

JUTF M-A User Guide

V2.0

4.3"/7" Color TFT LCD & Serial Control Module

T&1



(주)티앤원

목 차

1. 개요 -----	3
1.1 주요 사양(Hardware) -----	3
1.2 제품 주요 기능 -----	3
1.3 지원 이미지 파일 -----	5
2. 구성 -----	6
2.1 기본 구성 -----	6
2.2 추가 구성 -----	6
3. 제품 형상 및 치수 정보 -----	7
4. 외부 Host Controller Interface -----	8
5. Booting Mode -----	10
5.1 Normal Mode (일반동작모드) -----	10
5.2 USB Mass Storage Mode (USB 외장 저장장치모드) -----	11
6. 동작 상태 제어 -----	12
7. Control Commands -----	13
7.1 System Command -----	13
7.2 Primitive Command -----	14
7.3 Image Command -----	16
7.4 Input -----	18
7.5 Sound Command -----	18
7.6 Text out Command -----	19
8. Examples -----	20
8.1 Primitive Command -----	20
8.2 Image Command -----	21
8.3 Touch Test -----	22
[추가 자료] -----	23
A. Case 치수 정보	
B. 4.3인치 LCD용 브라켓 치수 정보	
C. 7인치 LCD용 브라켓 치수 정보	

알림

본 제품의 사용자 가이드는 사전통보없이 변경될 수 있습니다.

본 제품의 기능은 성능 개선 및 추가 등을 위해 사전통보없이 변경될 수 있습니다.

본 제품은 [전파법] 제58조의2 제3항에 따라 적합 등록된 KC인증 제품입니다.



- | | |
|-------------|------------------------|
| 1. 제품명칭 | : Touch TFT LCD Module |
| 2. 기본모델명 | : JUTF M-A |
| 3. 등록번호 | : R-R-JFs-JUTFM-A |
| 4. 제조자/제조국가 | : 주식회사 티앤원 / 한국 |

경고



제품의 설치 및 사용시 해당 작업장의 안전 수칙을 준수해 주시기 바랍니다.

조립 및 설치 작업을 하기 전 반드시 전원을 OFF 하여 주십시오.

제품 내부에 전기가 통할 수 있는 금속성 이물질 유입을 금지해 주십시오.

인명과 관련된 의료 시스템 및 고압, 고온, 폭발성 위험이 있는 작업장 등 위험한 작업 환경에서 사용 및 적용을 금지하여 주십시오.

주의



제품의 정격전압, 단자연결방법, 단자 위치 등을 확인한 후, 정확하게 배선 작업을 해주십시오.

제품 설치 및 단자 배선 시 해당 볼트/너트들의 체결을 단단하게 해 주십시오.

제품에 직접적인 진동 인가를 금지해 주십시오.

발화성, 가연성 물질이 있는 주변에서 설치 및 작업을 금지해 주십시오.

사용 환경의 조건에서 사용해 주십시오.

입출력단 연결 시 부하는 규정된 정격 이내로 설정하여 적용해 주십시오.

전원



전원 케이블 연결 후, 디스플레이와 스피커 출력이 안된다면 전원공급 문제입니다.

작동할 수 있도록 충분한 전원을 공급해주십시오.

관련 내용은본 가이드 매뉴얼의 8page-외부 Host Controller Interface 부분을 참조해 주십시오.

제품문의 : tech@tn1.co.kr, 051-791-2406

1. 개요

JUTF M-A은 TTL Level의 RS232통신으로 4.3"(해상도 480x272) 또는 7"(해상도 800x480) Color TFT LCD를 쉽게 사용할 수 있도록 한 제품입니다.

Arduino, Raspberry Pi 등의 Open Platform 외에도 직접 설계 제작한 8bit/16bit/32bit MCU가 적용된 장치에서도 TTL Level의 RS232 통신만 지원된다면, JUTF M-A 사용이 가능합니다.

USB To RS232 Convert Cable을 사용하면, PC와 JUTF M-A을 연결하여 사용할 수 있습니다.

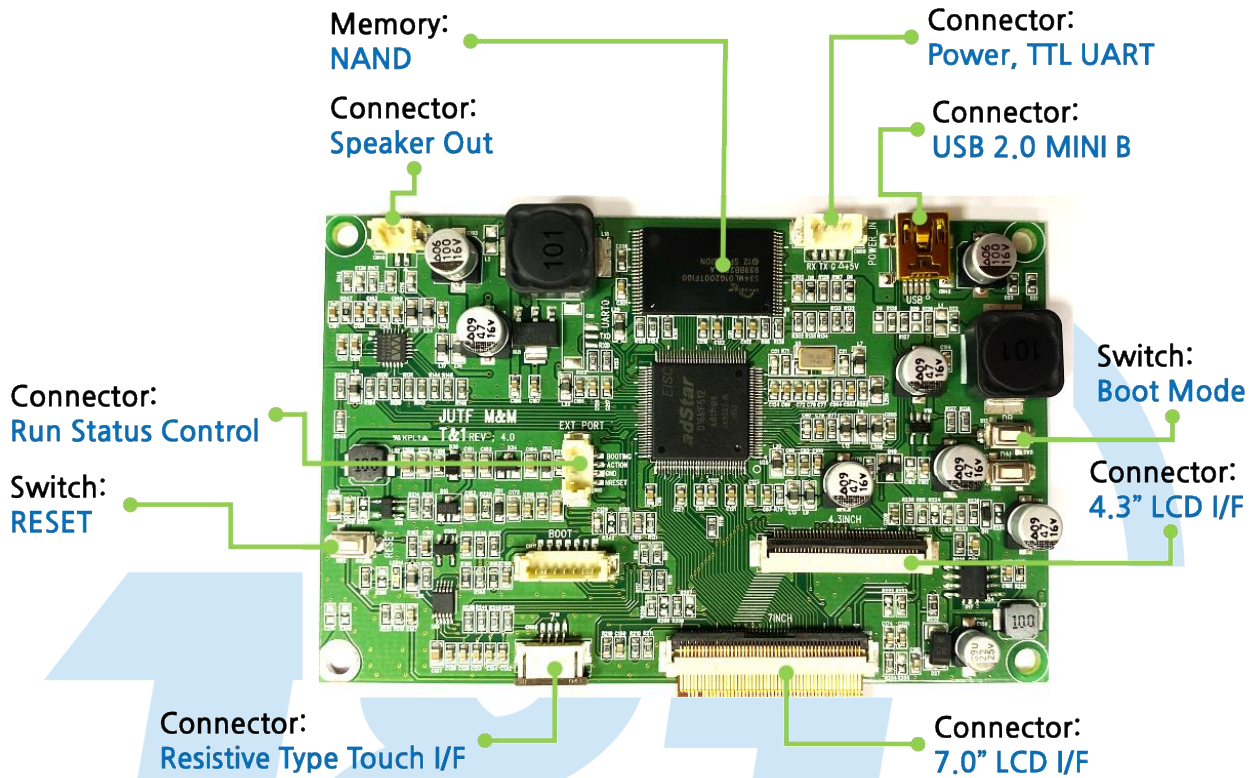
JUTF M-A을 통해 TFT LCD에 대한 GUI 구성 및 제어를 보다 쉽고 편하게 구현할 수 있습니다.

1.1 주요 사양 (Hardware)

주요 항목	주요 사양
MCU	<ul style="list-style-type: none"> - 32bit , EISC Microprocessor - Up to 108MIPS
RAM	<ul style="list-style-type: none"> - 2Kbyte Internal SRAM for Instruction - 30Kbyte Internal SRAM for Data - 16Mbyte SDRAM for Graphic Memory
NAND Memory	<ul style="list-style-type: none"> - NAND 128Mbyte - Image, Sound Wav 파일 및 Config 파일 저장 - 외장 USB 저장 장치 인식 (USB Mass Storage Boot Mode) - 외장 USB 접속 모드 : USB 2.0
LCD Interface	<ul style="list-style-type: none"> - 4 Wire : DC5V, GND, RxD, TxD - Backlight Control : PWM (Level :0~99) - 최대 지원 해상도 : 800x480 - 24bit RGB Mode, RGB888 or 565 Mode
Touch Panel Interface	<ul style="list-style-type: none"> - 기본 : 저항막 방식(Resistive Type) - 터치 입력 좌표 정보 자동 회신
Power	<ul style="list-style-type: none"> - DC 5V - 전류 : Min 1.5A

1.2 제품 주요 기능

1.2.1 제품 외관



* I/F : Interface

1.2.2 주요 부품별 기능

주요 구성	기능
Memory : NAND	<ul style="list-style-type: none"> - 디스플레이 이미지 저장 - 영문, 숫자 텍스트 저장 - 동작 설정 파일 저장
Connector: Power, TTL UART	<ul style="list-style-type: none"> - 제품 동작 전원 공급 (DC 5V, GND), 최소 공급 전류 1.5A - TTL UART Tx 및 Rx 신호(외부 MCU와 통신)
Connector: USB 2.0 MINI B	<ul style="list-style-type: none"> - GUI용 이미지 파일을 복사 저장하기 위해 PC와 연결 - 전원 부족시 외부 전원으로 연결가능(반드시 DC5V)
Switch: Boot Mode	<ul style="list-style-type: none"> - Boot Mode를 일반 동작 모드와 PC에서 외부메모리디스크로 인식하는 모드로 선택 - PC와 USB를 연결하고, Switch를 누른 상태에서 전원인가시 외부메모리디스크로 인식됨
Connector: 4.3" LCD Interface	<ul style="list-style-type: none"> - Touch Panel과 일체화된 4.3" Color TFT LCD 연결(40Pin) - 호환 LCD 별도 옵션 구매 가능
Connector: 7.0" LCD Interface	<ul style="list-style-type: none"> - 7" Color TFT LCD 연결(50Pin) - 호환 LCD 별도 옵션 구매 가능
Connector: Resistive Type Touch Interface	<ul style="list-style-type: none"> - 7" Color TFT LCD용 감압식 Touch Panel 연결 - 호환 LCD와 같이 조립 상태로 옵션 구매 가능
Switch: RESET	<ul style="list-style-type: none"> - 제품 동작 초기화
Connector: Run Status Control	<ul style="list-style-type: none"> - 부팅 단계별 동작 상태, MCU 동작 유무에 대한 신호 출력 - 외부 RESET 신호 입력
Connector: Speaker Out	<ul style="list-style-type: none"> - 사운드 출력 필요시 외부 모노 스피커 연결 - MCU 내장 코덱이 아닌 외부 별도 코덱(WM9874적용)









1.3 지원 이미지 파일

RGB888 또는 RGB565 조건으로 제작된 BMP(24bpp), JPG, PNG 파일 포맷을 지원합니다.


PNG 보다는 BMP 이미지가 파일 용량을 많이 차지하더라도, 이미지 디스플레이 시에는 보다 빠른 결과를 나타냅니다.

2. 구성

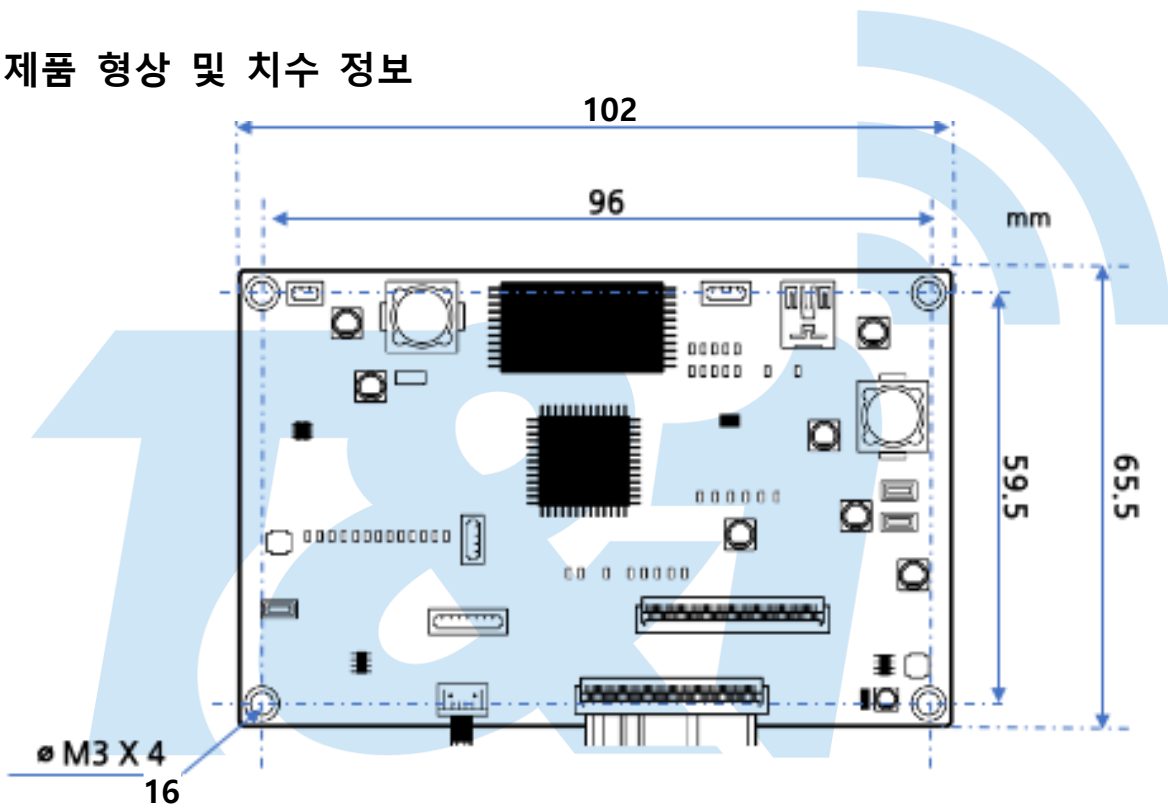
2.1 기본 구성

No	품명	수량	설명
1	제품 	1	- JUTF PCB 제품 - Serial TFT LCD Controller
2	케이블 	1	- 4P, 전원 및 TTL UART 케이블 - 외부 Host MCU 연결 - DC5V(R), GND(BK), Tx(G), Rx(W)
3	케이블 	1	- 4P, 제품 동작 상태 신호 케이블 - 외부 Host MCU 연결 - Booting Completed (B), Running(G), GND(BK), Ext.Reset(W)
4	케이블 	1	- 2P, 스피커 케이블 - 사운드 출력 필요 시 외부 모노 스피커 연결
5	케이스 	1	- 제품 보호용 케이스 - 재질 : 전기아연도금강판, T=1mm
6	볼트 	4	- 케이스, PCB 및 외부 브라켓 연결 고정 - 둥근머리나사, M3X12
7	브라켓 	1	- 4.3 / 7인치 LCD, 제품 및 케이스 고정 - 재질 : 전기아연도금강판, T=1mm
8	LCD 	1	- 7인치 Color TFT LCD - 4.3인치 Color TFT LCD - Touch Panel 포함

2.2 추가 구성

No	품명		설명
1	USB 케이블		- USB to Serial Cable

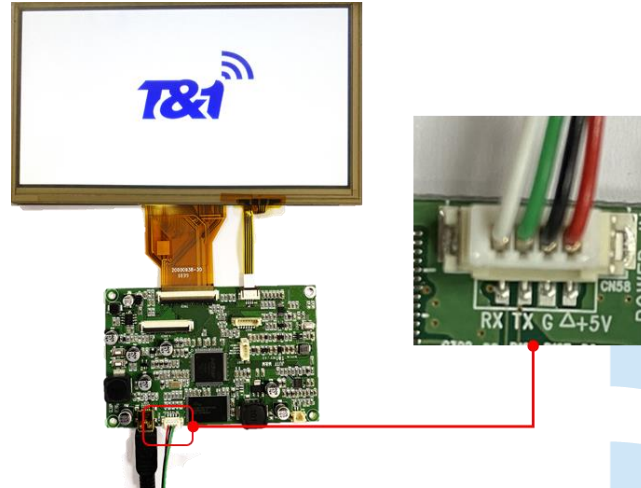
3. 제품 형상 및 치수 정보



* 제품 고정 및 설치를 위한 Bracket과 LCD는 별도 옵션 구매 가능합니다.

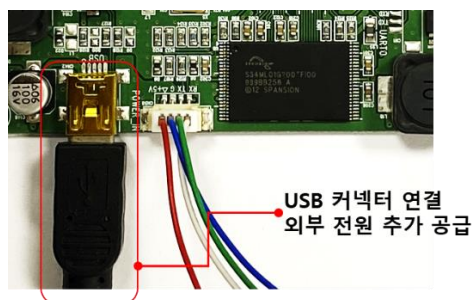
4. 외부 Host Controller Interface

JUTF M-A를 제어할 MCU와는 아래와 같이 4Pin의 Power, TTL UART Connector로 연결합니다.



Wire Color	구분	기능
RED	전원(DC5V)	외부에서 공급되는 전원 연결
BLACK	전원(GND)	외부에서 공급되는 전원의 GND 연결
GREEN	TTL UART(Tx)	외부 MCU의 TTL UART Rx와 연결(DC5V) 제품에서 외부 MCU로 터치 좌표 및 명령어 동작 결과 피드백 함
WHITE	TTL UART(Rx)	외부 MCU의 TTL UART Tx와 연결(DC5V) 외부 MCU에서 제품으로 디스플레이 제어 명령어를 전송함

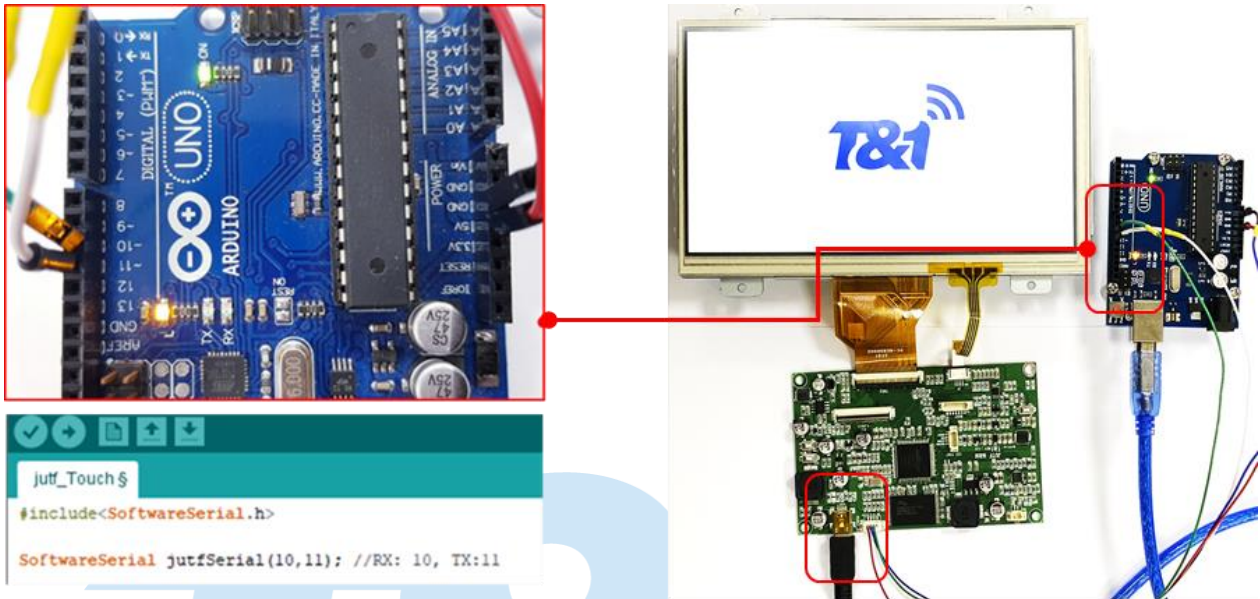
- * 외부에서 공급되는 전원의 경우, DC5V이어야 하며, 공급 전류는 최소 1.5A이어야합니다.
- * 스피커 사운드 출력 시 최소 공급 전류는 2A이어야합니다.
- * Arduino 같은 제어기와 연결하여 동작할 경우, Arduino에서 공급되는 전류가 부족하여, 정상 동작이 안 되는 경우가 있습니다. 이 때는 제품의 USB 커넥터를 이용하여 추가로 DC5V 외부 전원을 공급해 주어야 합니다.



[Arduino UNO와의 연결]

Arduino UNO와 연결하여, 제품을 사용할 경우, UNO 확장 포트에서 전원 및 통신 신호를 연결합니다. 이 경우, UNO에서 공급되는 전류가 LCD를 구동하기에는 부족하므로, 제품에 USB를 연결하여, 외부 전원을 추가적으로 공급을 해 주어야 합니다.

Arduino UNO의 기본 RS232 포트는 다운로드와 검용이 되므로, 안정적인 동작을 위해 SoftwareSerial 모드로 적용하여 사용하시기 바랍니다



5. Booting Mode

Booting Mode는 Normal Mode(일반동작모드)와 USB Mass Storage Mode(USB 외장 저장장치모드)로 2가지를 지원합니다.

5.1 Normal Mode(일반동작모드)

4 Wire 전원 및 통신 커넥터가 연결되고, 전원이 인가되면, JUTF M-A은 일반동작모드로 부팅이 됩니다.

단, 정상적인 Normal Mode의 부팅이 이루어지기 위해서는, NAND 메모리에 "config.txt" 파일과 "boot.bmp" 파일이 있어야 합니다.

부팅시, NAND에 저장되어 있는 "config.txt"의 정보에 따라, 통신 및 LCD Display 초기화를 진행합니다. "config.txt" 파일의 내용은 아래와 같이 RS232통신 통신속도(Baud Rate)와 LCD 해상도가 명시되어 있습니다.

반드시, 파일명과 파일내 영문자는 소문자가 되어야 하며, 아래와 같이 한 라인에 하나의 정보만 기록 저장되어야 합니다.

```
*config.txt - Windows 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말
baudrate=115200
resolution=800x480
```

지원 통신 속도:
9600, 19200, 38400, 57600, 115200

지원 LCD 해상도:
- 7 인치 : 800x480
- 4.3 인치 : 480x272

"boot.bmp"는 부팅 화면으로 적용 LCD의 해상도에 맞게 제작이 된 24bit bmp 포맷의 이미지 파일 이어야 합니다.

"config.txt" 파일내의 해상도 설정과 "boot.bmp" 이미지가 이상이 없다면, 정상적으로 "boot.bmp" 이미지가 LCD에 표시됩니다.

상품 구매시에는 (주)티앤원의 로고가 "boot.bmp"로 설정되어 있습니다.

사용자의 목적에 따라, "boot.bmp"의 사용자화가 가능합니다.



5.2 USB Mass Storage Mode(USB 외장 저장장치모드)

PC와 JUTF M-A을 USB Cable로 연결한 후, JUTF M-A의 NAND 메모리를 외장 USB 저장장치로 인식하여 사용하도록 하는 모드입니다.

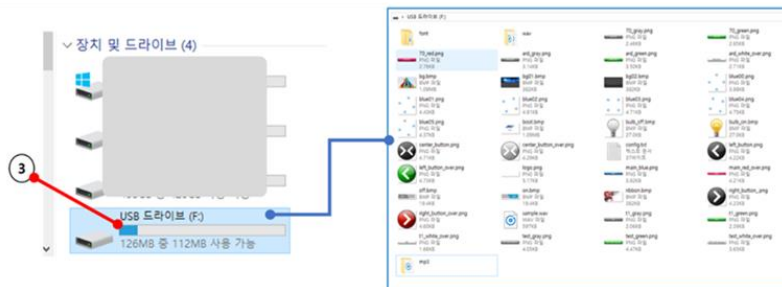
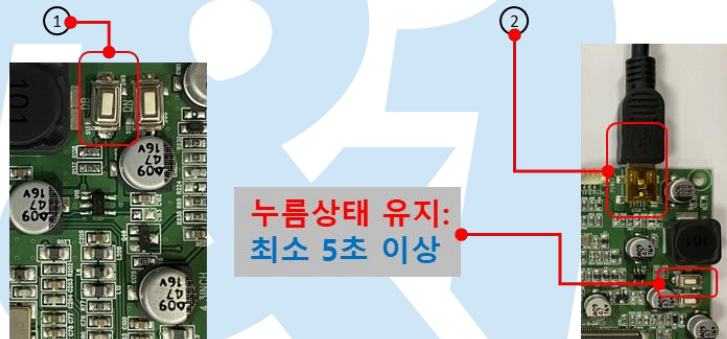
Format, Copy, Delete, Paste, Save 등 PC에서 사용하는 일반 파일 처리 기능을 사용할 수 있습니다.

단, Format(포맷)시에는 반드시 FAT16 또는 FAT32로 진행해야 합니다.

사용시 필요한 GUI용 배경 이미지, 버튼, 아이콘, 문자 등 이미지들을 저장하거나 삭제하기 위해 사용됩니다.

부팅모드 과정은 다음과 같습니다.

- ① 4Wire 전원/통신이 연결되어 있다면, 해당 케이블을 뽑고, JUTF M-A 전원을 해제한 후, PCB의 DB 버튼을 누릅니다.
- ② DB 버튼을 누른 상태에서 PC와 JUTF M-A간의 USB 케이블 연결합니다.
전원이 인가되더라도, DB 버튼을 최소 5초가 더 누른 상태로 유지해주시기 바랍니다.
- ③ DB 버튼에서 손을 땀 후, PC의 파일탐색기를 확인하면, USB 드라이브가 인식되어 있습니다.
해당 드라이브를 클릭하여 폴더를 열면, "config.txt", "boot.bmp"를 포함한 이미지 및 파일들을 확인할 수 있습니다.

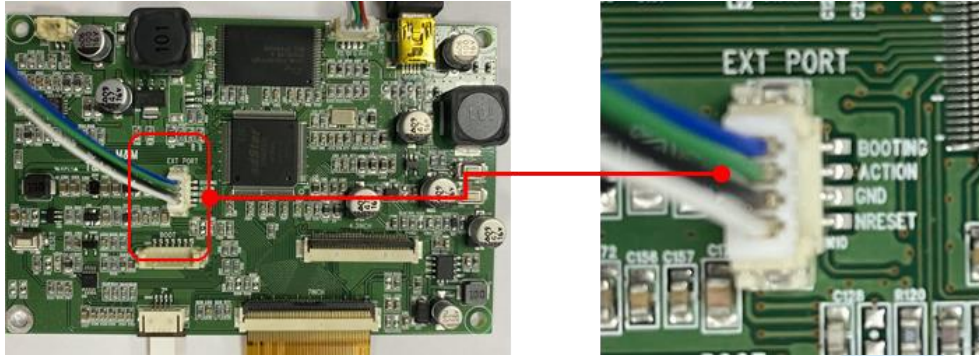


- ④ NAND 메모리의 내용 수정 변경 후에는, USB 케이블을 제거하고, 4Wire 전원/통신 케이블을 연결하여 사용하시면 됩니다.

NAND 메모리 내용 변경시에는 반드시 전원 재인가에 의한 재부팅이 진행이 되어야 합니다.

6. 동작 상태 제어

제품에는 제품의 동작 상태 확인 및 외부 리셋 기능을 제공하는 "EXT PORT"라는 4P Connector가 있습니다.



Wire Color	구분	기능
BLUE	BOOTING	- 전원인가 후, 부팅 상태를 출력하는 신호 - 부팅 중이면 HIGH(DC 5V) 신호 - 부팅이 완료되면 LOW(DC 0V) 신호 - 부팅 완료 신호상태에서, 외부 HOST MCU에서 제품으로 명령 송신하면 됨
GREEN	ACTION	- 제품의 MCU가 동작 중이면, 500Hz 구형파 연속 출력됨 - 지속적이거나 일정시간동안 HIGH나 LOW 신호만 유지될 경우 외부 리셋으로 재부팅 가능함
BLACK	GND	- GND
WHITE	NRESET	- 외부 HOST MCU에서 제품을 강제 리셋 할 수 있음 - 입력 신호가 HIGH에서 LOW로 변경시 리셋됨

7. Control Commands

제공되는 Control Command는 System, Primitive, Image, Sound, Touch 등 크게 6종 30개입니다.
Command Packet 구성은 아래와 같습니다.

Start Of Packet('\$')	Command Packet Body	End Of P('#')
-----------------------	----------------------------	---------------

Command Packet Body의 Type은 **string**이어야 하며, 각 Argument의 구분은 ', '로 합니다.

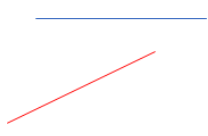


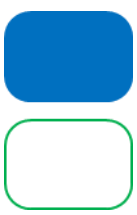
영문 대소문자 구분이 되어야 하므로, 아래의 Command Packet대로 전송이 되어야 합니다.

전송된 command가 error 없이 정상 실행이 되면 ' + ' 문자가 return 되고, 실패인 경우 ' - ' 문자가 return 됩니다.

7.1. System Command

No	Function	구분	내용
1	JUTF 연결상태 확인	Command	\$s,0#
		설명	연결 실패시 Response가 없을 수도 있음
2	Double Buffer Mode 설정	Command	\$s,3#
		설명	Drawing 시에서 Double Buffer Mode 를 적용함 Double Buffer Mode 적용시 모든 draw 관련 Command는 back frame buffer에 적용이 되어, 이를 Display하기 위해서는 반드시 "flip" (No.5 참조) Command를 사용해야 함 *'flip' Command : back frame buffer와 front frame buffer를 swap 함
3	Single Buffer Mode 설정	Command	\$s,4#
		설명	Drawing시 Single Buffer Mode 를 적용함 Single Buffer Mode 적용시 모든 draw 관련 Command는 front frame buffer에 위치하여 즉시 display 되지만, 용량이 큰 이미지의 경우, Double Buffer Mode에 비하여 처리 시간이 더 소요됨
4	LCD Backlight 밝기 조절	Command	\$s,5,db#"
		설명	db : Backlight 밝기 레벨(0~99), 정수형 정수형 데이터가 string으로 입력되어야 함
5	Screen Flip	Command	\$s,6#
		설명	Double buffer Mode에서 front frame buffer와 back frame buffer의 Swap하고, back frame buffer의 내용이 LCD에 표시됨
6	USB Mass Storage Device	Command	\$s,7#
		설명	일반동작모드로 부팅된 후 JUTF M-A을 외장 USB 저장장치로 사용. NAND 내용 수정 완료 후 반드시 재부팅이 되어야 수정 변경 결과가 반영됨
7	System Reset	Command	\$s,9#
		설명	NAND 메모리의 내용이 수정/변경되었을 때, 시스템을 강제 Reset/Reboot 함

7.2 Primitive Command

No	Function	구분	설명
1	표시 컬러 설정	Command	\$!,0,dR,dG,dB#
		설명	Drawing Command에서 대상의 Color를 RGB로 설정함 dR,dG,dB 는 각각 8bit(0~255)의 red, green, blue color 값을 나타내는 정수형이며, 반드시 string으로 변환 입력 되어야함
2	선그리기 	Command	\$!,1,dSx,dSy,dDx,dDy#
		설명	선 그리기 Command임 선을 그리기 위해서는 2개의 점(시작점과 끝점) 좌표가 필요함 dSx, dSy : 시작점의 x, y 좌표, 정수형 임 dEx, dEy : 끝점의 x, y 좌표, 정수형 임 입력 Argument들은 반드시 string으로 변환 입력되어야 함
3	사각형그리기 	Command	\$!,2,dSx,dSy,dw,dh,df#
		설명	사각형 그리기 Command임 dSx,dSy : 시작점, 정수형 dw, dh : 넓이, 높이, 정수형 df : Filled(1), or unFilled(0) -1 : 내부가 채워진 사각형 -0 : 외곽선만 있는 사각형 입력 Argument들은 반드시 string으로 변환 입력되어야 함
4	그라데이션 사각형 그리기 	Command	\$!,3,dSx,dSy,dw,dh,dSr,dSg,dSb,dEr,dEg,dEb#
		설명	Gradient 효과가 적용된 사각형 그리기 Command dSx,dSy : 시작점, 정수형 dw, dh : 넓이, 높이, 정수형 dSr,dSg,dSb : Gradient 효과 시작 RGB Color dEr,dEg,dEb : Gradient 효과 끝부분 RGB Color Color값은 정수형이며, 입력 Argument들은 반드시 string으로 변환 입력되어야 함
5	라운드 사각형 그리기 	Command	\$!,4,dSx,dSy,dw,dh,dR,df#
		설명	모서리 라운드 형태 사각형 그리기 Command dSx,dSy : 시작점, 정수형 dw, dh : 넓이, 높이, 정수형 dR : 모서리 라운드의 반경, 정수형 df : Filled(1), or unFilled(0) -1 : 내부가 채워진 사각형 -0 : 외곽선만 있는 사각형 입력 Argument들은 반드시 string으로 변환 입력되어야 함

No	Function	구분	설명
6	특정 두께의 선 그리기 	Command	\$l,5,dSx,dSy,dDx,dD,dT#
		설명	임의의 두께를 가지는 선 그리기 Command임 dSx, dSy : 시작점의 x, y 좌표, 정수형 임 dEx, dEy : 끝점의 x, y 좌표, 정수형 임 dT : 선두께, 정수형임 입력 Argument들은 반드시 string으로 변환 입력되어야 함
7	원그리기 	Command	\$l,6,dx,dy,dr,df#
		설명	임의의 반지름을 가지는 원 그리기 Command임 dx, dy : 원 중심 x, y 좌표, 정수임 dr : 원의 반지름, 정수형 임 df : Filled(1), or unFilled(0) -1 : 내부가 채워진 사각형 -0 : 외곽선만 있는 사각형 입력 Argument들은 반드시 string으로 변환 입력되어야 함
8	타원그리기 	Command	\$l,7,dx,dy,dXr,dYr,df#
		설명	임의의 x축, y축 방향 반지름을 가지는 타원 그리기 Command임 dx, dy : 타원 중심 x, y 좌표, 정수임 dXr : 타원의 x축방향(수평방향) 반지름, 정수형 임 dYr : 타원의 y축방향(수직방향) 반지름, 정수형 임 df : Filled(1), or unFilled(0) -1 : 내부가 채워진 사각형 -0 : 외곽선만 있는 사각형 입력 Argument들은 반드시 string으로 변환 입력되어야 함

7.3 Image Command

No	Function	구분	설명
1	Image Load	Command	\$i,1,di,sFileName#
		설명	NAND 메모리에 저장된 이미지 파일을 특정 인덱스 번호를 부여하여 그래픽 메모리로 로드함 di : 인덱스 번호(0~1023), 정수형 sFileName : NAND 메모리에 저장된 이미지의 Full Name(확장자 포함) String 입력 Argument들은 반드시 string으로 변환 입력되어야 함 가용 그래픽 메모리가 약 14MByte이며, 이미지 로딩시 메모리 문제가 발생할 경우, "Image Release" Command를 사용하여, 로딩되어 있는 임의의 이미지를 언로딩 한 후, 그래픽 메모리 공간 확보하여 사용 가능
2	Draw Image	Command	\$i,2,di,dx,dy#
		설명	임의의 좌표 (dx, dy)에 로딩된 임의의 이미지를 LCD상에 표시함 di : 표시할 이미지의 인덱스 번호, 정수형 dx, %dy : 이미지를 표시를 시작할 LCD상의 x축 및 y축 좌표, 정수형 입력 Argument들은 반드시 string으로 변환 입력되어야 함
			LCD
3	Draw Image Rectangle	Command	\$i,3,di,dx,dy,dXs,dYs,dw,dh#
		설명	이미지의 임의의 영역을 사각형 모양으로 선택하여, 선택 영역만 표시함 di : 표시할 이미지의 인덱스 번호, 정수형 dx, dy : 이미지를 표시를 시작할 LCD상의 x축 및 y축 좌표, 정수형 dXs, dYs : 이미지에서 선택할 사각형 영역의 시작 위치, 정수형 dw, dh : 선택한 영역의 넓이와 높이, 정수형 입력 Argument들은 반드시 string으로 변환 입력되어야 함
			LCD
No	Function	구분	설명

4	Draw Image Scale	Command	\$i,4,di,dx,dy,dw,dh#
		설명	<p>그래픽 메모리에 로딩된 이미지 파일을 임의의 위치에서 임의의 크기로 변경하여 표시함</p> <p>di : 인덱스 번호(0~1023), 정수형 dx, dy : 시작 위치 dw, dh : 임의의 크기에 대한 넓이 및 높이 입력 Argument들은 반드시 string으로 변환 입력되어야 함</p>
5	Draw Image Scale Rectangle	Command	\$i,5,di,dx,dy,dw,dh,dSx,dSy,dSw,dSh#
		설명	<p>그래픽 메모리에 로딩되어 있는 이미지의 특정 사각 영역을 선택하여, 해당 영역을 임의의 위치에 임의의 크기로 변경하여 표시함</p> <p>di : 표시할 이미지의 인덱스 번호, 정수형 dx, dy : 이미지를 표시를 시작할 LCD상의 x축 및 y축 좌표, 정수형 dw, dh : 이미지 표시의 넓이와 높이 dSx, dSy : 원본 이미지에서 표시할 영역의 시작 위치 dSw, dSh : 원본 이미지에서 표시할 영역의 넓이 및 높이 입력 Argument들은 반드시 string으로 변환 입력되어야 함</p>
6	Image Release	Command	\$i,6,di#
		설명	<p>그래픽 메모리에 로딩되어 있는 임의의 이미지를 메모리에서 언로딩함</p> <p>di : 언로딩할 이미지의 인덱스 번호, 정수형 입력 Argument들은 반드시 string으로 변환 입력되어야 함</p>

7.4 Touch Input

터치된 임의의 위치 좌표가 JUTF M-A에서 연결된 Host에 Return됨

Return Packet	\$t,ds,dx,dx#
설명	ds : Status - Touch 상태(누르고 있는 상태) : 1 - Touch 후 No Touch 상태(누른 후 맨 상태) : 0 dx, dy : 터치 좌표, 정수형 입력 Argument들은 반드시 string으로 변환 입력되어야 함 Ex) \$t,1,100,200# : 터치 패드 누른 상태에서 좌표는 100,200 \$t,0,0,0# : 터치 후 맨 상태

7.5 Sound Command

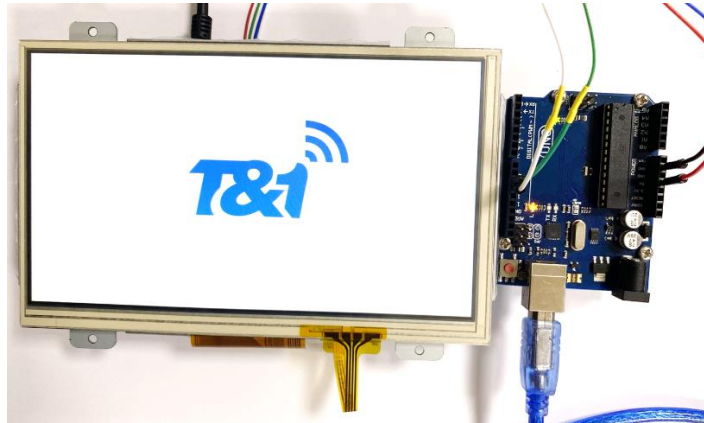
No	Function	구분	설명
1	Sound Load	Command	\$w,1,di,sFileName#
		설명	NAND 메모리에 저장된 임의의 Wave File을 특정 인덱스를 부여하여 메모리로 로드함 di : 인덱스 번호(0~63), 정수형 sFileName : 로딩할 Wave File명, 확장자 포함 입력 Argument들은 반드시 string으로 변환 입력되어야 함
2	Sound Play	Command	\$w,2,di#
		설명	메모리에 로딩된 Wave File을 Speak 단자로 출력함 di : 메모리에 로딩된 Wave File 중 출력할 File의 인덱스 번호, 정수형 입력 Argument들은 반드시 string으로 변환 입력되어야 함
3	Sound Stop	Command	\$w,3,di#
		설명	출력중인 Wave File의 출력을 정지함 di : 정지할 Wave File 인덱스 번호, 정수형 입력 Argument들은 반드시 string으로 변환 입력되어야 함
4	Sound Volume	Command	\$w,4,dv#
		설명	출력 볼륨 조절 dv : 볼륨크기(0~255), 정수형 입력 Argument들은 반드시 string으로 변환 입력되어야 함
5	Speaker On/Off	Command	\$w,5,ds#
		설명	Speaker 단자 출력 On/Off ds : 0(=Speaker Off), 1(=Speaker On) 입력 Argument들은 반드시 string으로 변환 입력되어야 함
6	Sound Release	Command	\$w,6,di#
		설명	메모리에 로딩되어 있는 임의의 인덱스가 부여된 Wave File을 메모리에서 언로딩함 di : 언로딩할 Wave File 인덱스 번호, 정수형 입력 Argument들은 반드시 string으로 변환 입력되어야 함

7.6 Text-Out Command

No	Function	구분	설명
1	Text Out	Command	\$f,0,sString,dx,dy#
		설명	임의의 위치에 영어 또는 숫자로 문자/문자열 표시 영어는 8x16 크기의 폰트 적용됨 sString : 표시할 문자/문자열 dx, dy : 문자/문자열 표시 시작 위치, 정수형 입력 Argument들은 반드시 string으로 변환 입력되어야 함
2	Text Color	Command	\$f,1,dR,dG,dB#
		설명	표시할 문자/문자열의 컬러를 각 RGB Color 값의 조합으로 설정함 dR : Red Color (0~255), 정수형 dG : Green Color(0~255), 정수형 dB : Blue Color(0~255), 정수형 입력 Argument들은 반드시 string으로 변환 입력되어야 함
3	Text Out	Command	\$F,0,sString,dx,dy#
		설명	임의의 위치에 영어 및 숫자를 32x32 크기의 문자/문자열 표시 sString : 표시할 문자/문자열 dx, dy : 문자/문자열 표시 시작 위치, 정수형 입력 Argument들은 반드시 string으로 변환 입력되어야 함
4	Text Color	Command	\$F,1,dR,dG,dB#
		설명	표시할 문자/문자열의 컬러를 각 RGB Color 값의 조합으로 설정함 dR : Red Color (0~255), 정수형 dG : Green Color(0~255), 정수형 dB : Blue Color(0~255), 정수형 입력 Argument들은 반드시 string으로 변환 입력되어야 함
5	Text Out	Command	\$F,10,sString,dx,dy#
		설명	임의의 위치에 영어 및 숫자를 64x64 크기의 문자/문자열 표시 sString : 표시할 문자/문자열 dx, dy : 문자/문자열 표시 시작 위치, 정수형 입력 Argument들은 반드시 string으로 변환 입력되어야 함
6	Text Color	Command	\$F,11,dR,dG,dB#
		설명	표시할 문자/문자열의 컬러를 각 RGB Color 값의 조합으로 설정함 dR : Red Color (0~255), 정수형 dG : Green Color(0~255), 정수형 dB : Blue Color(0~255), 정수형 입력 Argument들은 반드시 string으로 변환 입력되어야 함

8. Examples

- Host : Arduino UNO (Baud Rate : 9600)
- Programming Tool : Sketch of Arduino



8.1. Primitive Command

1) 선 그리기

```

#include<SoftwareSerial.h>

SoftwareSerial jutfSerial(10,11); //RX: 10, TX:11
int nCounter = 0;
void setup() {
  // put your setup code here, to run once:
  jutfSerial.begin(9600);
}

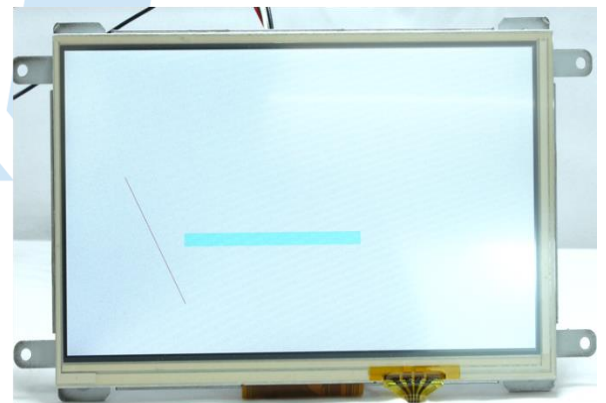
void loop() {
  // put your main code here, to run repeatedly:

  if(nCounter == 0)
  {
    //배경을 White Color로 클리어 (채움)
    jutfSerial.write("$s,3#"); // double buffer mode
    jutfSerial.write("$1,0,255,255,255#"); //color : white
    jutfSerial.write("$1,2,0,0,800,480,1#"); //clear with white color

    //Red Color 선그리기
    jutfSerial.write("$1,0,255,0,0#"); //set red color
    jutfSerial.write("$1,1,100,200,200,400#"); //drawing a line

    //Blue Color 두께선그리기
    jutfSerial.write("$1,0,0,255,255#"); //set blue color
    jutfSerial.write("$1,5,200,300,500,300,20#"); //drawing a line with thick

    //double buffer mode에서 Front Frame Buffer와 Back Frame Buffer간의 swap
    jutfSerial.write("$s,6#");
    nCounter++;
    delay(1000);
  }
}
    
```



2) 사각형 그리기

```
#include<SoftwareSerial.h>

SoftwareSerial jutfSerial(10,11); //RX: 10, TX:11
int nCounter = 0;
void setup() {
  // put your setup code here, to run once:
  jutfSerial.begin(9600);
}

void loop() {
  // put your main code here, to run repeatedly:

  if(nCounter == 0)
  {
    //배경을 White Color로 클리어 (채움)
    jutfSerial.write("$s,4#"); // single buffer mode
    jutfSerial.write("$l,0,255,255,255#"); //color : white
    jutfSerial.write("$l,2,0,0,800,480,1#"); //clear with white color

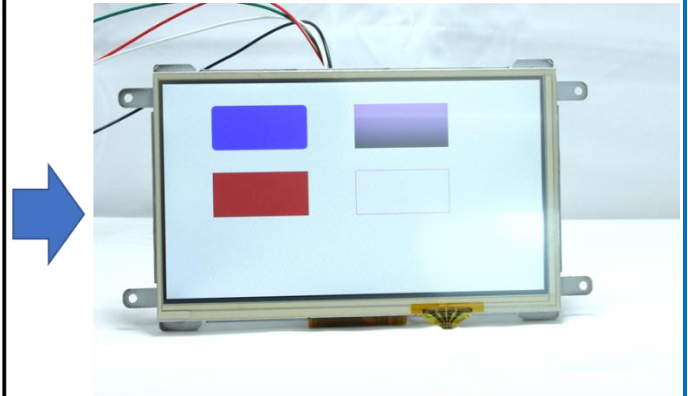
    //Red Color Box 그리기 (filled)
    jutfSerial.write("$l,0,255,0,0#"); //set red color
    jutfSerial.write("$l,2,100,200,200,100,1#"); //drawing a box

    //Red Color Box 그리기 (no filled)
    jutfSerial.write("$l,0,255,0,0#"); //set red color
    jutfSerial.write("$l,2,400,200,200,100,0#"); //drawing a box

    //Gradient Box 그리기
    jutfSerial.write("$l,3,400,50,200,100,200,100,200,50,50,50#"); //drawing a box

    //Rounded Rectangl Box 그리기 (filled)
    jutfSerial.write("$l,0,0,0,255#");
    jutfSerial.write("$l,4,100,50,200,100,10,1#"); //drawing a box

    nCounter++;
    delay(1000);
  }
}
```



3) 원/타원 그리기

```
#include<SoftwareSerial.h>

SoftwareSerial jutfSerial(10,11); //RX: 10, TX:11
int nCounter = 0;
void setup() {
  // put your setup code here, to run once:
  jutfSerial.begin(9600);
}

void loop() {
  // put your main code here, to run repeatedly:

  if(nCounter == 0)
  {
    //배경을 White Color로 클리어 (채움)
    jutfSerial.write("$s,4#"); // single buffer mode
    jutfSerial.write("$l,0,255,255,255#"); //color : white
    jutfSerial.write("$l,2,0,0,800,480,1#"); //clear with white color

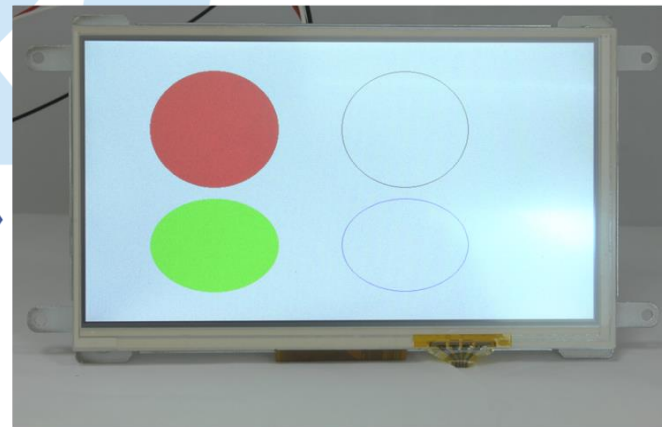
    //Circle 그리기 (filled)
    jutfSerial.write("$l,0,255,0,0#"); //set red color
    jutfSerial.write("$l,6,200,200,100,1#"); //drawing a filled

    //Circle 그리기 (no filled)
    jutfSerial.write("$l,0,0,0,0,0#");
    jutfSerial.write("$l,6,500,200,100,0#"); //drawing a box

    //Ellipse 그리기 (filled)
    jutfSerial.write("$l,0,0,255,0#");
    jutfSerial.write("$l,7,200,350,100,80,1#"); //drawing a filled

    //Ellipse 그리기 (no filled)
    jutfSerial.write("$l,0,0,0,255#");
    jutfSerial.write("$l,7,500,350,100,80,0#"); //drawing a filled

    nCounter++;
    delay(1000);
  }
}
```



8.2. Image Command

```

#include <SoftwareSerial.h>

SoftwareSerial jutfSerial(10,11);
int nCounter = 0;

void setup() {
  // put your setup code here, to run once:
  jutfSerial.begin(9600);
}

void loop() {
  // put your main code here, to run repeatedly:

  if(nCounter == 0)
  {
    //이미지 로드
    jutfSerial.write("$i,1,0,bg.bmp"); // 이미지 로드 (인덱스 0번)
    jutfSerial.write("$i,1,1,ribbon.bmp"); // 이미지 로드 (인덱스 1번)

    //배경을 White Color로 클리어 (채움)
    jutfSerial.write("$s,4#"); // single buffer mode
    jutfSerial.write("$l,0,255,255,255#"); // color : white
    jutfSerial.write("$l,2,0,0,800,480,1#"); // clear with white color

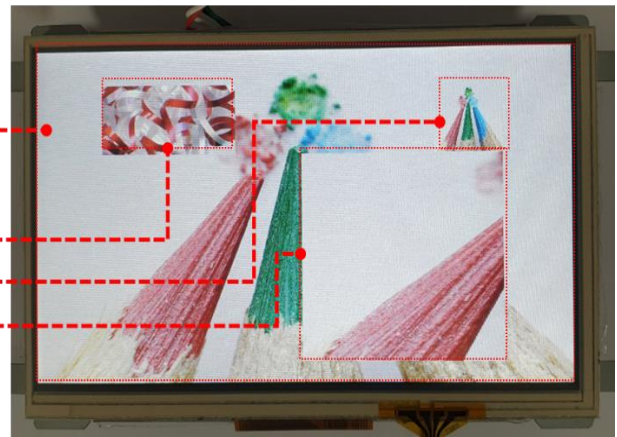
    jutfSerial.write("$i,2,0,0,0#"); // 이미지 그리기
    jutfSerial.write("$i,3,1,100,50,30,30,200,100#"); // 부분 이미지 그리기
    jutfSerial.write("$i,4,0,600,50,100,100#"); // 축소 이미지 그리기
    jutfSerial.write("$i,5,0,400,150,300,300,100,100,200,300#"); // 부분 확대 이미지 그리기

    nCounter++;
    delay(1000);
  }
}
  
```

bg.bmp



ribbon.bmp



8.3. Touch Test

```

void loop() {
  // put your main code here, to run repeatedly:
  if(jutfSerial.available() > 0)
  {
    /*-----통신수신 구문-----*/
    rx_dummy = jutfSerial.read();

    if(rx_dummy == '$')
    {
      rx_flag = 1;
      rx_buffer[0] = rx_dummy;
      rx_cnt = 1;
    }
    else if(rx_dummy == '#')
    {
      rx_flag = 0;
      rx_buffer[rx_cnt] = rx_dummy;
      strcpy(rx_temp0, rx_buffer);
      rx_cnt = 0;
      rx_complete = 1;
    }
    else if(rx_flag == 1)
    {
      rx_buffer[rx_cnt] = rx_dummy;
      rx_cnt++;
    }
  }
  /*-----터치 데이터 처리 구문-----*/
  if(rx_complete == 1)
  {
    rx_complete = 0;

    if(rx_temp0[3] == '0')
    {
      sscanf(rx_temp1, "$t,%d,%d,%d#", &Index, &cX, &cY);
      Serial.println(Index);
      Serial.println(cX);
      Serial.println(cY);
      sprintf(send_data, "$1,6,%d,%d,20,1#", cX, cY);
      jutfSerial.print(send_data);
    }
    else
    {
      strcpy(rx_temp1, rx_temp0);
    }
  }
}
}

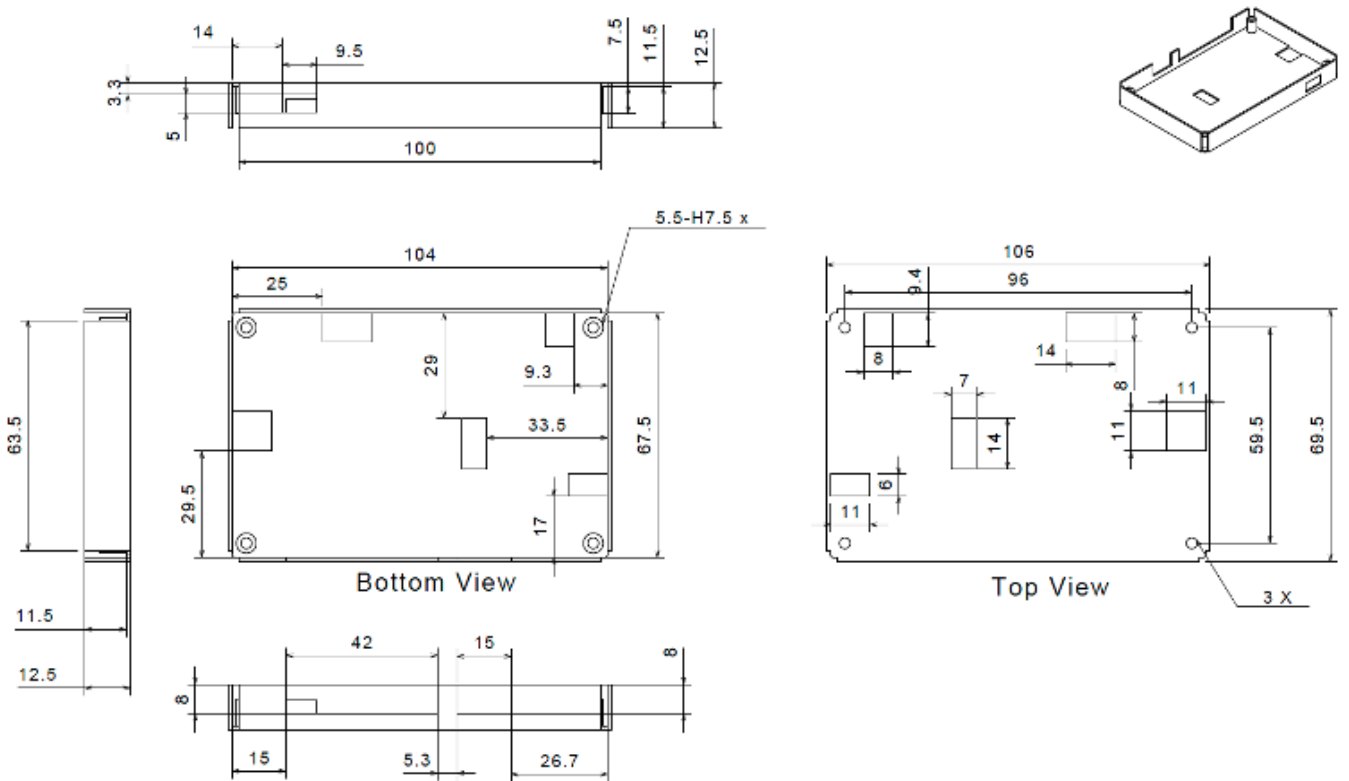
```



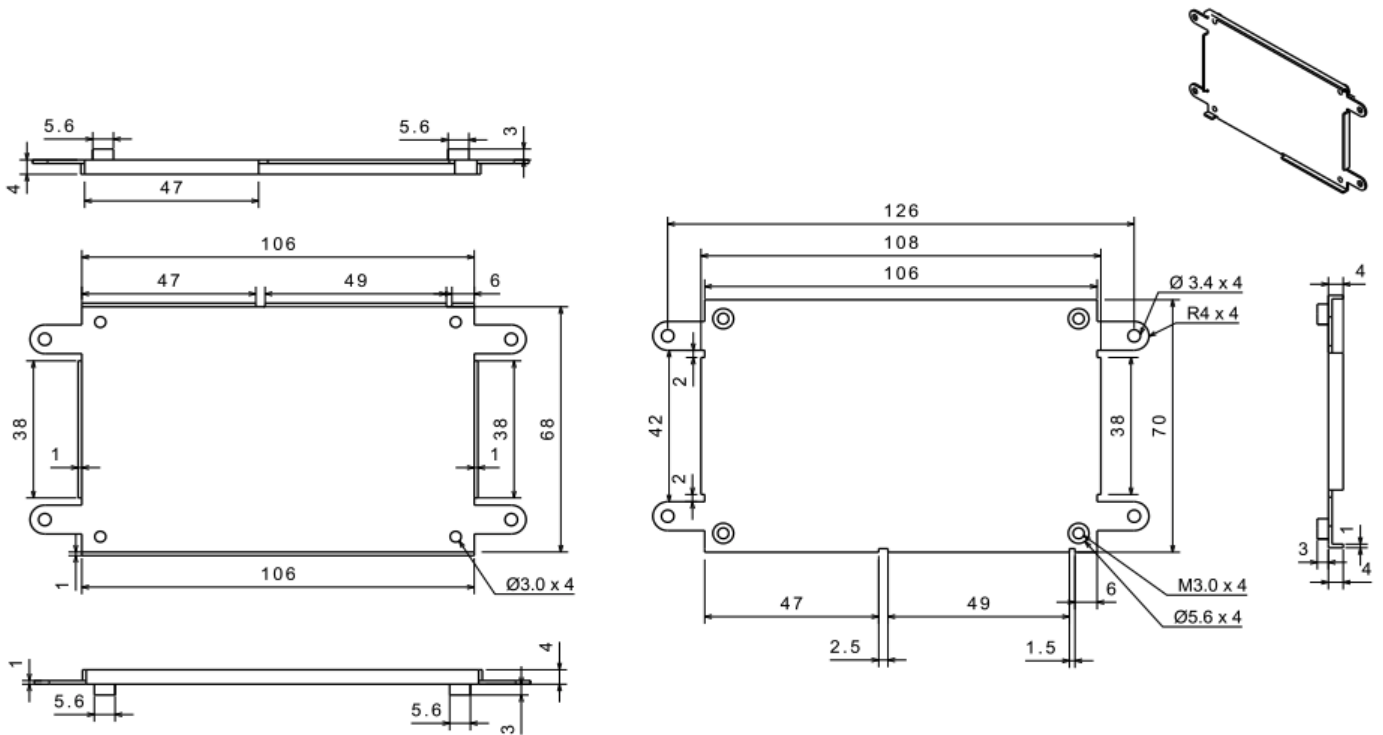
Touch Test

[추가 자료]

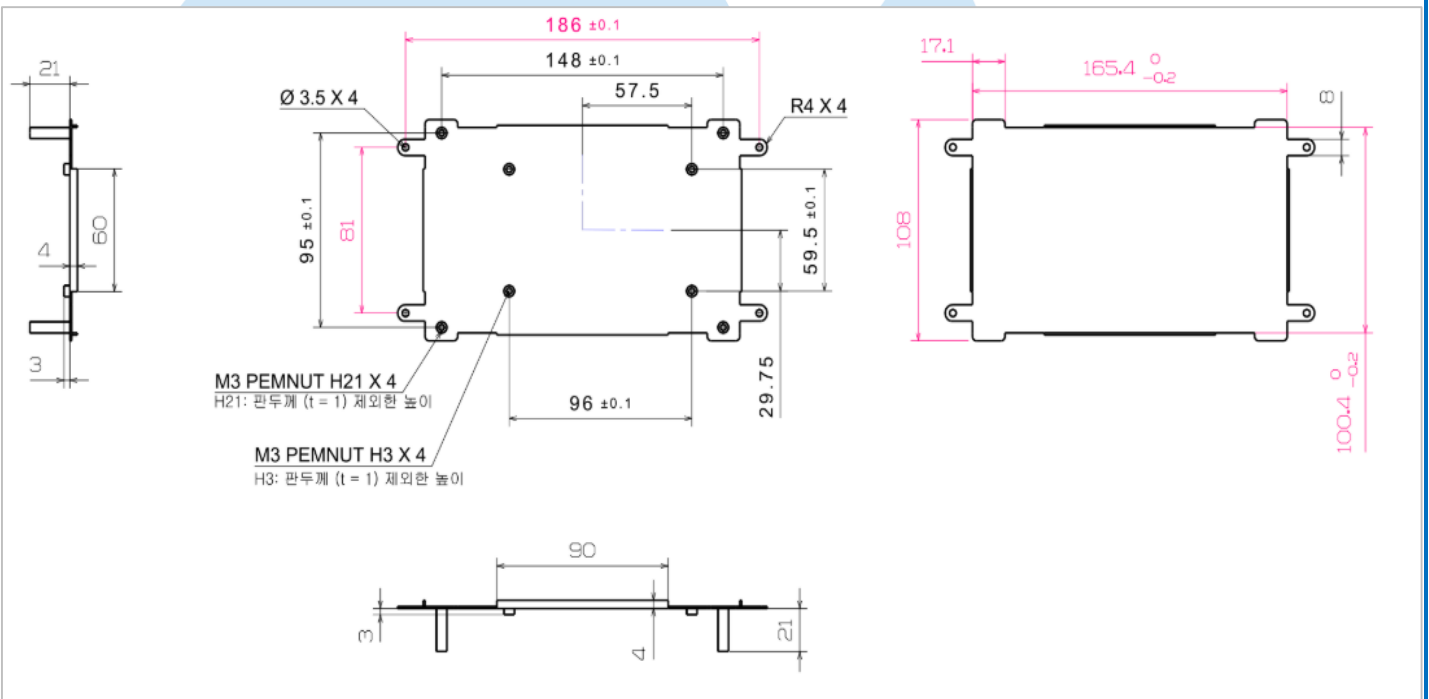
A. Case 치수 정보



B. 4.3인치 LCD용 브라켓 치수 정보



C. 7인치 LCD용 브라켓 치수 정보



[추가 자료]

RGB 16색상 정보

Color	dR,dG,dB	Color	dR,dG,dB	Color	dR,dG,dB	Color	dR,dG,dB
Red	255,0,0	Vermilion	255,85,78	Blue	0,0,255	Skyblue	100,255,255
Orange	255,145,0	Naples yellow	255,185,120	purple	150,0,255	Lavender	255,125,255
Yellow	255,255,0	Chrome yellow	255,207,25	Black	0,0,0	Pink	255,0,255
Green	0,255,0	Mint	160,255,190	White	255,255,255	Ivory	250,245,215

