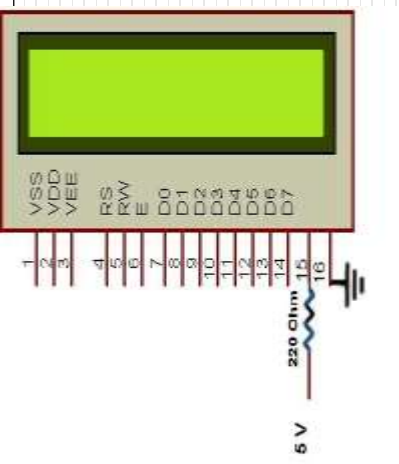
IoT

Arduino-LCD-Keypad Interfacing

Dr. Sarwan Singh
Deputy Director
NIELIT Chandigarh



Agenda

- LCD- Introduction
  - Working
  - Interfacing
- Keypad – Introduction
  - Working
  - Interfacing

LCD Interfacing

- Liquid Crystal Displays (LCDs)
- cheap and easy way to display text
- Various configurations (1 line 20 X char - 8 lines X 80).
- Integrated controller
- The display has two register
  - command register
  - data register
- By RS you can select register
- Data lines (DB7-DB0) used to transfer data and commands

Alphanumeric LCD Interfacing

- RS – Register Select
  - RS = 0 → Command Register
  - RS = 1 → Data Register
- R/W = 0 → Write , 1 → Read
- E – Enable
  - Used to latch the data present on the data pins.
- D0 – D7
  - Bi-directional data/command pins.
  - Alphanumeric characters are sent in ASCII format.

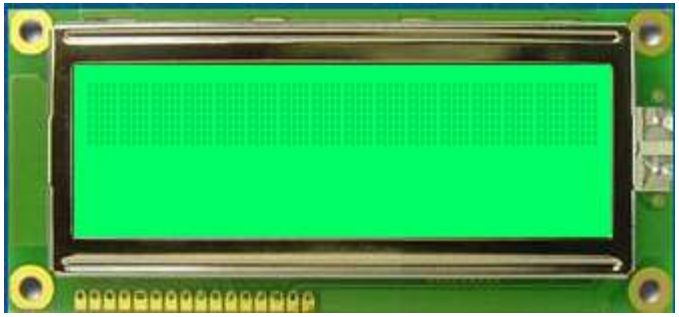
Pinout

- 8 data pins D7:D0
- RS (Register Select) : Data or Command
- R/W : Read or Write
- E : Enable (Latch data)

Basic LCD Working

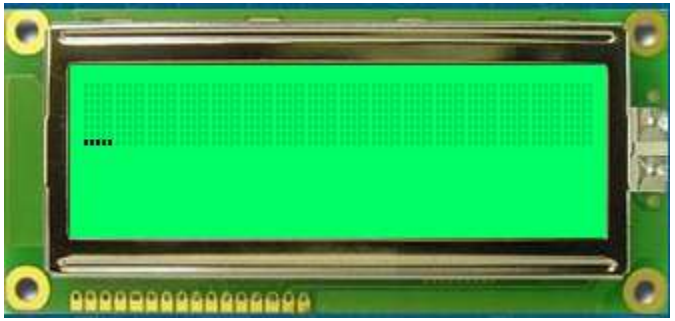
instr (0x0F) ;

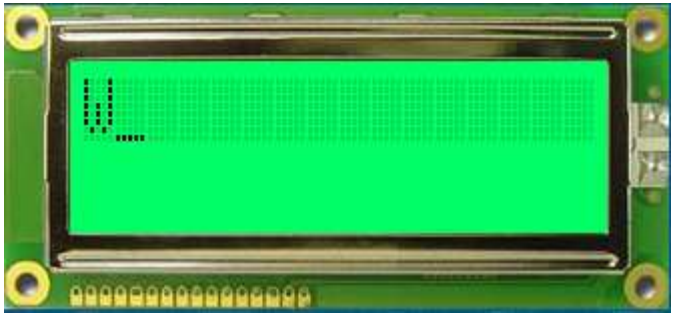
```
instr ( char data) {
        RS = 0;
        Enable = 1 ;
        data on controller Pins
        Enable = 0 ;
}
```



data ('W') ;

```
data ( char data ) {
        RS = 1;
        Enable = 1 ;
        data on controller pins
        Enable = 0 ;
}
```





LCD Commands

- The LCD's internal controller can accept several commands and modify the display accordingly. These commands would be things like:
  - Clear screen
  - Return home
  - Decrement/Increment cursor

- After writing to the LCD, it takes some time for it to complete its internal operations. During this time, it will not accept any new commands or data.
  - We need to insert time delay between any two commands or data sent to LCD

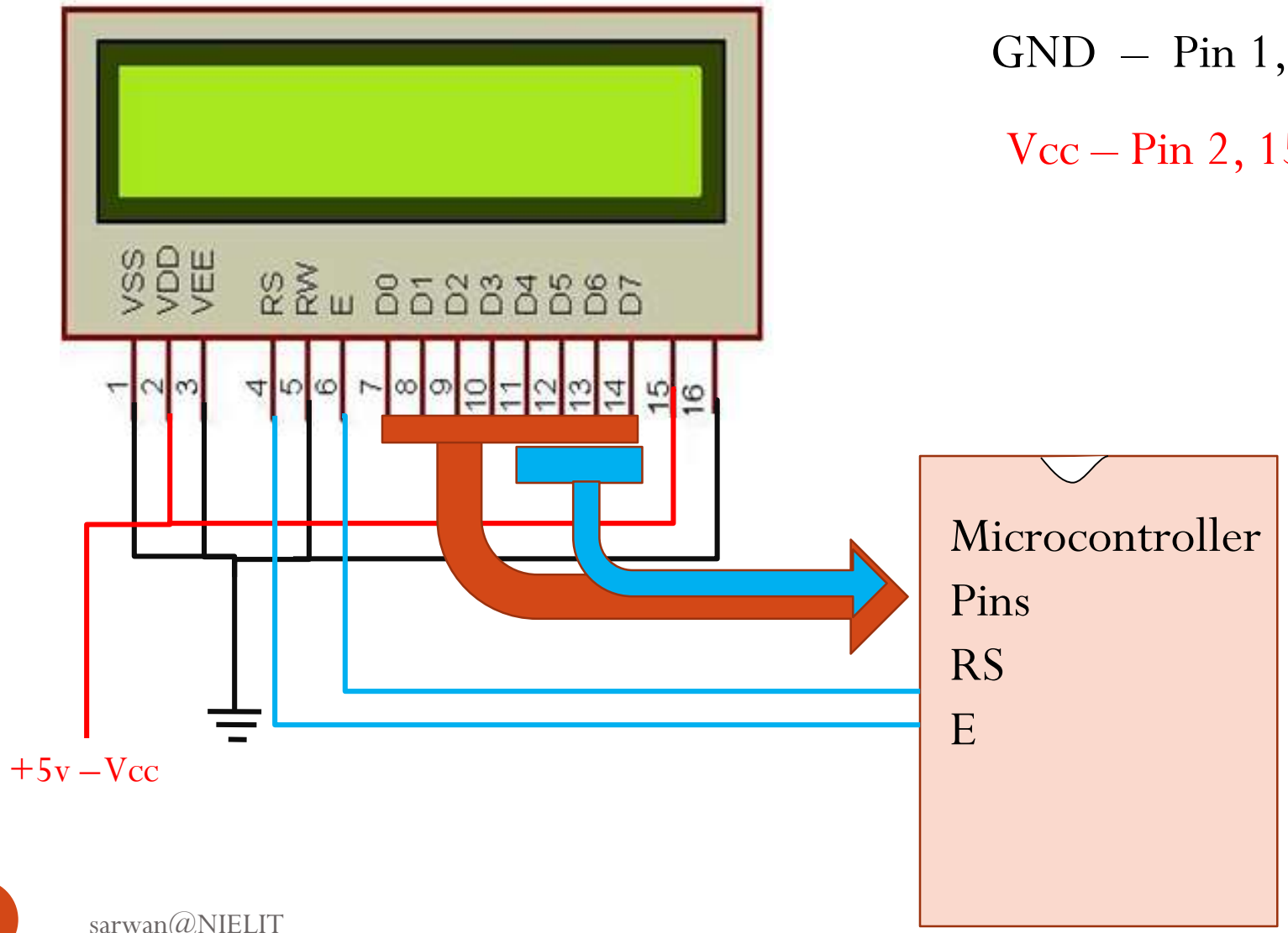
Pin No.	Name	Input / Ouput	Description
Pin no. 1	V_{SS}	-	Power supply (GND)
Pin no. 2	V_{CC}	-	Power supply (+5V)
Pin no. 3	V_{EE}	-	Power supply to control **Contrast**
Pin no. 4	**RS**	Input	0 = Instruction input 1 = Data input
Pin no. 5	**R/W**	Input	0 = Write to LCD module 1 = Read from LCD module
Pin no. 6	**EN**	Input / Output	Enable signal
Pin no. 7	**D0**	Input / Output	Data bus line 0 (LSB)
Pin no. 8	**D1**	Input / Output	Data bus line 1
Pin no. 9	**D2**	Input / Output	Data bus line 2
Pin no. 10	**D3**	Input / Output	Data bus line 3
Pin no. 11	**D4**	Input / Output	Data bus line 4
Pin no. 12	**D5**	Input / Output	Data bus line 5
Pin no. 13	**D6**	Input / Output	Data bus line 6
Pin no. 14	**D7**	Input / Output	Data bus line 7 (MSB)
Pin no. 15	**Backlight**	Input	+5v for LED backlight (+5V)
Pin no. 16	**Backlight**	Input	Ground for LED backlight (GND)

LCD Commands

Code (Decimal)	Code (Hex)	Command to LCD Instruction Register
1	0x01	Clear display screen
2	0x02	Return Home
4	0x04	Decrement cursor (shift cursor to left)
6	0x05	Increment cursor (shift cursor to right)
6	0x06	shift display right
7	0x07	shift display left
8	0x08	Display off, cursor off
10	0x0A	Display off, cursor on
12	0x0C	Display on, cursor off
14	0x0E	Display on, cursor on
15	0x0F	Display on, cursor blinking
16	0x10	Shift cursor position to left
20	0x14	Shift cursor position to right
24	0x18	Shift the entire display to the left
30	0x1C	Shift the entire display to the right
128	0x80	Force cursor to the beginning of 1st line
192	0xC0	Force cursor to the beginning of 2nd line
56	0x38	2 lines and 5 x 7 matrix

sarwan@NIELIT

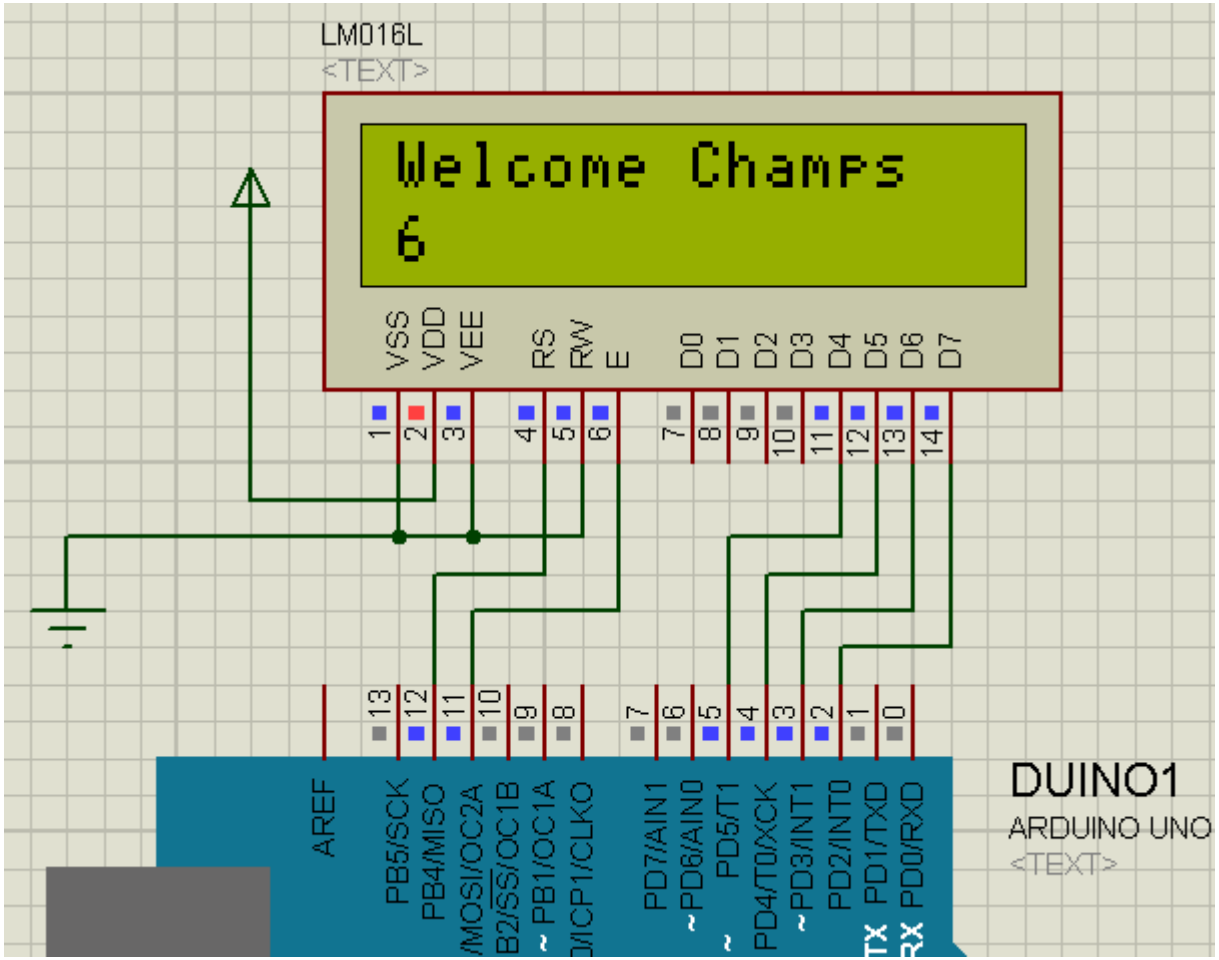
Circuit Diagram



GND – Pin 1,3,5,16

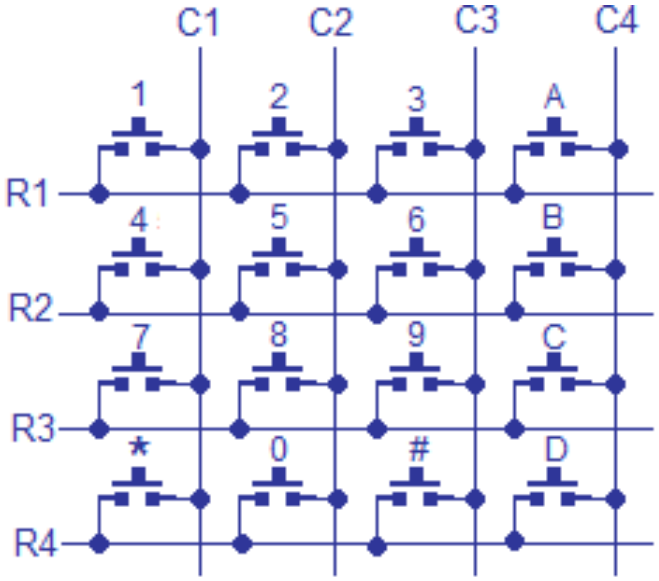
Vcc – Pin 2, 15

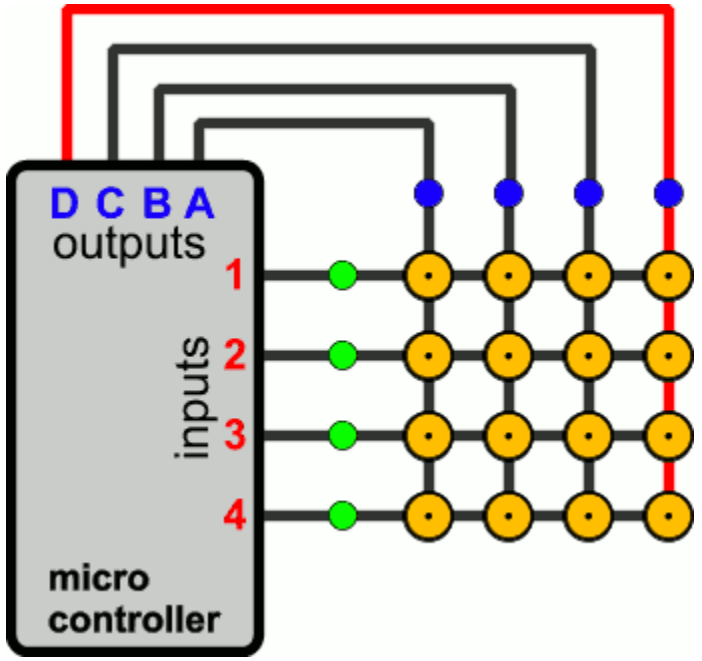
LCD Interfacing



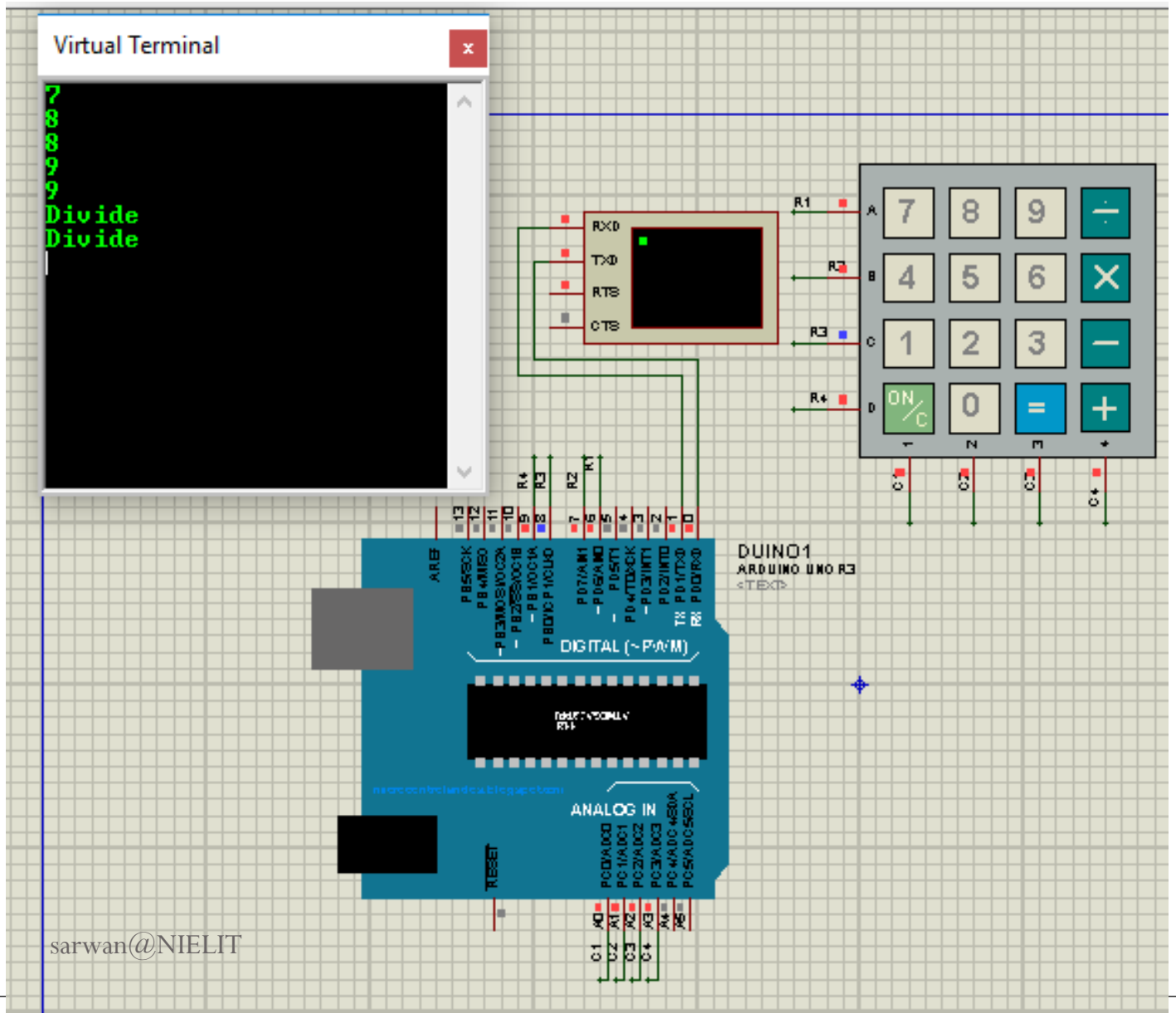
```
// include the library code:
#include <LiquidCrystal.h>

// initialize the library by associating any needed LCD interface pin
// with the arduino pin number it is connected to
        const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;
        LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
void setup() {
        // set up the LCD's number of columns and rows:
        lcd.begin(16, 2);
        // Print a message to the LCD.
        lcd.print("Welcome Champs");
}
void loop() {
        // set the cursor to column 0, line 1   (counting begins with 0)
        lcd.setCursor(0, 1);
        // print the number of seconds since reset:
         lcd.print(millis() / 1000);
}
```

Keypad interfacing





Keypad interfacing with Serial Monitor



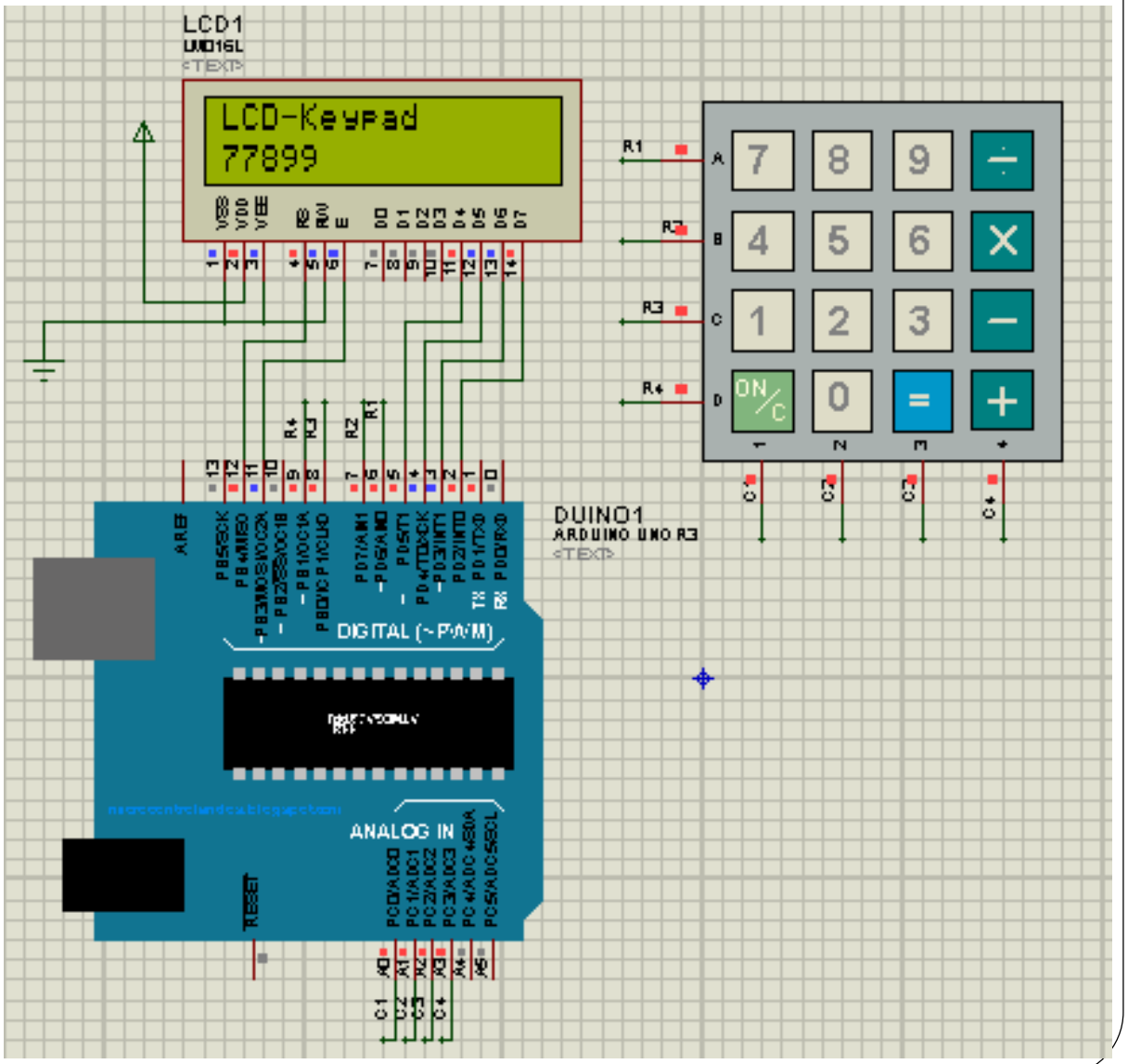
Coding

```
int r1=6, r2=7,r3=8,r4=9;
int c1=A0,c2=A1,c3=A2,c4=A3;
int colm1,colm2,colm3,colm4;

void setup()
{
    pinMode(r1,OUTPUT);  pinMode(r2,OUTPUT);
    pinMode(r3,OUTPUT);  pinMode(r4,OUTPUT);
    pinMode(c1,INPUT);  pinMode(c2,INPUT);
    pinMode(c3,INPUT);  pinMode(c4,INPUT);
    Serial.begin(9600);
    digitalWrite(c1,HIGH);  digitalWrite(c2,HIGH);
    digitalWrite(c3,HIGH);  digitalWrite(c4,HIGH);
}
```

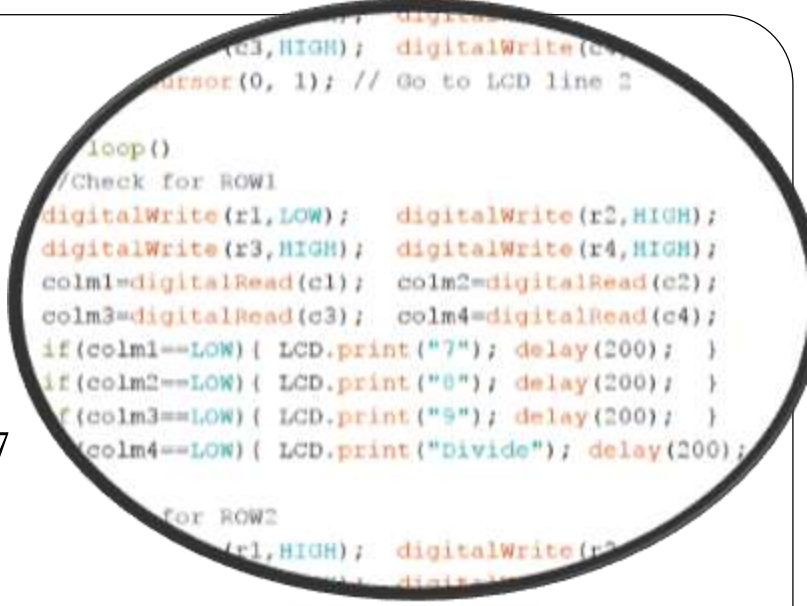
Coding

```
void loop()
      {   //Check for ROW1
       digitalWrite(r1,LOW);   digitalWrite(r2,HIGH);
       digitalWrite(r3,HIGH);  digitalWrite(r4,HIGH);
       colm1=digitalRead(c1);  colm2=digitalRead(c2);
       colm3=digitalRead(c3);  colm4=digitalRead(c4);
       if(colm1==LOW){ Serial.println("7"); delay(200);  }
       if(colm2==LOW){ Serial.println("8"); delay(200);  }
       if(colm3==LOW){ Serial.println("9"); delay(200);  }
       if(colm4==LOW){ Serial.println("Divide"); delay(200);  }
          //Check for ROW2
          //Check for ROW3
          //Check for ROW4
}
```

LCD - Keypad


LCD1
LCD-Keypad
77899
DUINO1
ARDUINO UNO R3
DIGITAL (~PWM)
ANALOG IN
RESET
ON/C

Coding



```
#include <LiquidCrystal.h>
int r1=6, r2=7,r3=8,r4=9;
int c1=A0,c2=A1,c3=A2,c4=A3;
int colm1,colm2,colm3,colm4;
// initialize the library - RS, E, D4, D5, D6, D7
LiquidCrystal LCD(12, 11, 5, 4, 3, 2);

void setup()
{     LCD.begin(16, 2);
      LCD.print("LCD-Keypad");
      pinMode(r1,OUTPUT);  pinMode(r2,OUTPUT);
      pinMode(r3,OUTPUT);  pinMode(r4,OUTPUT);
      pinMode(c1,INPUT);  pinMode(c2,INPUT);
      pinMode(c3,INPUT);  pinMode(c4,INPUT);
      Serial.begin(9600);
      digitalWrite(c1,HIGH);  digitalWrite(c2,HIGH);
      digitalWrite(c3,HIGH);  digitalWrite(c4,HIGH);
      LCD.setCursor(0, 1); // Go to LCD line 2
}
```

Happy Coding

Journey begins from here……