LECTURE 29 - LCD INTERFACING

* LCD PIN:

Pin Number	Name	Description
1	VSS	Power supply (GND)
2	VCC	Power supply (+5V)
3	VEE	Contrast adjust
4	RS	0 = Instruction input 1 = Data input
5	R/W	0 = Write to LCD module 1 = Read from LCD module
6	EN	Enable signal
7	D0	Data bus line 0 (LSB)
8	D1	Data bus line 1
9	D2	Data bus line 2
10	D3	Data bus line 3
11	D4	Data bus line 4
12	D5	Data bus line 5
13	D6	Data bus line 6
14	D7	Data bus line 7 (MSB)

* LCD COMMANDS:

Table - Frequently used commands and instructions for LCD.

No.	Instruction	Hex	Decimal
1	Function set: 8-bit, 1 Line, 5x7 Dots	0x30	48
2	Function set: 8-bit, 2 Line, 5x7 Dots	0x38	56
3	Function set: 4-bit, 1 Line, 5x7 Dots	0x20	32
4	Function set: 4-bit, 2 Line, 5x7 Dots	0x28	40
5	Entry mode	0x06	6

6	Display off Cursor off (clearing display without clearing DDRAM content)	0x08	8
7	Display on Cursor on	0x0E	14
8	Display on Cursor off	0x0C	12
9	Display on Cursor blinking	0x0F	15
10	Shift entire display left	0x18	24
11	Shift entire display right	0x1C	30
13	Move cursor left by one character	0x10	16
14	Move cursor right by one character	0x14	20
15	Clear Display (also clear DDRAM content)	0x01	1
16	Set DDRAM address or cursor position on display	0x80 + add*	128 + add*
M	Set CGRAM address or set pointer to CGRAM location	0x40 + add**	64 + add**

* STEPS TO PRINT A CHARACTER - LCD:

① Set P0.12 to P0.15 LCD Data (D4 to D7). P0.4, 5, 6 as RS, RW and EN as O/P

```
IO0DIR = 0x0000FFF0;
```

LCD Initialize

② Set the Function Set (4-bit interface, two line, 5x8 dots)

```
LCD_CMD (0x28);
```

③ Clear LCD Display

```
LCD_CMD (0x01);
```

④ Initialize Cursor to Home Position

```
LCD_CMD (0x02);
```

⑤ Set the Entry Mode (Cursor Increment, Display shift off)

```
LCD_CMD (0x06);
```

⑥ Display on, Cursor off and Blink off

```
LCD_CMD (0x0C);
```

⑦ Set the Cursor to First Line or Second Line

```
LCD_CMD (0x80 + P00); Or LCD_CMD (0xC0 + P00);
```

⑧ Display the character

```
LCD_CHAR (c);
```

* STEP 1 - IODIR = 0x0000FFF0;

* STEP 2 - LCD_CMD (0x28);

Function set Command

Instruction	Instruction Code										Description
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	
Function set	0	0	0	0	1	DL	N	F	-	-	Set interface data length (DL: 8-bit/4-bit), numbers of display line (N: 2-line/1-line) and, display font type (F: 5×11 dots/5×8 dots)

0	0	1	0	1	0	0	0
2				8			

* STEP 3 - LCD_CMD (0x01);

Clear Display Command

Instruction	Instruction code										Description
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	
Clear Display	0	0	0	0	0	0	0	0	0	1	Write "00H" to DDRAM and set DDRAM address to "00H" from AC.

0	0	0	0	0	0	0	1
0				1			

* STEP 4 - LCD_CMD (0x02);

Cursor Home Command

P70

Instruction	Instruction Code										Description
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	
Return Home	0	0	0	0	0	0	0	0	1	–	Set DDRAM address to "00H" from AC and return cursor to its original position if shifted. The contents of DDRAM are not changed.

0	0	0	0	0	0	1	0
0				2			

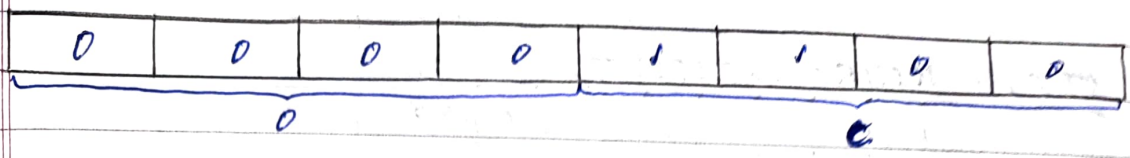
* STEP 5 - LCD_CMD (0x06);
Entry mode set command

Instruction	Instruction Code										Description
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	
Entry Mode Set	0	0	0	0	0	0	0	1	I/D	SH	Assign cursor moving direction and enable the shift of entire display.

0	0	0	0	0	1	1	0
0				6			

* STEP 6 - LCD_CMD (0x0C);
Display control command

Instruction	Instruction Code										Description
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	
Display ON/OFF Control	0	0	0	0	0	0	1	D	C	B	Set display (D), cursor (C) and blinking of cursor (B) on/off control bit.



* STEP 7 - LCD_CMD (0x80 + Pos); (or) LCD_CMD (0xC0 + Pos);

* STEP 8 - LCD_CHAR (c);

* LCD COMMAND WRITE (4-BIT):

① Set LCD_RS = 0 & LCD_RW = 0
```
IO0CLR = 0x00000030;
```
② Split MSB 4-Bits of LCD Command
```
cmd_MSB = cmd & 0xF0;
cmd_MSB = cmd_MSB >> 4;
```
③ Move MSB nibble to PORT0 - P0.15 to P0.12
```
IO0PIN = cmd_MSB;
```
④ Give LCD Enable signal
```
IO0SET = 0x00000040; // EN = 1
delay_ms(5);
IO0CLR = 0x00000040; // EN = 0
```
⑤ Split LSB 4-Bits of LCD Command
```
cmd_LSB = cmd & 0x0F;
cmd_LSB = cmd_LSB << 12;
```
⑥ Move LSB nibble to PORT0 - P0.15 to P0.12
```
IO0PIN = cmd_LSB
```
⑦ Give LCD Enable signal
```
IO0SET = 0x00000040; // EN = 1
delay_ms(5);
IO0CLR = 0x00000040; // EN = 0
```

* STEP 1 - IOCLR = 0x00000030;

* STEP 2 - cmd_MSB = cmd & 0xF0;
cmd_MSB = cmd_MSB << 8;

cmd 0100 1001
&0xF0 1111 0000
cmd-MSB 0100 0000

cmd_MSB 0000 0000 0100 0000
<<8
cmd_MSB 0100 0000 0000 0000

* STEP 3 - IOPIN = cmd_MSB;

PORT 0
15 14 13 12
IOPIN = 0100 0000 0000 0000

* STEP 4 - IOSET = 0x00000040; // EN=1
delay_ms(5);
IOCLR = 0x00000040; // EN=0

* STEP 5 - cmd_LSB = cmd & 0x0F;
cmd_LSB = cmd_LSB << 12;

cmd 0100 1001
&0x0F 0000 1111
cmd-LSB 0000 1001

cmd_LSB 0000 0000 0000 1001
<< 12
cmd_LSB 1001 0000 0000 0000

* STEP 6 - IOPIN = cmd_LSB;

PORT 0
15 14 13 12
IOPIN = 1001 0000 0000 0000

* STEP 7 - IOOSET = 0x00000040; // EN=1
  delay_ms(5);
  IOOCLR = 0x00000040; // EN=0

* LCD DATA WRITE (4-BIT):

① Set LCD_RS = 1 & LCD_RW = 0
  IOOSET = 0x00000010; // RS=1
  IOOCLR = 0x00000020; // RW=0

② Split MSB 4-Bits of LCD command
  data_MSB = data & 0xF0;
  data_MSB = data_MSB << 8;

③ Move MSB nibble to PORT0 - P0.15 to P0.12
  IOOPIN = data_MSB;

④ Give LCD Enable signal
  IOOSET = 0x00000040; // EN=1
  delay_ms(5);
  IOOCLR = 0x00000040; // EN=0

⑤ Split LSB 4-Bits of LCD command
  data_LSB = data & 0x0F;
  data_LSB = data_LSB << 12;

⑥ Move LSB nibble to PORT0 - P0.15 to P0.12
  IOOPIN = data_LSB;

⑦ Give LCD Enable signal
  IOOSET = 0x00000040; // EN=1
  delay_ms(5);
  IOOCLR = 0x00000040; // EN=0

* STEP 1 - IOOSET = 0x00000010; // RS=1
  IOOCLR = 0x00000020; // RW=0

* STEP 2 - data_MSB = data & 0xF0;
  data_MSB = data_MSB << 8;

data 0100 1001 data_MSB 0000 0000 0100 0000
0xF0 1111 0000 << 8
data_MSB 0100 0000 data_MSB 0100 0000 0000 0000

* STEP 3 - IODPIN = data_MSB;

PORTD
15 14 13 12
IODPIN = 0100 0000 0000 0000

* STEP 4 - IOSET = 0x00000040; //EN=1
  delay_ms(5);
  IOCLR = 0x00000040; //EN=0

* STEP 5 - data_LSB = data & 0x0F;
  data_LSB = data_LSB << 12;

data 0100 1001 data_LSB 0000 0000 0000 1001
&0x0F 0000 1111 << 12
data_LSB 0000 1001 data_LSB 1001 0000 0000 0000

* STEP 6 - IODPIN = data_LSB;

PORTD
15 14 13 12
IODPIN = 1001 0000 0000 0000

* STEP 7 - IOSET = 0x00000040; //EN=1
  delay_ms(5);
  IOCLR = 0x00000040; // EN=0