

GANASYS TFT LCD Control Board

Model: GL-70L-I

Issue Date : 2018.01.02

Revision History :

Date	Version	Page	Summary
2017.12.18	0.1.0		초판 완성
2018.01.02	0.1.1		구성 변경
2018.12.2	0.1.2		명령어 'debug' 수정

목차

- 1. General Information 4
 - 1.1 System Features 4
 - 1.2 Mechanical information 4
 - 1.3 Absolute Maximum Ratings 4
 - 1.3.1 Environmental Absolute Ratings 4
 - 1.3.2 Electrical Characteristics 4
- 2. Block diagram 5
- 3. Connect information 6
- 4. Control Board 치수 8
- 5. 프레임 치수 8
- 6. Application Example 9
- 7. TFT LCD Board 규격 10
 - 7.1 TFT LCD Board 에 그림 넣기(Copy) : Memory 동작 방법 10
 - 7.2 Debug Mode / MPU mode : UART interface mode 11
 - 7.3 명령어 12
 - 7.3.1 그림 그리기 : 명령어 'i' 13
 - 7.3.2 빠른 Image 출력 : 명령어 'ia', 'io', 'ix', 'imc' 13
 - 7.3.1 Sound 제어 : 명령어 's' 15
 - 7.3.2 File list 보기 : 명령어 'ls' 15
 - 7.3.3 명령어 실행 결과 출력 disable / enable : 명령어 'debug' 15
 - 7.3.4 USB [이동식 Memory 인식] : 명령어 'u' 16
 - 7.3.5 Back-light 제어 : 명령어 'l' 16
 - 7.3.6 내장 Font 사용 : 명령어 'f', 'fc' 16
 - 7.3.7 Color table 예 17
 - 7.4 Touch 좌표 출력 18
 - 7.5 지원 Image format 18
 - 7.6 Watchdog Timer 19
- 8. TFT LCD Board 사용 예제 20
 - 8.1 TFT LCD Board에 그림 Data 저장 20
 - 8.2 그림 그리기 21
 - 8.3 그림 그리기 : Double frame 사용 21
 - 8.4 빠른 Image 출력 22
 - 8.5 File list 보기 24
 - 8.6 명령어 실행 결과 출력 disable / enable 24
 - 8.7 Sound 제어 25
 - 8.8 Back-light 제어 26
 - 8.9 Font 출력 26
 - 8.10 Touch 좌표 27

그림 목차

그림 1. Block diagram.....	5	
그림 2. Ext. Connect 위치.....	6	
그림 3. J1 Connect PIN map.....	7	
그림 4. TFT LCD Control Board 치수.....	8	
그림 5. 7" TFT LCD Frame 치수.....	8	
그림 6 Control Board with Frame	그림 7. Control Board with Frame.....	9
그림 8. TFT LCD Control Board and MPU Board.....	9	
그림 9. TFT LCD Control Board and PC.....	9	
그림 10. 좌표 예제.....	12	
그림 11. Color table 예.....	17	
그림 12. 응용 프로그램 그림판의 색 편집 메뉴.....	18	
그림 13. TFT LCD Control Board와 PC 연결.....	20	

표 목차

표 1. J1 connect PIN map.....	7
------------------------------	---

1. General Information

1.1 System Features

- 지원 TFT-LCD : Innolux AT070TN94 7" TFT-LCD 지원
- Memory : NAND 128MByte
- Image : BMP, JPG, PNG, TGA format 지원 : JPG HW Decoder / RGB 565 output
- Back-light control : Dynamic PWM Control
- Touch : 4선 저항막 방식 / Capacitive 방식 (Option)
- Sound : 1ch (Option)
- UART : 2ch [Debug channel / MPU channel] : RS-232 Level
 - ✓ Baud rate : 115200
 - ✓ Data bit : 8
 - ✓ Stop bit : 1
 - ✓ Parity bit : None
- USB : 1ch Device USB (type : micro USB)
- Power : DC +5V
- TFT & Control Board Frame(Optional)

1.2 Mechanical information

Item		Min	Type	Max	Unit	Note
Module size	Horizontal		123.00		mm	
	Vertical		75.00		mm	
Dot Pitch			0.0642(w)x0.1790(h)		mm	Note 1.
Weight			TBD		g	

Note 1 : Dot Pitch 가로x세로 비율 차이에 의하여, 화면 출력 비율이 다를 수 있다.

1.3 Absolute Maximum Ratings

1.3.1 Environmental Absolute Ratings

Item	Min	Max	Unit	Note
Storage temperature	-30	70	°C	
Operating temperature	-20	60	°C	

1.3.2 Electrical Characteristics

Item	Symbol	Min	Type	Max	Unit	Note
Power supply voltage	VDD	-0.3	5	5.5	V	GND = 0
UART voltage	RxIN			± 25	V	
	TxOUT		± 13.2		V	
DC input High voltage	VOH	2.5			V	
DC input Low voltage	VOL			0.8	V	
Current Consumption (TFT LCD Back-light MIN)	IccMIN	360			mA	VDD=+5V
Current Consumption (TFT LCD Back-light MAX + Sound MAX)	IccMAX			750	mA	VDD=+5V

2. Block diagram

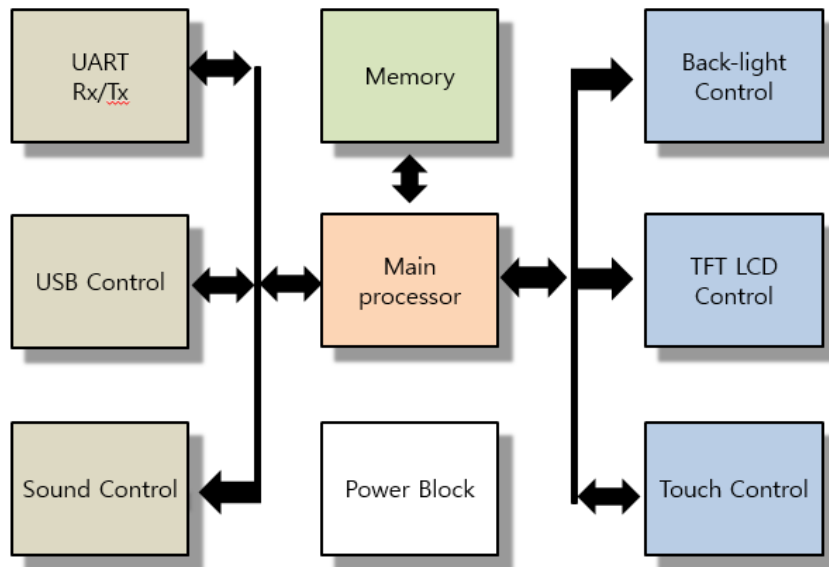


그림 1. Block diagram

3. Connect information

TFT Control board에는 USB, UART, Power, TFT-LCD, 그리고 back-light에 관한 connector가 있다. USB, UART 그리고 Power connector는 보드동작을, TFT-LCD connect와 back-light connector는 화면출력을 위한 것이다. 각각의 기능은 다음과 같다.

- USB : image data를 PC에서 내부 memory로 저장
- UART : 수행 명령어 전달 및 touch 좌표 출력
 - Debug channel : 명령어 전달, touch좌표 output 그리고 모니터링 기능 [pin 6, pin 7]
 - MPU channel : 명령어 전달, touch좌표 output [pin 8, pin 9]
- Power : 동작을 위한 전원 공급 (DC +5V)
- TFT-LCD interface : 7" TFT LCD와 연결
- Touch interface : 7" Touch와 연결

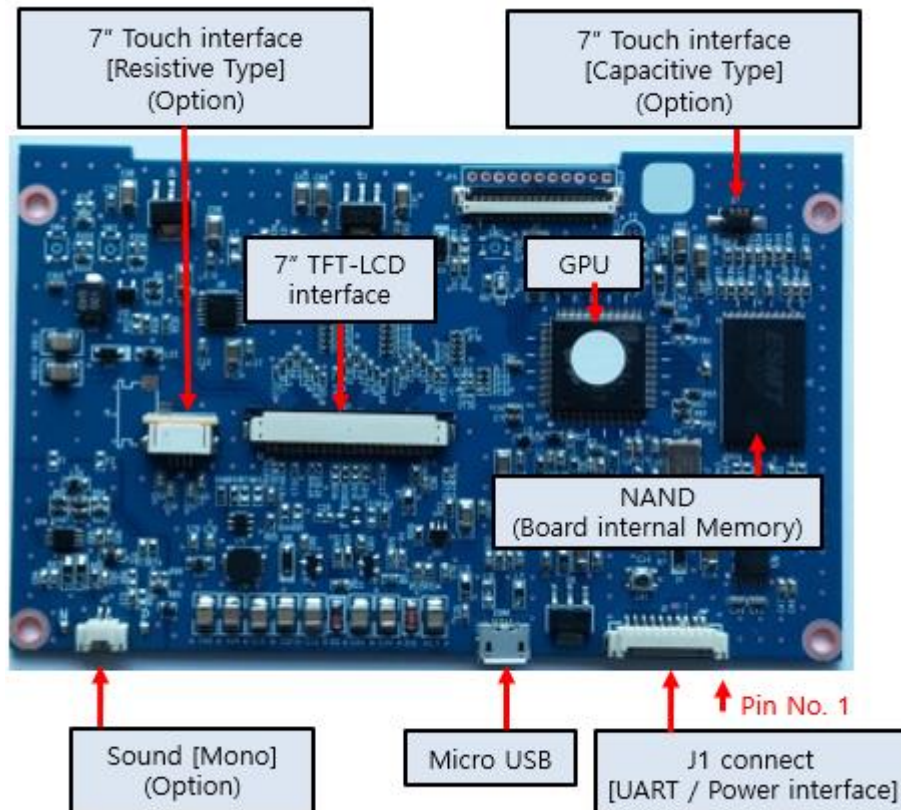


그림 2. Ext. Connect 위치

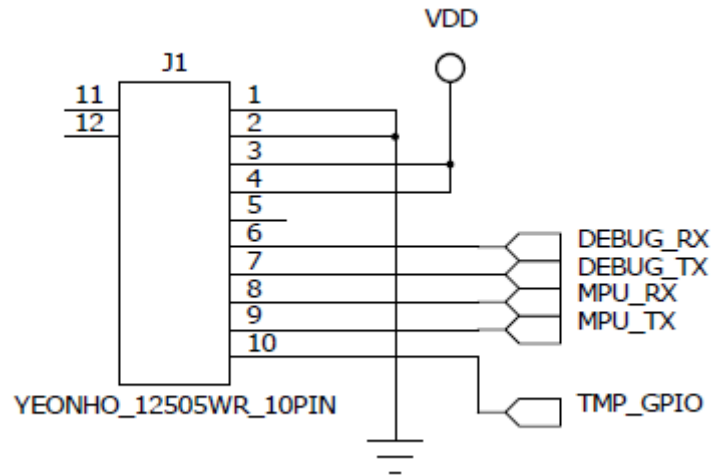


그림 3. J1 Connect PIN map

PIN No.	Symbol	Description	I/O
1	GND	Ground	
2	GND	Ground	
3	VDD	Power DC +5V	I
4	VDD	Power DC +5V	I
5	NC	No connect	
6	DEBUG_RX	Debug UART Rx	I
7	DEBUG_TX	Debug UART Tx	O
8	MPU_RX	MPU UART Rx	I
9	MPU_TX	MPU UART Tx	O
10	TMP_GPIO	Not used	

표 1. J1 connect PIN map

4. Control Board 치수

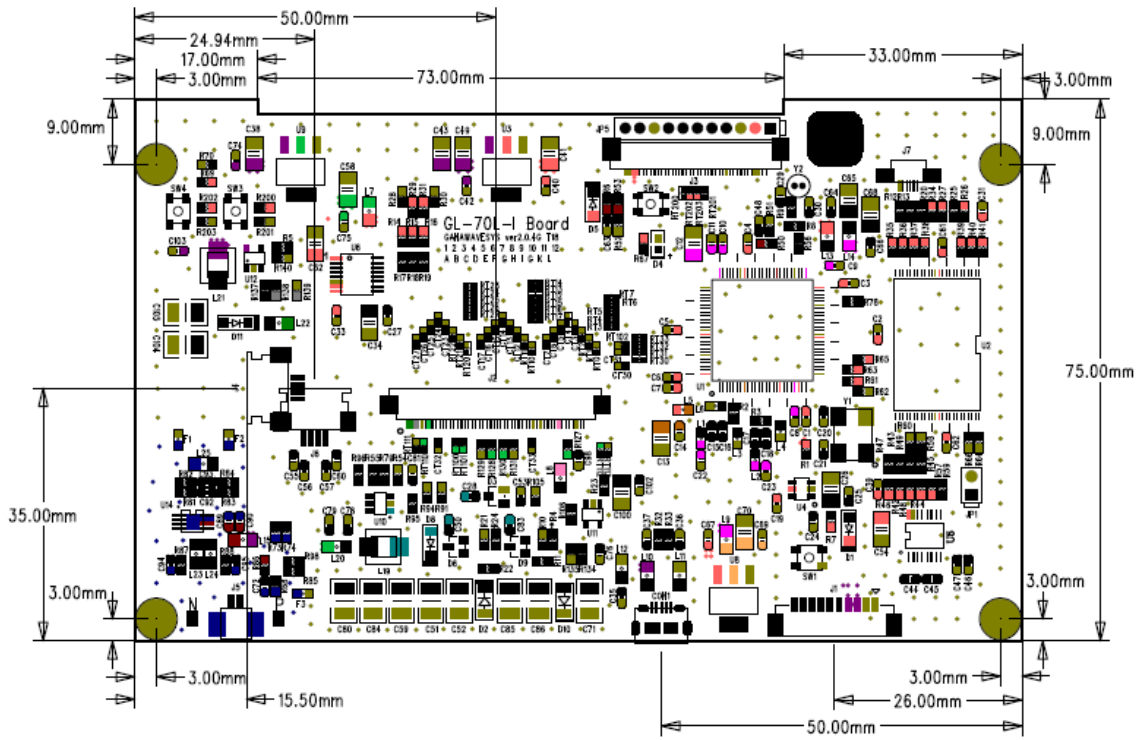


그림 4. TFT LCD Control Board 치수

5. 프레임 치수

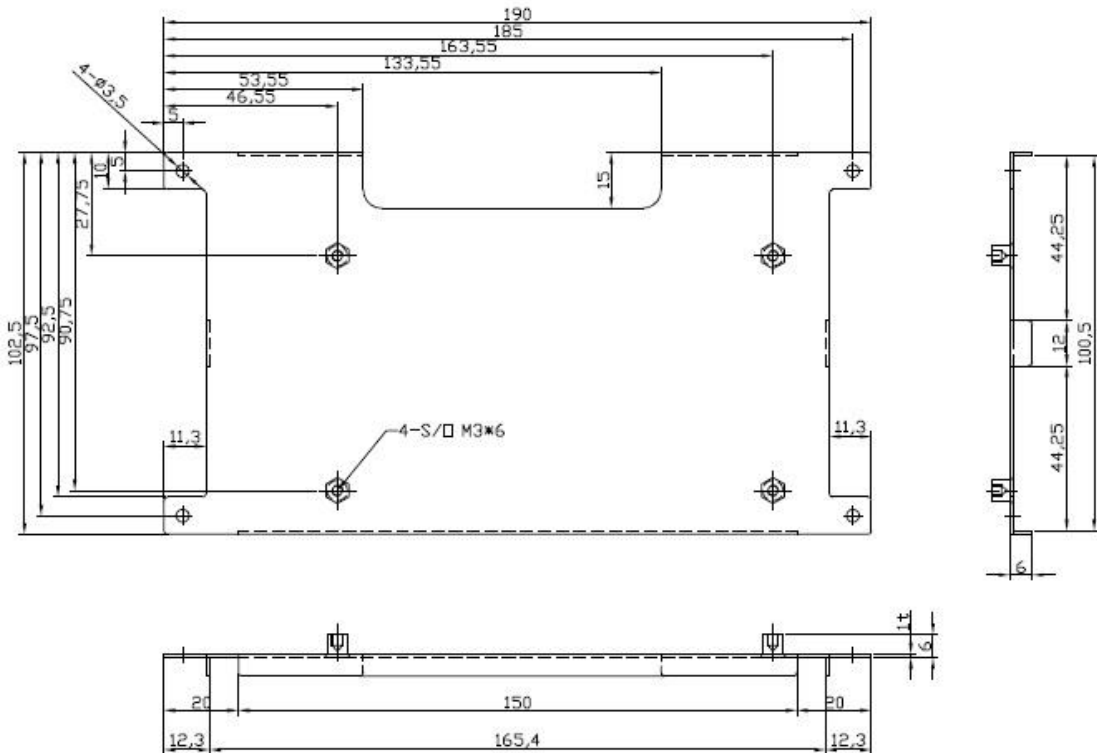


그림 5. 7" TFT LCD Frame 치수



그림 6 Control Board with Frame



그림 7. Control Board with Frame

6. Application Example

TFT LCD control Board는 UART 방식이 가능한 기기 또는 시스템과 연동 동작이 가능하다.

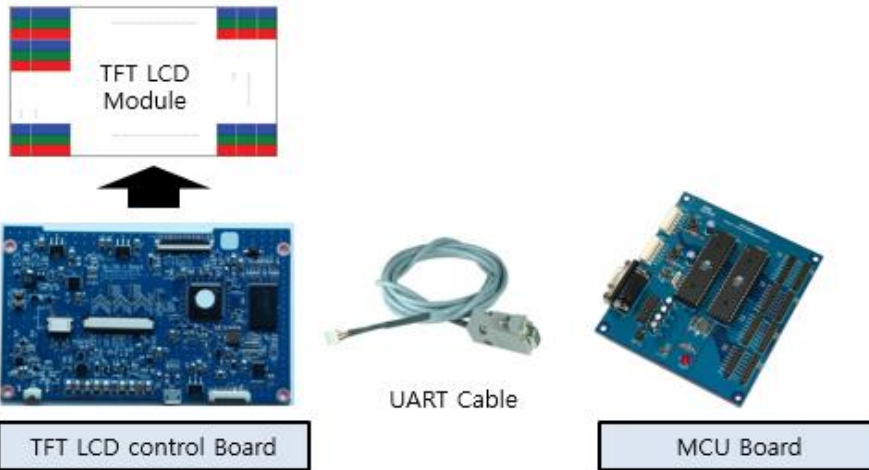


그림 8. TFT LCD Control Board and MPU Board

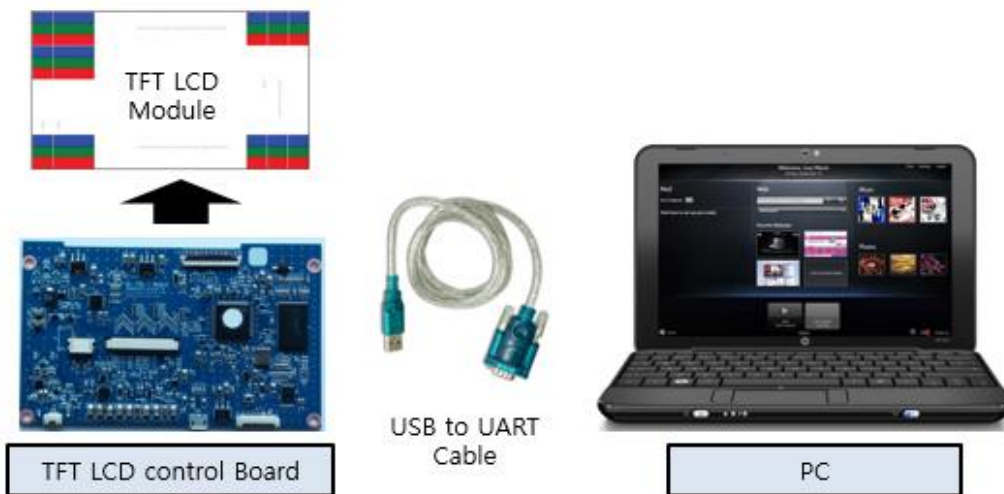


그림 9. TFT LCD Control Board and PC

7. TFT LCD Board 규격

7.1 TFT LCD Board 에 그림 넣기(Copy) : Memory 동작 방법

TFT LCD Board 에 그림을 넣기(Copy) 위해서는 아래의 작업이 선행되어야 한다.

- USB Cable 준비 : Connect 부분 Type 확인
 - Micro USB : TFT LCD Board
- J1 Connect 연결 Cable : UART 통신과 Power 연결한다.
 - 참조 : [표 1] J1 Connect PIN Map
- UART 통신 Cable : RS232 통신 Cable 또는 장비
- UART 통신 프로그램 : 하이퍼터미널 (windows PC 프로그램)
- Power : +5V (1A 이상)

선행작업, 즉, Power와 필요한 Cable이 준비가 되어 있으면, 아래의 순서로 진행한다.

1. USB Cable 연결 : TFT LCD Board에 있는 micro USB 와 PC USB 단자와 연결
2. UART 통신 : TFT LCD Board 의 J1 Connect(Debug Rx/Tx, GND) 와 RS232 Level (UART 통신) 케이블 연결
 - A. 참조 : [표 1] J1 Connect PIN Map
3. PC 응용 프로그램 : UART 통신 응용 프로그램을 실행 한다.
 - A. 예 : 하이퍼터미널 프로그램
4. Power 연결 : TFT LCD Board 의 J1 Connect 에 Power 연결
 - A. 전압 : +5V
5. 연결 확인 : UART 통신 응용 프로그램 창에 문자가 뜬다.
6. 'u' 명령어 입력 : UART 통신 응용 프로그램에 소문자 'u' 를 입력하고 'enter' 키를 누른다.
7. 폴더 확인 : PC 의 폴더 탐색기에서 추가된 '이동식 디스크'를 확인한다.
8. 그림 파일 copy : 준비된 그림 파일을 '이동식 디스크'에 복사 한다.
 - A. 주의 : copy 한 그림 파일의 파일명과 파일유형은 메모해 둔다.
9. Power OFF : 준비된 그림 파일을 모두 Copy 했으면, Power를 Off 한다.
10. 그림 파일 확인
 - A. TFT LCD 연결 상태 확인 : 연결 상태가 아니면 보드와 연결한다.
 - B. UART 연결 상태 확인 : 연결 상태가 아니면 보드와 연결한다.
 - C. Power ON : 보드에 Power 를 연결 한다.
 - D. UART 통신 응용 프로그램에서 'i 파일명.파일유형,0,0' 입력하고 'enter' 키를 누른다.
 - i. 참조 : 6.3.1 그림 그리기 명령어
 - E. Copy 한 그림이 x좌표=0, y좌표=0 에 그려진다.
 - i. 그림 좌표 기준 : 그림 왼쪽, 상단

7.2 Debug Mode / MPU mode : UART interface mode

TFT LCD Board에서는 기본 2개의 UART interface를 제공한다. 각각의 UART는 Debug UART와 MPU UART로 명기하며 기능은 명령어 입력, touch 좌표 출력 그리고 상태 정보 출력이다. 여기서 상태 정보 출력 기능은 Debug UART 만 있는 기능이다.

기능	내용	Debug UART	MPU UART
명령어 입력	외부 명령자로부터 명령어 받음	O	O
Touch 좌표 출력	외부 시스템에 touch panel 좌표 전달	O	O
상태 정보 출력	외부 시스템에 명령어 처리 상태 및 TFT LCD board의 상태를 전달	O	X

※ 통신 설정 (Debug / MPU) : Default

- Baud rate : 115200
- Data bit : 8
- Stop bit : 1
- Parity bit : None

※ MPU UART는 Baud rate를 변경하여 사용할 수 있다. 변경 방법은 TFT LCD board의 메모리 내에 있는 'setup' 폴더 안의 파일명을 수정하면 된다.

다음은 MPU UART에서 지원하는 Baud rate와 설정하고자 하는 Baud rate에 해당하는 파일명 이다.

MPU UART Baud rate	파일 이름, 형식	비고
2400 bps	UART_2400.gbs	
4800 bps	UART_4800.gbs	
9600 bps	UART_9600.gbs	
14400 bps	UART_14400.gbs	
19200 bps	UART_19200.gbs	
38400 bps	UART_38400.gbs	
57600 bps	UART_57600.gbs	
115200 bps	UART_115200.gbs	

- 다음은 MPU UART Baud rate 수정 방법이다.

1. TFT LCD board를 이동식 디스크로 인식
 - A. Debug UART를 이용하여 USB 사용 명령어 'u'를 전달한다.
 - B. 참조 : TFT LCD Board에 그림 Data 저장 예제

2. 'setup' 폴더를 연다.



3. 폴더 내의 파일 확인

A. 만약, 'UART_115200.gbs' 파일이 존재하면 baud rate은 115200으로 설정되어 있다는 의미

이름	수정한 날짜	유형	크기
UART_115200.gbs	2012-04-05 오후...	GBS 파일	1KB

4. MPU UART Baud rate 변경 (예 : 19200 bps)

A. 19200 bps로 변경 : UART_19200.gbs 로 파일명을 수정

이름	수정한 날짜	유형	크기
UART_19200.gbs	2012-04-05 오후...	GBS 파일	1KB

7.3 명령어

명령어는 TFT LCD board를 동작을 수행하는데 사용한다. 명령어를 ASCII code로 구성하여 UART 를 통해 TFT LCD board로 전달하면, 명령어를 해독하여 해당 명령어를 수행한다. 명령어는 그림 그리기, file list 보기, Sound 제어, USB 사용, back-light 제어 등이 있다. 각각의 명령어에 대한 세부 사항은 해당 부분을 참조하기 바란다.

※ 주의 : 명령 마지막에는 꼭 'wr'를 전달한다. 'wr'은 ASCII code의 의미로 'carriage return'이며, PC의 UART 응용 프로그램에서는 enter와 같은 의미이다. TFT LCD board는 'wr'의 ASCII code를 전달받으면, 명령어 문장의 마지막이라는 뜻과 명령어를 실행하라는 의미로 해석하여 명령어를 해석하고 동작하게 된다.

※ 참조 : 그림 출력에 대한 좌표는 아래 그림을 참조하여 X, Y 값을 설정하면 된다.

ex) 그림을 (10,10) 의 좌표에 그린다.

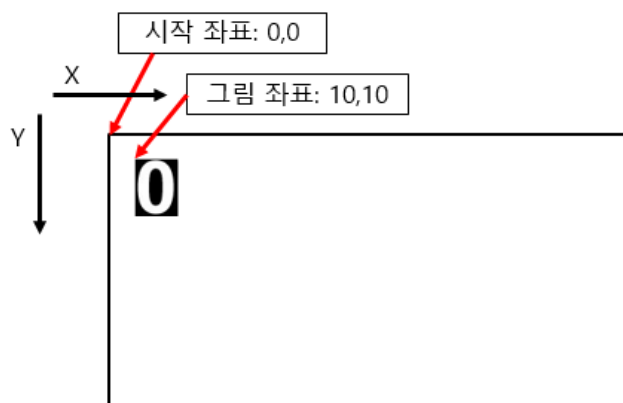


그림 10. 좌표 예제

7.3.1 그림 그리기 : 명령어 'i'

명령어	내용	비고
i[빈칸]image.bmp,x,yWr	image.bmp : 출력 image x : 그림 그리기 x 좌표 y : 그림 그리기 y 좌표	Path 구성 : a/bmp/img.format name : 출력 그림 파일명 format : 출력 그림 유형
<DWr	Double frame start Back frame에 그림 그리기	대문자 명령어
!DWr	Double frame end Back frame 출력	대문자 명령어

※ 그림을 출력하기 위해 대상 그림의 위치(폴더), 그림 이름 그리고 그림의 유형을 명령어 입력에 넣어주어야 한다. 그림의 위치가 폴더 안에 있으면 path 명을 써주어야 하며, 하위 폴더 표시는 '/' 이다.

※ 본 보드에서 지원하는 그림의 유형은 BMP, JPG, PNG, TGA 이다. [참조 '지원 image format' 장]

※ 그림의 유형별 출력 시간은 JPG, BMP가 PNG, TGA에 비해 빠르게 출력된다. (같은 크기 그림 일 경우)

※ <참조> : 특정 응용 프로그램으로 제작된 JPG 파일이 출력 되지 않는 경우가 있다. 이와 같은 경우에는, "Windows OS" 에서 제공하는 응용프로그램 "그림판" 으로, 해당파일을 "다른 이름으로 저장" 에서 "JPEG" 로 저장하고 사용하면 출력된다.

※ 그림 그리기의 x 좌표, y 좌표는 다음과 같다.

→ x 좌표, y 좌표는 그림의 좌측 상단을 기준으로 그림을 그리는 위치를 의미한다.

→ 기준 위치 즉, (0,0)은 TFT LCD 의 좌측 상단이며, 좌표 값은, x 좌표는 우측으로 증가, y 좌표는 아래로 증가한다.

※ 저장/삭제 [그림 파일] : 작업을 하기 위해서는 'u' 명령어를 사용한다. [USB 명령어 참조]

※ Double frame 명령어 "<D"와 "!D"는 꼭, 같이 사용하도록 한다.

7.3.2 빠른 Image 출력 : 명령어 'ia', 'io', 'ix', 'imc'

명령어	내용	비고
ia[빈칸] [address],image.bmpWr	저장 : image 정보를 SDRAM에 저장 image를 [address]에 저장	Path 구성 : a/bmp/img.bmp Address : 0 ~ 510
io[빈칸] [address],x,yWr	출력 : SDRAM에 저장된 image 출력 x : 그림 그리기 x 좌표 y : 그림 그리기 y 좌표	
ix[빈칸] [address]Wr	제거 : SDRAM에 저장된 image 정보 제거	모두 제거 : address 대신 'all'
imcWr	사용 address 확인	

- Image 정보를 SDRAM에 저장하여 출력하는 방식
- SDRAM에 저장되는 image는 address를 부여하여 관리
- 저장 / 출력 / 제거 명령어 [ia / io / ix]
 - 저장 : image 정보를 SDRAM에 address를 부여하여 저장
 - 출력 : address를 호출하여 원하는 x, y 좌표에 출력
 - 제거 : address를 호출하여 저장된 image 정보를 제거
- 사용 중인 address 확인 명령어 [imc]
 - 명령어를 입력하면 사용중인 address가 출력된다.
- 저장된 정보는 전원이 차단되거나 reset 되면 삭제된다.

※ 빠른 image 명령어 사용시 주의 사항

1. SDRAM image 저장 소요 시간

- image 정보를 SDRAM에 저장하기 위해서는 시간이 필요하다. 따라서 여러 개의 image를 저장 하는 경우, 시간 간격을 조절하여 저장명령어를 사용한다.
- 저장되는 시간은 image의 Type 즉, BMP, JPG, PGA 그리고 TGA 에 따라 다르며, 출력 소요 시간에 근접한 시간이 필요하다.

2. SDRAM image 저장 공간

- Image을 저장하기 위해 할당되는 메모리 공간은 파일의 크기에 따른 것이 아닌 image 크기에 의해 결정된다.
- BMP, JPG : 800x480 크기의 image [MAX 12장]
- PNG, TGI : 800x480 크기의 image [MAX 6장]
- 추가 내용
 - ✓ 명령어에 의해 SDRAM으로 전달되는 image 정보는, 파일 형식(BMP, JPG)에 관계 없이, pixel 당 칼라 값으로 변환되어 저장된다.
 - ✓ PNG와 TGI는 pixel 칼라 정보와 투명도 정보가 SDRAM에 저장되어, BMP나 JPG에 비교해 약 2배의 공간이 필요하다.
 - ✓ SDRAM에는 image 정보뿐만 아니라, control board의 System에서 사용되는 정보가 임시 저장된다. 따라서 저장 할 수 있는 image 정보의 양이 다소 유동적이다.
 - ✓ SDRAM의 저장공간보다 많은 image를 저장하면 system이 reset 된다. (메모리 overflow가 발생하면, system 정보에 이상이 발생하게 되며 보드는 이를 방지하기 위해 reset동작을 한다.)
 - ✓ SDRAM에 저장된 image는 Power OFF 되거나, Reset 이 되면 사라지기 때문에, 다시 Power ON 을 진행하면 저장명령어로 재 저장과정을 진행해야 한다.

3. SDRAM image 출력 시간

- image의 크기가 작을수록 빠름.
- [BMP, JPG]가 [PNG, TGI] 보다 약 2배 정도 빠름 : 같은 크기일 경우

7.3.1 Sound 제어 : 명령어 's'

※ 옵션 기능

명령어	내용	비고
sp[빈칸] audio.wavWr	Sound 재생 : audio.wav 파일을 재생한다. 주의 : wav 파일만 재생가능	Path 구성 : a/wav/audio.wav
ssWr	Sound 정지 : 재생 중인 Sound 를 중지한다.	
sv[빈칸]valueWr	Sound 볼륨	볼륨 조정 Level : 0~255 -. 가장 큰 볼륨은 255 이다 (default 볼륨 : 255)

※ 지원하는 Audio Format 은 wav(PCM) 이며, mono 로 출력된다.

※ Sound 재생에서 지연시간이 발생하는 경우는, 재생파일의 재생 시간이 길기 때문에 발생할 수 있다. 즉, 재생 시간이 길게 되는 파일은 재생 파일의 크기가 크다. 따라서 파일을 loading 하는 시간이 많이 필요하며 이에 따른 지연시간이 발생하는 것이다.

※ Sound의 Spec.은 연결되는 Speaker unit과 선에 따라 다르지만, 일반적인 Spec.은 4Ω의 Speaker unit에서 3W의 출력, THD+N는 10%이다.

7.3.2 File list 보기 : 명령어 'ls'

명령어	내용	비고
ls[빈칸] /Sub-DirectoryWr	Sub-Directory 내의 File list 확인	Main path : / ls[빈칸]Wr : 현 폴더 내용 lsWr : main 폴더 내용

※ 명령어 수행 결과에 따른 File list의 결과는 Debug UART와 연결된 기기를 통해 확인이 가능함.

7.3.3 명령어 실행 결과 출력 disable / enable : 명령어 'debug'

명령어	내용	비고
debugWr	명령어 수행 - enable : 결과 전달 수행 - disable : 결과 전달 안함	처리 결과 : <disable에서 enable로 전환> <enable에서 disable로 전환>
debug+onWr	명령어 수행 결과 -. enable : 결과 전달 수행	처리 결과 : <enable 기능 수행>
Debug+offWr	명령어 수행 결과 -. disable : 결과 전달 안함	처리 결과 : <disable 기능 수행>

- 명령어 수행 결과를 UART로 전달 : enable 일 경우

- 명령어 수행 결과를 UART로 전달 하지 않음 : disable 일 경우
- ※ 명령어 실행 결과 출력을 disable 하면, 통신 data의 감소로 image 출력 속도가 개선된다.
- ※ disable 상태에서도 일부 명령어 수행결과와 오류에 대한 메시지는 출력이 된다.

7.3.4 USB [이동식 Memory 인식] : 명령어 'u'

명령어	내용	비고
uWr	PC 연동	PC에서 이동식 memory로 인식

- PC 의 USB 단자와 연결 : 본 보드의 USB 단자와 PC USB 단자를 연결한다.
- 명령어 'u' 수행 결과 : 정상적인 연결상태에서 명령어를 수행하면, PC 의 파일 탐색기에서 본 보드가 이동식 Memory 가 연결되어 있는 것을 볼 수 있다.
- ※ 명령어 'u' 가 정상적으로 동작하면, 다른 명령어 수행이 이루어지지 않는다.
 - ▶ 다른 명령어를 수행하기 위해서는, TFT Control Board 전원을 'Power Off → Power ON' 작업을 진행한다.

7.3.5 Back-light 제어 : 명령어 'l'

명령어	내용	비고
[빈칸] nWr	Back-light 밝기 n Min : 0 Max : 99	100단계 조절
[빈칸] +/-Wr	+ : back-light ON - : back-light OFF	

7.3.6 내장 Font 사용 : 명령어 'f', 'fc'

좌표 출력	내용	비고
f[빈칸]문자,x,yWr	문자를 x, y 좌표에 출력 문자 : 출력하고자 하는 문자 및 기호 ¹⁾ x : 문자 출력 x 좌표 y : 문자 출력 y 좌표	특수문자:침표[,]와 사선 [] 출력 ¹⁾
fc[빈칸]r,g,bWr	문자 color 설정 r : 문자 Color Red 값 g : 문자 Color Green 값 b : 문자 Color Blue 값	Color 값은 0~255 Default : 검정색

- 영문 및 기호는 ASCII Code로 16x24 크기

- 한글은 KSC5601 Code로 24x24 크기
- 문자색 설정 명령어로 문자색 변경가능
- 하나의 폰트 종류만 지원

※ 1) 특수 문자로 사용되는 쉼표[,]와 사선[/] 출력

예제: 쉼표 사용

>> f[빈칸]test : 가나다라/,ABCD,100,100Wr

[결과] test : 가나다라,ABCD

예제 : 사선 사용

>> f[빈칸]C://AAA//출력 폰트,50,100Wr

[결과] C://AAA/출력 폰트

7.3.7 Color table 예

다음은 기본 그리기 함수에 사용되는 color 값은 R, G, B 즉, red, green, blue로 이루어졌으며 각각의 color는 0~255의 값으로 설정하여 다양한 종류의 color를 표현할 수 있다.













			
Black 0, 0, 0	Dark Red 128, 0, 0	Red 255, 0, 0	Pink 255, 0, 255
			
Teal 0, 128, 128	Green 0, 128, 0	Bright Green 0, 255, 0	Turquoise 0, 255, 255
			
Dark Blue 0, 0, 128	Violet 128, 0, 128	Blue 0, 0, 255	Gray 25% 192, 192, 192
			
Gray 50% 128, 128, 128	Dark Yellow 128, 128, 0	Yellow 255, 255, 0	White 255, 255, 255

그림 11. Color table 예

원하는 color 값을 알고자 할 경우 Microsoft Windows의 그림판을 이용해서 간단하게 color table 정보를 알 수 있다. 그림판 응용 프로그램을 실행하고 '색 편집' 메뉴를 클릭하면 다음의 창이 뜬다. 우측 하단에 빨강, 녹색, 파랑이라고 명시된 부분은 색을 지정하면 10진수의 값이 나온다. 이 값을 기본 그리기 함수의 color 값인 R, G, B에 맞추어서 넣어주면 원하는 색상으로 점, 선, 원, 타원, 사각형, 라운드 사각형 등을 그릴 수 있다.

단, TFT LCD module에 따라 PC의 모니터에 출력된 색상과 100% 일치하지 않을 수 있으므로 주의로 요한다.

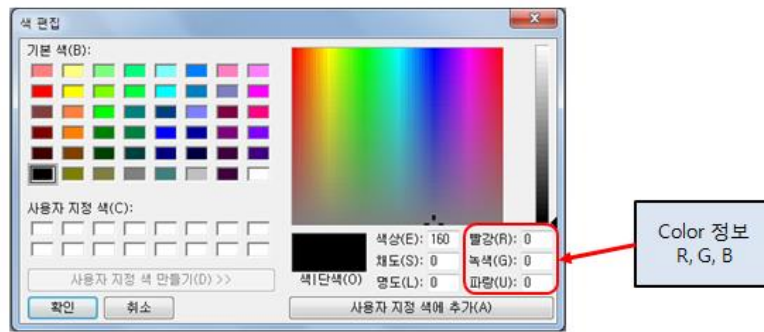


그림 12. 응용 프로그램 그림판의 색 편집 메뉴

※ 주위 : RGB 565 출력이므로, RGB 888로 표현된 color가 모두 LCD에 출력되지 않는다.

7.4 Touch 좌표 출력

Touch 좌표는 touch panel의 값을 좌표 값으로 전환하여 UART로 전달한다. 전달되는 data의 구성은 ASCII code이다.

출력되는 프로토콜은 다음과 같다.

좌표 출력	내용	비고
t(x,y)	Touch x,y 좌표 값을 전달 min : 0 max : TFT LCD size	Output value : ASCII code
<t>	Touch 누름 : Touch data start	Output value : ASCII code
!t!	Touch 띄움 : Touch data end	Output value : ASCII code

7.5 지원 Image format

TFT LCD Board 에서 지원하는 image format 은 BMP, JPG, PNG 그리고 TGA이다.

※ 주의 : 각각의 image format에 대해 지원하는 pixel 당 bit는 다음과 같다. 따라서 사용하고자 하는 image에 대한 정보를 확인하여 지원되는 format과 pixel 당 bit인지 체크한다. 만약 지원되지 않는 image 인 경우, PC에서 '그림판'과 같은 image 편집 응용 프로그램을 사용하여 수정하도록 한다.

※ 주의 : RGB 565 출력이므로, RGB 888로 표현된 color가 모두 LCD에 출력되지 않는다.

- BMP : 24bit/pixel
- JPG : 24bit/pixel
- PNG : 32bit/pixel
- TGA : 32bit/pixel

7.6 Watchdog Timer

예상하지 못한 System 이상으로 TFT LCD control board가 정상적인 동작이 진행되지 않을 경우, 이를 control board 내의 watchdog timer가 감지하여 정상상태로 복귀하는 기능이다. Watchdog Timer가 동작을 하면 control board는 자동 reset를 하여 board의 초기 상태가 된다.

8. TFT LCD Board 사용 예제

8.1 TFT LCD Board에 그림 Data 저장

- ① 준비
 - ✓ PC : Microsoft Windows 운영체제
 - ✓ 저장 image 준비 : 지원하는 format에 맞는 image를 PC 내에 준비
 - ✓ 응용 프로그램 준비 : PC UART 응용 프로그램 (예 : 하이퍼터미널)
 - ✓ Cable 준비
 - USB cable 준비 : data 저장용
 - TFT LCD control board 연결은 micro USB Type
 - UART cable 준비 : 명령어 전달용
 - PC port에 따라 Serial(RS-232) 혹은 USB-to-Serial
- ② TFT LCD Board 연결
 - ✓ USB cable 연결 : micro USB Type
 - ✓ UART cable 연결 : J1 connect 참조
 - ✓ Power 연결 : DC +5v 전원 연결 (500mA 이상) : J1 connect 참조
- ③ 명령어 입력 : UART 사용
 - ✓ 응용 프로그램 실행 : PC에서 UART 응용 프로그램 실행 (예 : 하이퍼터미널)
 - ✓ UART 명령어 입력 : UART 응용 프로그램으로 'USB 사용'명령어 입력
 - 'u[엔터]'
- ④ PC 작업
 - ✓ Microsoft Windows 탐색기 실행
 - ✓ 탐색된 외부 메모리에 그림 data 저장
- ⑤ 완료
 - ✓ 저장 작업 완료
 - ✓ TFT LCD Control Board의 전원을 OFF
- ⑥ 저장 data 사용
 - ✓ TFT LCD Control Board의 전원을 ON
 - ✓ 명령어를 입력하여 저장된 image 사용



그림 13. TFT LCD Control Board와 PC 연결

8.2 그림 그리기

그림 그리기 명령어는 TFT LCD에 그림을 출력 할 때 사용한다. 그림을 그리기 위해서는 그림 파일의 정보 즉, 파일의 폴더, 이름, 그리고 확장자가 필요하며 그리고자 하는 위치를 명령어와 같이 TFT LCD Board로 보낸다.

※ 그림 파일은 보드의 내부 메모리에서 불러온다. 따라서 '그림 그리기' 명령어에 사용할 그림 파일은 사전에 'USB 사용' 명령어를 참조하여 내부 메모리에 저장되어 있어야 한다.

A. PC의 UART 응용 프로그램 사용

□ A.bmp 그림을 x 좌표 50, y 좌표 80에 그리기

- >> i A.bmp,50,80[엔터]

□ IMG 폴더 내에 B.bmp 그림을 x 좌표 30, y 좌표 50에 그리기

- >> i IMG/B.bmp,30,50[엔터]

B. Micom 의 UART 사용

□ A.bmp 그림을 x 좌표 50, y 좌표 80에 그리기

□ IMG 폴더 내에 B.bmp 그림을 x 좌표 30, y 좌표 50에 그리기

main()

{

printf("i A.bmp,50,80\r");

printf("i IMG/B.bmp,30,50\r");

}

8.3 그림 그리기 : Double frame 사용

Double frame 명령어를 사용하여 그림을 그린다. 그림을 그리기 위해서는 7.2절에서 설명했듯 그림이 board 내부 메모리에 저장되어 있어야 한다. 내부 메모리 저장 방법은 'USB 사용 명령어' 부분을 참조한다.

Double frame 명령어를 사용하여 그림을 그리면, 여러 개의 조그만 그림을 같은 순간에 그리도록 할 수 있다.

A. PC의 UART 응용 프로그램 사용

□ A1.bmp는 (10,10), A2.bmp는 (100,10), A3.bmp는 (200,10), A4.bmp는 (300,10)에 그리기

- >> <D[엔터]

- >> i A1.bmp,10,10[엔터]

- >> i A2.bmp,100,10[엔터]

- >> i A3.bmp,200,10[엔터]

- >> i A4.bmp,300,10[엔터]

- >> !D[엔터]

B. Micom 의 UART 사용

□ A1.bmp는 (10,10), A2.bmp는 (100,10), A3.bmp는 (200,10), A4.bmp는 (300,10)에 그리기
main()

```

{
    printf("<DWr");
    printf("i A1.bmp,10,10Wr");
    printf("i A2.bmp,100,10Wr");
    printf("i A3.bmp,200,10Wr");
    printf("i A4.bmp,300,10Wr");
    printf("!DWr");
}

```

8.4 빠른 Image 출력

빠른 Image 출력 명령어를 사용하여 그림을 그린다. SDRAM에 저장하고 불러오는 순서로 명령어를 입력하여 원하는 위치에 그림을 그린다. SDRAM의 용량이 크지 않기 때문에 계속 사용하지 않는 그림 정보는 제거 명령어로 제거한다.

※ 그림 파일은 보드의 내부 메모리에서 불러온다. 따라서 '빠른 Image 출력' 명령어에 사용할 그림 파일은 사전에 'USB 사용' 명령어를 참조하여 내부 메모리에 저장되어 있어야 한다.

A. PC의 UART 응용 프로그램 사용

□ A1.bmp를 Address 0에 저장하고 X 좌표 10, Y 좌표 10에 출력 : X 좌표 20, Y 좌표 40 출력

1. A1.bmp을 Address 0에 저장
⇒ >> ia 0,A1.bmp[엔터]
2. A1.bmp가 저장되어 있는 Address 출력. X 좌표 : 10, Y 좌표 : 10
⇒ >> io 0,10,10[엔터]
3. A1.bmp가 저장되어 있는 Address X 좌표 : 20, Y 좌표 : 40
⇒ >> io 0,20,40[엔터]

□ A2.bmp를 Address 1에 저장하고 X 좌표 50, Y 좌표 100 에 출력 : X 좌표 30, Y 좌표 120 출력

1. A2.bmp을 Address 1에 저장
⇒ >> ia 1,A2.bmp[엔터]
2. A2.bmp가 저장되어 있는 Address 출력. X 좌표 : 50, Y 좌표 : 100
⇒ >> io 1,50,100[엔터]
3. A2.bmp가 저장되어 있는 Address 출력. X 좌표 : 30, Y 좌표 : 120
⇒ >> io 1,30,120[엔터]

□ 저장되어 있는 Address 하나씩 제거 : A1.bmp→Address 0 / A2.bmp→Address 1

1. 저장되어 있는 모든 Address 제거
⇒ >> ix 0[엔터]
⇒ >> ix 1[엔터]

- 'imc' 명령어로 저장되어 있는 Address 확인
 1. A1.bmp, A2.bmp, A3.bmp을 Address 5,6,7 에 저장
 - ⇒ >> ia 5,A1.bmp[엔터]
 - ⇒ >> ia 6,A2.bmp[엔터]
 - ⇒ >> ia 7,A3.bmp[엔터]
 2. 저장되어 있는 Address 확인
 - ⇒ >> imc[엔터]
 - ⇒ #+ Image SDRAM Address
 - ⇒ [5] [6] [7]

- 'all'을 입력하여 저장되어 있는 모든 Address 한번에 제거
 2. 저장되어 있는 모든 Address 제거
 - ⇒ >> ix all[엔터]
 - ⇒ #+ All SDRAM Clear #
 3. 저장되어 있는 Address 확인
 - ⇒ >> imc
 - ⇒ #+ Image SDRAM Address #
 - ⇒ ++ All SDRAM : clear ++

B. Micom 의 UART 사용

- A1.bmp를 Address 0에 저장, A2.bmp를 Address 1에 저장
- Address 0을 (10,10) 에 출력, Address 1을 (50,100)에 출력

```
main()
{
    printf("ia 0,A1.bmpWr");
    printf("ia 1,A2.bmpWr");
    printf("io 0,10,10Wr");
    printf("io 1,50,100Wr");
}
```

- Address 0, Address 1을 하나씩 제거

```
Delete_Address()
{
    printf("ix 0Wr");
    printf("ix 1Wr");
}
```

- 'imc' 명령어로 저장되어 있는 Address 확인

```
main()
{
    printf("imcWr");           // 저장되어 있는 Address가 return 된다
}
```

```
}
```

□ 'all'을 입력하여 저장되어 있는 모든 Address 한번에 제거

```
main()
```

```
{
```

```
    printf("ix allWr");
```

```
}
```

8.5 File list 보기

File list 보기 명령어는 Board 내부 메모리의 File 정보를 읽어 올 때 사용한다. Debug UART를 통해 정보를 읽어 오기 때문에 사전에 Debug UART 와 연결이 되어야 한다. 읽어온 정보는 ASCII code로 이루어져 있다.

A. PC의 UART 응용 프로그램 사용

□ 메인 폴더 File list 보기

- >> ls /[엔터]

□ IMG 폴더 내 File list 보기

- >> ls /IMG [엔터]

B. Micom 의 UART 사용

□ 인 폴더 File list 보기

□ IMG 폴더 내 File list 보기

```
main()
```

```
{
```

```
    printf("ls /Wr");
```

```
    printf("ls /IMGWr");
```

```
}
```

8.6 명령어 실행 결과 출력 disable / enable

TFT control board는 명령어 입력, 그리고 입력된 명령어의 실행이 정상적으로 진행되었는지 잘못되었는지의 상태를 출력하는 것이 기본으로 되어 있다. 그러나 이와 같은 출력은 UART통신으로 주고받는 data양의 증가로 전체 시스템의 진행에 영향을 줄 수 있다. 명령어 실행 결과 출력을 disable 하면 시스템의 진행이 개선된다.

C. PC의 UART 응용 프로그램 사용

□ 명령어 실행 결과 출력이 enable인 상태일 경우 명령어를 입력 하면 disable 된다.

- >> debug[엔터]

- #+ OK Command [Debug Output disable] #

- >> [엔터] : [키 입력이 나타나지 않는다]

- >> [명령어 수행 결과가 나타나지 않는다]

- 명령어 실행 결과 출력이 disable인 상태일 경우 명령어를 입력 하면 enable 된다.
 - >> debug[엔터]
 - #+ OK Command [Debug Output enable] #
 - >> i A.bmp,0,0[엔터] : [키 입력이 나타난다]
 - >> #+ OK Command [Image Draw] # : [명령어 수행 결과가 나타난다]

※ 일부 명령어 수행결과와 오류에 대한 메시지는 출력이 된다.

D. Micom 의 UART 사용

- 명령어 실행 결과 출력이 enable인 상태일 경우 명령어를 입력 하면 disable 된다.

```
main()
```

```
{
```

```
    printf("debugWr");
```

```
}
```

- 명령어 실행 결과 출력이 disable인 상태일 경우 명령어를 입력 하면 enable 된다.

```
main()
```

```
{
```

```
    printf("debugWr");
```

```
}
```

8.7 Sound 제어

Sound 사용 명령어는 GPU에 있는 DAC을 사용하여 Sound를 만들 때 사용한다. 지원되는 Format은 wav 이며, Mono 로 출력된다..

A. PC의 UART 응용 프로그램 사용

- A.wav 의 Audio 파일을 재생
 - >> sp A.wav[엔터]
- wav 폴더 안에 있는 B.wav 의 Audio 파일을 재생
 - >> sp wav/B.wav[엔터]
- 재생 중인, Sound 볼륨 조정
 - >> sv 200[엔터] // 200 으로 볼륨 조정
 - >> sv 100[엔터] // 100 으로 볼륨 조정
- 재생 중인, Sound 정지
 - >> ss[엔터]

B. Micom 의 UART 사용

- A.wav 의 Audio 파일을 재생 후, 1000msec 후에 정지
- wav 폴더 안에 있는 B.wav 의 Audio 파일을 재생 후, 볼륨 100으로 조정
- 2000msec 후에, 재생 중인, Sound 정지

```
main()
{
    printf("sp A.wavWr");
    delayms(1000);
    printf("ssWr");
    printf("sp wav/B.wavWr");
    delayms(100);
    printf("sv 100Wr");
    delayms(2000);
    printf("ssWr");
}
```

8.8 Back-light 제어

Back-light 사용 명령어는 TFT LCD module 의 back-light의 밝기 및 on/off를 제어할 때 사용한다.

A. PC의 UART 응용 프로그램 사용

Back-light off

- >> | -[엔터]

Back-light on

- >> | +[엔터]

Back-light 밝기 80 단계

- >> | 80[엔터]

B. Micom 의 UART 사용

Back-light off

Back-light on

Back-light 밝기 80 단계

```
main()
{
    printf("l -Wr");
    delayms(1000);
    printf("l +Wr");
    printf("l 80Wr");
}
```

8.9 Font 출력

TFT-LCD board에서 제공하는 기본 Font를 사용하여 원하는 문자나 기호를 출력한다. 제공되는 Font의 종류는 하나이다. 문자의 크기는 영어와 기호는 16x24, 한글은 24x24 이다. 문자의 칼라를 설정하여 문자의 색을 바꿀 수 있다.

※ 영어와 기호 code : ASCII code

※ 한글 code : KSC5601 code

A. PC의 UART 응용 프로그램 사용

- 'abcd-ABCD-0123' 을 x 좌표 50, y 좌표 70에 검은색 출력
 - >>f abcd-ABCD-0123,50,70[엔터]
- '가나다라각난달랄' 을 x 좌표 50, y 좌표 150에 검은색 출력
 - >>f 가나다라각난달랄,50,150[엔터]
- '가나다라각난달랄' 을 x 좌표 50, y 좌표 150에 빨간색 출력
 - >>fc 255,0,0[엔터]
 - >>f 가나다라각난달랄,50,150[엔터]
- 쉽표[,] 사용 : '가,나,다,라' 을 x 좌표 10, y 좌표 0에 검은색 출력
 - >>fc 0,0,0[엔터]
 - >>f 가,나,다,라,10,0[엔터]
- 사선[/] 사용 : '0.01 = 0/100 [결과]' 을 x 좌표 10, y 좌표 40에 검은색 출력
 - >>fc 0,0,0[엔터]
 - >>f 0.01 = 0/100 [결과],10,40 [엔터]

B. Micom 의 UART 사용

- 'abcd-ABCD-0123' 을 x 좌표 50, y 좌표 0에 검은색 출력
- '가나다라각난달랄' 을 x 좌표 50, y 좌표 30에 검은색 출력
- '마바사자차카하' 을 x 좌표 50, y 좌표 60에 빨간색 출력
- 쉽표[,] 사용 : '가,나,다,라' 을 x 좌표 50, y 좌표 90에 검은색 출력
- 사선[/] 사용 : '0.01 = 0/100 [결과]' 을 x 좌표 50, y 좌표 120에 검은색 출력

```
main()
{
    printf("f abcd-ABCD-0123,50,0Wr");
    printf("f 가나다라각난달랄,50,30Wr");
    printf("fc 255,0,0Wr");
    printf("f 마바사자차카하,50,60Wr");
    printf("fc 0,0,0Wr");
    printf("f 가,나,다,라,50,90Wr");
    printf("f 0.01 = 0//100 [결과],50,120Wr");
}
```

8.10 Touch 좌표

TFT-LCD board는 Touch panel에서 받은 Touch data를 위치 정보인 x 좌표 값과 y 좌표 값으로 변환하여 UART로 출력한다. 출력되는 Touch 값은 다음의 3부분으로 되어있다. Touch data 전송 시작 부분과 끝 부분 그리고 좌표 값 부분이다. 시작 부분은 최초 Touch panel를 눌렀을 때, 출력되는 data로 최초 한번 출력된다. 끝 부분은 Touch panel를 놓는 마지막 순간에 한번 출력된다. 좌표 값은 Touch panel를 누르고 있는 동안 n개가 출력되며, 현 Touch를 누르고 있는 위치 정보를 담고 있다. 따라서 좌표 값은 n개가 출력된다. 각각의 출력되는 data는 ASCII code의 규격에 따라 UART로 출력된다.

구성 : [touch 시작(눌림)] [touch 좌표 data n개] [touch 끝(놓음)]

□ touch 시작(눌림) : <t>

□ touch 좌표 : t(x,y)

□ touch 끝(놓음) : !t!

예) <t>t(343,92)t(343,91)t(343,92)t(347,95)t(345,92)t(344,92)t(346,94)!t!

Touch 값은 Debug UART와 MPU UART로 전달 받을 수 있으며 data format은 ASCII code로 되어 있다.

주의

1. 지원되지 않는 모든 기기와의 연결과정에서 발생하는 모든 불량은 당사에서 책임지지 않습니다. (교환 및 AS 불가)
2. 제품 구동에 필요한 액세서리는 기본만 제공되며, 그 외에는 별도 구매하셔야 합니다.
3. 제품에 대한 초기불량은 교환. 제품의 Spec.을 만족하는 환경에서 사용 중, 발생하는 불량에 대해 3개월(제품 구매 후)까지 발생하면 무상 AS 처리합니다.
4. 제품이 물리적인 충격에 의해 파손될 경우 교환 및 AS 불가합니다.
5. 제품 구성 및 사용설명서 내용은 사전통보 없이 변경될 수 있습니다.
6. 본 문서의 내용은 보드제작회사에서 권리를 포함하고 있으므로, 허가 없이 무단으로 사용, 복제, 배포하는 행위는 엄격히 금지합니다.

보드 제작 회사 정보

주소 : 서울시 금천구 가산동 481-4 벽산디지털밸리 6차 1209호

전화 : 02-2681-5611

홈페이지 : www.ganasys.co.kr

Copyright 2013. Ganasys CO.,LTD.