

CLCD Series



1. 제품의 개요

본 제품은 시리얼 방식으로 LCD를 제어할 수 있는 “시리얼 영문 LCD 모듈”입니다. 기존의 LCD모듈은 12가닥 정도의 많은 선과 복잡한 제어 방식을 이해해야만 사용할 수 있는 “병렬접속” 방식이었지만, 본 제품은 단 한 선으로 시리얼 방식의 데이터를 송신하여, LCD상에 글자를 표현할 수 있는 방식으로 LCD 모듈에 대한 구체적인 이해 없이도 손쉽게 사용할 수 있도록 만든 제품입니다.

2. 제품의 특징

- 16 by 2라인부터 20 by 4라인까지 다양한 사이즈
- UART 또는 I2C(CUNET) 인터페이스 방식
- 위치조정(LOCATE) 및 화면 클리어(CLS), 커서 관리(CSR ON, OFF)기능 내장

3. 제품의 종류

모델명	표시 가능 수	비탕색
CLCD216-G	16 by 2	녹색
CLCD420-B	20 by 4	파란색
CLCD420-G	20 by 4	녹색
CLCD420-BIG-G	20 by 4	녹색

* 대량 구입시 가격은 본사로 문의하시기 바랍니다. 대량구입시 LCD를 제외한 콘트롤 모듈만 구입하실 수 있습니다.

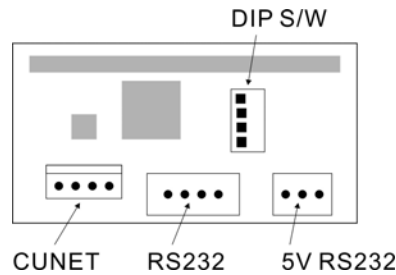
* I2C 통신은 2가닥 통신 프로토콜입니다. SCL, SDA 단자를 사용하며, CUBLOC에서는 CUNET 이라고도 부릅니다.
CUNET = I2C 같은 뜻입니다.

4. 제품의 외형



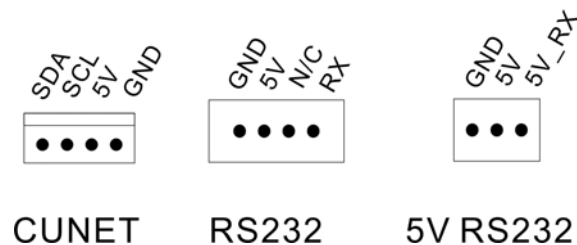
[뒷면]

CLCD 모듈의 뒷면에는 콘트롤보드가 부착되어 있습니다. 이 콘트롤 보드가 CUNET 신호를 받아서 LCD 모듈을 구동합니다.










5. 연결방법

CLCD는 CUNET통신 이외에도 RS232신호를 받을 수 있도록 되어 있습니다. RS232에는 2종류의 콘넥터가 있는데, 3핀 콘넥터 5V레벨의 RS232신호를 연결하는 콘넥터입니다. PC와 직접 연결하고자 할 때에는 +/- 12V 레벨의 RS232신호를 연결하는 4핀 콘넥터를 사용합니다.



덱스위치로 보레이트 및 I2C 슬레이브 어드레스를 설정할 수 있습니다. DIP스위치 중 4번 스위치는 사용하지 않는 스위치입니다.

DIP 스위치 상태	RS232 보레이트	I2C 의 SLAVE ADDRESS
ON 	2400	0
ON 	4800	1
ON 	9600	2
ON 	19200	3
ON 	28800	4
ON 	38400	5
ON 	57600	6
ON 	115200	7

6. 동작방법

CUNET과 RS232는 동시에 동작하도록 되어 있으나, 실제로 사용할 때에는 둘 중 하나만 연결해서 사용하시기 바랍니다.

다음은 CLCD의 명령코드 표입니다.

명령	전송예 (16 진)	바이트 수	수행시간	설 명
ESC 'C'	1B 43	2	15mS	화면을 클리어 합니다. 이 명령 수행 후 반드시 15mS 를 딜레이 해주어야 합니다.
ESC 'S'	1B 53	2		커서표시를 ON 합니다.(디폴트)
ESC 's'	1B 73	2		커서표시를 OFF 합니다.
ESC 'B'	1B 42	2		백라이트를 ON 합니다. (디폴트)
ESC 'b'	1B 62	2		백라이트를 OFF 합니다.
ESC 'H'	1B 48	2		LOCATE 0,0 위치로 갑니다.
ESC 'L'	1B 4C x y	4	100 uS	커서위치를 변경합니다.
ESC 'D'	1B 44 code 8byte	11		캐릭터코드 8 부터 15 까지 8개 영역은 유저가 정의할 수 있는 영역입니다. Code 에 8 부터 15 사이의 코드를 하나 선택하시고, 이어서 8개의 데이터를 보내주십시오. 이 코멘드를 수행한 후에는 LOCATE 0,0 명령을 반드시 수행하여 주십시오.
ESC 'K'	1B 4B X Y X Y data 0			문자를 표시합니다. 같은 X, Y 좌표를 두번 보냅니다. 이후 문자코드를 보냅니다. 맨끝에는 0 을 보냅니다.

7. CUBLOC에서의 사용방법

다음은 CB280모듈에서 CLCD모듈을 CUNET으로 접속하여 표시하는 샘플 프로그램입니다. 본 프로그램을 실행시키면 LCD상에 숫자가 증가되는 상황을 볼 수 있습니다.

```
Const Device = Cb280
Set Display 2,0,0,50
Dim i As Integer
Delay 100           ' 최초 딜레이 (CLCD 모듈이 START UP 하는 시간을 기다려줌)
Cls
Delay 200           ' CLCD 가 클리어 하는데 걸리는 시간을 기다림
Csroff
Clcdout 5,2, "Start!!!"
Delay 500
Cls
Delay 100
Do
    Incr i
    Clcdout 0,0, "COMFILE"
    Clcdout 1,3 "CUBLOC ",Dec i
    delay 100
Loop
```

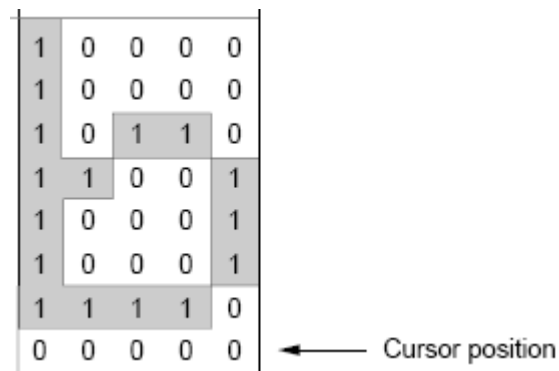
- CLCDOUT 은 좌표와 함께 캐릭터를 표시하는 명령입니다. (2011 년 1 월 이후출시분부터 포함)
- CLCDOUT 은 CUBLOC STUDIO V3.1.2 이후 버전부터 지원합니다.

8. 캐릭터정의 사용방법

CLCD에는 유저정의 캐릭터를 사용할 수 있습니다. 화살표등과 같은 특수문자를 유저가 임의대로 정의하여 사용할 수 있습니다. 코드8부터 15까지 총 8개의 캐릭터를 정의하여 사용할 수 있습니다.

```
Const Device = CB320
Set Display 2,0,0,80
Wait 100
Cls
Wait 100
Print 0x1b,"D",8,0k,0,0,0,15,15,0,0,0    '8 번 캐릭터를 정의
Print 0x1b,"D",9,0,0x10,0x18,0x1c,0x1c,0x18,0x10,0    '9 번 캐릭터를 정의
Locate 0,0 ' 정의가 끝나면 반드시 Locate 명령어를 수행해야합니다.
Print "abc"
Print 8,9    '정의한 캐릭터를 표시
Do
Loop
```

정의가 끝난후에는 해당 코드를 보내면, 정의된 캐릭터가 표시됩니다.



위 그림처럼 5 x 8 캐릭터를 정의하실 수 있습니다, 실제로는 5x7 만 사용할 수 있습니다. 8번째 줄은 커서가 위치해야되므로, 반드시 비워두시기 바랍니다.

8. 인터럽트 상황에서의 LOCATE 사용

기존 LOCATE, PRINT를 이용하는 표시방법으로는 인터럽트 상황에서 LCD표시화면이 깨지는 현상이 발생할 수 있습니다. 왜냐하면, 메인 루틴에서 LOCATE명령어를 실행한 직후, 인터럽트가 발생되어 다른 LOCATE로 지정되는 상황이 생기기 때문입니다.

```
Do
    Locate 0,0
    ' ← 이 사이에서 인터럽트 발생될 수 있습니다.
    Print "Comfile"
    Delay 100
Loop

OtherInt:
    Locate 1,1    ' 인터럽트 루틴을 실행하면 locate 를 건드리게 됩니다.
    Print "Technology"
    Return
```

이러한 상황을 피하기 위해서는 CLCDOUT명령어로 위치까지 동시에 지정하는 방법을 사용해야 합니다.

CLCDOUT 3,2, "COMFILE"

이렇게 하면 명령이 다 실행된 뒤 인터럽트가 발생되므로, 화면표시위치가 흐트러지는 일이 발생되지 않습니다. 위 프로그램을 아래처럼 변경하면, 인터럽트 루틴실행과 상관없이 항상 일정한 위치에 캐릭터가 표시됩니다.

```
Do
    Clcdout 0, 0, "Comfile"
    Delay 100
Loop

OtherInt:
    Clcdout 1, 1, "Technology"
    Return
```

또한, CLCDOUT은 일정한 포맷을 갖춘 명령이므로, 노이즈로 인해 엉뚱한 캐릭터가 LCD상에 표시되는 것도 막아줍니다. 일단 CLCDOUT 명령어를 사용한이후에는 계속 CLCDOUT명령만 사용해야합니다. 다시 LOCATE, PRINT를 사용하는 방법으로 돌아갈 수 없습니다. 왜냐하면 노이즈로 판단해서 표시를하지 않기 때문입니다.

따라서, LOCATE 와 PRINT명령어를 사용하실 분은 CLCDOUT 명령어를 단 한번이라도 사용하지 마십시오.
CLCDOUT명령어를 사용하실 분은 PRINT명령어를 사용하지 않으면됩니다.

* LOCATE 와 PRINT명령은 기존에 만들어두신 제품에 지속적으로사용하시려는 분들위해 계속 지원되지만, 새롭게 프로그램을 작성하는 분들이라면 CLCDOUT 명령어를 사용하시기 바랍니다.

9. MCU 에서 C 언어를 이용하여 CLCD 사용

다른 MCU, 예를들면 PIC, AVR 등에서도 CLCD 를 사용하실 수 있습니다.

다음은 CCS-C (CCS 사의 PIC 전용 C 컴파일러)를 사용해서 구동한 예입니다. (디바이스는 PIC16C711, 클록은 4MHz 사용, RS232 전송 방법)

```
#include <16c711.h>

#use delay(clock=4000000)    // 4MHz 사용 시
#use rs232(baud = 19200, xmit = PIN_B1, rcv=PIN_B0)

#byte TRISB = 0x85
#byte PORTB = 5

void main()
{
    Char i;
    TRISB = 1;

    delay_ms(200);           // lcd 모듈이 초기화하는 시간을 기다림
    printf("%c%c",0x1b,0x43); // cls
    delay_ms(200);           // lcd모듈이 클리어하는 시간을 기다림
    while(1) {
        printf("%c%c%c%c",0x1b,0x4c,0,0);    // locate 0,0
        printf("COMFILE SERIAL ");
        printf("%c%c%c%c"0x1b,0x4c,0,1);    // locate 0,1
        printf("LCD MODULE %d ",i);
        i++;
    }
}
```