基本FPGA實習

Lab8 Text LCD-JHD 162A

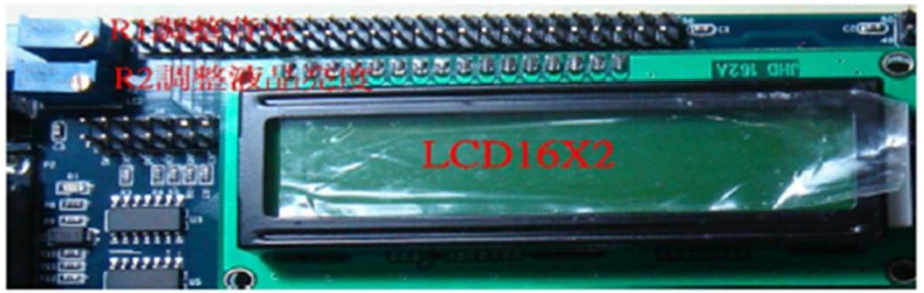
Text LCD

LCD16X2

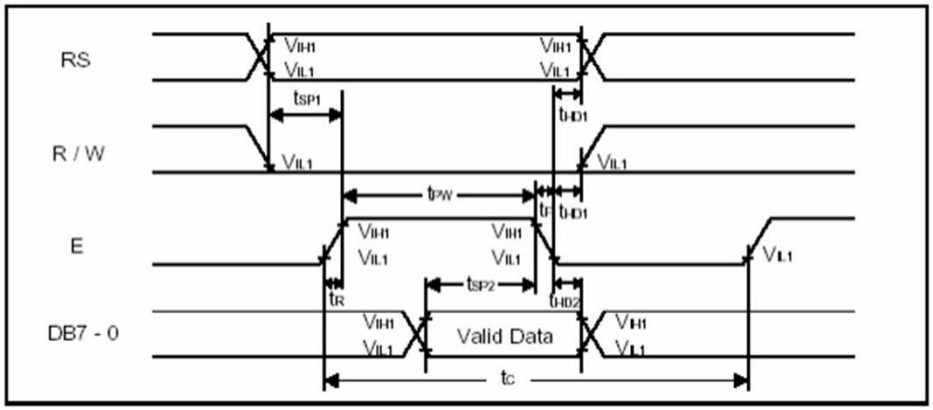
LCD16X2 是一個每行 16 字, 共 2 行的文字型液晶顯示器, 可以顯示 ASCII 字型.

● **Pin assignment**

Pin NO.	Symbol	Function		Remark
1	GND	Power supply	0V	
2	Vdd		+5V	
3	VS		For LCD	Variable
4	RS	Register Select(H=Data,L=Instruction)		
5	R/W	Read/Write L=MPU to LCM,H=LCM to MPU		
6	E	Enable		
7	DB0	Data bus bit 0		
8	DB1	Data bus bit 1		
9	DB2	Data bus bit 2		
10	DB3	Data bus bit 3		
11	DB4	Data bus bit 4		
12	DB5	Data bus bit 5		
13	DB6	Data bus bit 6		
14	DB7	Data bus bit 7		
15	A	Anode of LED Unit		
16	K	Cathode of LED Unit		



• **WRITE OPERATION**



• **WRITE MODE**

Characteristics	Symbol	Limit			Unit	Test Condition
		Min.	Typ.	Max.		
E Cycle Time	tc	1000	-	-	ns	Pin E
E Pulse Width	tPW	450	-	-	ns	Pin E
E Rise/Fall Time	tR, tF	-	-	25	ns	Pin E
Address Setup Time	tSP1	60	-	-	ns	Pins: RS, R/W, E
Address Hold Time	tHD1	20	-	-	ns	Pins: RS, R/W, E
Data Setup Time	tSP2	195	-	-	ns	Pins: DB7 - 0
Data Hold Time	tHD2	10	-	-	ns	Pins: DB7 - 0

Text LCD

13. INSTRUCTION SET

COMMAND	COMMAND CODE										COMMAND CODE	E-CYCLE f_{osc}=250KHz
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
SCREEN CLEAR	0	0	0	0	0	0	0	0	0	1	Screen Clear, Set AC to 0 Cursor Reposition	1.64ms
CURSOR RETURN	0	0	0	0	0	0	0	0	1	*	DDRAM AD=0, Return, Content Changeless	1.64ms
INPUT SET	0	0	0	0	0	0	0	1	I/D	S	Set moving direction of cursor, Appoint if move	40us
DISPLAY SWITCH	0	0	0	0	0	0	1	D	C	B	Set display on/off,cursor on/off, blink on/off	40us
SHIFT	0	0	0	0	0	1	S/C	R/L	*	*	Remove cursor and whole display,DDRAM changeless	40us
FUNCTION SET	0	0	0	0	1	DL	N	F	*	*	Set DL,display line,font	40us
CGRAM AD SET	0	0	0	1	ACG						Set CGRAM AD, send receive data	40us
DDRAM AD SET	0	0	1	ADD							Set DDRAM AD, send receive data	40us
BUSY/AD READ CT	0	1	BF	AC							Executing internal function, reading AD of CT	40us
CGRAM/ DDRAM DATA WRITE	1	0	DATA WRITE								Write data from CGRAM or DDRAM	40us
CGRAM/ DDRAM DATA READ	1	1	DATA READ								Read data from CGRAM or DDRAM	40us
	I/D=1: Increment Mode; I/D=0: Decrement Mode S=1: Shift S/C=1: Display Shift; S/C=0: Cursor Shift R/L=1: Right Shift; R/L=0: Left Shift DL=1: 8D DL=0: 4D N=1: 2R N=0: 1R F=1: 5x10 Style; F=0: 5x7 Style BF=1: Execute Internal Function; BF=0: Command Received										DDRAM: Display data RAM CGRAM: Character Generator RAM ACG: CGRAM AD ADD: DDRAM AD & Cursor AD AC: Address counter for DDRAM & CGRAM	E-cycle changing with main frequency. Example: If fcp or f_{osc}=270KHz 40us x 250/270 =37us

No	HEX Value	COMMAND TO LCD
1	0x01	Clear Display Screen
2	0x30	Function Set: 8-bit, 1 Line, 5x7 Dots
3	0x38	Function Set: 8-bit, 2 Line, 5x7 Dots
4	0x20	Function Set: 4-bit, 1 Line, 5x7 Dots
5	0x28	Function Set: 4-bit, 2 Line, 5x7 Dots
6	0x06	Entry Mode
7	0x08	Display off, Cursor off
8	0x0E	Display on, Cursor on
9	0x0C	Display on, Cursor off
10	0x0F	Display on, Cursor blinking
11	0x18	Shift entire display left
12	0x1C	Shift entire display right
13	0x10	Move cursor left by one character
14	0x14	Move cursor right by one character
15	0x80	Force cursor to beginning of 1st row
16	0xC0	Force cursor to beginning of 2nd row

Text LCD

14. FONT TABLE

```v
module Lab8(LCD_DB,LCD_EN,LCD_RW,LCD_RS,clk,rst);
output reg [7:0]LCD_DB;
output reg LCD_EN,LCD_RS;
output LCD_RW;
input clk,rst;

assign LCD_RW=1'b0;//Only Write

parameter idle=4'd0,setup_mode=4'd1,write_mode=4'd2,setup_display=4'd3,write_display=4'd4,setup_clear=4'd5,write_clear=4'd6;
parameter setup_position=4'd7,write_position=4'd8,setup_data=4'd9,write_data=4'd10;
reg [3:0]state;
reg [2:0]datacnt;

wire [7:0]romdata[6:0];
assign romdata[0]=8'h30;//0
assign romdata[1]=8'h31;//1
assign romdata[2]=8'h32;//2
assign romdata[3]=8'h33;//3
assign romdata[4]=8'h34;//4
assign romdata[5]=8'h35;//5
assign romdata[6]=8'h36;//6

reg [19:0]divclk;//40MHz=25ns  25ns*1024*1024 ~= 25ms
always @(posedge clk or negedge rst)begin
    if(~rst)
        divclk<=20'd0;
    else
        divclk<=divclk+1;
end

//Meely.............
always @(posedge divclk[19] or negedge rst)begin
    if(~rst)begin
        state<=idle;
        datacnt<=3'd0;
    end
    else begin
        case (state)
            idle:begin
                datacnt<=3'd0;
                state<=setup_mode;
            end
            //----Mode Command--------
            setup_mode:state<=write_mode;
            write_mode:state<=setup_display;
            //----Display Command-----------
            setup_display:state<=write_display;
            write_display:state<=setup_clear;
            //----Clear Command-------
            setup_clear:state<=write_clear;
            write_clear:state<=setup_position;
             //----Position Command----------
             setup_position:state<=write_position;
             write_position:state<=setup_data;
             //----Data-----------------
             setup_data:state<=write_data;
             write_data:begin
                 if(datacnt<3'd6)begin
                     datacnt<=datacnt+1;
                     state<=setup_data;
                 end
                 else
                     state<=state;
             end
             default:state<=idle;
        endcase
    end
end
```

```v
//Moore
always @(state)begin
    case (state)
        idle:begin
            LCD_EN<=1'b1;
            LCD_RS<=1'b0;//Command Mode
            LCD_DB<=8'd0;
        end
        //----Mode Command--------
        setup_mode:begin
            LCD_EN<=1'b1;
            LCD_RS<=1'b0;//Command Mode
            LCD_DB<=8'b0011_1000;//Data-8Bit Mode,2 Rows,5x7 Font
        end
        write_mode:begin
            LCD_EN<=1'b0;//Neg
            LCD_RS<=1'b0;//Command Mode
            LCD_DB<=8'b0011_1000;//Data-8Bit Mode,2 Rows,5x7 Font
        end
        //----Display Command-----------
        setup_display:begin
            LCD_EN<=1'b1;
            LCD_RS<=1'b0;//Command Mode
            LCD_DB<=8'b0000_1110;//Display On,Cursor On,Blink Off
        end
        write_display:begin
            LCD_EN<=1'b0;//Neg
            LCD_RS<=1'b0;//Command Mode
            LCD_DB<=8'b0000_1110;//Display On,Cursor On,Blink Off
        end
        //----Clear Command-------
        setup_clear:begin
            LCD_EN<=1'b1;
            LCD_RS<=1'b0;//Command Mode
            LCD_DB<=8'b0000_0001;//Clear
        end
        write_clear:begin
            LCD_EN<=1'b0;//Neg
            LCD_RS<=1'b0;//Command Mode
            LCD_DB<=8'b0000_0001;//Clear
        end
        //----Postion Command-------
        setup_position:begin
            LCD_EN<=1'b1;
            LCD_RS<=1'b0;//Command Mode
            LCD_DB<=8'b1000_0000;//0x80+0
        end
        write_position:begin
            LCD_EN<=1'b0;//Neg
            LCD_RS<=1'b0;//Command Mode
            LCD_DB<=8'b1000_0000;//0x80+0
        end
        //----Data-------
        setup_data:begin
            LCD_EN<=1'b1;
            LCD_RS<=1'b1;//Data Mode
            LCD_DB<=romdata[datacnt];//Rom Data
        end
        write_data:begin
            LCD_EN<=1'b0;//Neg
            LCD_RS<=1'b1;//Data Mode
            LCD_DB<=romdata[datacnt];//Rom Data
        end

        default:begin
            LCD_EN<=1'b1;
            LCD_RS<=1'b0;//Command Mode
            LCD_DB<=8'd0;
        end
    endcase
end

endmodule
```

作業

作業題目：將u+學號7位數字，總共8個字元顯示在LCD上。

作業內容需包含：

1.題目

2.Verilog Code **& FSM Meely/Moore Machine** 圖**(**參考**Lab7)**

3.RTL View

4.TestBench Setting & Waveform Result