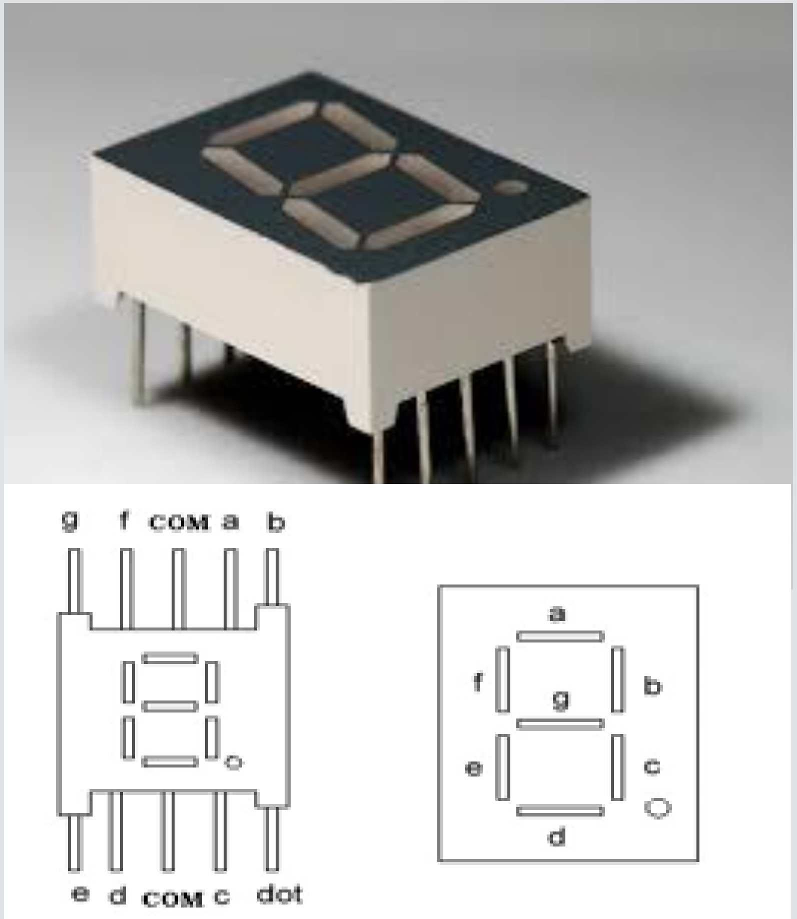
NIT MEGHALAYA

EMBEDDED SYSTEM DESIGN AND APPLICATION (CS 313)

NIT MEGHALAYA

LAB 2 :- DISPLAYING IN 7-SEGMENT LED AND LCD

Displaying in 7 segment LED



Seven-Segment Display

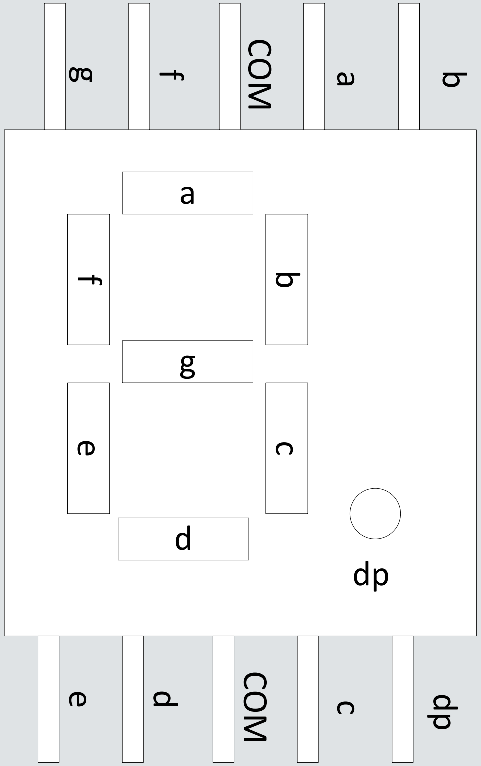
- The Common pin (COM) is connected to GND or VCC depending upon whether the display unit has common anode or common cathode.
- In this lab, we use the common anode display unit where COM is connected to VCC.

Displaying in 7 segment LED

- To interface the 7-segment display unit with nucleo board, we connect the digital pins of the board to the seven segment pins.
- If we send a low signal to a segment pin, the corresponding segment LED glows.
- A suitable resistor is connected via the COM pin to the VCC to limit the maximum current flowing.

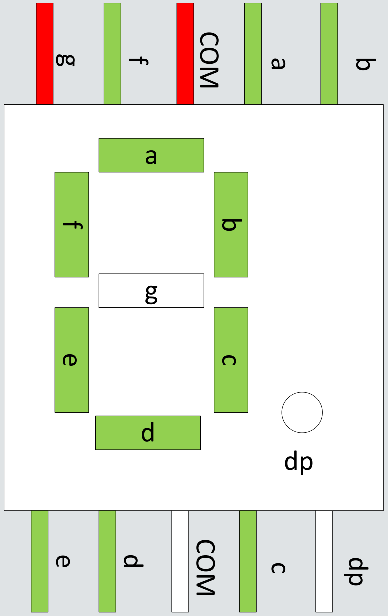
Interfacing to Digital Output

- Each segment can be interfaced to the digital output pins of the STM 32 Nucleo F401RE as:
  - DigitalOut a(D2);
  - DigitalOut b(D3);
  - DigitalOut c(D4);
  - DigitalOut d(D5);
  - DigitalOut e(D6);
  - DigitalOut f(D7);
  - DigitalOut g(D8);



Interfacing to Digital Output (contd.)

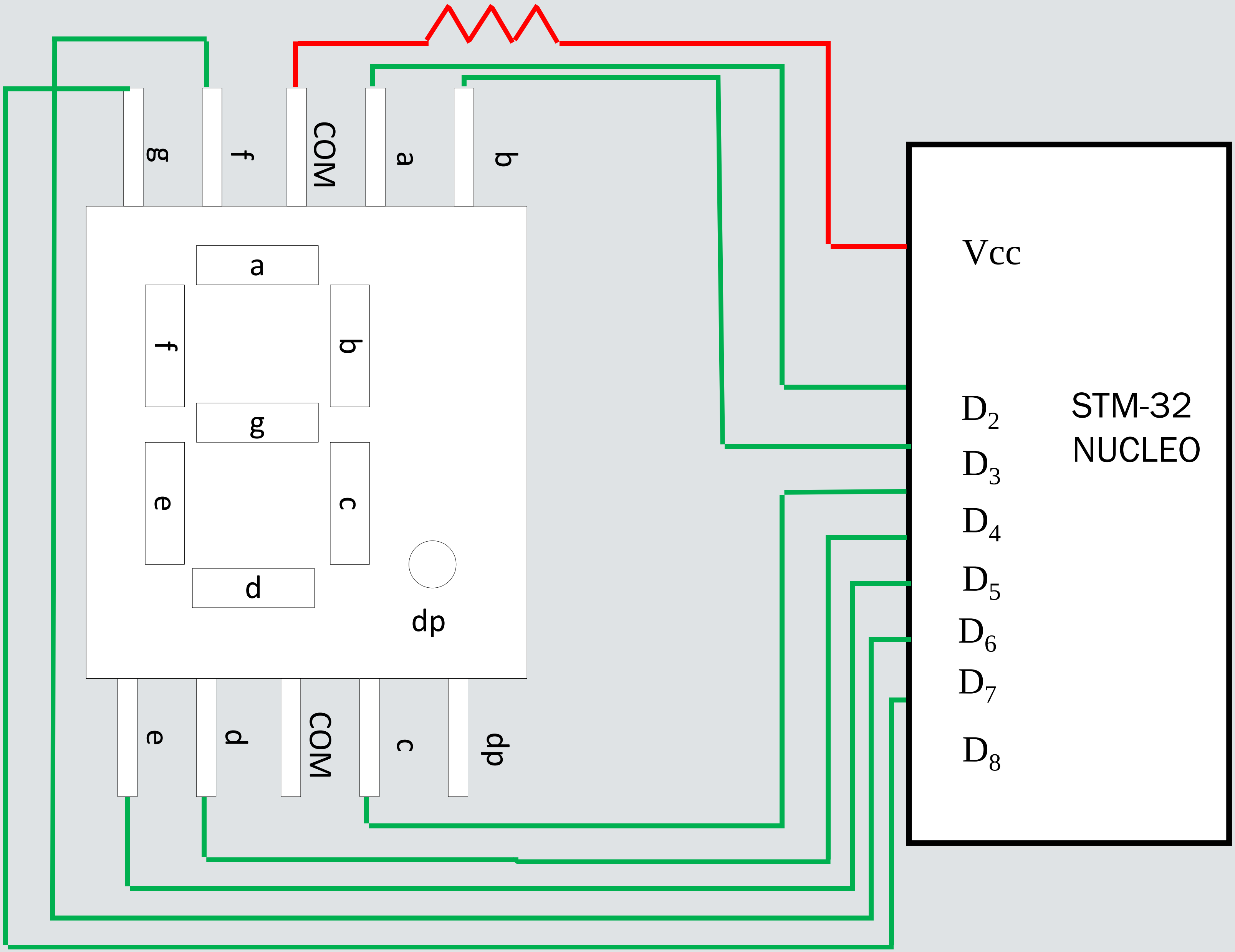
- For example, to display the digit 0 we have to set the digital output as
  - a = 0, b = 0, c = 0, d = 0, e = 0, f = 0, g = 1.
  - Segments a, b, c, d, e, f glows.



To display the digits 0 to 9

Digit	Code to Output
0	a=0, b=0, c=0, d=0, e=0, f=0, g=1
1	a=1, b=0, c=0, d=1, e=1, f=1, g=1
2	a=0, b=0, c=1, d=0, e=0, f=1, g=1
3	a=0, b=0, c=0, d=0, e=1, f=1, g=0
4	a=1, b=0, c=0, d=0, e=1, f=0, g=0
5	a=0, b=1, c=0, d=0, e=1, f=0, g=0
6	a=0, b=1, c=0, d=0, e=0, f=0, g=0
7	a=0, b=0, c=0, d=0, e=0, f=0, g=0
8	a=0, b=0, c=0, d=0, e=0, f=0, g=0
9	a=0, b=0, c=0, d=0, e=1, f=0, g=0

Connection Diagram



Program to display 0 to 9 repeatedly

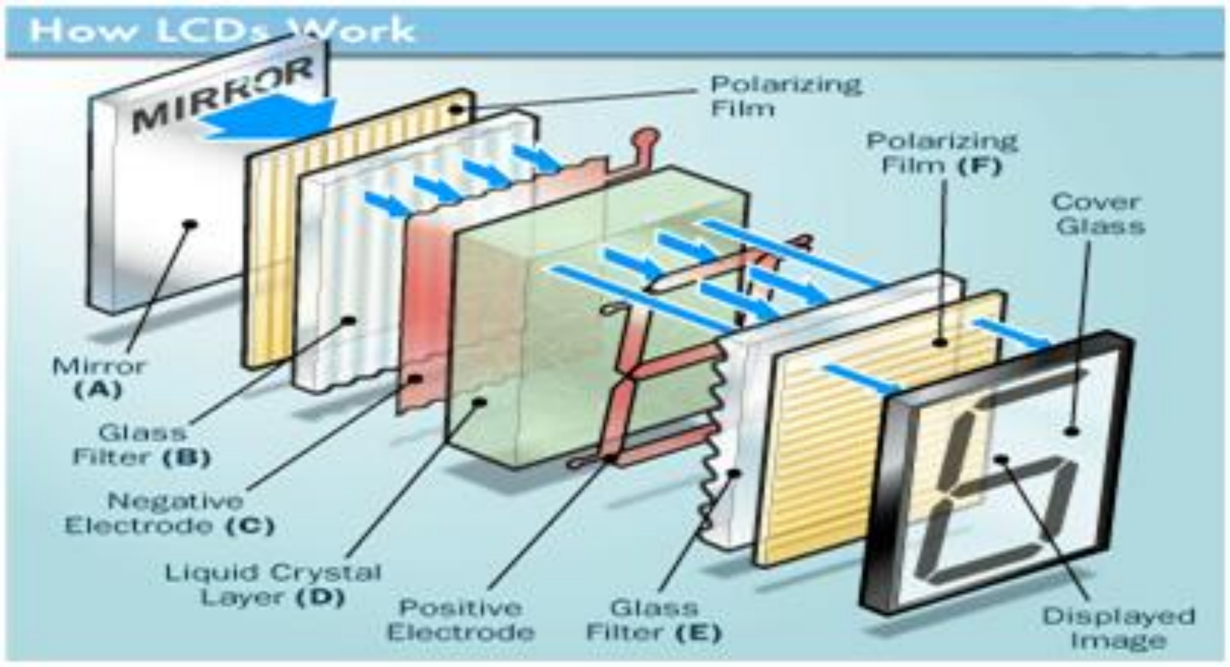
- The code can be written using a switch case statement as shown.

```
switch (display){
  case 0 : a = 0; b=0; c=0; d=0; e =0; f=0; g=1; break;
  …
  case 9 : a = 0; b=0; c=0; d=0; e =1; f=0; g=0; break;
}
display = 0;
while (1){
    Output segment values to 7-segment;
    wait (0.5);
    display++;
    If (display==10) display =0;
    wait(0.5);
}
```

Liquid Crystal Display (LCD)

- An LCD is a type of screen that is used in many computers, TVs, digital cameras, tablets, and cell phones.
- LCDs are very thin but are actually composed of several layers.
- Those layers include two polarized panels, with a liquid crystal solution between them.
- Light is projected through the layer of liquid crystals and is colorized, which produces the visible image.

Layers in LCD

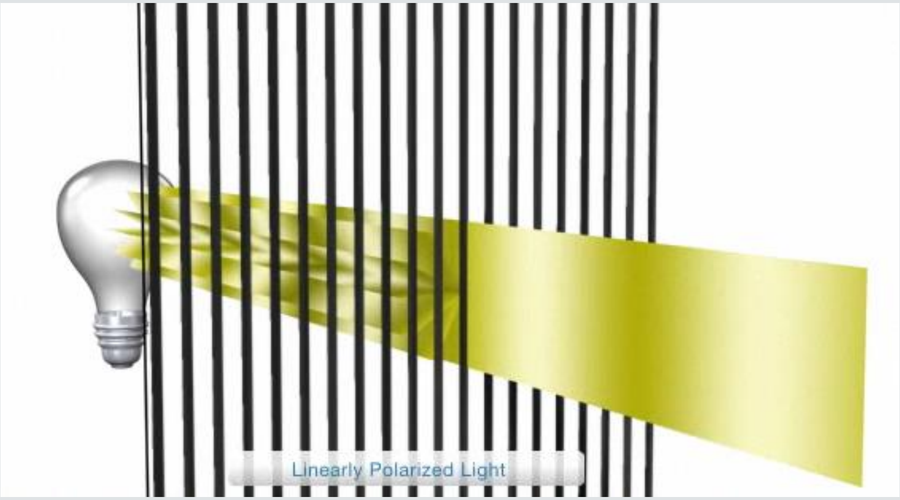


Working Principle

- The principle behind the LCD's is that when an electrical current is applied to the liquid crystal molecule, the molecule tends to untwist.
- This causes a change in the angle of light which is passing through the molecule of the polarized glass and also cause a change in the angle of the top polarizing filter.
  - *Little light is allowed to pass the polarized glass through a particular area of the LCD.*
  - *That particular area will become dark compared to others*
- In a LCD, a reflected mirror is arranged at the back.
- An electrode plane is made of indium-tin oxide which is kept on top and a polarized glass with a polarizing film is also added on the bottom of the device.
- The complete region of the LCD has to be enclosed by a common electrode and above it should be the liquid crystal matter.

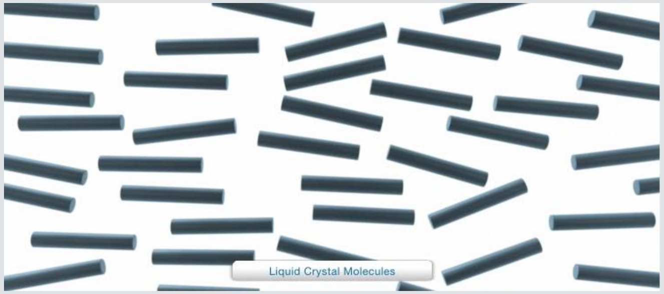
Linear Polarized Light

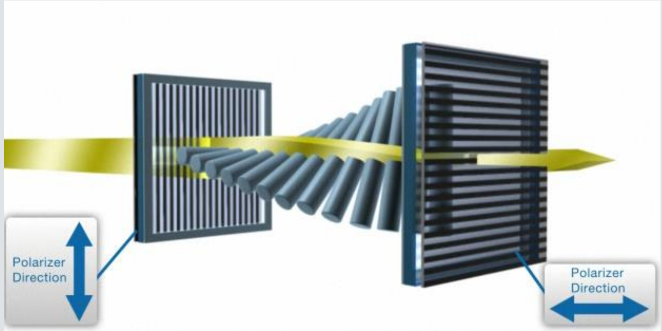
- Light usually vibrates in all directions.
- A linear polarized light limits the vibration to one direction.
- It absorbs the component of light that vibrate in all other direction.
- LCD require light to vibrate in one direction.

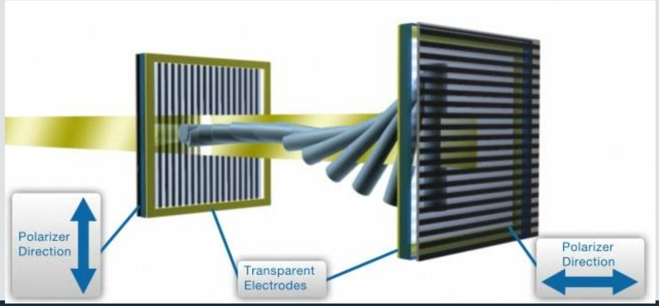


About Liquid Crystal

- Liquid crystal molecules can move freely while maintaining their orientation.
- It align itself to a polyimide film to the inside of a panel glass.
- When the two glass panels are not aligned the liquid crystal twists accordingly.
- The liquid crystal will also align to electric field.

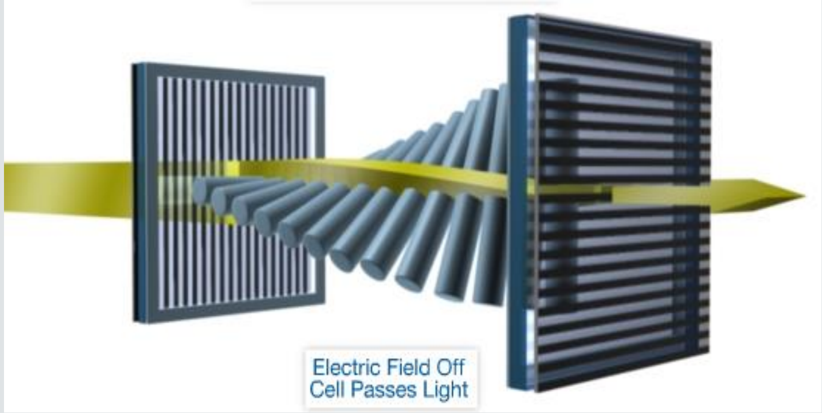






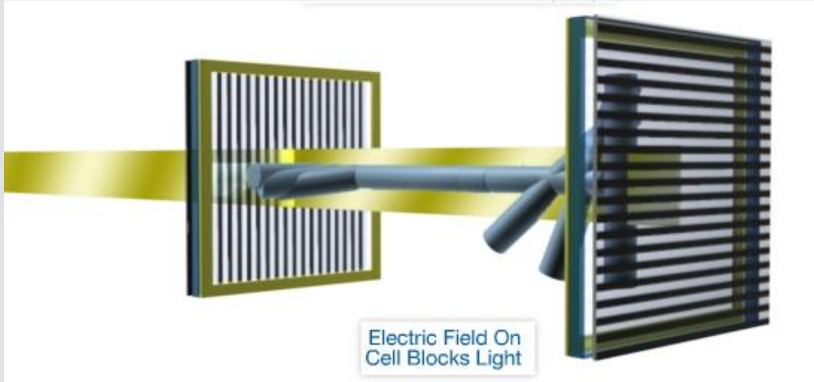
Twisted Nematic (TN) Display

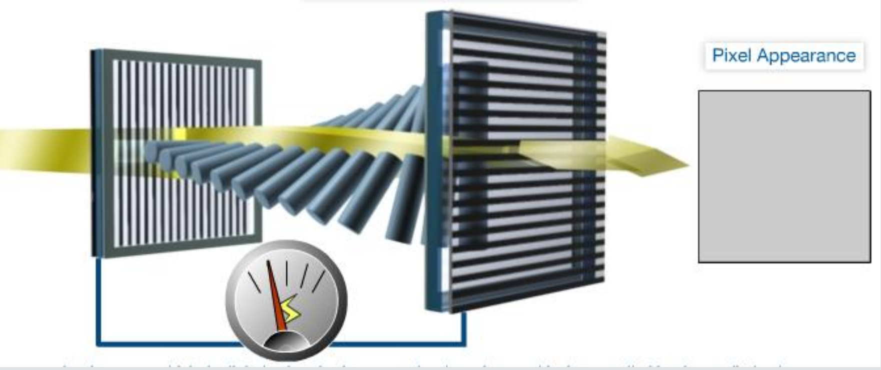
- Is the most common LCD display.
- The two alignments layers for the liquid crystal material are orthogonal.
- The light entering the polarize panel rotates by the twist in the liquid crystal and allowing it to pass through the second polarize.



Twisted Nematic (TN) Display

- The electric field is applied:
  - Liquid crystal loses its twist.
  - Align to the electric field.
  - Prevents the rotation of the polarized light.
  - The second polarizer absorbs the light.
- The applied voltage controls the absorbed and transmitted light.

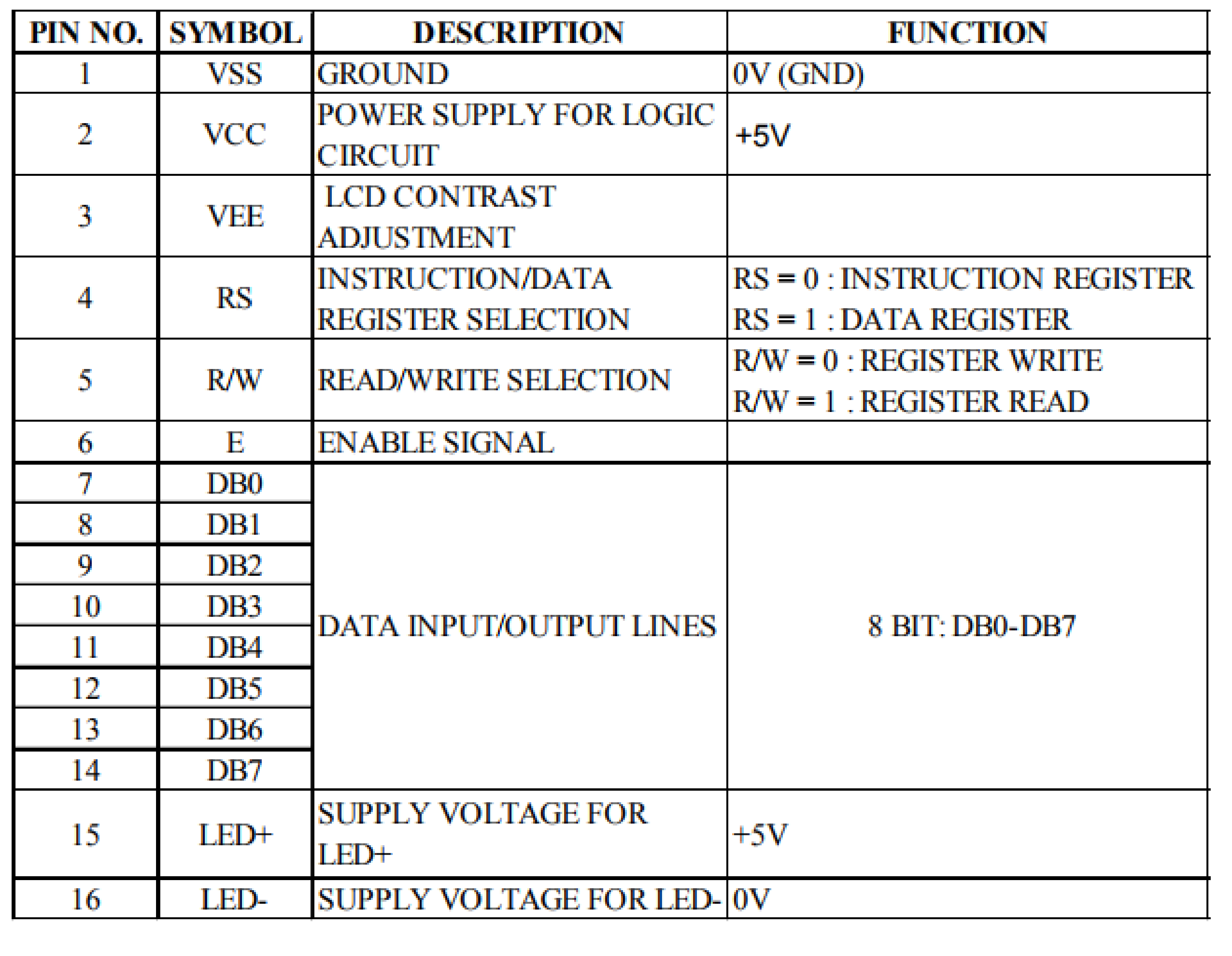




The JHD162A LCD

- In this Lab we will use the JHD162A LCD.
- This LCD unit consists of 16 pins.
  - Two lines can be displayed in this LCD.
  - Each line can hold 16 alphanumeric characters.

Pin Description of JHD162A



PIN NO.	SYMBOL	DESCRIPTION	FUNCTION
1	VSS	GROUND	0V (GND)
2	VCC	POWER SUPPLY FOR LOGIC CIRCUIT	+5V
3	VEE	LCD CONTRAST ADJUSTMENT	

4	RS	INSTRUCTION/DATA REGISTER SELECTION	RS = 0 : INSTRUCTION REGISTER
			RS = 1 : DATA REGISTER

5	R/W	READ/WRITE SELECTION	R/W = 0 : REGISTER WRITE R/W = 1 : REGISTER READ
6	E	ENABLE SIGNAL	
7	DB0	DATA INPUT/OUTPUT LINES	8 BIT: DB0-DB7
8	DB1		
9	DB2		
10	DB3		
11	DB4		
12	DB5		
13	DB6		
14	DB7		
15	LED+	SUPPLY VOLTAGE FOR LED+	+5V
16	LED-	SUPPLY VOLTAGE FOR LED-	0V

LCD Mbed Program

- In this program we will interface the JHD162A to STM-32 Nucleo F401RE.
- We will display a scrolling text on the 1^{st} line of the LCD.
- We will display some simple text on the 2^{nd} line of the LCD.

Accessories Required

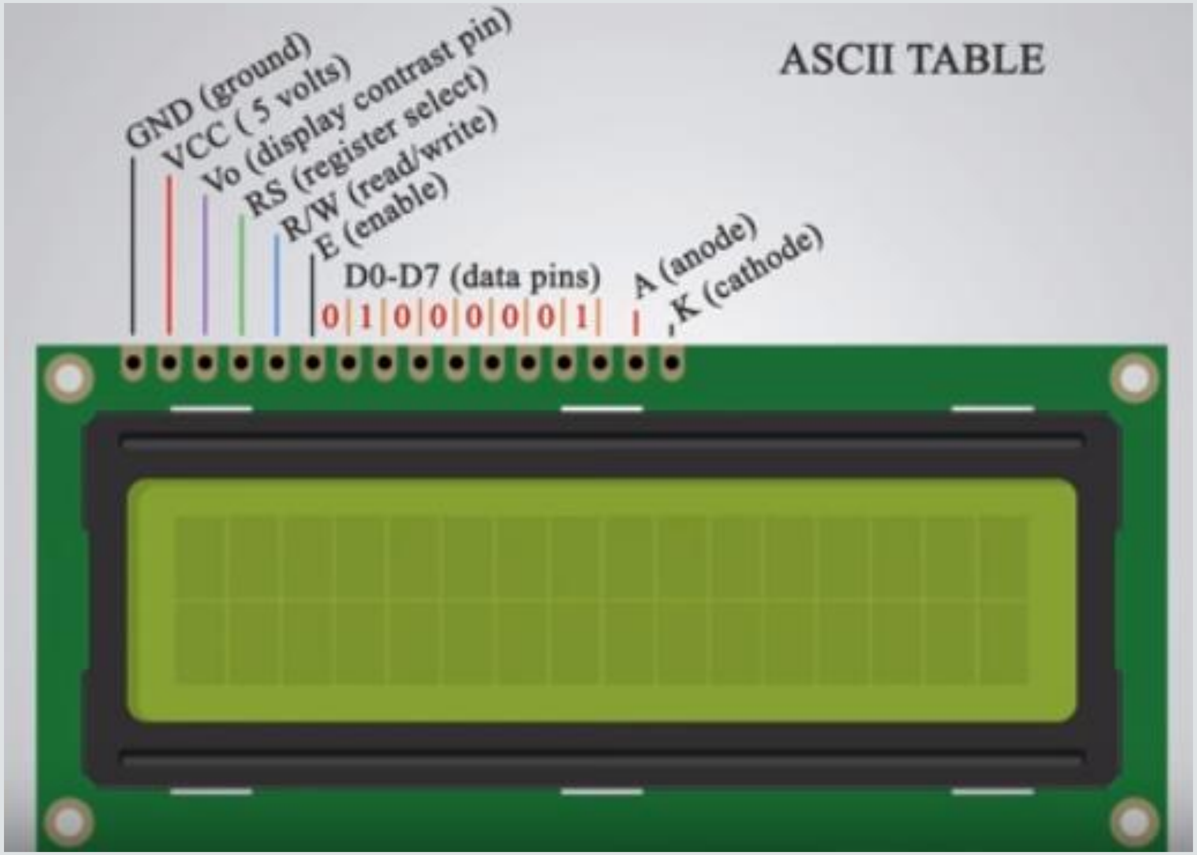
- Jumper Wires.
- 3 Resistors: 8.2 Kohms, 4.7 Kohms, 47 ohms.
- 1 JHD162A LCD.
- 1 STM-32 Nucleo-F401RE kit.

Connection

- LCD
  - 1st pin VSS is connected to GND.
  - 2nd pin VCC is connected to +5V supply.
  - 3rd pin VEE that is connected to Variable Resistor. This is used to control the contrast of the LCD.
  - 4th pin is for Register Select (RS). This is used to determine whether we will send data or command to the LCD.
    - If RS==0, Command is send.
    - If RS==1, Data is send.
  - 5th pin is for Read/Write R/W.
    - If Read, R/W=1.
    - IF Write, R/W=0.
  - 6th pin is Enable (EN) thatis used to enable the sending of data.

Connection (contd.)

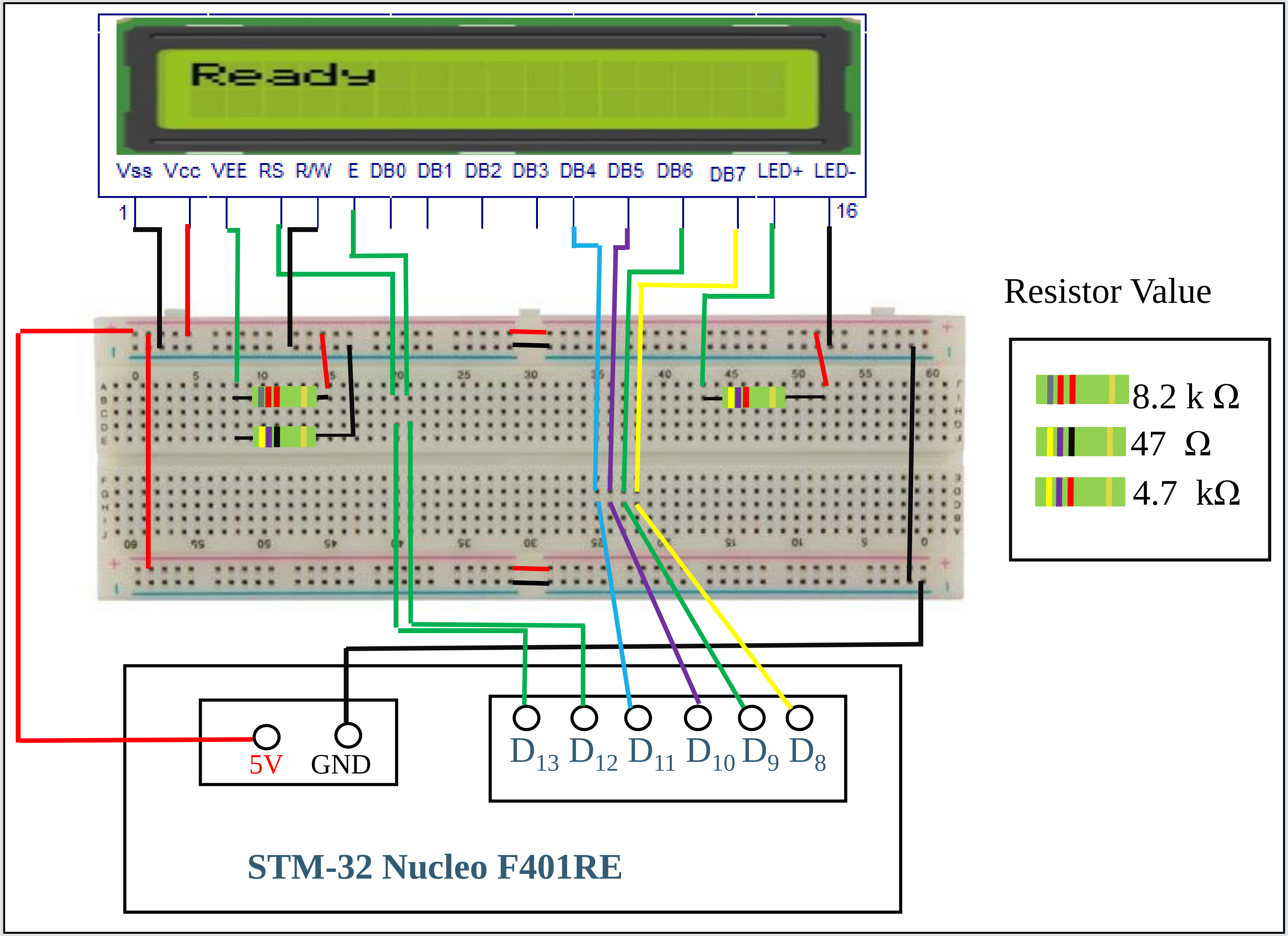
- The next 8 pins (7 to 14) are used to send data.
- The last two pins are for LED backlight. Pin 13 is anode, connected to VCC via a resistor, and pin 14 is cathode connected to ground.



LCD connections in 4-bit Mode

- The difference between 4-bit and 8-bit operations is that data are sent out as nibbles instead of bytes.
- The common steps are:
  - Mask lower 4-bits
  - Send to the LCD port
  - Send enable signal
  - Mask higher 4-bits
  - Send to the LCD port
  - Send enable signal

Circuit Diagram



Resistor Value

8.2 k Ω

47 Ω

4.7 kΩ

Library Modules to be Included

- mbed
- TextLCD
- TextLCDScroll

TextLCDScroll

- Function Definition of TextLCDScroll

```
TextLCDScroll::TextLCDScroll(PinName rs, PinName e,
PinName d4, PinName d5, PinName d6, PinName d7,
TextLCD::LCDType type):
TextLCD(rs,e,d4,d5,d6,d7,type)
```

- In the TextLCDScroll we have to pass 7 arguments:-

1. RS pin – The pin in STM32 kit used as Register Select.
2. EN pin - The pin in STM32 kit used as Data Enable.

3-6. D4- D7 pin – The 4-bits data pins that are interfaced to the kit.

7. Finally, the LCD type. In our case, the LCD type is TextLCD::LCD16x2.

Example:

```
TextLCDScroll lcd(D13, D12, D11, D10, D9, D8,
TextLCD::LCD16x2);
```

Basic Commands

- cls: used to clear screen.
- setSpeed(n) where n>=1: used to set the scrolling speed of the text.
- setLine(line_number, “Text”) where line number = 0 or 1 which indicates line 1 or 2 respectively. In the next field any text can be supplied.
- If the number of characters exceed 16. Then, the text will scroll through the screen.

cls: Clear Screen

- 0x01 is the hexcode command to clear screen

```
void TextLCD::cls() {
    writeCommand(0x01); // cls, and set cursor to 0
    wait(0.00164f);     // This takes 1.64 ms
    locate(0, 0);
}
```

Scrolling Speed

```
bool TextLCDScroll::setSpeed( int speed)
{
    if ((speed >= 0.1) && (speed <= 10)) {
        tick.detach();
        _speed = speed;
        startTicker();      // How fast the screen ticker
        return(true);
    } else {
        return(false);
    }
}

void TextLCDScroll::startTicker(){
  if (_mode == leftright)

     tick.attach(this,&TextLCDScroll::ScrollRightLeft,
               1.0/_speed);
  else
     tick.attach(this,&TextLCDScroll::ScrollLeft,
               1.0/_speed);
}
```

Sample Program

```
#include "mbed.h"
#include "TextLCDScroll.h“

TextLCDScroll lcd (D13, D12, D11, D10, D9, D8,
                    TextLCD::LCD16x2);
int main() {
      lcd.cls();
 lcd.setLine(0,“Hello");
}
```

Thank you