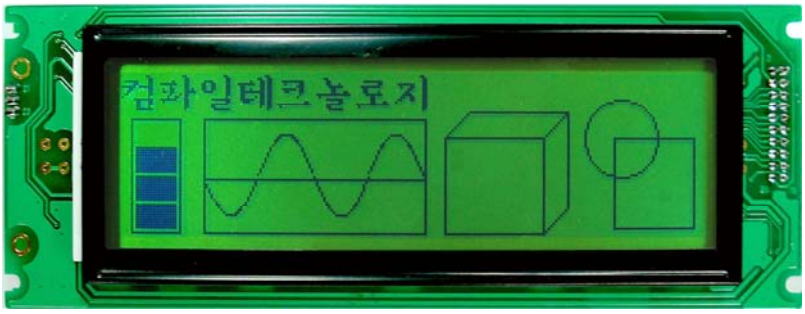
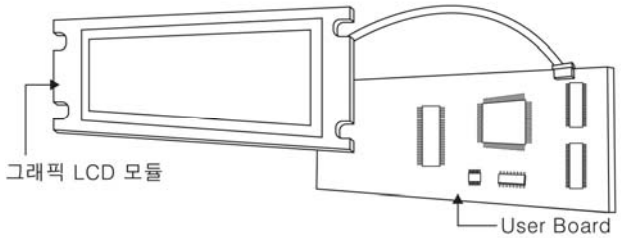
그래픽 한글 LCD 디스플레이 모듈 사용설명서

GHA-246A

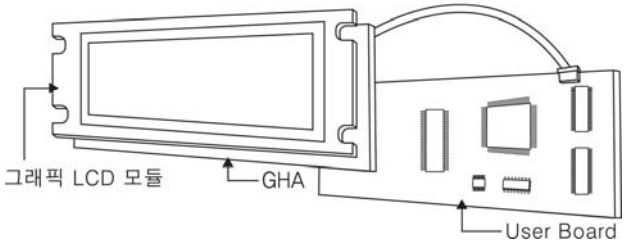


1. 제품의 개요

GHA는 간단한 시리얼 접속만으로 LCD상에 그래픽과 한글을 표시할 수 있는 제품입니다. 보통의 경우 아래 그림처럼 유저 보드에서 직접 그래픽 LCD를 제어하는 방식을 채택해야만 합니다. 이 경우 프로그래머는 그래픽LCD를 콘트롤 하기 위해 그래픽LCD 콘트롤러를 공부 해야 하고, 그래픽 및 한글을 표시하기 위해 많은 시간을 투자 해야만 합니다. 특히 한글 폰트 작성은 많은 부담이 따르는 작업이 아닐 수 없습니다.



GHA는 이러한 그래픽LCD 사용상의 여러 가지 불편함을 해소시켜 주는 제품입니다. 아래 그림처럼 그래픽 LCD와 유저 보드 사이에 GHA를 사용함으로써, 유저는 그래픽LCD에 관한 사항을 따로 공부할 필요 없이 단순한 명령만으로 그래픽과 한글, BMP파일 표시등의 결과를 얻을 수 있습니다. GHA는 각종 그래픽 함수와 한글 폰트, BMP파일 저장을 위한 플레쉬 영역 등이 있어서 프로그래머의 작업을 쉽게 도와 드립니다. 즉 본 제품은 사용자보드와 그래픽 LCD사이에서 여러 가지 다양한 일을 수행해 주는 "그래픽 LCD 콘트롤 엔진"이라고 볼 수 있습니다.



GHA는 그래픽 및 문자를 표시하기 위해서, RS232 및 I2C 연결이 필요합니다. 연결 케이블을 통해서 일련의 명령과 데이터를 전송해준다면, LCD상에는 그래픽 및 문자가 표시될 것입니다.

GHA에 데이터를 전송해주는 HOST 프로세서는 PC및 CUBLOC, 마이컴 (RS232 별도 부착 후) 등이 선택될 수 있습니다. 만약 여러분이 설계하고 있는 제품에 그래픽 LCD를 추가하고 싶을 때, 프로세서를 바꾸실 필요 없이 RS232 및 I2C 포트 하나만 할당해주는 것으로 간단히 그래픽LCD를 구동할 수 있다는 것을 의미합니다.

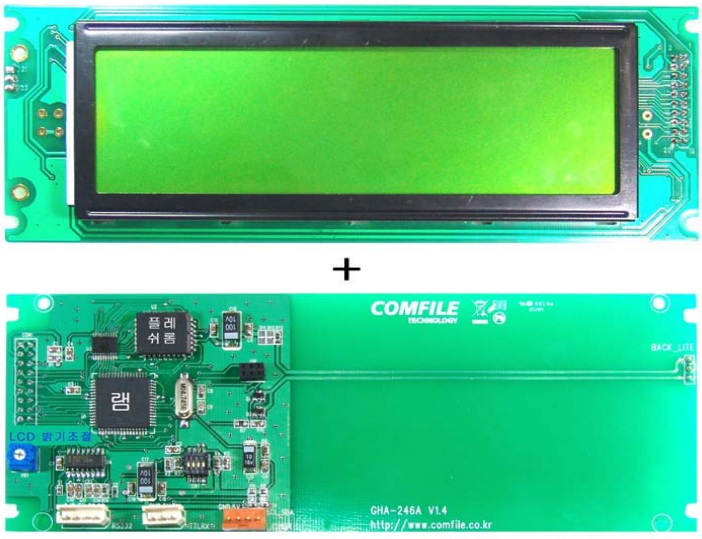
2. 제품의 주요 기능

1) 점, 선, 원, 타원, 박스, 호 등의 그래픽 표시
2) 한글, 영문, 숫자, 기호, 특수문자 표시
3) 한글 사용 시 조합형 및 완성형 코드 선택
4) 한글 (16x16)및 ASII 코드(8x16) 각각 2폰트 내장 (고딕, 명조)
5) 커서(밑줄형, 박스형), 글자 반전, 글자 밑줄
6) 폰트 사이즈 조절 (가로 2배, 세로 2배, 가로세로 2배)
7) 특수문자를 사용자가 만들어서 사용 가능
8) 그림(BMP) 파일 표시 (240x64 Size 115장 까지 가능, 245Kbyte)
9) 소프트웨어적으로 Backlight ON/OFF 가능
10) RS232 통신속도 선택 가능 (4800 ~ 115200bps)
11) I2C 어드레스 선택 가능 (0 ~ 7)

3. 제품 사양

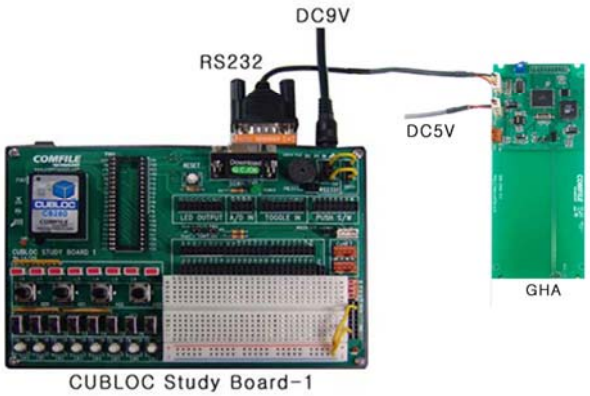
1) 전원: DC5V / 390mA (Backlight OFF: 80mA)
2) 사용 온도: 0 ~ 50℃
3) 사용 습도: 35~85 %RH (단, 결로하지 않을 것)
4) 외부 인터페이스: ±12V RS232, 5V RS232, 5V CuNET(I2C)
5) 플레쉬 메모리 수명: 10,000 회

4. 제품 구성

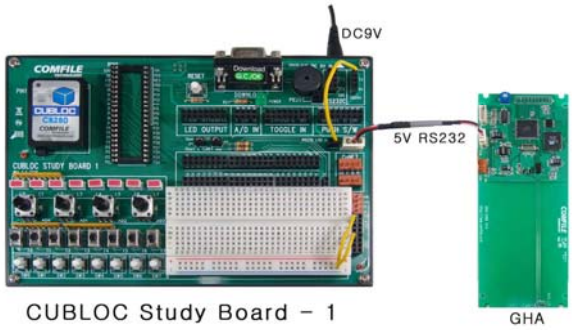


5. 제품 연결 방법

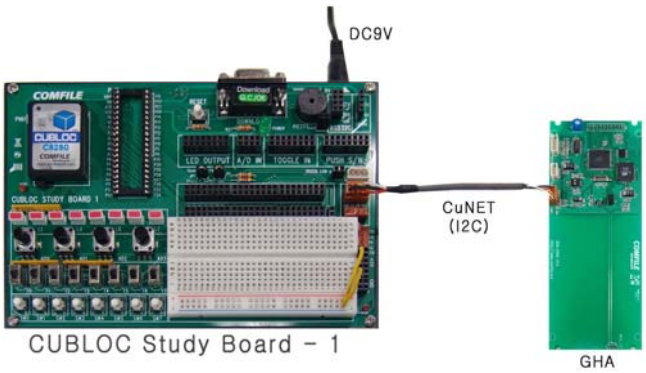
1) 12V RS232 (MAX232에 연결)



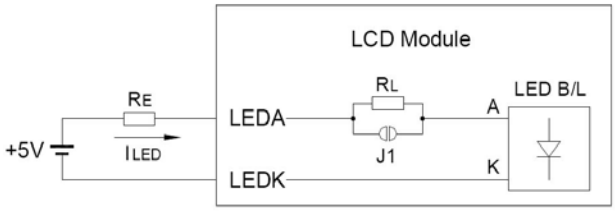
2) 5V RS232 (ATmega128에 직접 연결)

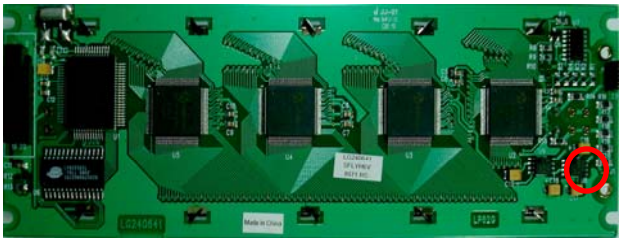


3) CuNET(I2C)



6. LED Backlight





GHA에서 Re값을 설정해 놓았기 때문에 반듯이 LCD 모듈에 있는 J1을 연결 시켜야 합니다.
(* J1을 연결 하지 않을 시 백라이트 밝기가 어두워집니다)

7. 스위치 설정 방법

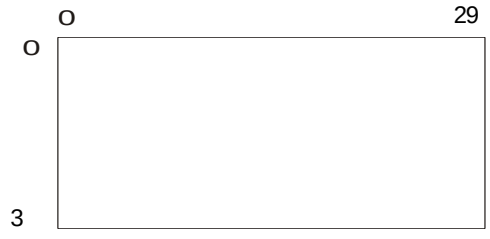
SW1, SW2, SW3 (O=ON, X=OFF)	SW4=OFF (RS232) RS232 통신속도	SW4=ON (I2C) I2C 어드레스
X, X, X	19200	0
X, X, O	4800	1
X, O, X	9600	2
X, O, O	19200	3
O, X, X	28800	4
O, X, O	38400	5
O, O, X	57600	6
O, O, O	115200	7

8. 좌표계 설명

텍스트와 그래픽을 표시 시 서로 다른 좌표 계를 사용합니다. 텍스트 에서는 문자(8x16) 단위, 그래픽에서는 도트(1x1) 단위입니다.
(* 유효 범위를 넘는 좌표를 지정할 경우 엉뚱한 위치에 표시되거나, 표시가 안 되는 경우가 있으므로 주의하시기 바랍니다.)

▌텍스트 화면의 좌표계

텍스트 좌표계는 문자 단위이며 X좌표가 0~29, Y좌표가 0~3까지 유효 합니다. 영문, 숫자, 기호의 크기는 8x16 크기이므로 가로 30자, 세로 4자를 표시할 수 있고 특수문자,한글의 크기는 16x16 이므로 가로 15자, 세로 4자를 표시할 수 있습니다.



▌그래픽 화면의 좌표계

그래픽 좌표계는 도트 단위이며 X좌표가 0~239, Y좌표가 0~63까지 유효합니다.

9. 제어 명령 설명

GHA의 제어 명령은 ESC코드 (HEX 1B)를 선두로, 2개의 영문자를 보내 동작을 지시한 뒤, 그에 필요한 각종 인수를 보내는 형식으로 되어 있습니다. 잘못된 명령을 보냈을 시 잘못된 명령 코드를 ASII로 화면에 표시합니다. (ex. Error : ESC,D,L)

▌CUBLOC에서 사용하기 위한 설정 방법

Set Display n1, n2, n3, n4

- 인수 설명 (n1): 1=GHLCD
- 인수 설명 (n2): 0=CuNET(I2C), 1=RS232 CH1, 2=RS232 CH2, 3=RS232 CH3
- 인수 설명 (n3): RS232 통신속도 및 CuNET(I2C) 어드레스
- 인수 설명 (n4): 송신버퍼 크기
- 전송 예: RS232 CH1을 이용해서 Set Dislay 1, 1, 19200, 200

▌커서 타입 설정

ESC	CM	n
1 바이트	2 바이트	1 바이트

- 인수 설명(n): 0=박스형, 1=밑줄형
- 동작 설명: 커서의 모양을 결정합니다.
- 전송 예: 커서 박스 타입
  (16진수) 1B 43 4D 00
  (CUBLOC) Cmode 0

▌커서 ON/OFF

ESC	CS	n
1 바이트	2 바이트	1 바이트

- 인수 설명(n): 0=OFF, 1=ON
- 동작 설명: 커서를 ON/OFF 합니다.
- 전송 예: 커서 표시
  (16진수) 1B 43 53 01
  (CUBLOC) Csron (Csroff : 커서 표시하지 않음)

▌BACK LIGHT ON/OFF

ESC	BL	n
1 바이트	2 바이트	1 바이트

- 인수 설명 (n): 0=OFF, 1=ON
- 동작 설명: LCD 백 라이트를 ON/OFF 합니다.
- 전송 예: 백라이트 ON
  (16진수) 1B 42 4C 01
  (CUBLOC) Light 1

지우기

ESC	CL	n
1 바이트	2 바이트	1 바이트

- 인수 설명 (n): 0= 전체 지움, 1= 부분 지움, 2= 지우지 않음
- 동작 설명: 전체 및 부분 화면을 지웁니다.
- 전송 예 1: 전체 화면 지움

  (16진수) 1B 43 4C 00

  (CUBLOC) Clear 0 (또는 CLS)

  * 이 명령 사용 후 약 200ms 지연 시간 필요
- 전송 예 2: 부분 지움

```
(CUBLOC) Clear  1                    ' 이 명령 사용 후
         Boxfill  0,0,100,100,0      ' 그래픽 명령을 사용 시 그려지지 않고 지워짐
         Clear  2                    ' 지우지 않음으로 다시 설정
```

(* 모든 그래픽 명령을 사용해서 지울 수 있음)

한글 코드 설정

ESC	CD	n
1 바이트	2 바이트	1 바이트

- 인수 설명 (n): 0=완성형(KS5601)코드, 1=조합형 코드
- 동작 설명: 한글을 사용 시 코드 방식을 결정합니다.
- 전송 예: 조합형 코드 선택

  (16진수) 1B 43 44 01

  (CUBLOC) Print 27, 67, 68, 1

폰트 설정

ESC	FO	n1	n2
1 바이트	2 바이트	1 바이트	1 바이트

- 인수 설명 (n1): 0=한글명조체, 1=한글고딕체
- 인수 설명 (n2): 0=영문명조체, 1=영문고딕체
- 동작 설명: 한글/영문 폰트를 설정한다.
- 전송 예: 한글명조체, 영문고딕체 설정

  (16진수) 1B 46 4F 00 01

  (CUBLOC) Font 0, 1

폰트 스타일 설정

ESC	FS	n1	n2	n3
1 바이트	2 바이트	1 바이트	1 바이트	1 바이트

- 인수 설명 (n1): 0=반전해제, 1=반전
- 인수 설명 (n2): 0=밑줄해제, 1=밑줄
- 인수 설명 (n3): 0= 확대 없음, 1= 가로 두 배, 2= 세로 두 배, 3= 가로, 세로 두 배
- 동작 설명: 폰트에 밑줄 및 반전 효과를 나타나게 하고 글자 사이즈를 확대합니다.
- 전송 예: 반전해제 및 밑줄 표시, 가로로 두 배 크게

  (16진수) 1B 46 53 00 01 01

  (CUBLOC) Style 0, 1, 1

▌글자 및 기호 좌표 설정

ESC	LO	n1	n2
1 바이트	2 바이트	1 바이트	1 바이트

- 인수 설명 (n1): LCD 가로축 (좌표계 설명 참조)
- 인수 설명 (n2): LCD 세로축(좌표계 설명 참조)
- 동작 설명: 화면에 글자 및 기호 표시 시 위치를 설정해줍니다.
- 전송 예: 가로축 0, 세로축 0

  (16진수) 1B 4C 4F 00 00

  (CUBLOC) Locate 0, 0

▌글자 표시 방법 (CUBLOC 기준)

1) RS232

 # 예제 1

```
Set Display 1, 1, 19200, 200          ' LCD 및 통신 방법 선택
Print 27, 67, 68, 0                   ' 코드설정
Font 0, 0                             ' 폰트설정
Style 0, 0, 0                         ' 폰트스타일 설정
Locate 0, 0                           ' 좌표 설정
Print "ComfileTechnology (컴파일)"
Locate 0, 1                           ' 좌표 설정
Print "www.comfile.co.kr"
Locate 0, 2                           ' 좌표 설정
Print "Tel. 02) 711-2592"
```

 # 예제 2

```
Set Display 1, 1, 19200, 200          ' LCD 및 통신 방법 선택
Print 27, 67, 68, 0                   ' 코드설정
Font 1, 1                             ' 폰트설정
Style 0, 0, 0                         ' 폰트스타일 설정
Print Loc, 0, 0, "ComfileTechnology (컴파일)"
Print Loc, 0, 1, "www.comfile.co.kr"
Print Loc, 0, 2, "Tel. 02) 711-2592"
```

 # 예제 3

```
Set Display 1, 1, 19200, 200          ' LCD 및 통신 방법 선택
Print 27, 67, 68, 0                   ' 코드설정
Font 0, 0                             ' 폰트설정
Style 1, 0, 0                         ' 폰트스타일 설정
Locate 0, 0                           ' 좌표 설정
Print "ComfileTechnology (컴파일)"
Locate 0, 1                           ' 좌표 설정
Print "www.comfile.co.kr"
Locate 0, 2                           ' 좌표 설정
Print "Tel. 02) 711-2592"
```

 # 예제 4

```
Set Display  1, 1, 19200, 200          ' LCD 및 통신 방법 선택
Print 27, 67, 68, 0                    ' 코드설정
Font  0, 0                             ' 폰트설정
Style  0, 1, 0                         ' 폰트스타일 설정
Print Loc, 0, 0, "ComfileTechnology (컴파일)"
Print Loc, 0, 1, "www.comfile.co.kr"
Print Loc, 0, 2, "Tel. 02) 711-2592"
```

 # 예제 5

```
Set Display  1, 1, 19200, 200          ' LCD 및 통신 방법 선택
Print 27, 67, 68, 0                    ' 코드설정
Font  0, 0                             ' 폰트설정
Style  0, 0, 1                         ' 폰트스타일 설정
Locate  0, 0                           ' 좌표 설정
Print "ComfileTechnology (컴파일)"
Locate  0, 1                           ' 좌표 설정
Print "www.comfile.co.kr"
Locate  0, 2                           ' 좌표 설정
Print "Tel. 02) 711-2592"
```

 # 예제 6

```
Set Display  1, 1, 19200, 200          ' LCD 및 통신 방법 선택
Print 27, 67, 68, 0                    ' 코드설정
Font  0, 0                             ' 폰트설정
Style  0, 0, 2                         ' 폰트스타일 설정
Print Loc, 0, 0, "ComfileTechnology (컴파일)"
Print Loc, 0, 2, "www.comfile.co.kr"
```

 # 예제 7

```
Set Display  1, 1, 19200, 200          ' LCD 및 통신 방법 선택
Print 27, 67, 68, 0                    ' 코드설정
Font  0, 0                             ' 폰트설정
Style  0, 0, 3                         ' 폰트스타일 설정
Locate  0, 0                           ' 좌표 설정
Print "ComfileTechnology (컴파일)"
Locate  0, 2                           ' 좌표 설정
Print "www.comfile.co.kr
```

2) CuNET(I2C)

 # 예제 1

```
Set Display  1, 0, 0, 200          ' LCD 및 통신 방법 선택
Print 27, 67, 68, 0                ' 코드설정
Font  0, 0                         ' 폰트설정
Style  0, 0, 0                     ' 폰트스타일 설정
Print Loc, 0, 0, "ComfileTechnology (컴파일)"
Print Loc, 0, 1, "www.comfile.co.kr"
Print Loc, 0, 2, "Tel. 02) 711-2592"
```

▌특수 문자 만들기 (16x16)

ESC	D F	n0	n1	n2...n33
1 바이트	2 바이트	1 바이트	1 바이트	32바이트

- 인수 설명 (n0): 램 저장 위치 (0~49), 50개 저장
- 인수 설명 (n1): 사용 안함 (0)
- 인수 설명 (n2...n33): 특수 문자 값
- 동작 설명: Pixel Studio를 이용하여 만든 특수문자를 램에 저장합니다. (주의 사항 : 램에 특수 문자를 저장 하므로 전원을 껐다 켜면 특수 문자는 지워집니다.) Page12 특수 문자 만들기 참조
- 전송 예: 저장 위치 7에 특수문자 만들기

(16진수) 1B 44 46 07 00
00 00 01 00 01 80 01 C0 01 E0 01 F0 01 F8 01 F0
01 E0 01 C0 01 80 01 00 7F FE 3F FC 1F F8 00 00

```
(CUBLOC) DEFCHR &H07,_
         &H00,&H00,&H01,&H00, &H01,&H80,&H01,&HC0,_
         &H01,&HE0,&H01,&HF0, &H01,&HF8,&H01,&HF0,_
         &H01,&HE0,&H01,&HC0, &H01,&H80,&H01,&H00,_
         &H7F,&HFE,&H3F,&HFC, &H1F,&HF8,&H00,&H00
```

▌사용자가 만든 특수 문자 표시 (16x16)

ESC	D P	n0
1 바이트	2 바이트	1 바이트

- 인수 설명 (n0): 램에 저장되어 있는 특수문자 주소(0~49)
- 동작 설명: 램에 저장되어 있는 특수문자를 표시합니다.
- 전송 예: 램 주소 1번 문자 표시

(16진수) 1B 44 50 00

```
(CUBLOC) Print 27, 68, 80, 0
```

▌내장된 특수 문자 표시 (16x16)

ESC	E P	n0
1 바이트	2 바이트	1 바이트

- 인수 설명 (n0): 플래쉬 메모리에 저장되어 있는 특수문자 주소(0~255), Page14 표1 참조
- 동작 설명: 플레쉬 메모리에 저장되어 있는 특수문자를 표시합니다.
- 전송 예: 플레쉬 메모리 1번 문자 표시

(16진수) 1B 45 50 00

```
(CUBLOC) Print 27, 69, 80, 0
```

(* 위 화면은 여러 개의 특수문자를 표시한 겁니다.)

9-1 그래픽 관련 명령군

좌표는 word값이므로 LOW -> HIGH순으로 2byte를 보내야 합니다.

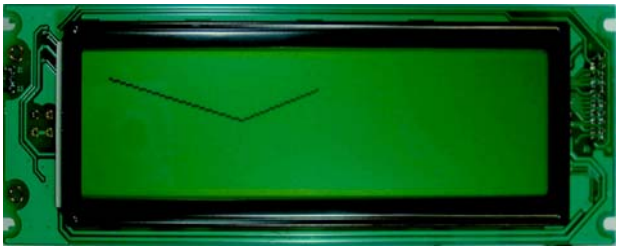
▌선 그리기

ESC	LI	X1	Y1	X2	Y2
1 바이트	2 바이트	2 바이트	2 바이트	2 바이트	2 바이트

- 인수 설명: X1, Y1=시작 포인트, X2, Y2=종료 포인트
- 동작 설명: X1, Y1에서 X2, Y2로 직선을 표시합니다.
- 전송 예: 좌표 10,10에서 80,30 에 선을 그릴 경우
  (16진수) 1B 4C 49 0A 00 0A 00 50 00 1E 00
  (CUBLOC) Line 10, 10, 80, 30

▌계속 이어서 그리기

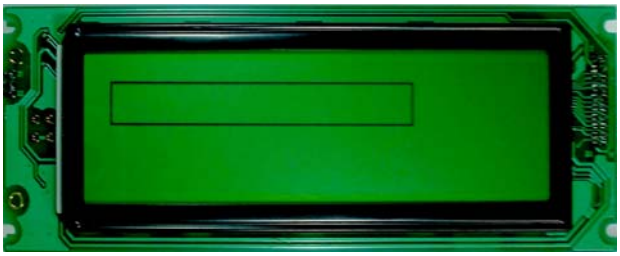
ESC	LT	X	Y
1 바이트	2 바이트	2 바이트	2 바이트



- 인수 설명: X, Y=종료 포인트
- 동작 설명: 계속해서 선을 그릴 경우에 사용합니다. 먼저 선을 그린 경우 끝난 지점이 자동적으로 시작점이 됩니다. 계속 이어진 선을 그릴 경우 종료 점을 반복해서 지정하는 불편함을 해소한 명령입니다.
- 전송 예: 좌표 120,15 에 선을 이어서 그릴 경우
  (16진수) 1B 4C 54 78 00 0F 00
  (CUBLOC) Lineto 120, 15

▌박스 그리기

ESC	B X	X1	Y1	X2	Y2
1 바이트	2 바이트	2 바이트	2 바이트	2 바이트	2 바이트



- 인수 설명: X1, Y1=시작 포인트, X2, Y2=종료 포인트
- 동작 설명: X1, Y1에서 X2, Y2를 사선으로 하는 박스를 그립니다.
- 전송 예: 좌표 10,10 ~ 170,30 박스를 그릴 경우
  (16진수) 1B 42 58 0A 00 0A 00 AA 00 1E 00
  (CUBLOC) Box 10, 10, 170, 30

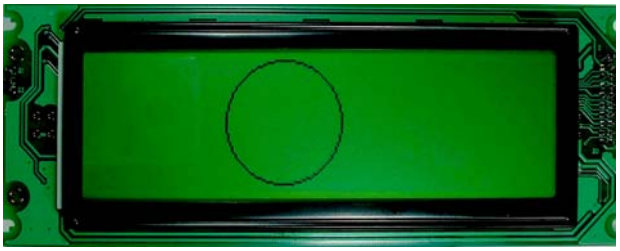
▌채워진 박스 그리기

ESC	B F	X1	Y1	X2	Y2	n
1 바이트	2 바이트	2 바이트	2 바이트	2 바이트	2 바이트	1 바이트

- 인수 설명: X1, Y1=시작 포인트, X2, Y2=종료 포인트,
  n : 0=그리기, 1=지우기
- 동작 설명: X1,Y1에서 X2,Y2를 사선으로하는 채워진 박스를 그립니다.
- 전송 예: 좌표 10,10~170,30 에 채워진 박스를 그릴 경우
  (16진수) 1B 42 46 0A 00 0A 00 AA 00 1E 00 00
  (CUBLOC) Boxfill 10, 10, 170, 30, 0

원 그리기

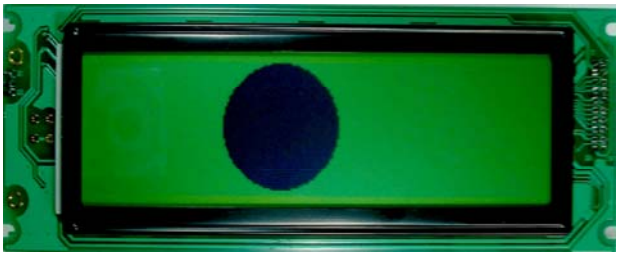
ESC	CI	X	Y	R
1 바이트	2 바이트	2 바이트	2 바이트	2 바이트



- 인수 설명: X, Y=중심 포인트, R=반지름
- 동작 설명: X, Y를 중심점으로 하고, R을 반지름으로 하는 원을 그립니다.
- 전송 예: 좌표 100,30에 반지름 30의 원을 그릴 경우
  (16진수) 1B 43 49 64 00 1E 00 1E 00
  (CUBLOC) Circle 100, 30, 30

채워진 원 그리기

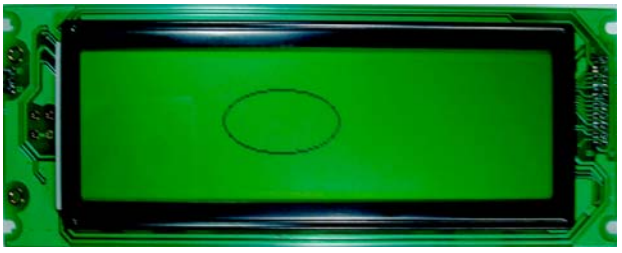
ESC	CF	X	Y	R
1 바이트	2 바이트	2 바이트	2 바이트	2 바이트



- 인수 설명: X, Y=중심 포인트, R=반지름
- 동작 설명: X, Y를 중심점으로 하고, R을 반지름으로 하는 채워진 원을 그립니다.
- 전송 예: 좌표 100,30에 반지름 30의 채워진 원을 그릴 경우
  (16진수) 1B 43 46 64 00 1E 00 1E 00
  (CUBLOC) Circlefill 100, 30, 30

타원 그리기

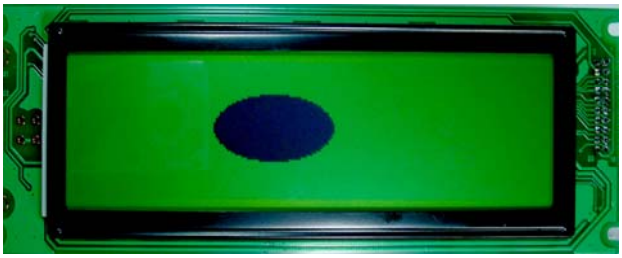
ESC	EL	X	Y	R1	R2
1 바이트	2 바이트	2 바이트	2 바이트	2 바이트	2 바이트



- 인수 설명: X, Y=중심 포인트, R1=가로 반지름, R2=세로 반지름
- 동작 설명: X, Y를 중심점으로 하고, R1을 가로 반지름, R2를 세로 반지름으로 하는 타원을 그립니다.
- 전송 예: 좌표 100,30에 가로 반지름 30, 세로 반지름 15의 타원을 그릴 경우
  (16진수) 1B 45 4C 64 00 1E 00 1E 00 0F 00
  (CUBLOC) Ellipse 100, 30, 30, 15

채워진 타원 그리기

ESC	EF	X	Y	R1	R2
1바이트	2바이트	2바이트	2바이트	2바이트	2바이트



- 인수 설명: X, Y=중심 포인트, R1=가로 반지름, R2=세로 반지름
- 동작 설명: X, Y를 중심점으로 하고, R1을 가로 반지름, R2를 세로 반지름으로 하는 채워진 타원을 그립니다.
- 전송 예: 좌표 100,30에 가로 반지름 30, 세로 반지름 15의 채워진 타원을 그릴 경우
  (16진수) 1B 45 46 64 00 1E 00 1E 00 0F 00
  (CUBLOC) Elfill 100, 30, 30, 15

호 그리기

ESC	A R	X	Y	R	START	END
1 바이트	2 바이트	2 바이트	2 바이트	2 바이트	2 바이트	2 바이트

- 인수 설명: X, Y=중심 포인트, R=반지름, START=시작 각도, END=종료 각도
- 동작 설명: X, Y를 중심점으로 하고, R을 반지름으로 하는 원에서 START와 END 각 사이의 호를 그립니다. 각도는 0~360사이의 값 이여야 합니다.
- 전송 예: 좌표 100,10에서 반지름=30, START=70, END=190 으로 지정하여 호를 그릴 경우
  (16진수) 1B 41 52 64 00 0A 00 1E 00 46 00 BE 00
  (CUBLOC) Arc 100, 10, 30, 70, 190

점 그리기

ESC	P S	n1	n2
1 바이트	2 바이트	2 바이트	2 바이트

- 인수 설명 (n1): LCD 가로축 (좌표계 설명 참조)
- 인수 설명 (n2): LCD 세로축(좌표계 설명 참조)
- 동작 설명: 화면에 점하나를 그립니다.
- 전송 예: 10,10에 점하나 찍기
  (16진수) 1B 50 53 0A 00 0A 00
  (CUBLOC) Pset 10,10

9-2 BMP 관련 명령군

화면에 BMP파일을 직접 표시하거나, 플레쉬 메모리에 저장한 뒤 표시할 수 있습니다. PC상에서 각종 그래픽 툴을 이용하여 그린 그림을 화면에 손쉽게 표시할 수 있는 일련의 명령군에 대한 설명입니다.

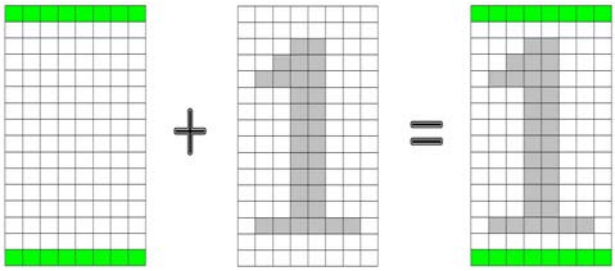
BMP PRINT

ESC	B P	X	Y	n1	n2
1 바이트	2 바이트	2 바이트	2 바이트	1 바이트	1 바이트

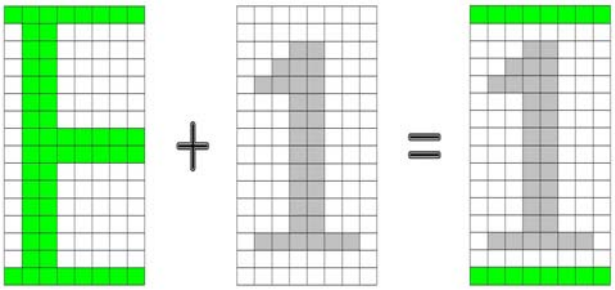
- 인수 설명: X, Y = BMP 파일을 표시할 위치
  n1 = 플레쉬 메모리에 저장되어 있는 BMP파일 번호
  n2 = 사용 안함(0)
- 동작 설명: 플레쉬 메모리에 저장되어 있는 BMP파일 중 n1번째 기록된 파일을 원하는 위치 X,Y에 표시 합니다.
  BMP파일을 플레쉬 메모리로 다운로드 하기 위해서는 BMP Downloader 프로그램이 있어야 합니다.
- 전송 예: 0,0에 3번째 파일 표시
  (16진수) 1B 42 50 00 00 00 00 02 00
  (CUBLOC) Bmp 0, 0, 2, 0

▌그래픽 및 BMP 파일 위에 한글 및 영문, 숫자, 기호 뿌리기

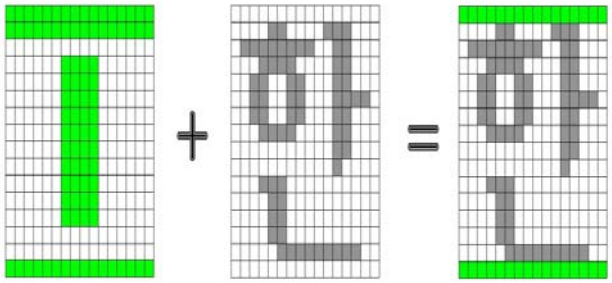
Print 명령을 사용하여 한글, 영문, 숫자, 기호를 표시할 때 기존에 그려져 있던 것과 부분적(위아래 1줄)으로 합쳐져서 표시됩니다. 아래 그림에서 보듯이 그래픽으로 위아래 선을 그려 놓고 숫자 1을 써 넣으면 그래픽과 숫자 1이 합쳐져서 표시되는 것을 알 수 있습니다.



만약 아래 그림과 같이 위아래 1줄을 제외한 나머지 부분에 그래픽 및 그림파일이 있으면 지우면서 뿌려줍니다.



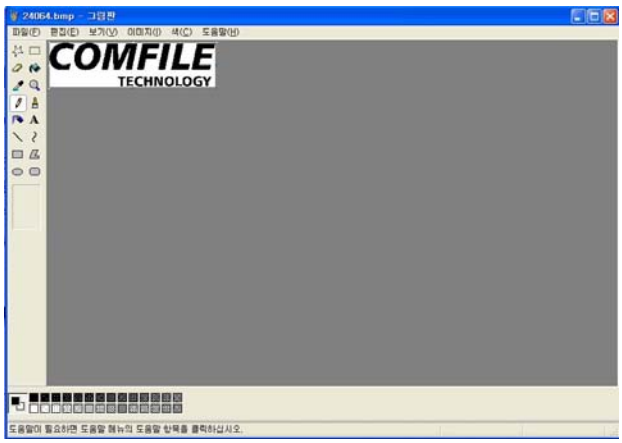
한글의 경우는 아래와 같습니다.



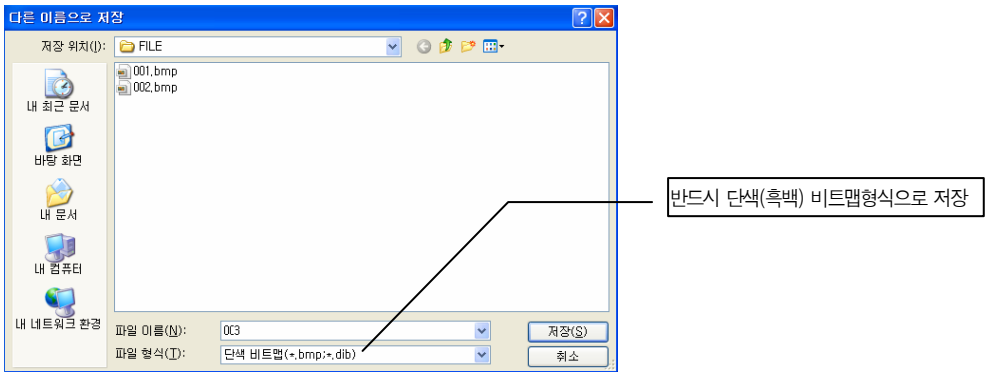
10. BMP 파일 만드는 방법

1) PC에서 각종 그래픽 툴을 이용해서 BMP파일을 만들면 되는데 여기에서는 윈도우XP에 있는 그림판을 이용해서 만들도록 하겠습니다.

2) 그림판 → 이미지 → 특성에 너비와 높이는 LCD SIZE가 240x64이므로 240x64를 초과할 수 없습니다.
단위는 픽셀, 색은 흑백에 체크되어 있어야 합니다.



3) 저장 방식은 반드시 "단색 (흑백) 비트맵" 으로 선택해야 합니다.

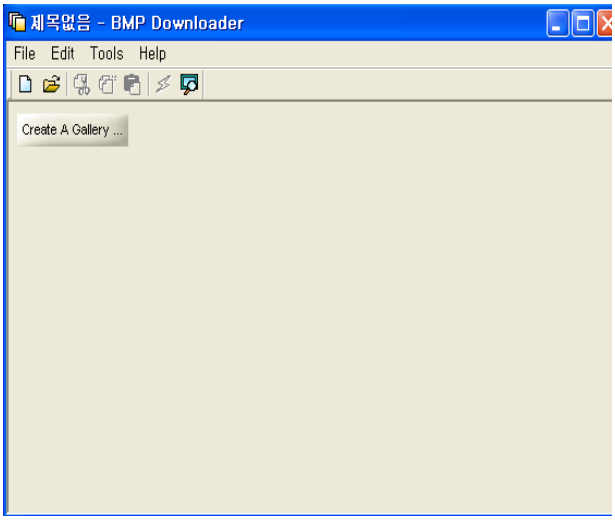


4) 다운로드 할 BMP파일을 하나의 폴더 안에 모아놓습니다. 파일명은 알파벳 순서대로 읽어서 다운로드 하기 때문에 위 그림에서처럼 숫자로 구분해 놓는 편이 좋습니다.

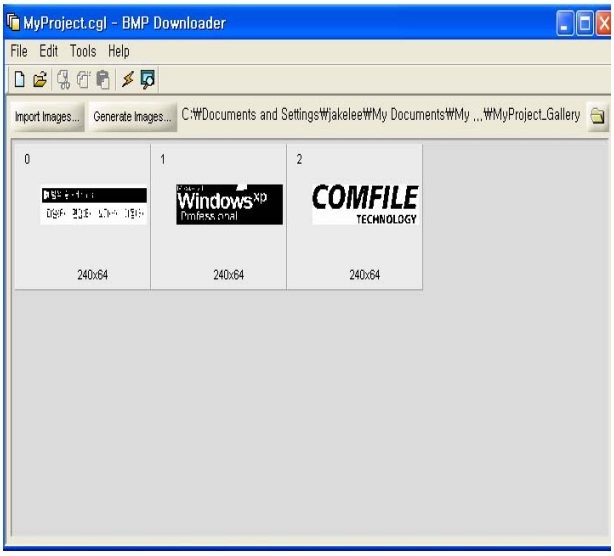
11. BMP 파일 다운로드

1) BMP Downloader v1.09 프로그램을 사용하여 다운로드 합니다. 이때 **PC와 GHA는 ALCD Cable로 연결이** 되어 있어야 하고 **GHA는 LCD가 반듯이 부착 되어** 있어야 합니다. 점검이 끝나면 전원을 인가한 후 다운로드를 실행합니다. (프로그램 위치 http://www.comfile.co.kr/ → 자료실 → DISPLAY → 소프트웨어 및 메뉴얼)

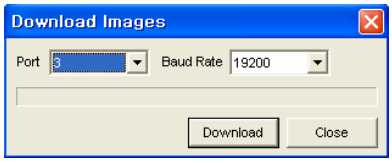
2) BMP Downloader v1.09 프로그램을 실행합니다.
Create A Gallery 버튼을 눌러서 BMP파일을 저장할 저장 폴더명과 저장 위치를 선택 후 Create Now를 누릅니다.



3) Import Images...을 눌러서 다운로드 할 BMP 파일을 불러옵니다.



4) Tools → Download Image를 누릅니다. (번개 아이콘을 눌러도 됨) 아래 그림처럼 시리얼 포트번호(Port)와 통신속도(BaudRate)를 맞혀주고 Download를 누릅니다. (통신속도는 Page3 스위치 설정방법 에 의해 설정된 통신속도랑 동일하게 설정해야 함)



5) 저장이 완료 되면 아래와 같은 메시지가 표시됩니다.

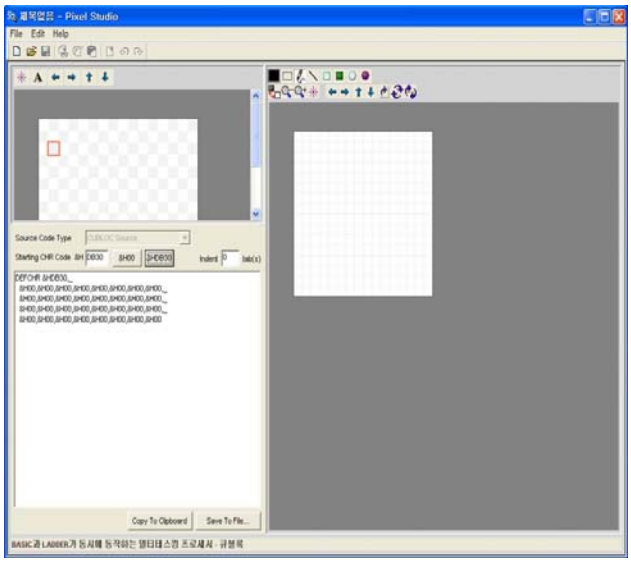
* 파일 용량이 245000byte를 초과해서는 안됩니다.

!!!. BMP Downloader 에서 보이는 BMP 파일은 LCD 에 표시 시,
흑백이 바뀌어서 표시가 됩니다.
흑백을 반대로 표시하고 싶으실 때는 BMP 파일을 만들 때
그림판 → 이미지 → 색반전을 눌러서 파일을 저장해야 합니다.

12. 특수 문자 만들기

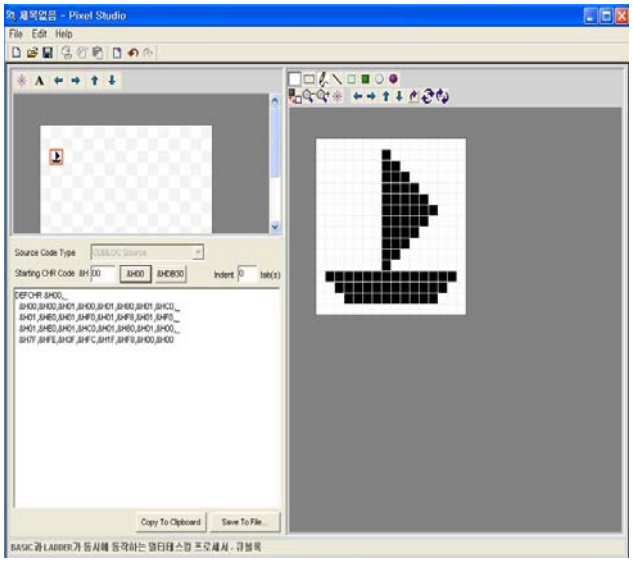
1) Pixel Studio v1.01 프로그램을 사용하여 특수문자를 만듭니다.
(프로그램 위치 http://www.comfile.co.kr/ → 자료실 → TOUCH CONTROL → 소프트웨어 및 메뉴얼)

2) Pixel Studio v1.01 프로그램을 실행합니다.



3) Starting CHR Code 값을 램 저장 위치(&H00)로 설정합니다.

4) 원하는 특수 문자를 그립니다.



5) Copy To Clipboard를 누른 후 소스 파일에 붙이셔서 사용하시면 됩니다.

#. 표 1

	0x00	0x01	0x02	0x03	0x04	0x05	0x06	0x07	0x08	0x09	0x0A	0x0B	0x0C	0x0D	0x0E	0x0F
0x00	※	☎	☎	☜	☞	㈜	㈜	℃	℉	○	●	◎	◇	◆	□	■
0x10	△	▽	◁	▷	▲	▼	◀	▶	↑	↓	←	→	↔	♤	♠	♡
0x20	♥	♧	♣	⊙	◈	▣	◐	◑	▤	▥	▨	▧	▦	♨	i	ii
0x30	iii	iv	v	vi	vii	viii	ix	x	I	II	III	IV	V	VI	VII	VIII
0x40	IX	X	μℓ	mℓ	dℓ	ℓ	kℓ	㎣	㎤	㎥	㎦	μm	㎟	㎠	㎡	㎢
0x50	μg	cal	kcal	m/s	m/s²	ps	ns	μs	ms	pV	nV	μV	mV	kV	MV	pA
0x60	nA	μA	mA	kA	μW	Hz	kHz	MHz	GHz	THz	Ω	kΩ	MΩ	μF	dB	mol
0x70	rad	rad/s	rad/s²	kPa	MPa	GPa	㉠	㉡	㉢	㉣	㉤	㉥	㉦	㉧	㉨	㉩
0x80	㉪	㉫	㉬	㉭	㉮	㉯	㉰	㉱	㉲	㉳	㉴	㉵	㉶	㉷	㉸	㉹
0x90	㉺	㉻	ⓐ	ⓑ	ⓒ	ⓓ	ⓔ	ⓕ	ⓖ	ⓗ	ⓘ	ⓙ	ⓚ	ⓛ	ⓜ	ⓝ
0xA0	ⓞ	ⓟ	ⓠ	ⓡ	ⓢ	ⓣ	ⓤ	ⓥ	ⓦ	ⓧ	ⓨ	ⓩ	①	②	③	④
0xB0	⑤	⑥	⑦	⑧	⑨	⑩	⑪	⑫	⑬	⑭	⑮	㉠	㉡	㉢	㉣	㉤
0xC0	㉥	㉦	㉧	㉨	㉩	㉪	㉫	㉬	㉭	㉮	㉯	㉰	㉱	㉲	㉳	㉴
0xD0	㉵	㉶	㉷	㉸	㉹	㉺	㉻	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)
0xE0	(j)	(k)	(l)	(m)	(n)	(o)	(p)	(q)	(r)	(s)	(t)	(u)	(v)	(w)	(x)	(y)
0xF0	(z)	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)	(15)