



深圳市拓普微科技开发有限公司

SHENZHEN TOPWAY TECHNOLOGY CO., LTD.

# HMT029BTA

## LCD Module User Manual

Prepared by:  Caiwei  Date: 2024-08-06	Checked by:    Date:	Approved by:    Date:
--	----------------------------------	-----------------------------------

Rev.	Descriptions	Edit	Release Date
0.1	Preliminary	Caiwei	2024-08-06

**Table of Content**

<b>1 Basic Specification .....</b>	<b>3</b>
1.1 General Specification .....	3
1.2 Block Diagram .....	3
1.3 Terminal Function .....	4
<b>2 Absolute Maximum Ratings .....</b>	<b>5</b>
<b>3 Electrical Characteristics .....</b>	<b>5</b>
3.1 DC Characteristics .....	5
<b>4 Function Specifications .....</b>	<b>6</b>
4.1 Basic Operation Function Descriptions .....	6
4.2 Quick Start Guide .....	7
4.3 Command Descriptions .....	7
<b>5 Optical Characteristics .....</b>	<b>8</b>
<b>6 LCD Module Design and Handling Precautions .....</b>	<b>10</b>
Warranty .....	11

# 1 Basic Specification

TOPWAY HMT029BTA is a Smart TFT Module with 32bit MCU on board. Its graphics engine provides numbers of outstanding features.It supports TOPWAY SGTools for preload and pre-design display interface that simplify the host operation and development time. Suitable for industry control, instrumentation, medical electronics, power electric equipment applications.

## 1.1 General Specification

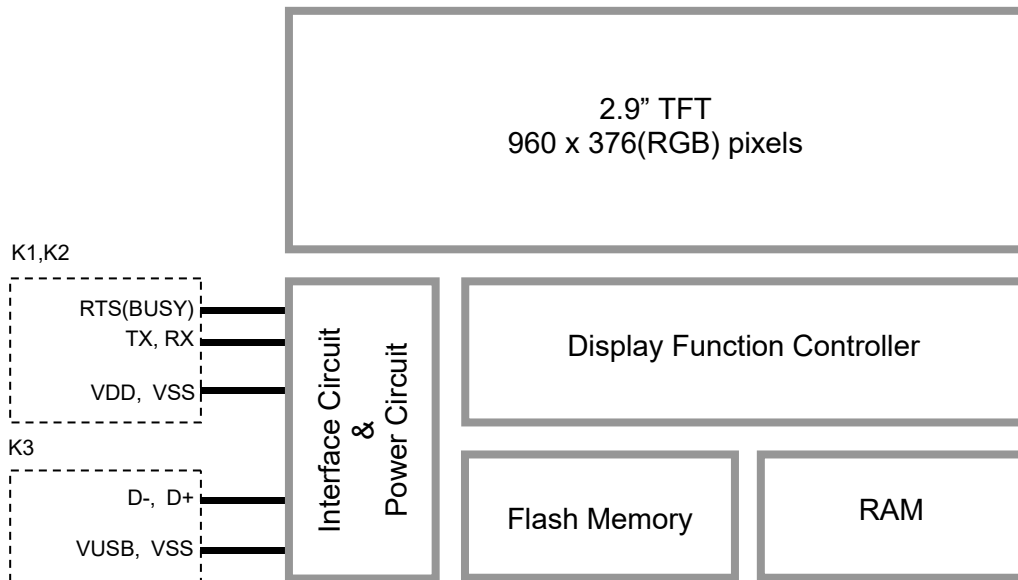
Screen Size(Diagonal) :	2.9"
Resolution :	960X376(RGB)
Color Depth :	65k color (16bit)
Pixel Configuration :	RGB Stripe
Display Mode :	Transmissive / Normal Black
Viewing Direction :	Full View
Outline Dimension :	99.0x 35.6x 11.5 (max)(mm) (see attached drawing for details)
Active Area :	67.68 x 26.51 (mm)
Backlight :	LED
Command I/F:	RS-232C
Project Download:	by PC or by U-Drive (with OTG cable) (*1)
Operating Temperature :	-20 ~ +70°C
Storage Temperature :	-30 ~ +80°C
Highlight:	Long strip,RTC without battery , Support 90 degrees rotation,Lua script engine

Note:

\*1. Section 1.3 for configuration.

\*2.Color tone may slightly change by Temperature and Driving Condition.

## 1.2 Block Diagram



### 1.3 Terminal Function

#### RS-232C Interface Terminal (K1,K2)

Pin No.	Pin Name	I/O	Descriptions
1, 2, 3	VDD	P	Power supply
4	NC	--	No connection, leave open
5	RX	I	Data Input (eg. to PC's RS-232C pin3 <9pin D-connector>)
6	TX	O	Data output (eg. to PC's RS-232C pin2 <9pin D-connector>)
7	RTS(BUSY)	O	Request To Send (could function as busy BUSY signal) (eg. to PC's RS-232C pin8 <9pin D-connector> )
8, 9, 10	VSS	P	Ground, (0V)

Note.

- \*1. User data and commands transfer through this terminal.
- \*2. HOST using command hand shake during communication is suggested.

#### USB Interface Terminal (K3)

Pin No.	Pin Name	I/O	Descriptions
A1,A12,B1,B12	VSS	P	Ground, (0V)
A4,A9,B4,B9	VUSB	P	Power supply(5.0 V)
A6,B6	D+	I/O	USB DATA positive signal
A7,B7	D-	I/O	USB DATA negative signal
A5,B5,A8,B8	NC	--	Not connected

Note.

- \*1. Display files preload through this terminal.
- \*2. Standard "USB-drive" functions provided.
- \*3. During the files transfer, all others display functions will be suspended.
- \*4. For PC direct download project: JP11=4.7R, JP12=NC,JP13=4.7R,JP14=NC.(default)  
For U-Drive to download project: JP11=NC, JP12=4.7R,JP13=NC,JP14=4.7R.

## 2 Absolute Maximum Ratings

Items	Symbol	Min.	Max.	Unit	Condition
Power Supply voltage	$V_{dd}$	-0.3	5.5	V	
Operating Temperature	$T_{OP}$	-20	70	°C	No Condensation
Storage Temperature	$T_{ST}$	-30	80	°C	No Condensation

Note:

- \*1. This rating applies to all parts of the module and should not be exceeded.
- \*2. The operating temperature only guarantees operation of the circuit. The contrast, response speed, and the other specification related to electro-optical display quality is determined at the room temperature,  $T_{OP}=25^{\circ}\text{C}$
- \*3. Ambient temperature when the backlight is lit (reference value)
- \*4. Any Stresses exceeding the Absolute Maximum Ratings may cause substantial damage to the device. Functional operation of this device at other conditions beyond those listed in the specification is not implied and prolonged exposure to extreme conditions may affect device reliability.

## 3 Electrical Characteristics

### 3.1 DC Characteristics

VSS=0V,  $T_{OP}=25^{\circ}\text{C}$

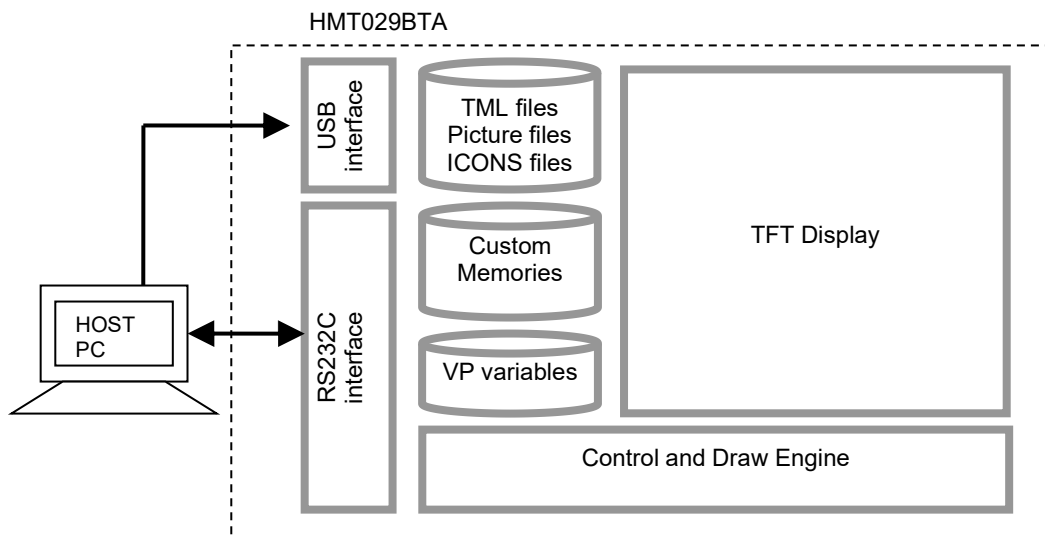
Items	Symbol	MIN.	TYP.	MAX.	Unit	Applicable Pin/FUNC
Operating Voltage	$V_{DD}$	4.8	5.0	5.2	V	VDD
Rx Input Hi	$V_{RxH}$	-3.0	-	-15.0	V	Rx
Rx Input Lo	$V_{RxL}$	+3.0	-	+15.0	V	Rx
Tx Output Hi	$V_{TxH}$	-3.0	-	-15.0	V	Tx
Tx Output Lo	$V_{TxL}$	+3.0	-	+15.0	V	Tx
RTS(BUSY) Output High	$V_{RTSH}$	-3.0	-	-15.0	V	RTS(BUSY)
RTS(BUSY) Output Low	$V_{RTSL}$	+3.0	-	+15.0	V	RTS(BUSY)
Operating Current	$I_{DD}$	-	260	-	mA	VDD (*1)
Operating Current (USB)	$I_{VUSB}$	-	180	-	mA	VUSB

Note.

- \*1. Normal display condition.

## 4 Function Specifications

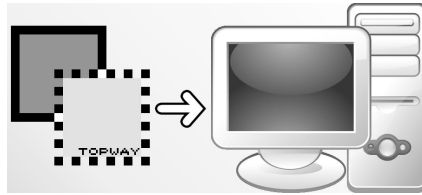
### 4.1 Basic Operation Function Descriptions



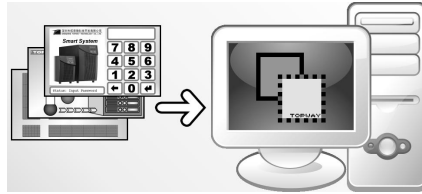
- TML files, Picture files, ICON files are stored inside FLASH memory area. They are preloaded to HMT029BTA for stand alone interface use.
- Those files are preloaded via USB interface (PC download or U-Drive).
- All the interface flow and the touch response are based on the preloaded TML files.
- VP variables memory is inside RAM area, it provides real time access via RS232C by the HOST or display onto the TFT by TML file.
- Custom Memories are inside FLASH memory area. It can be accessed via RS232C interface by the HOST.
- Control and Draw Engine executes HOST commands and response respectively.
- It also reports the real time Touch Key number to the HOST.

**4.2 Quick Start Guide**

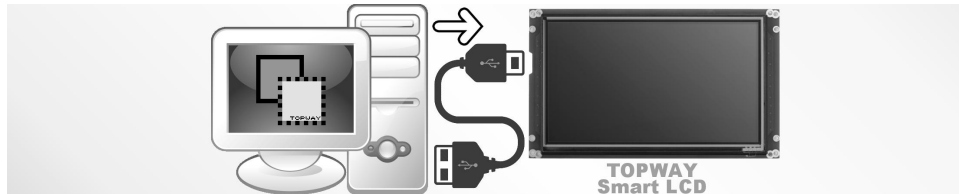
- 1. Install TOPWAY Graphics Editor



- 2. Import pictures design UI flow



- 3. Download to Smart LCD



- 4. power on & display



- 5. Connect to host Show real time data



**4.3 Command Descriptions**

Please refer to "SMART LCD Command Manual" .

## 5 Optical Characteristics

Item	Symbol	Condition	Min	Typ	Max	Unit	Remark
View Angles	$\theta T$	$CR \geq 10$	75	85	-	Degree	Note2,3
	$\theta B$		75	85	-		
	$\theta L$		75	85	-		
	$\theta R$		75	85	-		
Contrast Ratio	CR	$\theta=0^\circ$	1000	1500	-		Note 3
Response Time	$T_{ON}$	25°C	-	30	35	ms	Note 4
	$T_{OFF}$						
Chromaticity	White	x	Backlight is on	0.229	0.279	0.329	Note 1,5
		y		0.254	0.304	0.354	
	Red	x		0.534	0.584	0.634	Note 1,5
		y		0.305	0.355	0.405	
	Green	x		0.275	0.325	0.375	Note 1,5
		y		0.543	0.593	0.643	
	Blue	x		0.094	0.144	0.194	Note 1,5
		y		0.021	0.071	0.121	
Uniformity	U		80	-	-	%	Note 6
NTSC			55.0	55.1	-	%	Note 5
Luminance	L		250	300	-	cd/m <sup>2</sup>	Note 7

Note:

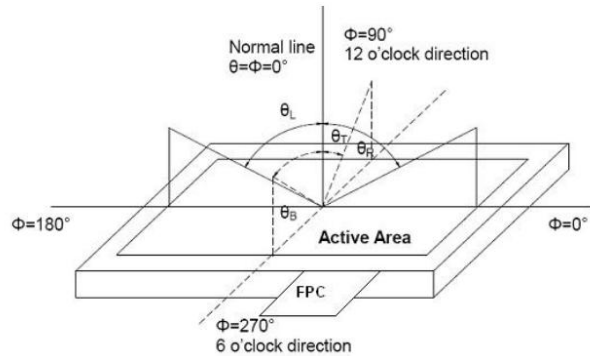
\*1.The ambient temperature is 25°C.

\*2. The test systems refer to Note 1 and Note 2.



**Note 1:**  
 The data are measured after LEDs are turned on for 5 minutes. LCM displays full white. The brightness is the average value of 9 measured spots. Measurement equipment SR-3A (1°)  
**Measuring condition:**  
 - Measuring surroundings: Dark room  
 - Measuring temperature: Ta=25°C.  
 - Adjust operating voltage to get optimum contrast at the center of the display.

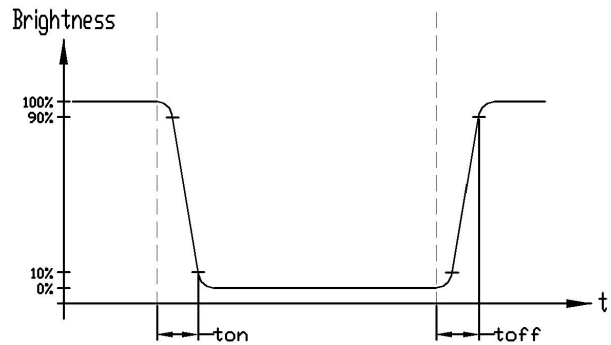
**Note 2:**  
 The definition of viewing angle:  
 Refer to the graph below marked by  $\theta$  and  $\phi$



**Note 3:**  
 The definition of contrast ratio (Test LCM using SR-3A (1°)):  

$$\text{Contrast Ratio (CR)} = \frac{\text{Luminance When LCD is at "White" state}}{\text{Luminance When LCD is at "Black" state}}$$
 (Contrast Ratio is measured in optimum common electrode voltage)

**Note 4:**  
 Definition of Response time. (Test LCD using BM-7A(2°)):  
 The output signals of photo detector are measured when the input signals are changed from "black" to "white" (falling time) and from "white" to "black" (rising time), respectively. The response time is defined as the time interval between the 10% and 90% of amplitudes. Refer to figure as below.



**Note 5:**  
 Definition of Color of CIE1931 Coordinate and NTSC Ratio.

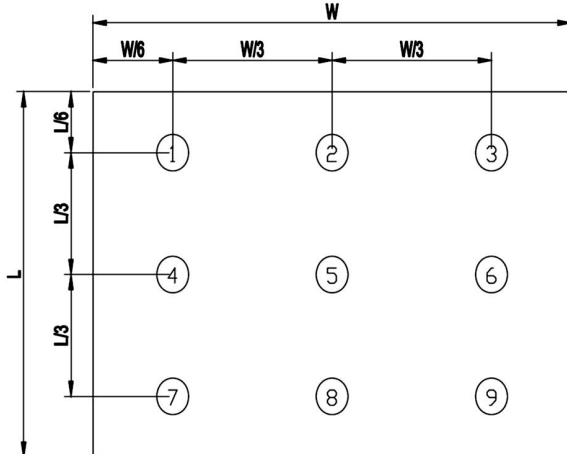
**Color gamut:**  

$$S = \frac{\text{Area of RGB triangle}}{\text{Area of NTSC triangle}} \times 100\%$$



**Note 6:**  
 The luminance uniformity is calculated by using following formula.  

$$\Delta Bp = Bp (\text{Min.}) / Bp (\text{Max.}) \times 100 (\%)$$
  
 Bp (Max.) = Maximum brightness in 9 measured spots  
 Bp (Min.) = Minimum brightness in 9 measured spots.



**Note 7:**  
 Measured the luminance of white state at center point..

## 6 LCD Module Design and Handling Precautions

- Please ensure V0, VCOM is adjustable, to enable LCD module get the best contrast ratio under different temperatures, view angles and positions.
- Normally display quality should be judged under the best contrast ratio within viewable area. Unexpected display pattern may come out under abnormal contrast ratio.
- Never operate the LCD module exceed the absolute maximum ratings.
- Never apply signal to the LCD module without power supply.
- Keep signal line as short as possible to reduce external noise interference.
- IC chip (e.g. TAB or COG) is sensitive to light. Strong light might cause malfunction. Light sealing structure casing is recommended.
- Make sure there is enough space (with cushion) between case and LCD panel, to prevent external force passed on to the panel; otherwise that may cause damage to the LCD and degrade its display result.
- Avoid showing a display pattern on screen for a long time (continuous ON segment).
- LCD module reliability may be reduced by temperature shock.
- When storing and operating LCD module, avoids exposure to direct sunlight, high humidity, high or low temperature. They may damage or degrade the LCD module.
- Never leave LCD module in extreme condition (max./min storage/operate temperature) for more than 48hr.
- Recommend LCD module storage conditions is 0 C~40 C <80%RH.
- LCD module should be stored in the room without acid, alkali and harmful gas.
- Avoid dropping & violent shocking during transportation, and no excessive pressure press, moisture and sunlight.
- LCD module can be easily damaged by static electricity. Please maintain an optimum anti-static working environment to protect the LCD module. (eg. ground the soldering irons properly)
- Be sure to ground the body when handling LCD module.
- Only hold LCD module by its sides. Never hold LCD module by applying force on the heat seal or TAB.
- When soldering, control the temperature and duration avoid damaging the backlight guide or diffuser which might degrade the display result such as uneven display.
- Never let LCD module contact with corrosive liquids, which might cause damage to the backlight guide or the

## 6 液晶显示模块设计和使用须知

- 请注意 V0, VCOM 的设定, 以确保液晶显示模块在不同的使用温度下以及在不同的视角和位置观察模块显示, 均能达到最佳对比度, 请务必将应用电路上设置为对比度可调。
- 请注意液晶显示模块的显示品质判定是指在正常对比度下以及视窗 (V. A) 范围内进行的, 非正常对比度下液晶可能会出现非预期的显示不良, 应注意区分。
- 请勿在最大额定值以外使用液晶显示模块。
- 请勿在没有接通电源的条件下, 给液晶显示模块输送信号。
- 请尽可能缩短信号线的连接, 以避免对液晶显示模块的信号干扰。
- 集成电路因 IC 芯片 (如 TAB 或 COG) 对紫外线极为敏感, 强光环境下可能会引起液晶显示模块功能失效, 故应采用不透光的外壳。
- 请在液晶显示模块与外壳之间保留足够的空间 (可使用衬垫), 以缓冲外力对液晶显示模块的损坏或因受力不均而产生的显示不均匀等异常现象。
- 避免液晶显示屏在某一画面下长时间点亮, 否则有出现残影的风险; 请通过软件每隔一段时间改变一次画面。
- 液晶显示模块的可靠性可能因温度冲击而降低。
- 请勿在阳光直射、高湿、高温或低温下储存和使用液晶显示模块, 这将造成液晶显示模块的损坏或失效。
- 请勿在极限环境 (最大/最小存储/工作温度) 下使用或放置液晶显示模块超过 48 小时以上。
- 液晶显示模块建议存储条件为: 0 C~40 C <80%RH。
- 请勿让液晶显示模块存储于带有 酸性, 碱性, 有害气体环境之中。
- 在运输过程中, 请勿让液晶显示模块跌落与猛烈震动, 同时避免 异常挤压, 高湿度, 与阳光照射。
- 液晶显示模块极易受静电损坏, 请务必保证液晶显示模块在防静电的工作环境中使用或保存。(如: 烙铁正确接地, 等)
- 拿取液晶显示模块时需注意操作人员的接地情况。
- 请手持液晶显示模块的边沿取放模块, 防止热压纸或 TAB 部位受力。
- 焊接液晶模块时, 请注意控制烙铁的温度、焊接时间, 以免烫坏导光板或偏光片, 导致显示不均匀等不良现象发生。
- 请勿使用洗板水等腐蚀性液体接触液晶模块, 以免腐蚀导光板或模块电路。

- electric circuit of LCD module.
- Only clean LCD with a soft dry cloth, Isopropyl Alcohol or Ethyl Alcohol. Other solvents (e.g. water) may damage the LCD.
  - Never add force to components of LCD module. It may cause invisible damage or degrade the module's reliability.
  - When mounting LCD module, please make sure it is free from twisting, warping and bending.
  - Do not add excessive force on surface of LCD, which may cause the display color change abnormally.
  - LCD panel is made with glass. Any mechanical shock (e.g. dropping from high place) will damage the LCD module.
  - Protective film is attached on LCD screen. Be careful when peeling off this protective film, since static electricity may be generated.
  - Polarizer on LCD gets scratched easily. If possible, do not remove LCD protective film until the last step of installation.
  - When peeling off protective film from LCD, static charge may cause abnormal display pattern. The symptom is normal, and it will turn back to normal in a short while.
  - LCD panel has sharp edges, please handle with care.
  - Never attempt to disassemble or rework LCD module.
  - If display panel is damaged and liquid crystal substance leaks out, be sure not to get any in your mouth, if the substance comes into contact with your skin or clothes promptly wash it off using soap and water.
- 仅可使用柔软的干布，异丙醇或乙醇清洁液晶屏表面，其他任何溶剂(如:水)都有可能损坏液晶模块。
  - 请勿挤压液晶显示模块上的元器件，以避免产生潜在的损坏或失效而影响产品可靠性。
  - 装配液晶显示模块时，请务必注意避免液晶显示模块的扭曲或变形。
  - 请勿挤压液晶显示屏表面，这将导致显示颜色的异常。
  - 液晶屏由玻璃制作而成，任何机械碰撞(如从高处跌落)均有可能损坏液晶显示模块。
  - 液晶屏表面带有保护膜，揭除保护膜时需要注意可能产生的静电。
  - 因液晶显示屏表面的偏光片极易划伤，安装完成之前请尽量不要揭下保护膜。
  - 请缓慢揭除保护膜，在此过程中液晶显示屏上可能会产生静电，此为正常情况，可在短时间内消失。
  - 请注意避免被液晶显示屏的边缘割伤。
  - 请不要试图拆卸或改造液晶显示模块。
  - 当液晶显示屏出现破裂，内部液晶液体可能流出；相关液体不可吞吃，绝对不可接触嘴巴，如接触到皮肤或衣服，请使用肥皂与清水彻底清洗。

## Warranty

This product has been manufactured to our company's specifications as a part for use in your company's general electronic products. It is guaranteed to perform according to delivery specifications. For any other use apart from general electronic equipment, we cannot take responsibility if the product is used in medical devices, nuclear power control equipment, aerospace equipment, fire and security systems, or any other applications in which there is a direct risk to human life and where extremely high levels of reliability are required. If the product is to be used in any of the above applications, we will need to enter into a separate product liability agreement.

- We cannot accept responsibility for any defect, which may arise from additional manufacturing of the product (including disassembly and reassembly), after product delivery.
- We cannot accept responsibility for any defect, which may arise after the application of strong external force to the product.
- We cannot accept responsibility for any defect, which may arise due to the application of static electricity after the product has passed our company's acceptance inspection procedures.
- When the product is in CCFL models, CCFL service life and brightness will vary according to the performance of the inverter used, leaks, etc. We cannot accept responsibility for product performance, reliability, or defect, which may arise.

We cannot accept responsibility for intellectual property of a third part, which may arise through the application of our product to our assembly with exception to those issues relating directly to the structure or method of manufacturing of our product.



深圳市拓普微科技开发有限公司

SHENZHEN TOPWAY TECHNOLOGY CO., LTD.

# SMART LCD

Command V6.12

Manual

Prepared by:  Date: 2023-11-13	Checked by:  Date:	Approved by:  Date:
--------------------------------------	--------------------------	---------------------------

Rev.	Descriptions	Enactment/Revision	Release Date
0.1	- Preliminary Draft release		2018-08-28
0.2	- add 0x94, 0x95		2018-11-22
0.3	- update section 2.1, 4.2.4, 4.2.7, 4.4.2, 4.4.9	chenjian	2019-06-28
0.4	- add 0x96,0x97,0xEE	liwenming	2023-11-13

## Table of Content

<b>1</b>	<b>Basic Specifications</b> .....	<b>4</b>
1.1	Hardware connection .....	4
<b>2</b>	<b>Command Structure</b> .....	<b>4</b>
2.1	Communication Packet Structure .....	4
2.1.1	Basic Packet: .....	4
2.1.2	Packet with length: .....	4
2.1.3	Packet with CRC: .....	4
2.2	Packet Timeout .....	5
2.3	Packet Acknowledgment .....	5
<b>3</b>	<b>Data arrangement</b> .....	<b>5</b>
3.1	Color Data Value Configuration .....	5
3.2	Data / Address / Page_ID / Location Values Configuration .....	5
<b>4</b>	<b>Command Descriptions</b> .....	<b>6</b>
4.1	Command table .....	6
4.2	Config/ Status Function Commands Details .....	7
4.2.1	hand_shake ( 0x30 ) .....	7
4.2.2	read_version ( 0x31 ) .....	7
4.2.3	read_pg_id ( 0x32 ) .....	8
4.2.4	touch_response ( 0x72/ 0x73/ 0x77/ 0x78/ 0x79 ) .....	8
4.2.5	set_sys_config ( 0xE0 ) .....	9
4.2.6	sel_project ( 0xE1 ) .....	10
4.2.7	touch_calib ( 0xE4 ) .....	10
4.2.8	screen_saver (0x5E) .....	10
4.2.9	backlight_ctrl ( 0x5F ) .....	10
4.2.10	buzzer_touch_sound ( 0x79 ) .....	10
4.2.11	buzzer_ctrl ( 0x7A ) .....	10
4.2.12	Flash_write ( 0x90 ) .....	11
4.2.13	Flash_read ( 0x91 ) .....	11
4.2.14	USR_bin_read ( 0x93 ) .....	11
4.2.15	Ymodem_transmit ( 0x96 ) .....	12
4.2.16	Checksum_calculate ( 0x97 ) .....	12
4.2.17	RTC_read ( 0x9B ) .....	12
4.2.18	RTC_set ( 0x9C ) .....	12
4.2.19	U_drv_format ( 0xE2 ) .....	13
4.2.20	U_drv_unlock ( 0xE3 ) .....	13
4.2.21	Reset ( 0xEE ) .....	13
4.3	Display Control Function Commands Details .....	13
4.3.1	disp_page ( 0x70 ) .....	13
4.3.2	set_element_fg ( 0x7E ) .....	13
4.3.3	set_element_bg ( 0x7F ) .....	14
4.3.4	set_codepage (0xE7) .....	14
4.3.5	suspend_vp_refresh (0xE8) .....	14
4.4	VP Function Commands Details .....	14
4.4.1	Successive_write ( 0x82 ) .....	14
4.4.2	Successive_read ( 0x83 ) .....	15
4.4.3	VP_Backup ( 0x94 ) .....	16
4.4.4	VP_Preload ( 0x95 ) .....	16
4.4.5	BP1_write ( 0x4B ) .....	16
4.4.6	BP1_write_compress ( 0x4C ) .....	17
4.4.7	G16_write ( 0x4D ) .....	17
4.4.8	G16_write_rotate ( 0x4E ) .....	17
4.4.9	Reg_Write ( 0x3B ) .....	18
4.4.10	Reg_Read ( 0x3C ) .....	18
4.4.11	STR_write ( 0x42 ) .....	18
4.4.12	STR_read ( 0x43 ) .....	19
4.4.13	STR_fill ( 0x46 ) .....	19
4.4.14	N16_write ( 0x3D ) .....	19
4.4.15	N16_read ( 0x3E ) .....	20
4.4.16	N16_fill ( 0x3F ) .....	20

---

4.4.17	N32_write ( 0x44 )	20
4.4.18	N32_read ( 0x45 )	21
4.4.19	N32_fill ( 0x47 )	21
4.4.20	N64_write ( 0x48 )	21
4.4.21	N64_read ( 0x49 )	22
4.4.22	N64_fill ( 0x4A )	22
<b>Appendix 1: CRC Calculate</b>		<b>23</b>

## 1 Basic Specifications

TOPWAY Smart LCD serial command is for real-time control and access. Host machine get the data which input through the Smart LCD interface or provide the data for display.

### 1.1 Hardware connection

Smart LCD serial UART interface are mainly base on RS232-C standard, by default, config as 8N1 115200bps.

## 2 Command Structure

### 2.1 Communication Packet Structure

TOPWAY SmartLCD offer 3 kinds of Communication Packet Structure, which can be defined in editor project setting.

#### 2.1.1 Basic Packet:

Seq	Code	Code type	Description
1	0xAA	Packet header	1byte
2	Cmd-code	Command code	1byte
3	Par-data	Parameter or Data	(*1)
4	0xCC	Packet tail	4byte
	0x33		
	0xC3		
	0x3C		

#### 2.1.2 Packet with length:

Seq	Code	Code type	Description
1	0xAA	Packet header	1byte
2	Len	Packet length	2byte(*2)
3	Cmd-code	Command code	1byte
4	Par-data	Parameter or Data	(*1)
5	0xCC	Packet tail	4byte
	0x33		
	0xC3		
	0x3C		

#### 2.1.3 Packet with CRC:

Seq	Code	Code type	Description
1	0xAA	Packet header	1byte
2	Len	Packet length	2byte(*2)
3	Cmd-code	Command code	1byte
4	Par-data	Parameter or Data	(*1)
5	0xCC	Packet tail	2byte
	0x33		
6	CRCL		2byte(*3)
	CRCH		

Note.

\*1. Unless otherwise specified,

all the multi-byte values, data, address' byte sequence are MSB first, LSB last.

\*2. Packet length: from Seq3 to the end. (no. of byte)

\*3. CRC Polynomial:  $x^{16}+x^{15}+x^2+1$ , Calculate the CRC value from Seq3 to Seq5. Please refer to appendix 1.

**2.2 Packet Timeout**

TOPWAY SmartLCD support Timeout setting, which can be defined in editor project setting. Timeout options: None, 1s, 2s, 3s, 5s, 10s, 20s. If timeout, The incomplete Packet will be discarded.

**2.3 Packet Acknowledgment**

Packet Acknowledgment is two byte in ASCII (module → host):

Response	code	Description
Command (in packet) executed and wait for next Command	":>"	In ASCII (0x3a, 0x3e)
Command (in packet) error and wait for next Command	"!>"	In ASCII (0x21,0x3e)

Note.

\*1. Packet Acknowledgement response to a valid packet only.

**3 Data arrangement**

**3.1 Color Data Value Configuration**

16 bit Color value

16 bit color value															
<b>R4</b>	<b>R3</b>	<b>R2</b>	<b>R1</b>	<b>R0</b>	<b>G5</b>	<b>G4</b>	<b>G3</b>	<b>G2</b>	<b>G1</b>	<b>G0</b>	<b>B4</b>	<b>B3</b>	<b>B2</b>	<b>B1</b>	<b>B0</b>
High byte (MSB)								Low byte (LSB)							
D7	D6	D5	D4	D3	D2	D1	D0	D7	D6	D5	D4	D3	D2	D1	D0

**3.2 Data / Address / Page\_ID / Location Values Configuration**

64bit value

64 bit number value							
<b>D63...D56</b>	<b>D55...D48</b>	<b>D47...D40</b>	<b>D39..D32</b>	<b>D31...D24</b>	<b>D23...D16</b>	<b>D15...D8</b>	<b>D7...D0</b>
Byte7 (MSB)							Byte0 (LSB)
D7...D0	D7...D0	D7...D0	D7...D0	D7...D0	D7...D0	D7...D0	D7...D0

32bit value

32 bit number value			
<b>D31...D24</b>	<b>D23...D16</b>	<b>D15...D8</b>	<b>D7...D0</b>
Byte3 (MSB)			Byte0 (LSB)
D7...D0	D7...D0	D7...D0	D7...D0

16bit value

16 bit number value	
<b>D15...D8</b>	<b>D7...D0</b>
High Byte (MSB)	Low Byte (LSB)
D7...D0	D7...D0



## 4 Command Descriptions

### 4.1 Command table

Functions	Name	Code	Description
Config/ Status Functions	hand_shake	0x30	Read a Hand Shake
	read_version	0x31	Read firmware version
	read_pg_id	0x32	Read Current page ID
	touch_response	0x72/0x73/ 0x77/0x78/ 0x79	see also set_sys_config
	set_sys_config	0xE0	System parameter configuration and Baud Rate
	sel_project	0xE1	Specify operating project folder
	touch_calib	0xE4	Touch panel calibration(only for RTP)
	screen_saver	0x5E	Screen saver (backlight dim down time out)
	backlight_ctrl	0x5F	backlight brightness control (64 levels)
	buzzer_touch_sound	0x79	buzzer enable time length (in 10ms step)
	buzzer_ctrl	0x7A	Buzzer control
	Flash_write	0x90	Write data to the flash
	Flash_read	0x91	Read data from the flash
	RTC_read	0x9B	Read the RTC values
	RTC_set	0x9C	Set the RTC
	USR_bin_read	0x93	Read data from the USR_bin
	Ymodem_transmit	0x96	Enter the Ymodem Transfer mode
	Checksum_calculate	0x97	Calculate the file Checksum by filename and path
	U_drv_format	0xE2	Format the U_drv
	U_drv_unlock	0xE3	Unlock the U_drv with pre-stored password
	Reset	0xEE	Reset the smart LCD by watchdog
Display Control Functions	disp_page	0x70	Display a pre-stored TML file (page)
	set_element_fg	0x7E	Set the foreground color of STR, N16, N32 or N64
	set_element_bg	0x7F	Set the background color of STR, N16, N32 or N64
	set_codepage	0xE7	Sets country character set and code-page character set
	suspend_vp_fresh	0xE8	Set the screen to pause the refresh and deactivate the touchkey or release the pause to refresh and enable the touchkey
VP Functions	Successive_write	0x82	Write successive value to VP_N16, VP_N32, VP_N64
	Successive_read	0x83	Read successive value from VP_N16, VP_N32, VP_N64
	VP_Backup	0x94	VP Backup to Flash or VP Restore from Flash
	VP_Preload	0x95	VP Preload from usr.bin
	BP1_write	0x4B	Write bit-map (1bpp) data to VP_BP1
	BP1_write_comp	0x4C	Write compressed bit-map (1bpp) data to VP_BP1
	G16_write	0x4D	Write 16bit (signed integer) graphic array to VP_G16
	G16_write_rotate	0x4E	Rotate the VP_G16 array data inside the module and write a 16bit (signed integer) value into end-of-array
	Reg_Write	0x3B	Write System Register
	Reg_Read	0x3C	Read System Register
	STR_write	0x42	Write string to VP_STR
	STR_read	0x43	Read string form VP_STR
	STR_fill	0x46	Fill strings to the VP_STR
	N16_write	0x3d	Write 16bit (signed integer) value to VP_N16
	N16_read	0x3e	Read 16bit (signed integer) value from VP_N16
	N16_fill	0x3f	Fill numbers to the VP_N16
	N32_write	0x44	Write 32bit (signed integer) value to VP_N32
	N32_read	0x45	Read 32bit (signed integer) value from VP_N32
	N32_fill	0x47	Fill numbers to the VP_N32
	N64_write	0x48	Write 64bit (signed integer) value to VP_N64
	N64_read	0x49	Read 64bit (signed integer) value from VP_N64
	N64_fill	0x4A	Fill numbers to the VP_N64

## 4.2 Config/ Status Function Commands Details

### 4.2.1 hand\_shake ( 0x30 )

seq	Cmd-code / Par-data	Descriptions
1	0x30	Read a Hand Shake

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

#### Response code:

Seq.	Content	Byte in Hex	Descriptions	
1 <sup>st</sup>	Header	0xAA	Communication packet header	
2 <sup>nd</sup>	Command	0x30	Command executed	
3 <sup>rd</sup>	"T"	0x54	"Topway HMT Ready\0" in ASCII	
4 <sup>th</sup>	"o"	0x6f		
5 <sup>th</sup>	"P"	0x70		
6 <sup>th</sup>	"w"	0x77		
7 <sup>th</sup>	"a"	0x61		
8 <sup>th</sup>	"y"	0x79		
9 <sup>th</sup>	" "	0x20		
10 <sup>th</sup>	"H"	0x48		
11 <sup>th</sup>	"M"	0x4d		
12 <sup>th</sup>	"T"	0x54		
13 <sup>th</sup>	" "	0x20		
14 <sup>th</sup>	"R"	0x52		
15 <sup>th</sup>	"e"	0x65		
16 <sup>th</sup>	"a"	0x61		
17 <sup>th</sup>	"d"	0x64		
18 <sup>th</sup>	"y"	0x79		'\0'(0x00): string end mark
19 <sup>th</sup>	\0	0x00		
20 <sup>th</sup>	Tail	0xCC	Communication packet tail	
21 <sup>st</sup>		0x33		
22 <sup>nd</sup>		0xC3		
23 <sup>rd</sup>		0x3C		

Note.

\*1. The Response code with communication packet format (see Communication Packet Structure Section for details)

### 4.2.2 read\_version ( 0x31 )

Seq	Cmd-code / Par-data	Descriptions
1	0x31	Read firmware version

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

#### Response code:

Seq.	Content	Byte in Hex	Descriptions
1 <sup>st</sup>	Header	0xAA	Communication packet header
2 <sup>nd</sup>	Command	0x31	Command executed
3 <sup>rd</sup>	"1"	0x31	"1.06\0" in ASCII Where firmware version is V1.06(example)
4 <sup>th</sup>	" "	0x2e	
5 <sup>th</sup>	"0"	0x30	
6 <sup>th</sup>	"6"	0x36	
7 <sup>th</sup>	\0	0x00	'\0'(0x00): string end mark
8 <sup>th</sup>	Tail	0xCC	Communication packet tail
9 <sup>th</sup>		0x33	
10 <sup>th</sup>		0xC3	
11 <sup>th</sup>		0x3C	

Note.

\*1. The Response code with communication packet format (see Communication Packet Structure Section for details)

**4.2.3 read\_pg\_id ( 0x32 )**

Seq	Cmd-code / Par-data	Descriptions
1	0x32	Read Current page ID

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**Response code:**

Seq.	Content	Byte in Hex	Descriptions
1 <sup>st</sup>	Header	0xAA	Communication packet header
2 <sup>nd</sup>	Command	0x32	Command executed
3 <sup>rd</sup>	Page ID	Page_IDh	Current Page ID in 16bit binary value
4 <sup>th</sup>		Page_IDl	
5 <sup>th</sup>	Tail	0xCC	Communication packet tail
6 <sup>th</sup>		0x33	
7 <sup>th</sup>		0xC3	
8 <sup>th</sup>		0x3C	

Note.

\*1. The Response code with communication packet format (see Communication Packet Structure Section for details)

**4.2.4 touch\_response ( 0x72/ 0x73/ 0x77/ 0x78/ 0x79 )**

seq	Cmd-code / Par-data	Descriptions
1	--	Use set_sys_config to config the functions

**Touch Release Coordinate Response code (0x72):**

Seq.	Content	Byte in Hex	Descriptions
1 <sup>st</sup>	Header	0xAA	Communication packet header
2 <sup>nd</sup>	Command	0x72	Touched release Coordinate
3 <sup>rd</sup>	X coordinate	Xh	Coordinate in 16bit binary value X = horizontal coordinate Y = vertical coordinate
4 <sup>th</sup>		Xl	
5 <sup>th</sup>	Y coordinate	Yh	Communication packet tail
6 <sup>th</sup>		Yl	
7 <sup>th</sup>	Tail	0xCC	Communication packet tail
8 <sup>th</sup>		0x33	
9 <sup>th</sup>		0xC3	
10 <sup>th</sup>		0x3C	

Note.

\*1. The Response code with communication packet format (see Communication Packet Structure Section for details)

**Touch Down Coordinate Response code ( 0x73 ):**

Seq.	Content	Byte in Hex	Descriptions
1 <sup>st</sup>	Header	0xAA	Communication packet header
2 <sup>nd</sup>	Command	0x73	Touched down Coordinate
3 <sup>rd</sup>	X coordinate	Xh	Coordinate in 16bit binary value X = horizontal coordinate Y = vertical coordinate
4 <sup>th</sup>		Xl	
5 <sup>th</sup>	Y coordinate	Yh	Communication packet tail
6 <sup>th</sup>		Yl	
7 <sup>th</sup>	Tail	0xCC	Communication packet tail
8 <sup>th</sup>		0x33	
9 <sup>th</sup>		0xC3	
10 <sup>th</sup>		0x3C	

Note.

\*1. The Response code with communication packet format (see Communication Packet Structure Section for details)

**Touch Key ID Response code ( 0x78 ):**

Seq.	Content	Byte in Hex	Descriptions
1 <sup>st</sup>	Header	0xAA	Communication packet header
2 <sup>nd</sup>	Command	0x78	Touched release Key_ID defined by TOPWAY TML Graphic Editor will be response to host
3 <sup>rd</sup>	Page_ID	Page_IDh	Page_ID = the touch key in page(16bit binary value)
4 <sup>th</sup>		Page_IDl	
5 <sup>th</sup>	Key_ID	Key_ID	Key_ID (8bit binary value)
6 <sup>th</sup>	Tail	0xCC	Communication packet tail
7 <sup>th</sup>		0x33	
8 <sup>th</sup>		0xC3	
9 <sup>th</sup>		0x3C	

Note.

\*1. The Response code with communication packet format (see Communication Packet Structure Section for details)

**Touch Key ID Response code ( 0x79 ):**

Seq.	Content	Byte in Hex	Descriptions
1 <sup>st</sup>	Header	0xAA	Communication packet header
2 <sup>nd</sup>	Command	0x79	Touched down Key_ID defined by TOPWAY TML Graphic Editor will be response to host
3 <sup>rd</sup>	Page_ID	Page_IDh	Page_ID = the touch key in page(16bit binary value)
4 <sup>th</sup>		Page_IDl	
5 <sup>th</sup>	Key_ID	Key_ID	Key_ID (8bit binary value)
6 <sup>th</sup>	Tail	0xCC	Communication packet tail
7 <sup>th</sup>		0x33	
8 <sup>th</sup>		0xC3	
9 <sup>th</sup>		0x3C	

Note.

\*1. The Response code with communication packet format (see Communication Packet Structure Section for details)

**Touch Key VP\_ADD+VP\_Value Response code ( 0x77 ):**

Seq.	Content	Byte in Hex	Descriptions
1 <sup>st</sup>	Header	0xAA	Communication packet header
2 <sup>nd</sup>	Command	0x77	Touch Key VP_ADD+VP_Value Response code
3 <sup>rd</sup>	VP_ADD	Addr3 (MSB)	VP Address
4 <sup>th</sup>		Addr2	0x080000 ~ 0x08FFFF:VP_N16 Address
5 <sup>th</sup>		Addr1	0x020000 ~ 0x02FFFF:VP_N32 Address
6 <sup>th</sup>		Addr0(LSB)	0x030000 ~ 0x03FFFF:VP_N64 Address 0x000000 ~ 0x01FFFF:VP_STR Address
7 <sup>th</sup>	Data	:	No.of byte
:		:	VP_N16: 2byte
:		:	VP_N32: 4byte
:		:	VP_N64: 8byte
:	Tail	:	VP_STR: string (with end mark ("'\0'(0x00)))
:		0xCC	Communication packet tail
:		0x33	
:		0xC3	
:	0x3C		

Note.

\*1. The Response code with communication packet format (see Communication Packet Structure Section for details)

**4.2.5 set\_sys\_config ( 0xE0 )**

seq	Cmd-code / Par-data	Descriptions
1	0xE0	Baud Rate and system parameter configuration
2	0x55	
3	0xAA	
4	0x5A	
5	0xA5	
6	Baud_Set	Baudrate Set: 0x00 = 1200bps 0x01 = 2400bps 0x02 = 4800bps 0x03 = 9600bps 0x04 = 19200bps 0x05 = 38400bps 0x06 = 57600bps 0x07 = 115200bps
7	sys_par1	Bit7 = 0: Touch panel function disable Bit7 = 1: Touch panel functions enable (*3)(default) Bit[1..0]: Touch actions configuration (*2, *3)
8	0x00	Reserved

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

\*2. Touch panel configuration:

Sys_par1 Bit7	Sys_par1 Bit1	Sys_par1 Bit0	Response To host	Descriptions
0	0	0	Null	Not touch panel functions
1	0	1	Coordinates	Touch down Coordinate will be response to host
1	1	0	Coordinates	Touch release Coordinate will be response to host
1	1	1	Key ID	Touch Key_ID defined by TOPWAY TML Graphic Editor will be response to host

\*3. see set\_touch section for response code

**4.2.6 sel\_project ( 0xE1 )**

seq	Cmd-code / Par-data	Descriptions
1	0xE1	Select project folder
2	Prj_ID	0~9, project ID 0: System execute the default project "THMT" 1~9: System execute the project "THMT01"~"THMT09"

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**4.2.7 touch\_calib ( 0xE4 )**

seq	Cmd-code / Par-data	Descriptions
1	0xE4	Touch panel calibration
2	0x55	
3	0xAA	
4	0x5A	
5	0xA5	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

\*2. Keep pressing the top right corner of touch panel during power on, could also trigger the touch\_calib function (only for RTP)

**4.2.8 screen\_saver (0x5E)**

seq	Cmd-code / Par-data	Descriptions
1	0x5E	Screen saver
2	Time1h	time out time in seconds, range: 0x0000 ~ 0xffff (0x0000: disable screen saver function) (*2)
3	Time1l	
4	PWM_LE	PWM_LE = 0 ~ 0x3F (default 0x19 in dim down), the backlight dimmed level in screen saving mode (*2) Screensavers brightness can not be greater than the backlight brightness.

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

\*2. default value defined by TML graphic editor configuration

**4.2.9 backlight\_ctrl ( 0x5F )**

seq	Cmd-code / Par-data	Descriptions
1	0x5F	backlight brightness control
2	PWM_LE	PWM_LE=0x00 ~ 0x3F (*2)

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

\*2. default value defined by TML graphic editor configuration

**4.2.10 buzzer\_touch\_sound ( 0x79 )**

seq	Cmd-code / Par-data	Descriptions
1	0x79	buzzer touch sound control
2	Time	Sounding time length (in 10ms), range 0x00~0x3F 0x00= disable (*2)

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

\*2. default value defined by TML graphic editor configuration

**4.2.11 buzzer\_ctrl ( 0x7A )**

seq	Cmd-code / Par-data	Descriptions
1	0x7A	Buzzer control
2	Loop count	Loop count, Range: 0x01 ~ 0xFF. 0xFF = buzzer infinite loop
3	T1	Buzzer play time 1 Range: 0x00 ~ 0xFF (0~25.5s)(unit 100ms)
4	T2	Buzzer play time 2 Range: 0x00 ~ 0xFF (0~25.5s)(unit 100ms)
5	Freq1	T1 time Buzzer frequency, Unit 100 Hz Ranges: 0x05 ~ 0x32 (500Hz ~ 5KHz) 0x00 = T1 time period buzzer turn off
6	Freq2	T2 time Buzzer frequency, Unit 100 Hz Ranges: 0x05 ~ 0x32 (500Hz ~ 5KHz) 0x00 = T1 time period buzzer turn off

Note:

1. The buzzer sound time is (T1 + T2)\*100ms

**4.2.12 Flash\_write ( 0x90 )**

seq	Cmd-code / Par-data	Descriptions
1	0x90	Write data to the flash at specified address
2	Address3(MSB)	the specified start address to write Address range =0x00000 ~ 0x03FFFF
3	Address2	
4	Address1	
5	Address0(LSB)	
6	Data_Lengthh	The no. of data byte to write. Length =0x0001 ~ 0x0400
7	Data_Lengthl	
8	Data	data to write.
:	:	
:	:	
:	:	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**4.2.13 Flash\_read ( 0x91 )**

seq	Cmd-code / Par-data	Descriptions
1	0x91	Read data from the flash at specified address
2	Address3(MSB)	the specified start address to write Address range =0x00000 ~ 0x03FFFF
3	Address2	
4	Address1	
5	Address0(LSB)	
6	Data_Lengthh	The no. of data byte to read Length =0x0001 ~ 0x0400
7	Data_Lengthl	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**Response code:**

Seq.	Content	Byte in Hex	Descriptions
1 <sup>st</sup>	Header	0xAA	Communication packet header
2 <sup>nd</sup>	Command	0x91	Command executed
3 <sup>rd</sup>	Data	data	Read back data
:		:	
:		:	
:		:	
:	Tail	0xCC	Communication packet tail
:		0x33	
:		0xC3	
:		0x3C	

Note.

\*1. The Response code with communication packet format (see Communication Packet Structure Section for details)

**4.2.14 USR\_bin\_read ( 0x93 )**

seq	Cmd-code / Par-data	Descriptions
1	0x93	Read USR_bin data from the flash at specified address
2	Address3(MSB)	the specified start address to write Address range = 0x00000 ~ 0x03FFFF
3	Address2	
4	Address1	
5	Address0(LSB)	
6	Data_Lengthh	The no. of data byte to read Length = 0x0001 ~ 0x0400
7	Data_Lengthl	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**Response code:**

Seq.	Content	Byte in Hex	Descriptions
1 <sup>st</sup>	Header	0xAA	Communication packet header
2 <sup>nd</sup>	Command	0x93	Command executed
3 <sup>rd</sup>	Data	data	Read back data
:		:	
:		:	
:		:	
:	Tail	0xCC	Communication packet tail
:		0x33	
:		0xC3	
:		0x3C	

Note.

\*1. The Response code with communication packet format (see Communication Packet Structure Section for details)

**4.2.15 Ymodem\_transmit ( 0x96 )**

seq	Cmd-code / Par-data	Descriptions
1	0x96	Enter the Ymodem Transfer mode
2	0x55	Screen display "Ymodem start" and serial port will cycle to receive character C
3	0xAA	Transmission waiting time <200s, otherwise screen display "End/Please reset the System"
4	0x5A	
5	0xA5	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**4.2.16 Checksum\_calculate ( 0x97 )**

seq	Cmd-code / Par-data	Descriptions
1	0x97	Calculates the checksum of the specified file
2	1 <sup>st</sup> character	32-byte filename (with path) and is case insensitive.
3	2 <sup>nd</sup> character	Filename length less than 32 bytes, fill 0.
4	3 <sup>rd</sup> character	1 <sup>st</sup> character = the first character of file name (contains path)
5	4 <sup>th</sup> character	
:	:	
:	:	
:	:	
32	:	
33	:	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**Response code:**

Seq.	Content	Byte in Hex	Descriptions
1 <sup>st</sup>	Header	0xAA	Communication packet header
2 <sup>nd</sup>	Command	0x97	Command executed
3 <sup>rd</sup>	File name	1 <sup>st</sup> character	Read back filename with path
:		:	
:		:	
35	checksum	1 <sup>st</sup> byte	Read back checksum with file The first character is the low byte
36		2 <sup>nd</sup> byte	
37		3 <sup>rd</sup> byte	
38		4 <sup>th</sup> byte	
39	Tail	0xCC	Communication packet tail
40		0x33	
41		0xC3	
42		0x3C	

Note.

\*1. The Response code with communication packet format (see Communication Packet Structure Section for details)

**4.2.17 RTC\_read ( 0x9B )**

seq	Cmd-code / Par-data	Descriptions
1	0x9B	Read the current RTC value

**Response code:**

Seq.	Content	Byte in Hex	Descriptions
1 <sup>st</sup>	Header	0xAA	Communication packet header
2 <sup>nd</sup>	Command	0x9B	Command executed
3 <sup>rd</sup>	Date	Year	Year: 00~99 (00=year 2000) (8bit binary value)
4 <sup>th</sup>		Month	Month: 01~12 (8bit binary value)
5 <sup>th</sup>		Day	Day: 01~31 (8bit binary value)
6 <sup>th</sup>	Time	Hour	Hour 00~23 (24hr format)(8bit binary value)
7 <sup>th</sup>		Minute	Minutes 00~59 (8bit binary value)
8 <sup>th</sup>		Second	Second 00~59 (8bit binary value)
9 <sup>th</sup>	Tail	0xCC	Communication packet tail
10 <sup>th</sup>		0x33	
11 <sup>th</sup>		0xC3	
12 <sup>th</sup>		0x3C	

Note.

\*1. The Response code with communication packet format (see Communication Packet Structure Section for details)

**4.2.18 RTC\_set ( 0x9C )**

seq	Cmd-code / Par-data	Descriptions
-----	---------------------	--------------

1	0x9C	Set the RTC
2	Year	Year = 00~99(2000 ~ 2099)
3	Month	Month = 00~12
4	Date	Date = 00~31
5	Hour	Hour = 00~23
6	Minute	Minute = 00~59
7	Second	Second = 00~59

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**4.2.19 U\_drv\_format ( 0xE2 )**

seq	Cmd-code / Par-data	Descriptions
1	0xE2	Format the USB drive.
2	0x55	All the files (include the security lock file) will be erased.
3	0xAA	
4	0x5A	
5	0xA5	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**4.2.20 U\_drv\_unlock ( 0xE3 )**

seq	Cmd-code / Par-data	Descriptions
1	0xE3	Unlock the USB drive of file read/write with pre-stored password.
2	PW	PW: password in ASCII Length = 127max. '\0'(0x00): string end mark
:	:	
:	:	
:	'\0'	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**4.2.21 Reset ( 0xEE )**

seq	Cmd-code / Par-data	Descriptions
1	0xEE	Reset the smart LCD by watchdog
2	0xAA	
3	0x55	
4	0xA5	
5	0x5A	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**4.3 Display Control Function Commands Details**

**4.3.1 disp\_page ( 0x70 )**

seq	Cmd-code / Par-data	Descriptions
1	0x70	Display a pre-stored TML file(page)
2	Page_IDh	Page_ID = 0~999
3	Page_IDl	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**4.3.2 set\_element\_fg ( 0x7E )**

Seq	Cmd-code / Par-data	Descriptions
1	0x7E	Set foreground colors of the STR, N16, N32 or N64
2	Element	0x00 = STR; 0x01 = N16, N32, N64
3	Page_IDh	Page_ID = 0~999
4	Page_IDl	
5	Element_ID	VP_STR = 0~127; N16, N32, N64 = 0~119
6	0x00	Reserve
7	FGh	Foreground color(0~0xffff)
8	FGl	

Note.



\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**4.3.3 set\_element\_bg ( 0x7F )**

Seq	Cmd-code / Par-data	Descriptions
1	0x7F	Set background color of the STR, N16, N32 or N64
2	Element	0x00 = STR; 0x01 = N16, N32, N64
3	Page_IDh	Page_ID = 0~999
4	Page_IDl	
5	Element_ID	VP_STR = 0~127, N16, N32, N64 = 0~119
6	Mode	0x00: non transparent; 0x01 : transparent
7	BGh	Background color(0 ~ 0xffff)
8	BGl	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**4.3.4 set\_codepage (0xE7)**

Seq	Cmd-code / Par-data	Descriptions
1	0xE7	Sets country character set and code-page character set
2	Country	1 ~ 11, country character set
3	Codepage	1 ~ 22, code-page character set

Note.

\*1. Country and CodePage table

Country Code	Descriptions	Code Page	Descriptions
1	USA	1	437 (OEM – United States)
2	France	2	737 (OEM – Greek 437G)
3	Germany	3	852 (OEM – Latin II)
4	United Kingdom	4	860 (OEM – Portuguese)
5	Demark I	5	863 (OEM – Canadian French)
6	Demark II	6	865 (OEM – Nordic)
7	Sweden	7	866 (OEM – Russian)
8	Italy	8	874 (ANSI/OEM – Thai)
9	Spain	9	932 (ANSI/OEM – Japanese Shift-JIS)
10	Japan	10	1250 (ANSI - Central Europe)
11	Norway	11	1251 (ANSI – Cyrillic)
--	--	12	1252 (ANSI – Latin I)
--	--	13	1253 (ANSI – Greek)
--	--	14	1254 (ANSI – Turkish)
--	--	15	1255 (ANSI – Hebrew)
--	--	16	1256 (ANSI – Arabic)
--	--	17	1257 (ANSI – Baltic)
--	--	18	1258 (ANSI – Viet Nam)
--	--	19	GB2312
--	--	20	GBK
--	--	21	EUC_KR
--	--	22	Big5

**4.3.5 suspend\_vp\_refresh (0xE8)**

Seq	Cmd-code / Par-data	Descriptions
1	0xE8	Set the screen to pause the refresh and deactivate the touchkey or release the pause to refresh and enable the touchkey
2	55	
3	AA	
4	5A	
5	A5	
6	Mode	0x00: release the pause to refresh and enable the touchkey 0x01: pause the refresh and deactivate the touchkey

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**4.4 VP Function Commands Details**

**4.4.1 Successive\_write ( 0x82 )**

Seq	Cmd-code / Par-data	Descriptions
1	0x82	Write successive value to VP_N16, VP_N32, VP_N64
2	Addr3(MSB)	VP_N16 Address = 0x080000 ~ 0x08FFFF (should be aligned every 2 byte)
3	Addr2	
4	Addr1	VP_N32 Address = 0x020000 ~ 0x02FFFF (should be aligned every 4 byte)
5	Addr0(LSB)	VP_N64 Address = 0x030000 ~ 0x03FFFF (should be aligned every 8 byte)

6	Length	The number of data to write (Length = 1~255)
7	Data 1(MSB)	the value to write
8	Data 2	No. of byte of Data:
9	Data 3	VP_N16 = Length *2,
:	:	VP_N32 = Length *4,
:	Data n(LSB)	VP_N64 = Length *8,

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

#### 4.4.2 Successive\_read ( 0x83 )

Seq	Cmd-code / Par-data	Descriptions
1	0x83	Read successive value to VP_N16, VP_N32, VP_N64
2	Addr3(MSB)	VP_N16 Address = 0x080000 ~ 0x08FFFF (should be aligned every 2 byte)
3	Addr2	VP_N32 Address = 0x020000 ~ 0x02FFFF (should be aligned every 4 byte)
4	Addr1	
5	Addr0(LSB)	VP_N64 Address = 0x030000 ~ 0x03FFFF (should be aligned every 8 byte)
6	Length	The number of data to write (Length = 1 ~ 255)

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

#### Response code:

Seq.	Content	Byte in Hex	Descriptions
1 <sup>st</sup>	Header	0xAA	Communication packet header
2 <sup>nd</sup>	Command	0x83	Command executed
3 <sup>rd</sup>	VP_ADD	Addr3(MSB)	VP Address 0x080000 ~ 0x08FFFF:VP_N16 Address 0x020000 ~ 0x02FFFF:VP_N32 Address 0x030000 ~ 0x03FFFF:VP_N64 Address
4 <sup>th</sup>		Addr2	
5 <sup>th</sup>		Addr1	
6 <sup>th</sup>		Addr0(LSB)	
7 <sup>th</sup>	Length	NO.	No. of data
8 <sup>th</sup>	Data	Data n(MSB)	No. of byte of Data: VP_N16 = Length *2, VP_N32 = Length *4, VP_N64 = Length *8,
:		:	
:		:	
(n-1) <sup>th</sup>		Data1	
n <sup>th</sup>	Data0(LSB)		
(n+1) <sup>th</sup>	Tail	0xCC	Communication packet tail
(n+2) <sup>th</sup>		0x33	
(n+3) <sup>th</sup>		0xC3	
(n+4) <sup>th</sup>		0x3C	

Note.

\*1. The Response code with communication packet format (see Communication Packet Structure Section for details)

**4.4.3 VP\_Backup ( 0x94 )**

seq	Cmd-code / Par-data	Descriptions
1	0x94	VP Backup to Flash or VP Restore from Flash
2	Dir	1: VP Restore 0: VP Backup
3	Addr3(MSB)	the specified Flash start address Address range = 0x000000 ~ 0x3FFFFFF
4	Addr2	
5	Addr1	
6	Addr0(LSB)	
7	VP Addr3(MSB)	VP Address
8	VP Addr2	0x080000 ~ 0x08FFFF:VP_N16 Address
9	VP Addr1	0x020000 ~ 0x02FFFF:VP_N32 Address
10	VP Addr0(LSB)	0x030000 ~ 0x03FFFF:VP_N64 Address 0x000000 ~ 0x01FFFF:VP_STR Address
11	VP_Lengthh	The no. of VP
12	VP_Lengthl	VP_Length = 0x0001 ~ 0x8000

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

\*2. Random data may be restore, before the first time VP backup operation.

**4.4.4 VP\_Preload ( 0x95 )**

seq	Cmd-code / Par-data	Descriptions
1	0x95	VP Preload from USR_bin
2	01	
3	Addr3(MSB)	the specified usr.bin start address . Address range = 0x000000 ~ 0x3FFFFFF
4	Addr2	
5	Addr1	
6	Addr0(LSB)	
7	VP Addr3(MSB)	VP Address
8	VP Addr2	0x080000 ~ 0x08FFFF:VP_N16 Address
9	VP Addr1	0x020000 ~ 0x02FFFF:VP_N32 Address
10	VP Addr0(LSB)	0x030000 ~ 0x03FFFF:VP_N64 Address 0x000000 ~ 0x01FFFF:VP_STR Address
11	VP_Lengthh	The no. of VP
12	VP_Lengthl	VP_Length = 0x0001 ~ 0x8000

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

\*2. If usr.bin read invalid, VP may not be update correctly.

**4.4.5 BP1\_write ( 0x4B )**

Seq	Cmd-code / Par-data	Descriptions
1	0x4B	Write raw bit-map data to the VP_BP1
2	Addr3(MSB)	VP_BP1 Address = 0x040000 ~ 0x05FFFF
3	Addr2	
4	Addr1	
5	Addr0(LSB)	
6	Length3(MSB)	
7	Length2	
8	Length1	
9	Length0(LSB)	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

\*2. After the above command issued, it follow with the raw data byte with out communication packet structure.

\*3. over all command flow

HOST	Flow	module
BP1_write Command (in communication packet structure)	→	Instruct to wait for data...
Raw 1bpp image data (without communication packet structure)	→	Store the data into VP_BP1
	←	Response code ":->" in ASCII (without communication packet structure)

**4.4.6 BP1\_write\_compress ( 0x4C )**

Seq	Cmd-code / Par-data	Descriptions
1	0x4C	Write compressed bit-map data to the VP_BP1
2	Addr3(MSB)	VP_BP1 Address = 0x040000 ~ 0x05FFFF
3	Addr2	
4	Addr1	
5	Addr0(LSB)	
6	Length3(MSB)	
7	Length2	
8	Length1	
9	Length0(LSB)	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

\*2. After the above command issued, it follow with the compressed data byte with out communication packet structure.

\*3. over all command flow

HOST	Flow	module
BP1_write Command (in communication packet structure)	→	Instruct to wait for data...
compressed 1bpp image data (without communication packet structure)	→	Store the data into VP_BP1
	←	Response code ":->" in ASCII (without communication packet structure)

**4.4.7 G16\_write ( 0x4D )**

Seq	Cmd-code / Par-data	Descriptions
1	0x4D	Write graph values to the VP_G16 array
2	Addr1_H	VP_G16 Address = 0x060000 ~ 0x07FFFF
3	Addr1l	
4	Addr2h	
5	Addr2l	
6	Sizeh	
7	Sizel	
8	Data(MSB)	16 bit data array (no. of byte = 2x array-size)
9	Data(LSB)	
10	Data(MSB)	
11	Data(LSB)	
:	:	
:	:	
:	:	
:	:	
:	:	
:	:	
:	:	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

\*2. Array-size = no. of 16bit values

\*3. Array-size suggest to be same at the size value defined in TML editor

**4.4.8 G16\_write\_rotate ( 0x4E )**

Seq	Cmd-code / Par-data	Descriptions
1	0x4E	Write graph values to the last position of VP_G16 array with rotation effect
2	Addr1_H	VP_G16 Address = 0x060000 ~ 0x07FFFF
3	Addr1_L	
4	Addr2_H	
5	Addr2_L	
6	Size_H	
7	Size_L	
8	Data(MSB)	16 bit data value to be add to the end-of-array
9	Data(LSB)	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

\*2. Array-size = no. of 16bit values

\*3. Array-size suggest to be same at the value defined in TML editor

**4.4.9 Reg\_Write ( 0x3B )**

seq	Cmd-code / Par-data	Descriptions
1	0x3B	System Register Write Command
2	Addr3(MSB)	Address = 0xFFFF00 : Address = 0FFFFFFF
3	Addr2	
4	Addr1	
5	Addr0(LSB)	
6	Data	the value to write

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**4.4.10 Reg\_Read ( 0x3C )**

seq	Cmd-code / Par-data	Descriptions
1	0x3C	System Register Read Command
2	Addr3(MSB)	Address = 0xFFFF00 : Address = 0FFFFFFF
3	Addr2	
4	Addr1	
5	Addr0(LSB)	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**Response code:**

Seq.	Content	Byte in Hex	Descriptions
1 <sup>st</sup>	Header	0xAA	Communication packet header
2 <sup>nd</sup>	Command	0x3C	Command executed
3 <sup>rd</sup>	Address	Addr3(MSB)	Address = 0xFFFF00 : Address = 0FFFFFFF
4 <sup>th</sup>		Addr2	
5 <sup>th</sup>		Addr1	
6 <sup>th</sup>		Addr0(LSB)	
7 <sup>th</sup>	Data	Data(1Byte)	the value of the register
8 <sup>th</sup>	Tail	0xCC	Communication packet tail
9 <sup>th</sup>		0x33	
10 <sup>th</sup>		0xC3	
11 <sup>th</sup>		0x3C	

Note.

\*1. The Response code with communication packet format (see Communication Packet Structure Section for details)

\*2. When Timer reach the 0x00000000 or 0x7FFFFFFF, a notification will be provided a 0x77 response code with the corresponding Timer Address and Value.(See touch\_response(0x77)for details)

**4.4.11 STR\_write ( 0x42 )**

Seq	Cmd-code / Par-data	Descriptions
1	0x42	Write string to VP_STR
2	Addr3(MSB)	the VP_STR Address = 0x00000 ~ 0x01FFFF (each VP_STR = 128 bytes) (address value must be divisible by 128)
3	Addr2	
4	Addr1	
5	Addr0(LSB)	
6	data	String to write
:	:	Total no. of byte in string ≤ 128
:	:	
:	'\0'	'\0'(0x00): string end mark

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**4.4.12 STR\_read ( 0x43 )**

Seq	Cmd-code / Par-data	Descriptions
1	0x43	Read string from VP_STR
2	Addr3(MSB)	the VP_STR Address = 0x00000 ~ 0x01FFFF (each VP_STR = 128 bytes) (address value must be divisible by 128)
3	Addr2	
4	Addr1	
5	Addr0(LSB)	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**Response code:**

Seq.	Content	Byte in Hex	Descriptions
1 <sup>st</sup>	Header	0xAA	Communication packet header
2 <sup>nd</sup>	Command	0x43	Command executed
3 <sup>rd</sup>	String data	data	String code
:		:	
:		:	
:		:	
:	\0	0x00	"\0"(0x00): string end mark
:	Tail	0xCC	Communication packet tail
:		0x33	
:		0xC3	
:		0x3C	

Note.

\*1. The Response code with communication packet format (see Communication Packet Structure Section for details)

**4.4.13 STR\_fill ( 0x46 )**

Seq	Cmd-code / Par-data	Descriptions
1	0x46	Write string to VP_STR
2	Addr3(MSB)	the VP_STR Address = 0x00000 ~ 0x01FFFF (each VP_STR = 128 bytes) (address value must be divisible by 128)
3	Addr2	
4	Addr1	
5	Addr0(LSB)	
6	Lengthh	the number of VP_STR (including the start address) to be filled
7	Lengthl	Length = 1 ~ 1024
8	data	String to write
:	:	Total no. of byte in string ≤ 128
:	:	
:	:	
:	'\0'	'\0'(0x00): string end mark

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**4.4.14 N16\_write ( 0x3D )**

Seq	Cmd-code / Par-data	Descriptions
1	0x3D	Write 16bit number to VP_N16
2	Addr3(MSB)	VP_N16 Address = 0x080000 ~ 0x08FFFF (each VP_N16 = 2 byte) (address value must be divisible by 2)
3	Addr2	
4	Addr1	
5	Addr0(LSB)	
6	High Byte	The 16 bit value to write
7	Low Byte	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**4.4.15 N16\_read ( 0x3E )**

Seq	Cmd-code / Par-data	Descriptions
1	0x3E	Read 16bit number from VP_N16
2	Addr3(MSB)	VP_N16 Address = 0x080000 ~ 0x08FFFF (each VP_N16 = 2 byte) (address value must be divisible by 2)
3	Addr2	
4	Addr1	
5	Addr0(LSB)	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**Response code:**

Seq.	Content	Byte in Hex	Descriptions
1 <sup>st</sup>	Header	0xAA	Communication packet header
2 <sup>nd</sup>	Command	0x3E	Command executed
3 <sup>rd</sup>	N16 value	Data1(MSB)	16 bit value
4 <sup>th</sup>		Data0(LSB)	
5 <sup>th</sup>	Tail	0xCC	Communication packet tail
6 <sup>th</sup>		0x33	
7 <sup>th</sup>		0xC3	
8 <sup>th</sup>		0x3C	

Note.

\*1. The Response code with communication packet format (see Communication Packet Structure Section for details)

**4.4.16 N16\_fill ( 0x3F )**

Seq	Cmd-code / Par-data	Descriptions
1	0x3F	Fill 16bit number to the VP_N16
2	Addr3(MSB)	VP_N16 Address = 0x080000 ~ 0x08FFFF (each VP_N16 = 2 byte) (address value must be divisible by 2)
3	Addr2	
4	Addr1	
5	Addr0(LSB)	
6	Lengthh	the number of VP_N16 (including the start address) to be filled Length = 1 ~ 32768
7	Lengthl	
8	High Byte	the 16 bit value to fill
9	Low Byte	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**4.4.17 N32\_write ( 0x44 )**

Seq	Cmd-code / Par-data	Descriptions
1	0x44	Write 32bit number to VP_N32
2	Addr3(MSB)	VP_N32 Address = 0x020000 ~ 0x02FFFF (each VP_N32 = 4 byte) (address value must be divisible by 4)
3	Addr2	
4	Addr1	
5	Addr0(LSB)	
6	Data3(MSB)	the 32 bit no. value write.
7	Data2	
8	Data1	
9	Data0(LSB)	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**4.4.18 N32\_read ( 0x45 )**

Seq	Cmd-code / Par-data	Descriptions
1	0x45	Read 32bit number from VP_N32
2	Addr3(MSB)	VP_N32 Address = 0x020000 ~ 0x02FFFF (each VP_N32 = 4 byte) (address value must be divisible by 4)
3	Addr2	
4	Addr1	
5	Addr0(LSB)	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**Response code:**

Seq.	Content	Byte in Hex	Descriptions
1 <sup>st</sup>	Header	0xAA	Communication packet header
2 <sup>nd</sup>	Command	0x45	Command executed
3 <sup>rd</sup>	N32 value	Data3(MSB)	32 bit value
4 <sup>th</sup>		Data2	
5 <sup>th</sup>		Data1	
6 <sup>th</sup>		Data0(LSB)	
7 <sup>th</sup>	Tail	0xCC	Communication packet tail
8 <sup>th</sup>		0x33	
9 <sup>th</sup>		0xC3	
10 <sup>th</sup>		0x3C	

Note.

\*1. The Response code with communication packet format (see Communication Packet Structure Section for details)

**4.4.19 N32\_fill ( 0x47 )**

Seq	Cmd-code / Par-data	Descriptions
1	0x47	Fill 32bit number to the VP_N32
2	Addr3(MSB)	VP_N32 Address = 0x020000 ~ 0x02FFFF (each VP_N32 = 4 byte) (address value must be divisible by 4)
3	Addr2	
4	Addr1	
5	Addr0(LSB)	
6	Lengthh	the number of VP_N32 (including the start address) to be filled Length = 1 ~ 16384
7	Lengthl	
8	Data3(MSB)	the 32 bit no. value to fill
9	Data2	
10	Data1	
11	Data0(LSB)	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**4.4.20 N64\_write ( 0x48 )**

Seq	Cmd-code / Par-data	Descriptions
1	0x48	Write 64bit number to VP_N64
2	Addr3(MSB)	VP_N64 Address= 0x030000 ~ 0x03FFFF (each VP_N64 = 8 byte) (address value must be divisible by 8)
3	Addr2	
4	Addr1	
5	Addr0(LSB)	
6	Data7(MSB)	the 64bit no. value write.
7	Data6	
⋮	⋮	
⋮	⋮	
12	Data1	
13	Data0(LSB)	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)



**4.4.21 N64\_read ( 0x49 )**

Seq	Cmd-code / Par-data	Descriptions
1	0x49	Read 64bit number from VP_N64
2	Addr3(MSB)	VP_N64 Address= 0x030000 ~ 0x03FFFF (each VP_N64 = 8 byte) (address value must be divisible by 8)
3	Addr2	
4	Addr1	
5	Addr0(LSB)	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

**Response code:**

Seq.	Content	Byte in Hex	Descriptions
1 <sup>st</sup>	Header	0xAA	Communication packet header
2 <sup>nd</sup>	Command	0x49	Command executed
3 <sup>rd</sup>	N64 value	Data7(MSB)	64 bit value
4 <sup>th</sup>		Data6	
:		:	
:		:	
9 <sup>th</sup>		Data1	
10 <sup>th</sup>		Data0(LSB)	
11 <sup>th</sup>		Tail	
12 <sup>th</sup>	0x33		
13 <sup>th</sup>	0xC3		
14 <sup>th</sup>	0x3C		

Note.

\*1. The Response code with communication packet format (see Communication Packet Structure Section for details)

**4.4.22 N64\_fill ( 0x4A )**

Seq	Cmd-code / Par-data	Descriptions
1	0x4A	Fill 64bit number to the VP_N64
2	Addr3(MSB)	VP_N64 Address= 0x030000 ~ 0x03FFFF (each VP_N64 = 8 byte) (address value must be divisible by 8)
3	Addr2	
4	Addr1	
5	Addr0(LSB)	
6	Lengthh	the number of VP_N64 (including the start address) to be filled Length = 1 ~ 8192
7	Lengthl	
8	Data7(MSB)	the 64 bit no. value to fill
9	Data6	
:	:	
:	:	
14	Data1	
15	Data0(LSB)	

Note.

\*1. Command should be transferred in communication packet structure (see Communication Packet Structure Section for details)

## Appendix 1: CRC Calculate

```

uint16_t const CRC16[256]={
/* 16: 8005 reflected */

0x0000,0xc0c1,0xc181,0x0140,0xc301,0x03c0,0x0280,0xc241,0xc601,0x06c0,0x0780,0xc741,0x0500,0xc5c1,0xc
481,0x0440,

0xcc01,0x0cc0,0x0d80,0xcd41,0x0f00,0xcfc1,0xce81,0x0e40,0x0a00,0xcac1,0xcb81,0x0b40,0xc901,0x09c0,0x0
880,0xc841,

0xd801,0x18c0,0x1980,0xd941,0x1b00,0xdbc1,0xda81,0x1a40,0x1e00,0xdec1,0xdf81,0x1f40,0xdd01,0x1dc0,0x1
c80,0xdc41,

0x1400,0xd4c1,0xd581,0x1540,0xd701,0x17c0,0x1680,0xd641,0xd201,0x12c0,0x1380,0xd341,0x1100,0xd1c1,0xd
081,0x1040,

0xf001,0x30c0,0x3180,0xf141,0x3300,0xf3c1,0xf281,0x3240,0x3600,0xf6c1,0xf781,0x3740,0xf501,0x35c0,0x3
480,0xf441,

0x3c00,0xfc01,0xfd81,0x3d40,0xff01,0x3fc0,0x3e80,0xfe41,0xfa01,0x3ac0,0x3b80,0xfb41,0x3900,0xf9c1,0xf
881,0x3840,

0x2800,0xe8c1,0xe981,0x2940,0xeb01,0x2bc0,0x2a80,0xea41,0xee01,0x2ec0,0x2f80,0xef41,0x2d00,0xedc1,0xe
c81,0x2c40,

0xe401,0x24c0,0x2580,0xe541,0x2700,0xe7c1,0xe681,0x2640,0x2200,0xe2c1,0xe381,0x2340,0xe101,0x21c0,0x2
080,0xe041,

0xa001,0x60c0,0x6180,0xa141,0x6300,0xa3c1,0xa281,0x6240,0x6600,0xa6c1,0xa781,0x6740,0xa501,0x65c0,0x6
480,0xa441,

0x6c00,0xacc1,0xad81,0x6d40,0xaf01,0x6fc0,0x6e80,0xae41,0xaa01,0x6ac0,0x6b80,0xab41,0x6900,0xa9c1,0xa
881,0x6840,

0x7800,0xb8c1,0xb981,0x7940,0xbb01,0x7bc0,0x7a80,0xba41,0xbe01,0x7ec0,0x7f80,0xbf41,0x7d00,0xbdc1,0xb
c81,0x7c40,

0xb401,0x74c0,0x7580,0xb541,0x7700,0xb7c1,0xb681,0x7640,0x7200,0xb2c1,0xb381,0x7340,0xb101,0x71c0,0x7
080,0xb041,

0x5000,0x90c1,0x9181,0x5140,0x9301,0x53c0,0x5280,0x9241,0x9601,0x56c0,0x5780,0x9741,0x5500,0x95c1,0x9
481,0x5440,

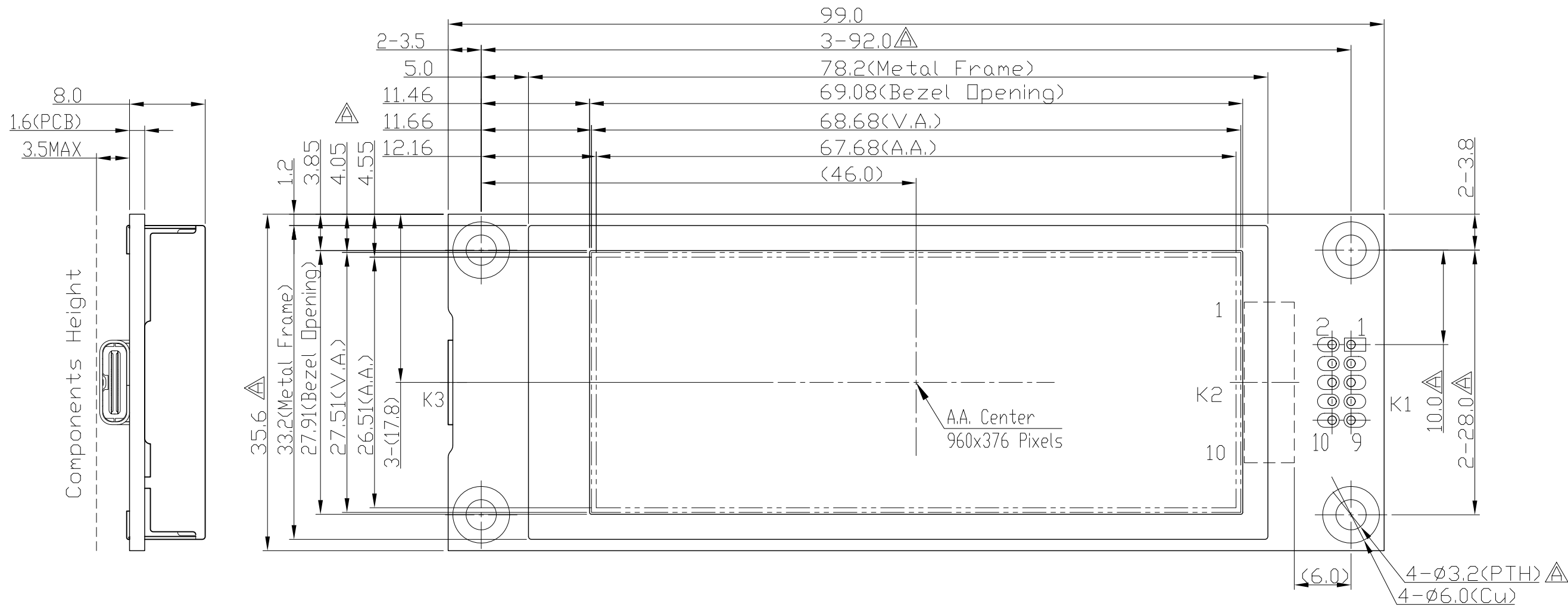
0x9c01,0x5cc0,0x5d80,0x9d41,0x5f00,0x9fc1,0x9e81,0x5e40,0x5a00,0x9ac1,0x9b81,0x5b40,0x9901,0x99c0,0x5
880,0x9841,

0x8801,0x48c0,0x4980,0x8941,0x4b00,0x8bc1,0x8a81,0x4a40,0x4e00,0x8ec1,0x8f81,0x4f40,0x8d01,0x4dc0,0x4
c80,0x8c41,

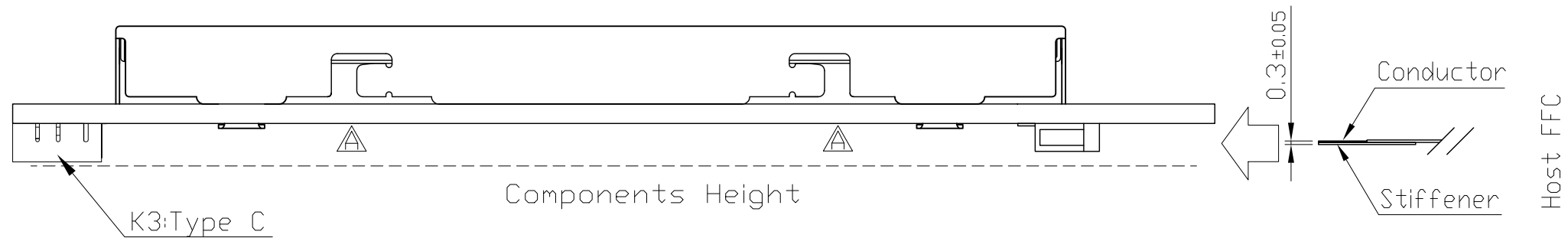
0x4400,0x84c1,0x8581,0x4540,0x8701,0x47c0,0x4680,0x8641,0x8201,0x42c0,0x4380,0x8341,0x4100,0x81c1,0x8
081,0x4040,
};

static __inline uint16_t rshiftu16(uint16_t value, int nb)
{
return (uint16_t)((value >> nb) & ~((( uint16_t) 0x8000) >> (nb-1)));
}
uint16_t crc16_calc(unsigned char *q, int len)
{
uint16_t crc = 0xffff;
while (len-- > 0)
    crc=(rshiftu16(crc,8) ^ CRC16[(crc ^ *q++) & 0xff]);
return crc;
}

```

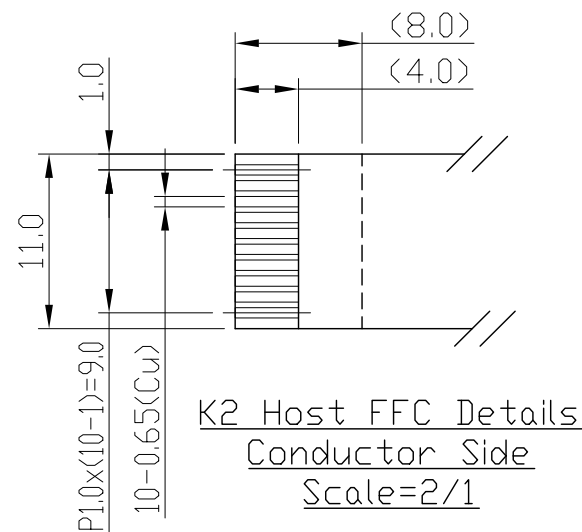
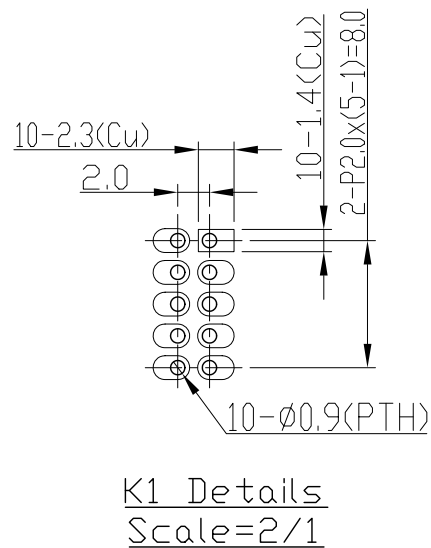


K1/K2 Terminal	
No.	Pin Name
1	VDD
2	VDD
3	VDD
4	NC
5	RX
6	TX
7	RTS(BUSY)
8	VSS
9	VSS
10	VSS



Note:

- \*1. LCD Display Type : TFT,Transmissive (Full View)
- \*2. Operating Voltage : 5.0V
- \*3. Backlight Color : White LEDs
- \*4. Pixel Arrangement : RGB-STRIPE
- \*5. Color Depth : 65k Colors
- \*6. Signal Interface : RS232C
- \*7. Connector:
  - K1: P2.0,2x5 PCB Pad
  - K2: 10Pin P1.0 FFC Socket Or Equivalent
  - K3: USB Type-C 16Pin
- \*8. Operating Temperature : -20°C~70°C
- \*9. Storage Temperature : -30°C~80°C



C		
B		
A	Revise PCB & Metal Frame	Luo Lin 2024-06-20
Rev/Note		Date
Dwg Title HMT029BTA Outline Dwg		
Dwg No. MK-008471a-1-1		Date 2024-05-22
Scale 2/1	Tol. ±0.3	Unit mm
Paper Size A3		Drawn Luo Lin
Approved	Checked	

**TOPWAY**