

4강. Text LCD

(또는 Character LCD)

박 원 업
010.5451.0113

목 차

1. Text LCD란?
2. Text LCD의 동작 원리
 - 4선 제어 방식 (진행할 방식)
 - 8선 제어 방식
3. Text LCD 동작시키기
 - 회로 구성
 - 코드 작성
4. Text LCD에 변수 출력하기
 - 실수형 변수 출력

○ Text LCD란?

- 영대소문자, 숫자, 특수문자등 모든 ASCII 문자를 출력할 수 있는 가장 단순한 형태의 디스플레이 장치. 거의 모든 Text LCD의 구동방식이나 핀 맵이 표준화 되어있어서 다루기 쉬움. 디스플레이 단위는 5*8 또는 5*10 dot의 한 문자이며, 변수의 값을 확인하거나 간단한 수치를 모니터링 하는데 사용됨. 느린 반응 속도와 표시할 수 있는 정보가 적다는 한계가 있음.



○ Text LCD란?

- LCD 모듈의 내부는 컨트롤러, 드라이버, LCD 패널, 백라이트 유닛 등으로 구성되며 이 기능들이 일체형으로 제작됨.
- LCD 컨트롤러 내부에는 표시할 문자 데이터를 저장하는 DD RAM, 이들 문자 코드를 실제로 표시할 문자 폰트로 변환하는 CG ROM, 사용자 정의 문자를 저장하는 CG RAM 등의 메모리가 존재. 또 LCD 모듈을 제어하는데 사용되는 명령을 저장하는 명령 레지스터 및 이를 해독하는 Instruction Decoder가 있으며, 각 문자의 폰트를 쉬프트 레지스터에 의하여 수평 구동신호와 수직 구동신호로 변환하는 회로를 내장.

- Text LCD란?

- LCD를 제어하기 위해서는 Data가 오고 갈 때의 동작 타이밍을 이해하고 그에 맞게 MCU를 이용하여 신호들을 제어해 주어야 함.
- LCD를 제어하기 위해서는 Instruction Set에 대하여 이해하여야 하고 명령어를 전달하기위한 제어신호 및 실행 시간에 대해서 이해해야 함.

Text LCD의 핀 구조

- Text LCD는 제조사마다 약간 다를수 있지만, 대체로 표준화되어있어서 동일한 핀 구조를 가짐(글자, 라인 수에 관계없음). 총 16개의 핀으로 구성되어 있으며 기능은 다음과 같음.

PIN CONNECTIONS			
Pin	Symbol	Level	Function
1	Vss	0V	GND
2	Vdd	+5V	Power supply for logic
3	Vo	--	Operating voltage for LCD
4	RS	H/L	H : Data L : Instruction code
5	R/W	H/L	H : Read L : Write
6	E	H,H \Rightarrow L	Enable signal
7	DB0	H/L	In 8-bit bus mode, used as low order bidirectional data bus. In 4-bit bus mode, open these pins.
8	DB1	H/L	
9	DB2	H/L	
10	DB3	H/L	
11	DB4	H/L	In 8-bit bus mode, used as high order bidirectional data bus. In 4-bit bus mode, used as both high and low order data bus.
12	DB5	H/L	
13	DB6	H/L	
14	DB7	H/L	
15	LEDA	+5V	Power supply for LED backlight
16	LEDK	0V	

Text LCD의 핀 구조

- 데이터시트를 확인해보면 데이터 비트가 총 8개인 것을 확인할 수 있으나, 구동 시 4비트 모드와 8비트 모드 중 하나를 선택할 수 있음. 4비트 모드로 동작 시, 하위 4개 비트는 사용 안 함.

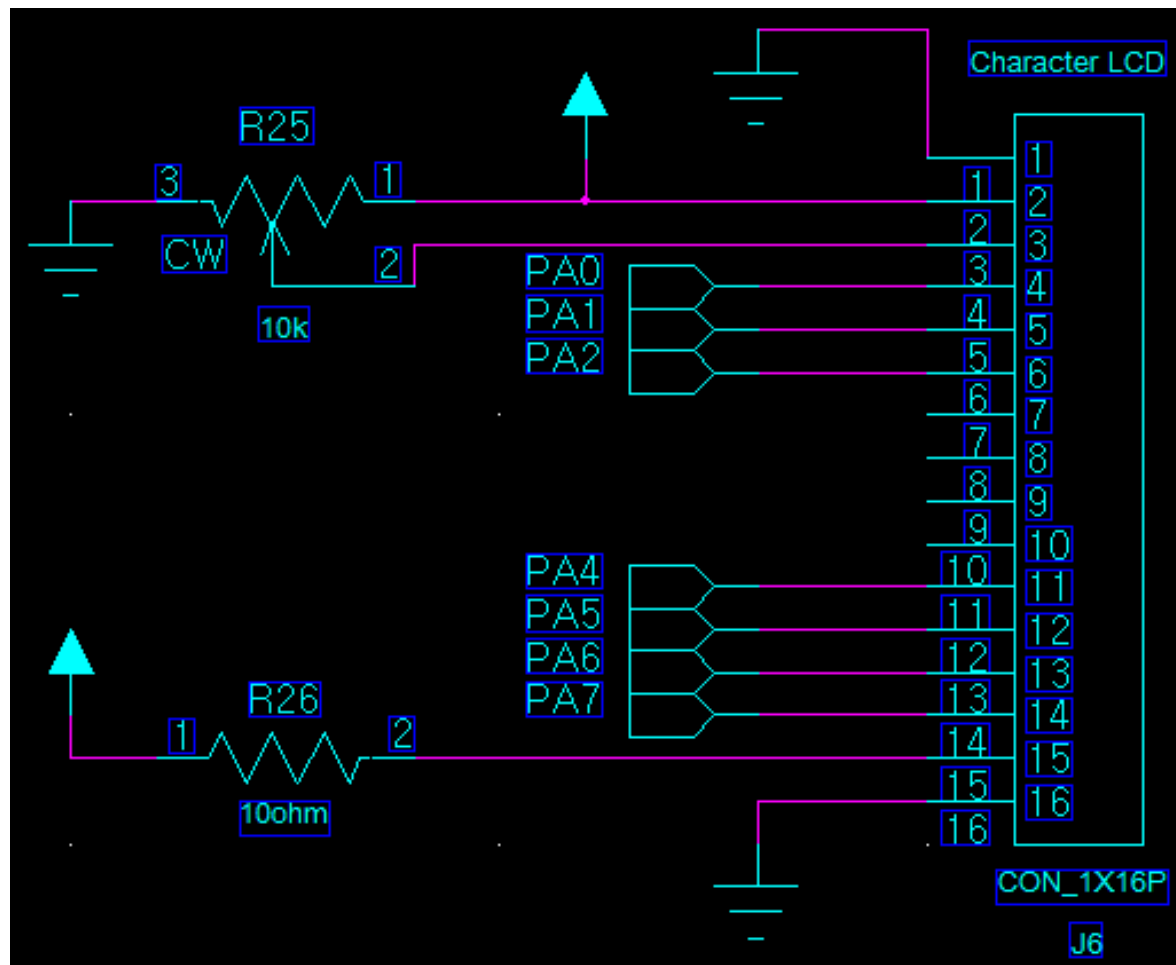
PIN CONNECTIONS			
Pin	Symbol	Level	Function
1	Vss	0V	GND
2	Vdd	+5V	Power supply for logic
3	Vo	--	Operating voltage for LCD
4	RS	H/L	H : Data L : Instruction code
5	R/W	H/L	H : Read L : Write
6	E	H, H \Rightarrow L	Enable signal
7	DB0	H/L	In 8-bit bus mode, used as low order bidirectional data bus. In 4-bit bus mode, open these pins.
8	DB1	H/L	
9	DB2	H/L	
10	DB3	H/L	
11	DB4	H/L	In 8-bit bus mode, used as high order bidirectional data bus. In 4-bit bus mode, used as both high and low order data bus.
12	DB5	H/L	
13	DB6	H/L	
14	DB7	H/L	
15	LEDA	+5V	Power supply for LED backlight
16	LEDK	0V	

4비트 모드 시,
사용하지 않는 핀

4강 Text LCD

- Text LCD의 핀 구조

- 따라서 다음과 같이 회로를 구성한다.

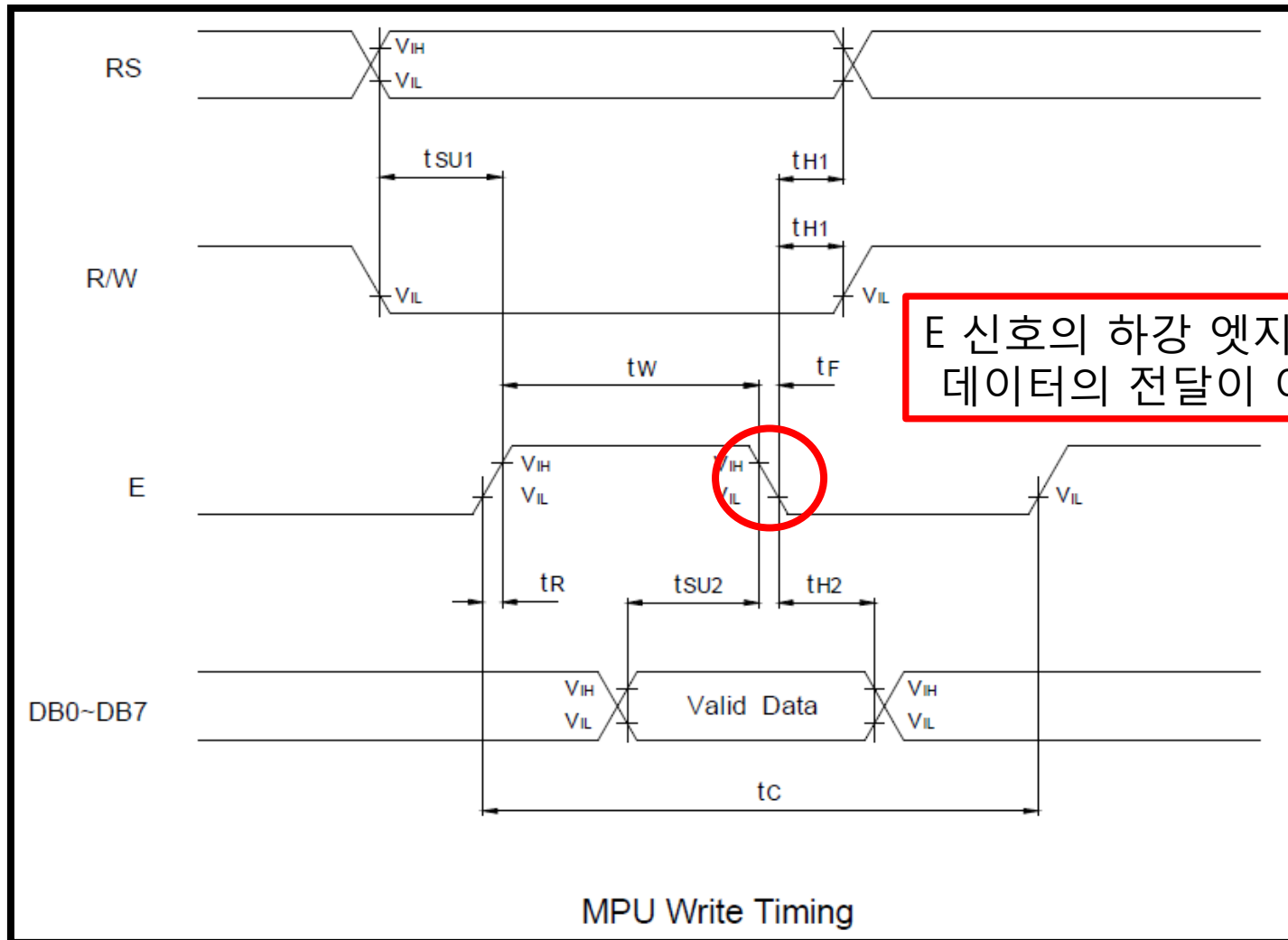


○ Text LCD의 타이밍도

- 역시 Text LCD는 제조사마다 약간 다를수 있지만, 대체로 표준화 되어있어서 비슷한 타이밍을 가짐.
- 타이밍도는 어떤 디지털 모듈 혹은 칩을 동작시키기 위해 꼭 필요한 특징이기 때문에 반드시 타이밍도를 정확히 이해해야 함.
(시간까지 정확히 이해하는 것이 좋음.)

4강 Text LCD

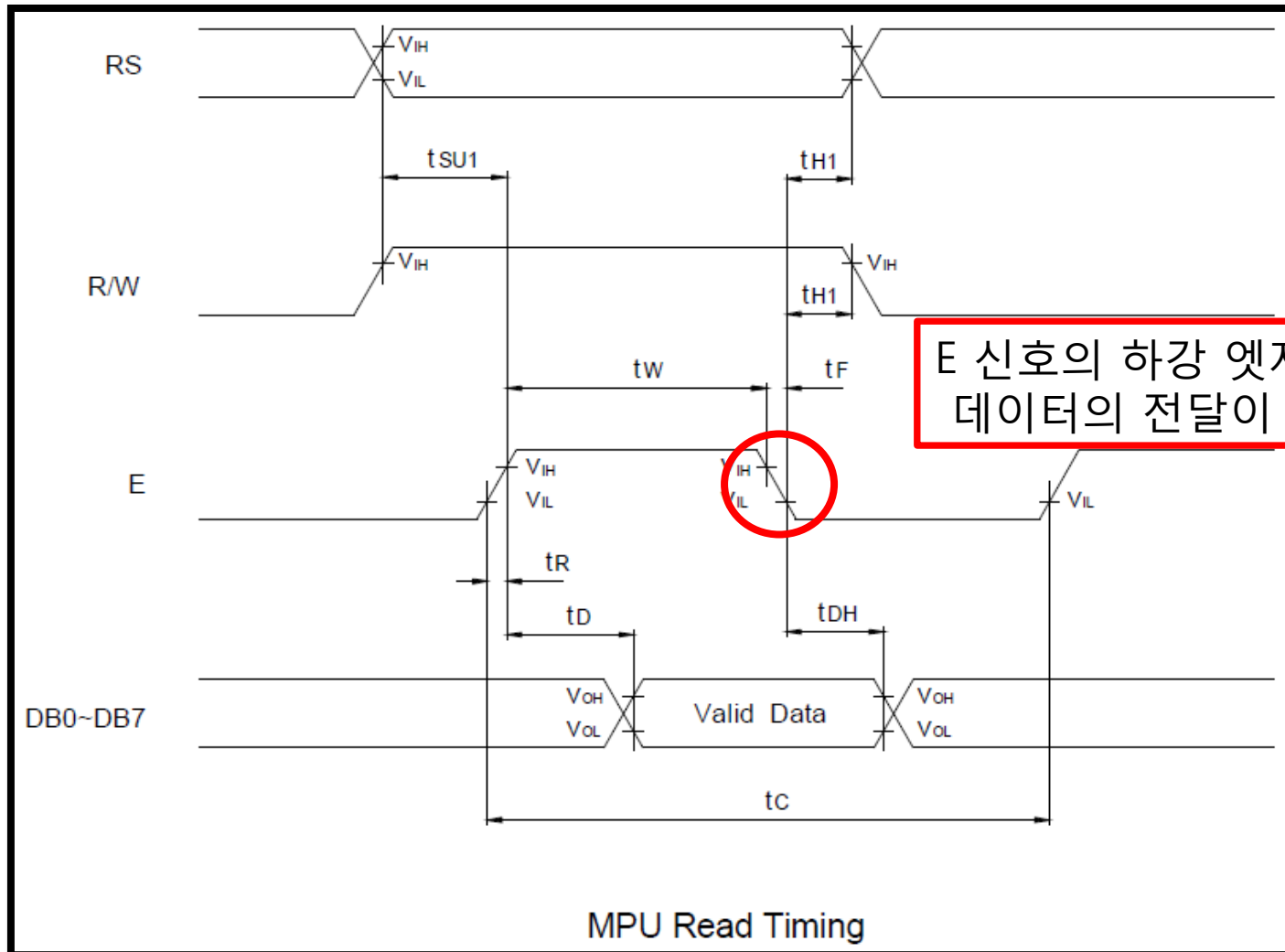
- Text LCD의 타이밍도
 - 쓰기 동작을 위한 타이밍도



E 신호의 하강 엣지에서
데이터의 전달이 이뤄짐!!

4강 Text LCD

- Text LCD의 타이밍도
 - 읽기 동작을 위한 타이밍도



- Text LCD의 타이밍도
 - 타이밍 기호와 시간을 나타낸 표

3.2 Interface Timing Chart			(VDD=5.0V±10%, Ta=25℃)			
Mode	Characteristic	Symbol	Min.	Typ.	Max.	Unit
Write Mode Refer to fig.1	E Cycle Time	t _c	500	--	--	ns
	E Rise/Fall Time	t _R , t _F	--	--	20	
	E Pulse Width (High,Low)	t _w	230	--	--	
	R/W and RS Setup Time	t _{SU1}	40	--	--	
	R/W and RS Hold Time	t _{H1}	10	--	--	
	Data Setup Time	t _{SU2}	80	--	--	
	Data Hold Time	t _{H2}	10	--	--	
Read Mode Refer to fig.2	E Cycle Time	t _c	500	--	--	ns
	E Rise/Fall Time	t _R , t _F	--	--	20	
	E Pulse Width (High,Low)	t _w	230	--		
	R/W and RS Setup Time	t _{SU}	40	--		
	R/W and RS Hold Time	t _H	10	--		
	Data Output Delay Time	t _D	--	--	120	
	Data Hold Time	t _{DH}	5	--	--	

4강 Text LCD

Text LCD를 동작시키기 위해서 가장 먼저 해야할 일은 **Text LCD를 초기화** 시키는 일이다. 초기화 과정은 데이터 시트에 나와있으므로 데이터 시트 참조.

데이터 시트 12 페이지 부터.

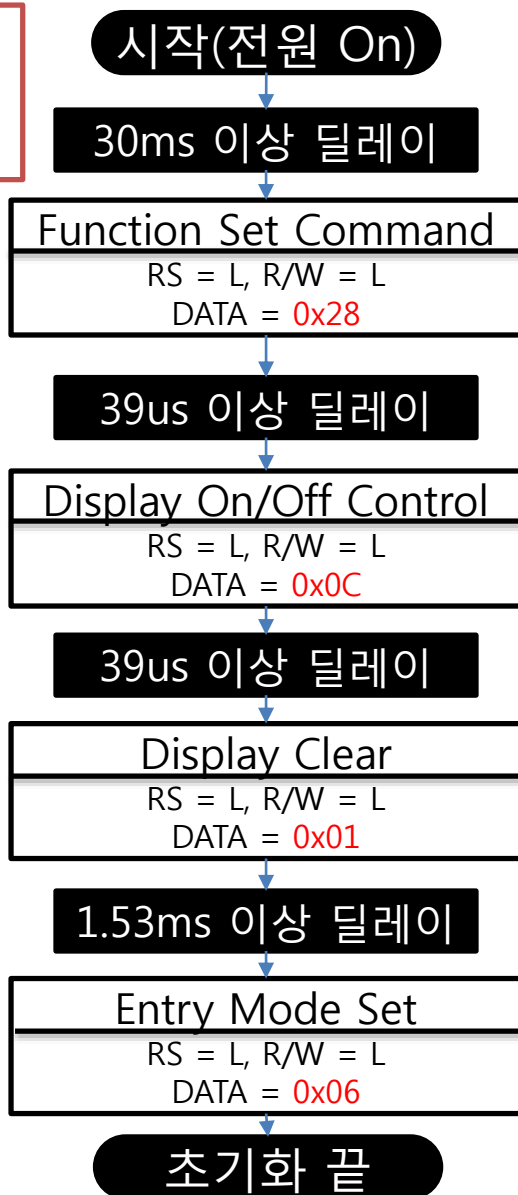
초기화 시키기 위해선 전원 인가 후, 몇 가지 명령어를 LCD에 입력시켜야 한다. 8bit 모드와 4bit 모드의 초기화 명령어가 다르므로, 이에 유의하여 프로그램을 작성해야 함.

데이터 시트의 명령어 부분을 참조.

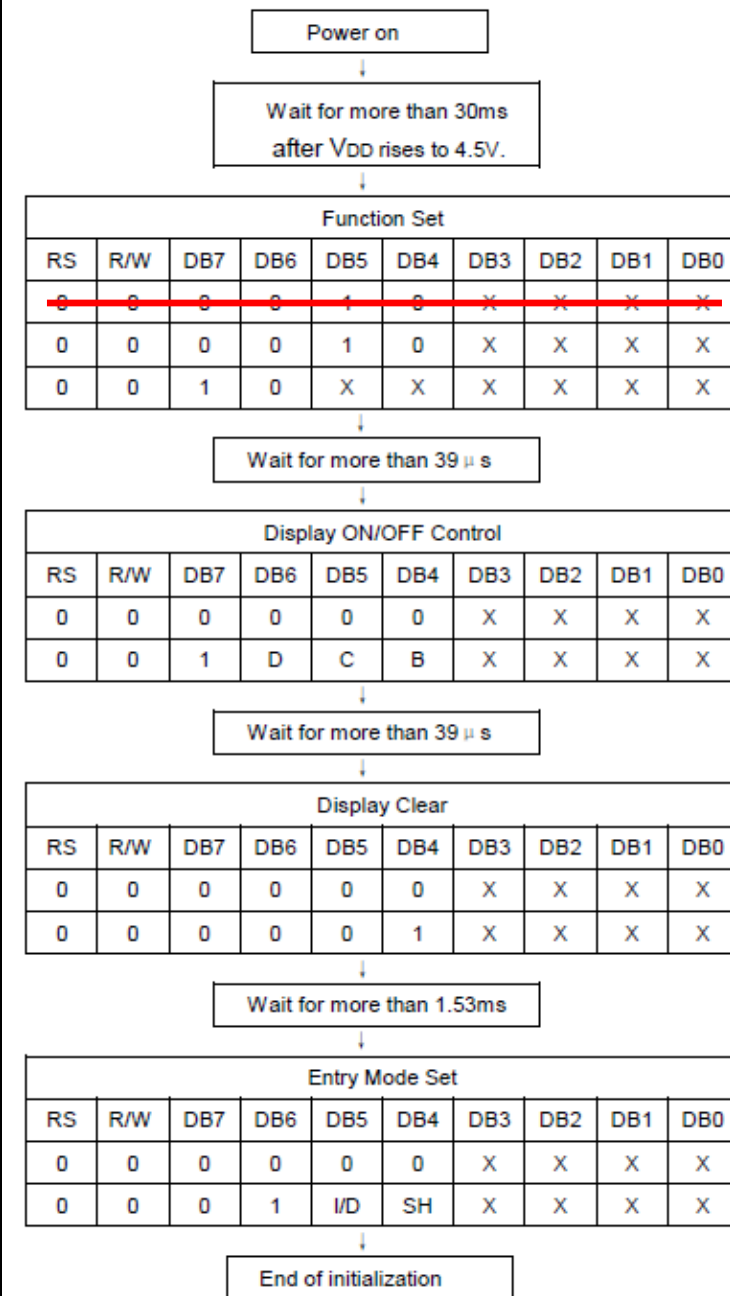
4강 Text LCD

4비트 제어 초기화 과정 (데이터시트 14페이지)

여기서 주의해야 할 점은
상위 4비트를 먼저 전송하고
하위 4비트를 나중에 전송하는
것이다!



4-Bit Initialization:



- Text LCD의 Instruction Set

- Text LCD를 제어하기 위한 명령어들. 데이터시트 16페이지 부터.

Table 6.1 Instructions												
Instruction	Instruction code										Description	Execution time (fosc=270KHz)
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
Clear Display	0	0	0	0	0	0	0	0	0	1	Clears entire display and sets DDRAM address to 00H.	1.53ms
Return Home	0	0	0	0	0	0	0	0	1	-	Sets DDRAM address to 00H in AC and returns shifted display to its original position. The contents of DDRAM remain unchanged.	1.53ms
Entry Mode Set	0	0	0	0	0	0	0	1	I/D	SH	Sets cursor move direction and enable the shift of entire display. These operations are performed during data write and read.	39 μ s
Display ON/OFF Control	0	0	0	0	0	0	1	D	C	B	Set ON/OFF of entire display (D), cursor ON/OFF(C), and blinking of cursor position character(B).	39 μ s
Cursor or Display Shift	0	0	0	0	0	1	S/C	R/L	-	-	Moves cursor and shifts display without changing DDRAM contents.	39 μ s

Text LCD의 Instruction Set

- Text LCD를 제어하기 위한 명령어들. 데이터시트 16페이지 부터.

Function Set	0	0	0	0	1	DL	N	F	-	-	Sets interface data length (DL: 8-bit/4-bit), numbers of display line (N: 2-line/1-line), and display font type (F: 5x11dots/5x8dots)	39 μ s
Set CGRAM Address	0	0	0	1	AC5	AC4	AC3	AC2	AC1	AC0	Set CGRAM address in address counter.	39 μ s
Set DDRAM Address	0	0	1	AC6	AC5	AC4	AC3	AC2	AC1	AC0	Set DDRAM address in address Counter.	39 μ s
Read Busy Flag and Address	0	1	BF	AC6	AC5	AC4	AC3	AC2	AC1	AC0	Reads busy flag (BF) indicating internal operation is being performed and reads address counter contents.	0 μ s
Write data to CG or DD RAM	1	0	D7	D6	D5	D4	D3	D2	D1	D0	Write data into internal RAM (DDRAM/CGRAM).	43us
Read data from CG or DD RAM	1	1	D7	D6	D5	D4	D3	D2	D1	D0	Read data from internal RAM (DDRAM/CGRAM).	43us

“-” : don't care

4강 Text LCD

위의 Text LCD 초기화 과정에 맞춰 초기화 신호를 보내주면 동작모드로 들어가고 LCD가 Clear된다.

그럼 Text LCD에 신호를 보내는 방법은?

- 위의 타이밍도에서 E 신호의 하강엣지에서 데이터의 전달이 이루어진다. 따라서 RS, R/W, DATA(8bits) 신호들을 미리 넣어준 후, 마지막에 E 신호에 클럭을 한번 넣어주면 된다.

다음 샘플 코드를 보면 이해가 갈 것이다. (8bit 모드로 동작 시, CodeVisionAVR사용)

```
void instruction_out(unsigned char b)
{
    CLCD_D7 = (b >> 7) & 0x01;
    CLCD_D6 = (b >> 6) & 0x01;
    CLCD_D5 = (b >> 5) & 0x01;
    CLCD_D4 = (b >> 4) & 0x01;
    CLCD_D3 = (b >> 3) & 0x01;
    CLCD_D2 = (b >> 2) & 0x01;
    CLCD_D1 = (b >> 1) & 0x01;
    CLCD_D0 = (b >> 0) & 0x01;

    CLCD_RS = 0;
    CLCD_RW = 0;
    CLCD_EN = 0;

    CLCD_EN = 1;
    CLCD_EN = 0;
}
```

4강 Text LCD

이제 RS, RW, EN, D0~D7에 해당하는 핀을 실제 ATmega128 핀에 매칭시켜 주어야 한다.

예를 들면,

```
#define CLCD_RS PORTA.0
#define CLCD_RW PORTA.1
#define CLCD_EN PORTA.2
#define CLCD_D7 PORTA.7
#define CLCD_D6 PORTA.6
#define CLCD_D5 PORTA.5
.
.
.
.
```

CodeVisionAVR에서는 위와 같이 매칭시켜줄 수 있다.
(다른 IDE 환경에서는 그에 맞는 매크로함수 등으로 정의하면 된다.)

4강 Text LCD

4bit 모드로 동작시키기 위해서는 8bit의 데이터를 상위 4bit와 하위 4bit 둘로 나눠서 데이터를 보내주어야 한다.

먼저 상위 4bit를 보낸 후, 그 다음 하위 4bit를 보내준다.

다음 예제코드를 보면 이해하기 쉽다. **명령어 전송 함수**의 예이다.
(4bit 모드로 동작 시, CodeVisionAVR사용)

상위 4bit 전송

하위 4bit 전송

```
void instruction_out(unsigned char b)
{
    CLCD_D7 = (b >> 7) & 0x01;
    CLCD_D6 = (b >> 6) & 0x01;
    CLCD_D5 = (b >> 5) & 0x01;
    CLCD_D4 = (b >> 4) & 0x01;
    CLCD_RS = 0;
    CLCD_RW = 0;
    CLCD_EN = 0;

    CLCD_EN = 1;
    CLCD_EN = 0;

    CLCD_D7 = (b >> 3) & 0x01;
    CLCD_D6 = (b >> 2) & 0x01;
    CLCD_D5 = (b >> 1) & 0x01;
    CLCD_D4 = b & 0x01;
    CLCD_RS = 0;
    CLCD_RW = 0;
    CLCD_EN = 0;

    CLCD_EN = 1;
    CLCD_EN = 0;
}
```

4강 Text LCD

초기화 과정이 끝났으면 문자를 출력해보자.
문자를 출력하기 위해서는 RS신호만 H로 바꿔주면 된다.

다음 예제코드를 보면 이해하기 쉽다. **문자 전송 함수**의 예이다.
(4bit 모드로 동작 시, CodeVisionAVR사용)

상위 4bit 전송

하위 4bit 전송

```
void char_out(unsigned char b)
{
    CLCD_D7 = (b >> 7) & 0x01;
    CLCD_D6 = (b >> 6) & 0x01;
    CLCD_D5 = (b >> 5) & 0x01;
    CLCD_D4 = (b >> 4) & 0x01;
    CLCD_RS = 1;
    CLCD_RW = 0;
    CLCD_EN = 0;

    CLCD_EN = 1;
    CLCD_EN = 0;

    CLCD_D7 = (b >> 3) & 0x01;
    CLCD_D6 = (b >> 2) & 0x01;
    CLCD_D5 = (b >> 1) & 0x01;
    CLCD_D4 = b & 0x01;
    CLCD_RS = 1;
    CLCD_RW = 0;
    CLCD_EN = 0;

    CLCD_EN = 1;
    CLCD_EN = 0;
}
```

4강 Text LCD

문0) 정수, 실수, 문자형 변수를 LCD에 출력. Sprintf() 함수를 사용.

문1) 타이머 기능을 사용하여 현재 시간을 시-분-초 단위로 Text LCD로 출력

문2) LCD 첫 번째 줄

- 1번 스위치를 누르면 시간-분-초 순서로 선택. 2번 스위치를 누르면 선택된 숫자를 1씩 증가. (표시할 수 있는 숫자를 넘어서면 다시 0부터 시작)

LCD 두 번째 줄

- 현재 선택된 것을 표시(시간, 분, 초 중 하나를 표시)