATION OF EX-NOR (Table5.6)

Hardware switch and LED STATUS			**Truth Table Boolean Expression Q=($\overline{A}$)$\overline{B}$**		
SW1(pin8)	SW2(pin9)	LED(pin7)	A	B	Q
OFF	OFF	OFF	0	0	1
OFF	ON	OFF	0	1	0
ON	OFF	OFF	1	0	0
ON	ON	ON	1	1	1
ON=Logic high, OFF=Logic low			Read as A EX-NOR B gives Q		

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

CONCLUSION:

The application and use of digital electronics in current context is very great because of the ease and the associative efficiency. The complex circuits often cause difficulty in replacement at applicable cost and repair. The associated difficulty of unavailabilty of specific replacement components leads to retrofitting at the best way and the effort is made to demonstrate the working of logic gates and Boolean algebra using practical based approach using arduino micro controller with associated hardware like LED's and switches. The proposed method is an innovatine teaching & learning technique as well as having advantages like clarity in reduction and understanding in presentation time . The theoretical explanation and added activity found very effective.

FUTURE SCOPE:

As a scope for the future work combinational and sequential circuit realization using the Arduino for the learning and effective teaching can be worked out. Then the exploration of the capability of ardinuo controller in the retrofitting works can be evidenced. The proposed method is an innovative teaching learning techniques. Future work related to building of the proposed reversible logic by using technologies such as CMOS, in a particular adiabatic CMOS, optical , thermodynamic technology.

REFERENCES:

1. Black, Michael Export to Arduino: A tool to teach processor design on real hardware. In Computer Science Faculty Publications. 2016, Paper 12.
2. R. Hari Sudhan, M. Ganesh Kumar, A. Udhaya Prakash, et al. Arduino atmega-328 microcontroller. International journal of Innovative Research in Electrical, Electronics, Instrumentation and control Engineering. 2015; 3(4).
3. Myo Su Su Theint, Zaw Myo Tun. Development of digital logic trainer for student's experimentation. International journal of scientific Engineering and Technology Research. 2014; 03(10): 1912-1916p.
4. Janaki Venukumar, Naveen. S.Ardinuo based door access control. International Journal of Research in Advent Technology. 2016; 4(8).

APPENDIX

ARDUINO CODE:

```
#include <keypad.h>

const byte ROWS = 4; //four rows

const byte COLS = 4; //four columns

char keys [ROWS] [COLS] = {

    {'0', '1', '2', '3'},

    {'4', '5', '6', '7'},

    {'8', '9', 'A', 'B'},

    {'C', 'D', 'E', 'F'},

};

byte rowPins[ROWS] = {5, 4, 3, 2}; //connect to the row pinouts of the keypad

byte colPins[COLS] =  {12, 11,10, 6}; //connect to the column pinouts of the keypad

Keypad keypad = Keypad( makeKeymap(keys), rowpins, colpins, ROWS, COLS );

int Out = 7;

int inA = 8;

int inB = 9;
```

```
int key;
int pinoutstate;
void setup ( ) {
    Serial.begin(9600) ;
pinMode(inA, INPUT) ;
pinMode(inB, INPUT) ;
pinMode(out, OUTPUT);
}
void loop ( ) {
   char key = keypad.getkey( ) ;
boolean inAState = digitalRead(inA) ;
boolean inBState = digitalRead(inB) ;
boolean outState = digitalRead(out);
if (key) {
     Serial.println(key) ;
}
```

```
delay(10);
//boolean inAState = digitalRead(inA) ;
//boolean inBState = digitalRead(inB) ;
//boolean outState = digitalRead(out) ;
if (key>3)
27{switch (key)
{
    case 0 :
   pinoutstate=inAState\inBState;
                digitalWrite (Out,pinoutstate);
                break;
    case 1:
    pinoutstate=inAState&inBState;
                 digitalWrite(Out,pinoutstate);
                 break;
   case 2 :      pinoutstate=!inAState;
                digitalWrite(OutState,pinoutstate);
                 break;
```

```
 case 3:        pinoutstate=inAState^inBState;
                digitalWrite(Outstate,pinoutstate);
                 break;
    case 4:       pinoutstate=! (inAState|inBState);
                  digitalWrite(OutState,pinoutstate);
  break;
   case 5:       pinoutstate=! (inAState&inBState);
                 digitalWrite(Out,pinoutstate);
                 break;
     case 6:      pinoutstate=! (inAState^inBState);
                 digitalWrite(OutState,pinoutstate);
                 break;
    default:     digitalWrite(13,LOW);

}
}
delay(2000);
}
```