

128 * 128 Pixel 그래픽 LCD + 정전용량 touch screen (8 Key)

일체형 모듈 사용자 설명서 (FW V2.1)

(주)우정하이텍

2019,04,05 (R10)

본 manual은 ㈜우정하이텍 의 “Graphic Touch LCD module” 의 hardware 사양, display 제어 와 Touch data 전송 등 Packet 사양 및 명령/응답 에 대하여 기술한다.

1. H/W 구조

- 1) 전송 용 Connector : J3
 - A. Pin 1 : VCC In = 5V input
 - B. Pin 2 : Touch Event = Touch 발생시 Low를 출력.
 - C. Pin 3 : RxD = UART 수신 데이터 라인, 5V TTL level.
 - D. Pin 4 : TxD = UART 송신 데이터 라인, 5V TTL level.
 - E. Pin 5 : Ground Line.
- 2) Program Download 용 connector : J1
해당 connector는 user가 사용하지 않습니다.
- 3) LCD Interface Connector : J2
- 4) Touch Node dusruf connector : J4
- 5) 전원 투입 표시 용 LED : LED2
- 6) Status 표시 용 LED : LED1
- 7) 제품 외형

<G, T, R, E, V>

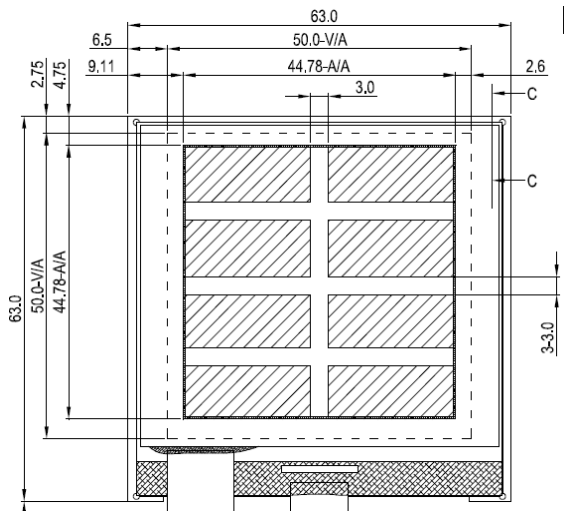


<그림 1 : 모듈 전면 부>

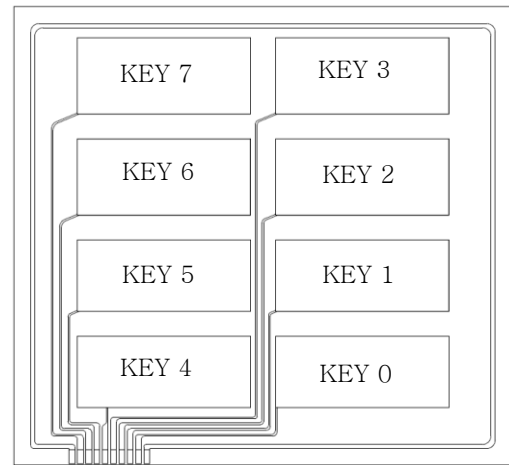


<그림 2 : 모듈 후면 부>

8) Touch 부 위치 및 size

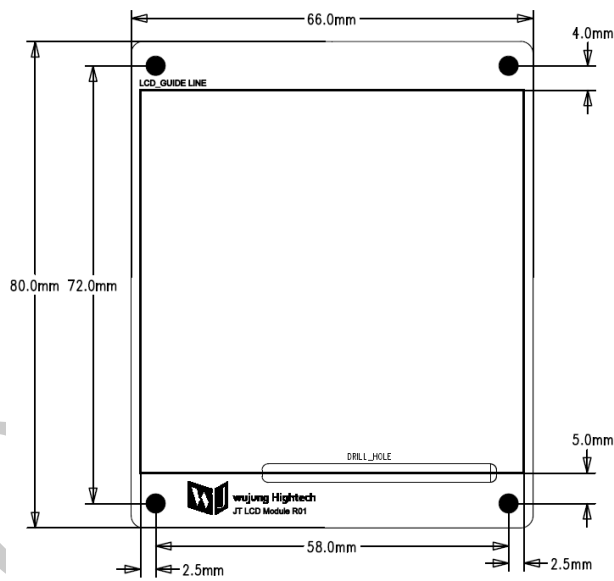


<그림 3 : Touch node 치수>



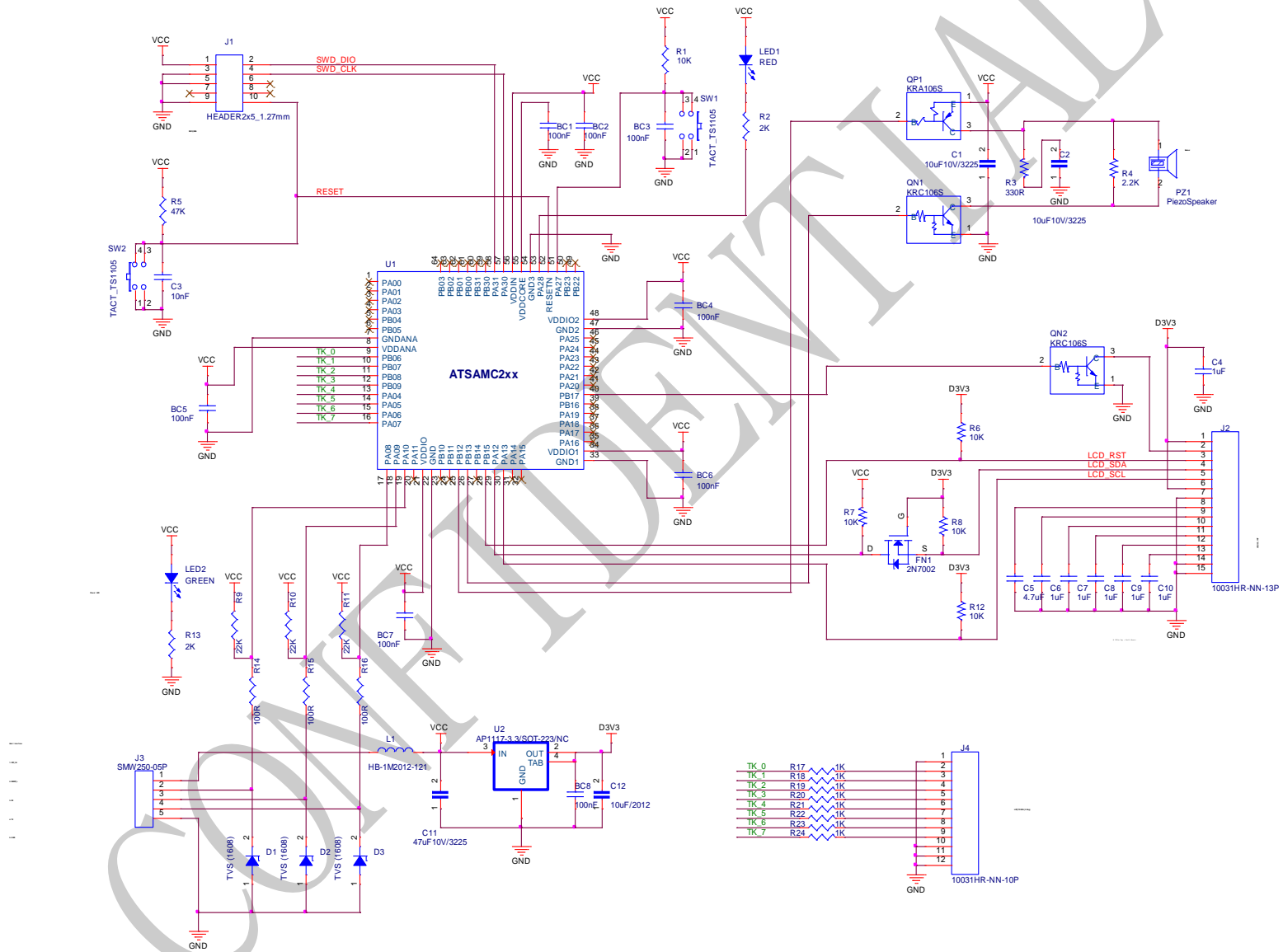
<그림 4 : Touch node Key 위치>

9) Board 치수 도



<그림 3 : Board 외형 치수>

10) 회로도



2. Protocol 구조

- 1) 전송 전압 : 5V TTL Level.
- 2) 전송 속도 : 57600 BPS
- 3) Byte 구조 : 8 data bit, 1 stop bit, non parity.

4) Transection flow.

HOST	Direction	LCD Touch Monitor
제어 명령	==>>	
	<<==	NAK
	전문 에러	
전문 재전송	==>>	
	<<==	ACK
	전문 정상 수신	
Touch data 요청	==>>	
	<<==	ACK
	<<==	Touch data 회신
NAK	==>>	
	전문 수신 에러	
	<<==	Touch data 재 전송
ACK	==>>	
	touch data 정상 전송	

3. Packet 구조

1) Data 열의 구조

STX	LENGTH	DATA byte 1	DATA byte 2	DATA byte n	ETX	BCC
0x02	30H+n	data 1	data 2		data n	0x03	SUM

(1) LENGTH : $0x30 + 2('LENGTH' \text{ 자신} + 'ETX') + \text{data byte 수}$.

최대 data 영역 길이는 128 byte.

예) 128 byte 전송시 'LENGTH' = $0x30 + 2 + 128 = 178 = 0xB2$

(2) Data : Command / Response에 정의 된 data 구조를 따름.

(3) BCC : Block Check Character 로서 'STX'를 제외한 모든 data를 sum 연산하여 하위 1 byte 만 사용한다.

(4) Packet 구조는 송신 과 수신 이 동일 구조 이다.

(5) 각 Packet에 대한 정상 수신 여부를 asc2 code의 ACK, NAK 로 응답 한다.

ACK = 0x06, NAK = 0x15.

2) 수신 Data 영역의 구조

COMMAND	CONTROL	좌표 X1	좌표 Y1	좌표 X2	좌표 Y2	DATA
---------	---------	-------	-------	-------	-------	------

- (1) COMMAND : 1 byte 로 0x41~x49 를 사용하며 상세내용은 각 명령의 설명을 참조.
- (2) CONTROL : 1 byte 로 0x31~0x39 를 사용하며 상세내용은 각 명령의 설명을 참조.
- (3) 좌표 X1 : 1 byte data로 시작점 가로 영역 pixel의 위치를 지정한다. Resolution은 128 이며, 0 ~127을 사용하고 여기에 0x30을 더하여 전송 한다.
- (4) 좌표 Y1 : 1 byte data로 시작점 세로 영역 pixel의 위치를 지정한다. Resolution은 128 이며, 0 ~127을 사용하고 여기에 0x30을 더하여 전송 한다.
- (5) 좌표 X2 : 내용은 X1 과 동일 하며 가로축의 끝점을 표시한다.
- (6) 좌표 Y2 : 내용은 Y1 과 동일 하며 세로축의 끝점을 표시한다.
- (7) DATA : 각 command에 따라 가변 장 이다.
- (8) COMMAND를 포함한 모든 data의 합은 128 byte를 이하이다.
- (9) 시작점 좌표가 끝점 좌표 보다 항상 작은 값이어야 한다.
- (10) Command의 구조는 command 의 종류에 따라 달라질 수 있으므로 각명령의 항목을 확인하여 실행 한다.

4. Command / Response

1) Graphic Display 'COMMAND' : 'A' (0x41)

: 정해진 좌표에 점, 선, 원, 사각 의 도형을 표시 한다.

(1) Point draw 'CONTROL' : 0x31

좌표상에 점을 표시하며, 좌표 X1-Y1 만 사용한다.

COMMAND	CONTROL	좌표 X1	좌표 Y1
0x41	0x31	X + 0x30	Y + 0x30
점의 좌표			

(2) Line draw 'CONTROL' : 0x32

: 좌표상에 선을 표시하며, 좌표 X1-Y1, X2-Y2 를 사용한다.

COMMAND	CONTROL	좌표 X1	좌표 Y1	좌표 X2	좌표 Y2
0x41	0x32	X1 + 0x30	Y1 + 0x30	X2 + 0x30	Y2 + 0x30
시작 좌표				끝 좌표	

(3) Circle draw 'CONTROL' : 0x33

: 좌표상에 원을 표시하며, 좌표 X1-Y1, 원의 직경 R 을 사용한다.

COMMAND	CONTROL	좌표 X1	좌표 Y1	반지름 R
0x41	0x33	$X1 + 0x30$	$Y1 + 0x30$	$R + 0x30$
		원의 중심 좌표		원의 반지름

반지름 R은 pixel 개수를 표시한다.

(4) Box draw 'CONTROL' : 0x34

: 좌표상에 사각 box를 표시하며, 좌표 X1-Y1, X2-Y2를 사용한다.

COMMAND	CONTROL	좌표 X1	좌표 Y1	좌표 X2	좌표 Y2
0x41	0x34	$X1 + 0x30$	$Y1 + 0x30$	$X2 + 0x30$	$Y2 + 0x30$
		좌측상단 좌표		우측하단 좌표	

(5) Area Clear 'CONTROL' : 0x35

: X1Y1, X2Y2 좌표상의 사각 box 영역의 display를 지운다.

COMMAND	CONTROL	좌표 X1	좌표 Y1	좌표 X2	좌표 Y2
0x41	0x35	$X1 + 0x30$	$Y1 + 0x30$	$X2 + 0x30$	$Y2 + 0x30$
		좌측상단 좌표		우측하단 좌표	

2) 문자 Display 'COMMAND' : 'B' (0x42)

: 정해진 좌표를 시작 점으로 내장 Font를 이용하여 영문, 기호, 한글을 화면에 Display 한다. 지정된 좌표는 첫 문자의 좌측 상단 시작 점이 된다.

Font의 해상도는 영문과 기호의 경우 8 X 16 pixel 이고 한글의 경우 16 X 16 pixel 이다. 문자의 2배 확대 기능을 사용할 수 있다.

(1) 정상 크기 문자 사용 'DISPLAY' : '1' (0x31)

COMMAND	CONTROL	좌표 X1	좌표 Y1	Character String
0x42	0x31	$X1 + 0x30$	$Y1 + 0x30$	(주) max 124 byte
		좌측 상단 시작 좌표		

(주) : 문자열 data 부분은 영문 과 기호는 ASCII code를 따르며 한글은 완성형 code를 사용 한다. 문자열의 byte 수 계산 시 한글은 1자당 2byte 이므로 주의.

(2) 2배 확대 문자 사용 'DISPLAY' : '2' (0x32)

: Command 구조는 정상 크기 문자열 과 동일하며, 위치 계산 시 한 문자 당 영문 & 기호는 16 X 32, 한글은 32 X 32 pixel 을 차지하므로 위치 계산 시 주의.

COMMAND	CONTROL	좌표 X1	좌표 Y1	Character String
0x42	0x32	$X1 + 0x30$	$Y1 + 0x30$	(주) max 124 byte
		좌측 상단 시작 좌표		

(주의) 영문과 기호 (ASCII code)의 확대문자는 별도의 font를 사용하며 “c”(섭씨)의 표현을 위해 “'” 기호를 font 변경하여 사용한다.

(3) Clear Screen 'CONTROL' : '3' (0x33)

: 전체 화면을 지운다.

COMMAND	CONTROL
0x42	0x33

(4) 16 * 8 pixel (작은 Font) 의 User Font를 'DISPLAY' : '4' (0x34)

: Command 구조는 정상 크기 문자열 display 와 동일 하며, Font size 또한 동일 하다. 모두 79개가 있으며 font number 1 ~ 79에 0x30을 더한 값을 “Character String”에 넣어 전송한다.

COMMAND	CONTROL	좌표 X1	좌표 Y1	Character String
0x42	0x34	$X1 + 0x30$	$Y1 + 0x30$	(주) max 120 byte
		좌측 상단 시작 좌표		

(5) 32 * 16 pixel (큰 Font) 의 User Font를 'DISPLAY' : '5' (0x35)

: Command 구조는 확대 크기 문자열 display 와 동일 하며, Font size 또한 동일 하다. 모두 79개가 있으며 font number 1 ~ 79에 0x30을 더한 값을 “Character String”에 넣어 전송한다.

COMMAND	CONTROL	좌표 X1	좌표 Y1	Character String
0x42	0x35	$X1 + 0x30$	$Y1 + 0x30$	(주) max 120 byte
		좌측 상단 시작 좌표		

3) Back light 밝기 제어 'COMMAND' : 'C' (0x43)

: LCD Back light 의 밝기를 설정하는 명령 이며, 0% = off 이고 100% 까지 1% 단위로 설정한다. 한번 설정하면 전원 Off->On 시에도 절정치가 유지 된다.

(1) Control code 는 0 ~ 100 % 의 값에 0x30을 더하여 전송한다.

COMMAND	CONTROL
0x43	(주)

(주) 0x30 = Backlight off

0x30 ~ 0x30 + 100 = Brightness set.

4) 터치 효과음 설정 'COMMAND' : 'D' (0x44)

: 효과 음 높이 별 28가지 와 효과음 Off 중에서 한가지를 설정 하며, 한번 설정 하면 전원 Off-> On 시에도 설정 유지.

(1) 효과 음 1번 'CONTROL' : 0x30 ~ 0x4C

COMMAND	CONTROL
0x44	(주)

(주) 효과음 선택

0x30 = 효과음 없음.

0x30 + 1 = 낮은 '도',

0x30 + 28 = 높은 '도' 사이에서 선택.

5) 정전 용량 터치 event data request 'COMMAND' : 'E' (0x45)

: Connector 2번에 연결된 "Touch event" 신호가 Low 상태가 되면 Touch의 상태가 변경 된 것을 의미하며, 이때는 즉시 "Touch Data Request Command"를 이용하여 Data를 읽는다. Data를 읽고 나면 "Event" 신호는 다시 High 로 되며 touch event process 가 종료 된다.

(1) Touch Data Request

COMMAND
0x45

(2) Touch Data Response

COMMAND	Touch data 1	Touch data 2
0x45	(주1)0x30~0x3f	(주2)0x30~0x3f

(주1) Touch Data 중 상위 4 bit data.

상위 4bit를 아래로 4bit shift 하여 0x30 을 더한값.

(주2) Touch Data 중 하위 4 bit data.

하위 4bit 에 0x30 을 더한값.

EX) touch data 가 0x81 (1000 0001b) 이면

Touch data 1 = 0x38

Touch data 2 = 0x31

이 된다.

6) 통신 속도 설정 'COMMAND' : 'F' (0x46)

: 하기의 5가지 통신 속도 중에 선택하여 설정 하며, 한번 설정 하면 전원 Off-> On 시에도 설정 유지. 변경된 설정은 power Off-> On 하면 적용 된다.

(초기 값은 57600 bps)

(1) 통신 속도 'CONTROL' : 0x30 ~ 0x34

COMMAND	CONTROL
0x46	(주)

(주) 통신 속도 선택

0x30 = 115200 bps

0x31 = 57600 bps (default).

0x32 = 38400 bps

0x33 = 19200 bps

0x34 = 9600 bps

7) LCD Contrast adjust 'COMMAND' : 'G' (0x47)

: Display가 너무 흐리지 않고 Crosstalk 이 없는 값으로 설정 한다.

(개인 이나 LCD 마다 적정 값이 다를 수 있으며, 출하 시 적정 값이 설정되어 있다)

(1) Contrast 값 직접 설정 CONTROL : 0x31 ('1')

Contrast 범위 Value : 0x31 ~ 0x70. (64 Step)

COMMAND	CONTROL	VALUE
0x47	0x31	0x31 ~ 0x70

(2) Contrast 값 1 Count Step Up Control : 0x32 ('2')

Contrast 조정 범위 내에서 값을 1 올린다.

COMMAND	CONTROL
0x47	0x32

(3) Contrast 값 1 Count Step Down Control : 0x33 ('3')

Contrast 조정 범위 내에서 값을 1 내린다.

COMMAND	CONTROL
0x47	0x33

8) 정전용량 터치 감도 설정 'COMMAND' : 'H' (0x48)

: 정전용량 터치 8키에 대하여 각각 감도를 설정하며, 설정 값이 낮을수록 감도가 높아진다. (설정 값은 저장 된다.)

COMMAND	Key	Sensitivity
0x48	1 ~ 8	15 ~ 120

(주) Key : 터치 키 번호 = (1 ~ 8) + 0x30 (8 key)

키 번호에 0x30 을 더하여 전송 한다.

(0x31 ~ 0x38)

(주) Sensitivity : 감도 설정 값 = (15 ~ 120) + 0x30

감도 값에 0x30을 더하여 전송 한다.

(0x3f ~ 0xa8)

9) 정전용량 터치 key의 활성화 설정 'COMMAND' : 'I' (0x49)

: 정전용량 터치 8키에 대하여 각각 사용 여부를 결정하며, 각 키에 해당하는 Bit 가 1 이면 active(활성화) 되고, 0 이면 Inactive(비 활성화) 된다.

이 설정은 저장 되지 않으며 필요 시 Touch 사용 전에 바로 설정 한다.

COMMAND	상위 4 key	하위 4 key
0x49	0x30 ~ 0x3F	0x30 ~ 0x3F

(주) 8 Key (B7:8번키 ~ B0:1번키) 의 8bit 0xFF 를 상위 4bit 와 하위 4 bit 로 분리하여 0x30을 더하여 전송 한다.

10) Power On 시 초기 화면 설정 'COMMAND' : 'J' (0x4A)

: 전원 On 시 처음 화면을 결정하는 명령 이다. Default 설정으로 “우정하이텍”의 기본 화면 과 Blank(빈 화면) 및 User screen 16개 중에 선택 할 수 있다.

이 설정은 저장 되어 Power on 과 rebooting 시 적용된다.

Command	Screen Number
J' (0x4a)	0x30~0x41

(주) Screen number는 0 ~17 까지의 10진수에 0x30을 더한 값을 전송 한다.

- 1) Screen 0 ~ 15 : 0x30 ~ 0x3f = User screen으로 사용자가 Load한 화면이다.
- 2) Screen 16 : 0x31 = Blank screen (빈 화면)
- 3) Screen 17 : 0x41 = “우정하이텍” 초기 화면으로 version 과 연락처가 표시된다.

11) Beep 음 발생 'COMMAND' : 'K' (0x4b)

: 이 명령을 수신하면 미리 설정된 sound 또는 4개옥타브의 음 중에 선택된 음을 발생 시킨다. 미리 입력 된 Sound는 “0 ~ 3”의 4가지가 있으며 여기에 사용자 음 지정 명령 “4”가 있으며, 명령 수신 즉시 음을 발생 시킨다.

Command	Beep Select	Sound Scale
K' (0x4b)	0x30~0x33, 0x34	0x30~0x5F

(주1) Beep Select : user sound 4가지 0 ~ 3 또는 사용자 지정 음 4 에 0x30을 더하여 전송 한다. 미리 설정된 음은 아래와 같다.

- a) 0x30 : “땡~”
- b) 0x31 : “띠당~”
- c) 0x32 : “띠디당~” (점차 높아짐)
- d) 0x33 : “띠디당~” (점차 낮아짐)
- e) 0x34 : 사용자 지정음 (Sound Scale 로 결정)

(주2) Beep Select 가 “0x34” 일 경우 Sound Scale에 해당하는 음을 발생 시킨다. Sound Scale은 아래와 같이 4개의 옥타브(4C ~ 7B)로 구성 되어 있다.

- (옥타브 1) 0x30(4C), 0x31(4CS), 0x32(4D), 0x33(4DS), 0x34(4E), 0x35(4F), 0x36(4FS), 0x37(4G), 0x38(4GS), 0x39(4A), 0x3A(4AS), 0x3B(4B),
- (옥타브 2) 0x3C(5C), 0x3D(5CS), 0x3E(5D), 0x3F(5DS), 0x40(5E), 0x41(5F), 0x42(5FS), 0x43(5G), 0x44(5GS), 0x45(5A), 0x46(5AS), 0x47(5B),
- (옥타브 3) 0x48(6C), 0x49(6CS), 0x4A(6D), 0x4B(6DS), 0x4C(6E), 0x4D(6F), 0x4E(6FS), 0x4F(6G), 0x50(6GS), 0x51(6A), 0x52(6AS), 0x53(6B),
- (옥타브 4) 0x54(7C), 0x55(7CS), 0x56(7D), 0x57(7DS), 0x58(7E), 0x59(7F), 0x5A(7FS), 0x5B(7G), 0x5C(7GS), 0x5D(7A), 0x5E(7AS), 0x5F(7B)

12) 선택한 User Screen을 display 'COMMAND' : 'L' (0x4c)

: 16개의 사용자 화면 image 중에 선택한 image를 display 한다.

Command	Screen Select
L' (0x4c)	0x30~0x3f

(주1) Screen Select : User screen 0 ~ 15 중에 선택하며, 그 값에 0x30을 더하여 전송 한다.

13) Touch BEEP active 'COMMAND' : 'M' (0x4d)

: Touch 시 효과음 유,무를 제어하는 명령 이다. 이 명령은 저장 되지 않으며 필요시 바로 사용한다.

Command	Control
M' (0x4d)	0x30 or 0x31

(주1) Control : 0x30 = disable (무음)
0x31 = enable (유음)

14) User Font Memory Upload 'COMMAND' : 'U' (0x55)

: 사용자가 작성한 Font를 memory에 upload 한다. Font는 32*16 와 16*8 을 선택 할 수 있으며 각 79개 씩을 upload 하여 사용할 수 있다.

Command	Font size Select	Font Select	Data
U' (0x55)	0x31,0x32 (주1)	0x31~0x7F(주2)	16 byte or 64 byte

(주1) Font size Select : 16*8 font 와 32*16 font 중에 선택 한다.

(a) 0x31 : 16 * 8 font를 선택 한다.

(b) 0x32 : 32 * 16 font를 선택 한다.

(주2) Font Select : 79개의 Font중에 upload할 font를 선택 한다.

font는 79개씩으로 1~79 까지 이며 여기에 0x30을 더하여 전송 한다.

Ex) 16*8 font에 5번 내용을 upload = 0x55,0x31,0x35

(주3) Data : font에 적용할 binary data 이며, font에 따라 data byte가 다르다.

16 * 8 font : 16 byte.

32 * 16 font : 64 byte.

15) User Screen Memory Upload 'COMMAND' : 'V' (0x56)

: 사용자가 미리 load 해 놓은 Screen(전체 화면) image를 display 한다.

User Screen은 총 15개를 사용할 수 있다. (screen load command 참조)

Command	Screen Select	Load Page	Data
U' (0x56)	0x31~0x3f (주1)	0x30~0x3f	128 byte

(주1) Screen Select : 15개의 Screen중에 display 할 Screen을 선택 한다.

Screen은 15개로 1~15 까지이며 여기에 0x30을 더하여 전송 한다.

Ex) 7번 screen을 select : 0x37

(주2) Load Page : 1개의 화면data 는 2048 byte이며 이것을 128 byte 단위로

16번에 나누어 전송하며, 이 128 byte 단위를 "Page"라고 한다.

Page는 0 ~ 15 까지 16개이며, 이 값에 0x30을 더하여 전송한다.

(주3) Data : 한 page 가 128 byte 이므로 128 byte 고정 길이로

binary data이다.

5. User Font & User Screen Download data 구조.

1) Normal font (16 * 8)

byte 0	byte 1	byte 2	byte 3	byte 4	byte 5	byte 6	byte 7
b0	b0	b0	b0	b0	b0	b0	b0
b1	b1	b1	b1	b1	b1	b1	b1
b2	b2	b2	b2	b2	b2	b2	b2
b3	b3	b3	b3	b3	b3	b3	b3
b4	b4	b4	b4	b4	b4	b4	b4
b5	b5	b5	b5	b5	b5	b5	b5
b6	b6	b6	b6	b6	b6	b6	b6
b7	b7	b7	b7	b7	b7	b7	b7
byte 8	byte 9	byte 10	byte 11	byte 12	byte 13	byte 14	byte 15
b0	b0	b0	b0	b0	b0	b0	b0
b1	b1	b1	b1	b1	b1	b1	b1
b2	b2	b2	b2	b2	b2	b2	b2
b3	b3	b3	b3	b3	b3	b3	b3
b4	b4	b4	b4	b4	b4	b4	b4
b5	b5	b5	b5	b5	b5	b5	b5
b6	b6	b6	b6	b6	b6	b6	b6
b7	b7	b7	b7	b7	b7	b7	b7

상기 그림과 같이 윗줄 왼쪽부터 byte0 이며 윗줄 8 byte 아랫줄 8 byte 이다.

이 16 byte를 원하는 font 번호에 load 하면 된다.

2) Large font (32 * 16)

byte 0	byte 1	byte 2	byte 3	byte 4	byte 5	byte 6	byte 7	byte 8	byte 9	byte 10	byte 11	byte 12	byte 13	byte 14	byte 15
b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0
b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1
b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2
b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3
b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4
b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5
b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6
b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7
b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0
b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1
b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2
b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3
b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4
b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5
b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6
b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7
byte 16	byte 17	byte 18	byte 19	byte 20	byte 21	byte 22	byte 23	byte 24	byte 25	byte 26	byte 27	byte 28	byte 29	byte 30	byte 31
byte 32	byte 33	byte 34	byte 35	byte 36	byte 37	byte 38	byte 39	byte 40	byte 41	byte 42	byte 43	byte 44	byte 45	byte 46	byte 47
b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0
b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1
b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2
b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3
b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4
b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5
b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6
b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7
b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0	b0
b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1	b1
b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2	b2
b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3	b3
b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4	b4
b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5	b5
b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6	b6
b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7	b7
byte 48	byte 49	byte 50	byte 51	byte 52	byte 53	byte 54	byte 55	byte 56	byte 57	byte 58	byte 59	byte 60	byte 61	byte 62	byte 63

상기 그림과 같이 윗줄 왼쪽부터 byte0 이며 윗줄 16 byte 둘째 줄 16 byte 셋째 줄 16 byte 넷째 줄 16 byte, 총 64 byte를 byte 번호 순서 대로 원하는 font 번호에 load 하면 된다.

3) User screen

byte 0	byte 1	byte 2	~	byte 125	byte 126	byte 127
bit 0	bit 0	bit 0		bit 0	bit 0	bit 0
bit 1	bit 1	bit 1		bit 1	bit 1	bit 1
bit 2	bit 2	bit 2		bit 2	bit 2	bit 2
bit 3	bit 3	bit 3		bit 3	bit 3	bit 3
bit 4	bit 4	bit 4		bit 4	bit 4	bit 4
bit 5	bit 5	bit 5		bit 5	bit 5	bit 5
bit 6	bit 6	bit 6		bit 6	bit 6	bit 6
bit 7	bit 7	bit 7		bit 7	bit 7	bit 7

상기 그림은 screen의 제일 윗줄 세로 8 pixel * 가로 128 pixel 의 image 영역 이며, 이 구간 즉 128 byte 단위의 Page 가 세로로 16개 로 구성 되어 128 * 128 화면을 구성 한다.

User 는 이 page 단위(128 byte) 로 page 번호를 넣어 upload 하면 되며 일부 page만 update 할 수 있다.

주의 본 data 는 Flash memory 영역을 사용하므로 RAM에서 사용할 때 와 같이 무한

write 가 가능하지 않다. 그러므로 user 의 응용 프로그램이 지속적인 쓰기가 발생하지 않도록 주의해야한다.

6. 시험 순서 및 방법

- (1) PC 에서 시험하기 위하여 시중에서 판매하는 USB to Serial convert를 준비한다.
- (2) “Docklight” (<https://docklight.de/downloads/>) 사의 Docklight V2.2 demo version 을 download 하여 설치 한다.
- (3) Serial port의 Tx 를 JT Module의 Rx에, Serial port의 Rx 를 JT Module의 Tx에 각각 연결 한다.
- (4) USB to Serial converter 의 전원 출력을 JT Module 에 연결 한다.
- (5) Docklight V2.2 를 실행 하고 함께 제공된 Project file (JT_Lcd_Protocol_Test.ptp) 을 load 한다. (HEX mode)
(file -> Open Project... 에서 file 선택)
- (6) 각 항목별로 send icon을 누르면 제목에 있는 내용의 data를 전송하며, JT Module 에서 실행 되는 것을 확인 할 수 있다.
- (7) 이 test는 통신 및 작동을 확인하기 위하여 일부 command 만 입력 되어있으며, 세부적인 제어는 본 사용자 메뉴얼을 참고하여 프로그램 하시면 됩니다.

--- END ---