

USER MANUAL

uSign 300

Technical Reference Manual



80109502-001-B
07-16-2012

Software & Documentation License Agreement

CAREFULLY READ ALL THE TERMS, CONDITIONS, AND RESTRICTIONS OF THIS LICENSE AGREEMENT BEFORE USING OR INSTALLING THE SOFTWARE. YOUR USE OR INSTALLATION OF THE SOFTWARE PRESUMES YOUR AGREEMENT WITH AND ACCEPTANCE OF THE TERMS, CONDITIONS, AND RESTRICTIONS CONTAINED IN THIS AGREEMENT. IF YOU DO NOT AGREE WITH THESE TERMS, CONDITIONS, AND RESTRICTIONS, PROMPTLY RETURN THE SOFTWARE AND RELATED DOCUMENTATION TO – ID TECH Support, 10721 Walker Street, Cypress, CA 90630.

TERMS, CONDITIONS AND RESTRICTIONS

ID TECH, Incorporated (the “Licensor”) owns and has the right to distribute the described software and documentation, collectively referred to as the “Software”.

LICENSE: Licensor grants you (the “Licensee”) the right to use the Software in conjunction with ID TECH products.

LICENSEE MAY NOT COPY, MODIFY OR TRANSFER THE SOFTWARE and DOCUMENTATION IN WHOLE OR IN PART EXCEPT AS EXPRESSLY PROVIDED IN THIS AGREEMENT. Licensee may not decompile, disassemble, or in any other manner attempt to reverse engineer the Software. Licensee shall not tamper with, bypass, or alter any security features of the software or attempt to do so.

TRANSFER: Licensee may not transfer the Software & Documentation or license the Software to another party without prior written authorization of the Licensor. If Licensee transfers the Software without authorization, all rights granted under this Agreement are automatically terminated.

COPYRIGHT: The Software is copyrighted. Licensee may not copy the Software except to archive the Software or to load the Software for execution purposes. All other copies of the Software are in violation of this Agreement.

TERM: This Agreement is in effect as long as Licensee continues the use of the Software. The Licensor also reserves the right to terminate this Agreement if Licensee fails to comply with any of the terms, conditions, or restrictions contained herein. Should Licensor terminate this Agreement due to Licensee’s failure to comply, Licensee agrees to return the Software to Licensor. Receipt of returned Software by the Licensor shall mark the termination.

LIMITED WARRANTY: Licensor warrants to the Licensee that the disk(s) or other media on which the Software is recorded to be free from defects in material or workmanship under normal use. THE SOFTWARE IS PROVIDED AS IS WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Because of the diversity of conditions and PC hardware under which the Software may be used, Licensor does not warrant that the Software will meet Licensee specifications or that the operation of the Software will be uninterrupted or free of errors.

IN NO EVENT WILL LICENSOR BE LIABLE FOR ANY DAMAGES, INCLUDING ANY LOST PROFITS, LOST SAVINGS OR OTHER INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE SOFTWARE. Licensee's sole remedy in the event of a defect in material or workmanship is expressly limited to replacement of the Software disk(s) if applicable.

GOVERNING LAW: If any provision of this Agreement is found to be unlawful, void or unenforceable, that provision shall be removed from consideration under this Agreement and will not affect the enforceability of any of the remaining provisions. This Agreement shall be governed by the laws of the State of California and shall insure to the benefit of ID TECH, Incorporated, its successors, or assigns.

ACKNOWLEDGMENT: LICENSEE ACKNOWLEDGES THAT HE HAS READ THIS AGREEMENT, UNDERSTANDS ALL OF ITS TERMS, CONDITIONS, AND RESTRICTIONS AND AGREES TO BE BOUND BY THEM. LICENSEE ALSO AGREES THAT THIS AGREEMENT SUPERSEDES ANY AND ALL, VERBAL AND WRITTEN, COMMUNICATIONS BETWEEN LICENSOR AND LICENSEE OR THEIR ASSIGNS RELATING TO THE SUBJECT MATTER OF THIS AGREEMENT.

QUESTIONS REGARDING THIS AGREEMENT SHOULD BE ADDRESSED IN WRITING TO ID TECH, INCORPORATED, ATTENTION: CUSTOMER SUPPORT, AT THE ABOVE ADDRESS OR E-MAILED TO: support@idtechproducts.com

Information Provided

The information contained herein is provided to the user as a convenience. While every effort has been made to ensure accuracy, ID TECH is not responsible for damages that might occur because of errors or omissions, including any loss of profit or other commercial damage, nor for any infringements or patents or other rights of third parties that may result from its use. The specifications & information described herein were current at the time of publication, but are subject to change at any time without prior notice.

Proprietary & Trademark Statements

This document contains proprietary information of ID TECH. Its receipt or possession does not convey any rights to reproduce or disclose its contents or to manufacture, use or sell anything it may describe. Reproduction, disclosure, or use without specific written authorization from ID TECH is strictly forbidden.

Copyright 2012, International Technologies & Systems Corporation. All rights reserved. ID TECH is a registered trademark of International Technologies & Systems Corporation. uSign 300 and Value through Innovation are trademarks of International Technologies & Systems Corporation.

ID TECH
10721 Walker Street
Cypress, CA 90630
(714) 761-6368
www.idtechproducts.com

Revision History

Revision	Date	Description of Changes	By
A	02/23/2012	Initial release	JW
B	07/16/2012	4.8 Select Font for Text Display - Added information on parameter definition for font Height, Width and Charset. 5.3 Create Region - Added information on DataLen and Data.	JW

Table of Contents

1.0	uSign 300 Command Description.....	8
2.0	Generic Commands	8
2.1	Get Firmware Version.....	8
2.2	Get Serial Number	9
2.3	Set Serial Number.....	9
2.4	Get Model Number.....	9
3.0	LED Commands	9
3.1	LED Control	9
4.0	Display Commands	10
4.1	Set Pen Width and Color.....	10
4.2	Draw Line.....	11
4.3	Draw Rectangle	11
4.4	Draw Arc	11
4.5	Set Brush Color	12
4.6	Fill Rectangle.....	12
4.7	Fill Arc.....	12
4.8	Select Font for Text Display	12
4.9	Set Text Color.....	13
4.10	Set Background Color	13
4.11	Select Background Mode for Display.....	13
4.12	Draw String in Rectangle.....	14
4.13	Draw String.....	14
4.14	Get Picture on the LCD	14
4.15	Show Picture on the LCD.....	15
4.16	Store Picture in the Device	15
4.17	Show Stored Picture on the LCD.....	16
4.18	Retrieve Stored Picture in the Device.....	16
5.0	Signature Commands.....	16
5.1	Calibrate Device	16
5.2	Set Clip Area.....	16
5.3	Create Region	17
5.4	Start Capture	19
5.5	Continue Capture.....	19
5.6	Pause Capture	19
5.7	Get Script Point Count.....	20
5.8	Clear Signature	20
5.9	Exit Capture.....	20
5.10	Get SIG Format Signature	20
5.11	Get CMP Format Signature	20
5.12	Get RAW Format Signature.....	20
5.13	Get BMP Format Signature	21
5.14	Set SIG Format Signature.....	21

uSign 300 Technical Reference Manual

5.15	Set CMP Format Signature	21
5.16	Set RAW Format Signature	21
5.17	Set BMP Format Signature	22
6.0	Error Code List	23

1.0 uSign 300 Command Description

All commands in this section have the following command and response structures:

Host to Device Command Protocol

<STX><LenL><LenH><CommandData><Lrc1><Lrc2><ETX>

STX: 0x02. 1 byte.

LenL+LenH: sizeof(CommandData). If Length of CommandData is less than 0x8000, LenL + LenH occupies 2 bytes, otherwise it occupies 3 bytes.

CommandData: main command string. Several bytes needed.

Lrc1: Exclusive or of CommandData. 1 byte.

Lrc2: Sum of CommandData. 1 byte.

ETX: 0x03. 1 byte.

Device to Host Command Protocol

<STX><LenL><LenH><ResponseStatus><ResponseData><Lrc1><Lrc2><ETX>

STX: 0x02. 1 byte.

LenL+LenH: sizeof(ResponseStatus+ResponseData). If Length of ResponseData is less than 0x8000, LenL + LenH occupy 2 bytes, otherwise it occupies 3 bytes.

ResponseStatus: Status of the response. 1 byte.

NAK: 0x15

ACK: 0x06

ResponseData: main response string. Several bytes needed.

Lrc1: Exclusive or of ResponseData. 1 byte.

Lrc2: Sum of ResponseData. 1 byte.

ETX: 0x03. 1 byte.

The <CommandData> and <ResponseData> is defined for each command.

All length described below is little endian.

2.0 Generic Commands

2.1 *Get Firmware Version*

COMMAND: <0x78><0x46><0x01>

Get uSign 300 firmware version.

PARAMETERS:

<0x78><0x46><0x01> is the command head.

RETURN:

IDTECH-uSign300 Vx.yz

Vx.yz is defined as the firmware version, such as V1.00 or V1.01.

2.2 *Get Serial Number*

COMMAND: <0x78><0x46><0x02>

Get uSign 300 serial number.

PARAMETERS:

<0x78><0x46><0x02> is the command head.

RETURN:

Serial number

2.3 *Set Serial Number*

COMMAND: <0x78><0x46><0x03><Serial number>

Set serial number.

PARAMETERS:

<0x78><0x46><0x03> is the command head.

<Serial number> The length must be eight.

RETURN:

<ACK>

2.4 *Get Model Number*

COMMAND: <0x78><0x46><0x20>

Get model of communication. IDUB-012500 is for RS232 and IDUB-015500 is for USB HID.

PARAMETERS:

<0x78><0x46><0x20> is the command head.

RETURN:

<ACK> IDUB-012500/ IDUB-015500

3.0 **LED Commands**

3.1 *LED Control*

COMMAND: <0x76><0x46><0x01><LED Control Code>

Control the Red led (R) and Green led (G).

PARAMETERS:

<0x76><0x46><0x01> is the command head.

< LED Control Code > is 1-byte length. The definition is:

MSB

LSB

B7 B6 B5 B4 B3 B2 B1 B0

Every Led use two bits:

Left led: Green& Red B6 B5 B4

Right led: Green &Red B2 B1 B0

Where

The B7 is reserved.

The B6 is switch of Left led.

B6=1: Left led is ON, Then B4 and B5 are selected.

B6=0: Left led is OFF, Then B4 and B5 are ignored.

The b5 is switch of Left Green led or Left Red led.

B5=1: Left Red led is selected.

B5=0: Left Green led is selected.

The b4 is switch of left led flashing.

B4=1: Left Green/Red Led flash. The B6 must be set when you want to use this function .

B4=0: Left Green/Red Led steady.

The B3 is reserved.

The B2 is switch of Right led.

B2=1: Right led is ON, Then B1 and B0 are selected.

B2=0: Right led is OFF, Then B1 and B0 are ignored.

The B1 is switch of Right Green led or Right Red led.

B1=1: Right Red led is selected.

B1=0: Right Green led is selected.

The B0 is switch of Right led flashing.

B0=1: Right Green/Red led flash, The b2 must be set when you want to use this function.

B0=0: Right Green/Red led steady,

RETURN:

<ACK>

4.0 Display Commands

4.1 *Set Pen Width and Color*

COMMAND: <0x8A><0x46><0x10><WIDTH><COLOR>

Se the pen color used to draw line on the LCD.

PARAMETERS:

<0x8A><0x46><0x10> is the command head.

<WIDTH> is the pen's width, 4 bytes long. WIDTH must be 0x01.

<COLOR> is the pen's color, defined as <RED><GREEN><BLUE>. Each is 1 byte.

RETURN:

<ACK>

4.2 Draw Line

COMMAND: <0x8A><0x46><0x11><X0><Y0><X1><Y1>

Draw line from point <X0><Y0> to <X1><Y1> using the pen.

PARAMETERS:

<0x8A><0x46><0x11> is the command head.

<X0> is X-coordinate of start point, 2 bytes.

<Y0> is Y-coordinate of start point, 2 bytes.

<X1> is X-coordinate of end point, 2 bytes.

<Y1> is Y-coordinate of end point, 2 bytes.

RETURN:

<ACK>

4.3 Draw Rectangle

COMMAND: <0x8A><0x46><0x12><X0><Y0><X1><Y1>

Draw rectangle defined by top left point <X0><Y0> and bottom right point <X1><Y1> using the pen.

PARAMETERS:

<0x8A><0x46><0x12> is the command head.

<X0> is X-coordinate of top left point, 2 bytes.

<Y0> is Y-coordinate of top left point, 2 bytes.

<X1> is X-coordinate of bottom right point, 2 bytes.

<Y1> is Y-coordinate of bottom right point, 2 bytes.

<X0 ><Y0 ><X1 ><Y1 > is The LCD coordinate, 0<="X0" < "X1"< 320, 0<="Y0" < "Y1" < 240.

RETURN:

<ACK>

4.4 Draw Arc

COMMAND: <0x8A><0x46><0x13> <X><Y><Radius>< StartAngle >< SweepAngle >

Draw arc defined by center point, radius, start angle and sweep angle using pen.

PARAMETERS:

<0x8A><0x46><0x13> is the command head.

<X> specifies the x-coordinate of the center of the related circle. 2 bytes.

<Y> specifies the y-coordinate of the center of the related circle. 2 bytes.

<X><Y> is the LCD coordinate, 0< "X" < 320, 0< "Y" < 240.

<Radius> specifies the radius of the related circle. 2 bytes.

<StartAngle> specifies the starting angle in degrees relative to the x-axis. Unit is 0.1, 2 bytes.

<SweepAngle> specifies the sweep angle in degrees relative to the starting angle. Unit is 0.1, 2 bytes.

RETURN:

<ACK>

4.5 *Set Brush Color*

COMMAND: <0x8A><0x46><0x20><COLOR>

Set the brush's color used to fill region on the LCD.

PARAMETERS:

<0x8A><0x46><0x20> is the command head.

<COLOR> is the brush's color, defined as <RED><GREEN><BLUE>. Each is 1 byte.

RETURN:

<ACK>

4.6 *Fill Rectangle*

COMMAND: <0x8A><0x46><0x22><X0><Y0><X1><Y1>

Fill rectangle define by top left point <X0><Y0> and bottom right point <X1><Y1> using the brush color.

PARAMETERS:

<0x8A><0x46><0x22> is the command head.

<X0> is X-coordinate of top left point, 2 bytes.

<Y0> is Y-coordinate of top left point, 2 bytes.

<X1> is X-coordinate of bottom right point, 2 bytes.

<Y1> is Y-coordinate of bottom right point, 2 bytes.

<X0 ><Y0 ><X1 ><Y1 > is the LCD coordinate, 0<="X0" < "X1"< 320, 0<="Y0" < "Y1" < 240.

RETURN:

<ACK>

4.7 *Fill Arc*

COMMAND: <0x8A><0x46><0x23> <X><Y><Radius>< StartAngle >< SweepAngle >

Draw arc defined by center point, radius, start angle and sweep angle use brush color.

PARAMETERS:

<0x8A><0x46><0x23> is the command head.

<X> specifies the x-coordinate of the center of the related circle. 2 bytes.

<Y> specifies the y-coordinate of the center of the related circle. 2 bytes.

<X><Y> is the LCD coordinate, 0< "X" < 320, 0< "Y" < 240.

<Radius> specifies the radius of the related circle. 2 bytes.

<StartAngle> specifies the starting angle in degrees relative to the x-axis. Unit is 0.1. 2 bytes.

<SweepAngle> specifies the sweep angle in degrees relative to the starting angle. Unit is 0.1. 2 bytes.

RETURN:

<ACK>

4.8 *Select Font for Text Display*

COMMAND: <0x8A><0x46><0x40><Height><Width><Weight><Italic><Underline><CharSet>

Select the font for text display on the LCD.

PARAMETERS:

<0x8A><0x46><0x40> is the command head.

<Height> specifies the height of a char. 1 byte. This value defines the vertical gap between characters. Height must be greater than or equal to the dot matrix height of the Charset specified.

<Width> specifies the width of a char. 1 byte. This value defines the horizontal gap between characters. Width must be greater than or equal to the dot matrix width of the Charset specified.

<Weight> specifies the weight of the char. 1 byte. Must be 0x00.

<Italic> specifies the italic of the char. 1 byte. Must be 0x00 means no Italic.

<Underline> specifies the underline or not of the char. 1 byte. Must be 0x00 means no underline.

<CharSet> specifies the char set. 1 byte. The valid size is 1 – 6, and the corresponding character dot matrix (Height x Width) is: 4x8, 8x16, 12x24, 16x32, 24x48, 32x64.

For example, if the Charset is 0x02, characters dot height is 16, characters dot width is 8, Height must be >=16 and Width >=8.

RETURN:

<ACK>

4.9 Set Text Color

COMMAND: <0x8A><0x46><0x41><COLOR>

Set the text color.

PARAMETERS:

<0x8A><0x46><0x41> is the command head.

<COLOR> is the text color, defined as <RED><GREEN><BLUE>. Each is 1 byte long.

RETURN:

<ACK>

4.10 Set Background Color

COMMAND: <0x8A><0x46><0x42><COLOR>

Set the background color when display text on the LCD.

PARAMETERS:

<0x8A><0x46><0x42> is the command head.

<COLOR> is the text background color, defined as <RED><GREEN><BLUE>. Each is 1 byte long.

RETURN:

<ACK>

4.11 Select Background Mode for Display

COMMAND: <0x8A><0x46><0x43><MODE>

Select the background mode for text display on the LCD.

PARAMETERS:

<0x8A><0x46><0x43> is the command head.

<MODE> specifies background mode. 1byte. 0x00 means OPAQUE and others means

TRANSPARENT.

RETURN:

<ACK>

4.12 Draw String in Rectangle

COMMAND: <0x8A><0x46><0x4E><X0><Y0><X1><Y1><Length><String>

Draw string using the selected font and colors on the LCD. The string will be displayed in the specified rectangle, from the top left of the rectangle to the right bottom of the rectangle.

PARAMETERS:

<0x8A><0x46><0x4E> is the command head.

<X0> specifies the x-coordinate of the top left point. 2 bytes.

<Y0> specifies the y-coordinate of the top left point. 2 bytes.

<X1> specifies the x-coordinate of the right bottom point. 2 bytes.

<Y1> specifies the y-coordinate of the right bottom point. 2 bytes.

<X0 ><Y0 ><X1 ><Y1 > is The LCD coordinate, 0<="X0" < "X1"< 320, 0<="Y0" < "Y1" < 240.

<Length> specifies the length of the string in characters. 2 bytes.

<String> specifies the string to be displayed.

RETURN:

<ACK>

4.13 Draw String

COMMAND: <0x8A><0x46><0x4F><X><Y><Length><String>

Draw string using the selected font and colors on the LCD.

PARAMETERS:

<0x8A><0x46><0x4F> is the command head.

<X> specifies the x-coordinate of the start point. 2 bytes.

<Y> specifies the y-coordinate of the start point. 2 bytes.

<X><Y> is the LCD coordinate, 0<="X" < 320, 0<="Y" < 240.

<Length> specifies the length of the string in chars. 2 bytes.

<String> specifies the string to be displayed.

RETURN:

<ACK>

4.14 Get Picture on the LCD

COMMAND: <0x8A><0x46><0x60><X0><Y0><X1><Y1>

Get picture on the LCD defined by top left point <X0><Y0> and bottom right point <X1><Y1>.

PARAMETERS:

<0x8A><0x46><0x60> is the command head.

<X0> is X-coordinate of top left point, 2 bytes.

<Y0> is Y-coordinate of top left point, 2 bytes.

<X1> is X-coordinate of bottom right point, 2 bytes.

<Y1> is Y-coordinate of bottom right point, 2 bytes.

<X0 ><Y0 ><X1 ><Y1 > is the LCD coordinate, 0<="X0" < "X1"< 320, 0<="Y0" < "Y1" < 240.

RETURN:

<ACK><Picture data>

Picture is arranged as top left point first and bottom right end. Each point occupies three bytes defined as: RED GREEN BLUE.

4.15 *Show Picture on the LCD*

COMMAND: <0x8A><0x46><0x61><X0><Y0><X1><Y1><Picture Data>

Show picture on the LCD defined by top left point <X0><Y0> and bottom right point <X1><Y1>, the data is RAW format.

PARAMETERS:

<0x8A><0x46><0x61> is the command head.

<X0> is X-coordinate of top left point, 2 bytes.

<Y0> is Y-coordinate of top left point, 2 bytes.

<X1> is X-coordinate of bottom right point, 2 bytes.

<Y1> is Y-coordinate of bottom right point, 2 bytes.

<X0 ><Y0 ><X1 ><Y1 > is the LCD coordinate, 0<="X0" < "X1"< 320, 0<="Y0" < "Y1" < 240.

<Picture Data> is the data of raw format, they can read out from the picture file.

RAW format: It is three bytes to display a point on LCD, all bytes are messages of point. This format has no redundant data in the head and the end of the file.

RETURN:

<ACK>

4.16 *Store Picture in the Device*

COMMAND: <0x8A><0x46><0x70><ID><TYPE><Picture Data>

Store picture in the device.

PARAMETERS:

<0x8A><0x46><0x70> is the command head.

<ID> is the identifier for the picture. 2 bytes.

<TYPE> is the picture's type. 2 bytes. 0x00 means RAW format, 0x01 means 24-bit true color BMP format, 0x02 means JPEG format,

<Picture Data> is the data of jpeg format or 24-bit bmp format or raw format, they can read out from the picture file.

RAW format: It is three bytes to display a point on LCD, all bytes are messages of point. This format has no redundant data in the head and the end of the file.

RETURN:

<ACK>

NOTE: <Picture Data>24-bit bmp format or raw format *must be less than 230k bytes and jpeg format must be less than 32k bytes.*

4.17 Show Stored Picture on the LCD

COMMAND: <0x8A><0x46><0x71><ID><X0><Y0><X1><Y1>

Show stored picture on the LCD defined by top left point <X0><Y0> and bottom right point <X1><Y1>.

PARAMETERS:

<0x8A><0x46><0x71> is the command head.

<ID> is the identifier for the picture. 2 bytes.

<X0> is X-coordinate of top left point, 2 bytes.

<Y0> is Y-coordinate of top left point, 2 bytes.

<X1> is X-coordinate of bottom right point, 2 bytes.

<Y1> is Y-coordinate of bottom right point, 2 bytes.

<X0 ><Y0 ><X1 ><Y1 > is the LCD coordinate, 0<="X0" < "X1"< 320, 0<="Y0" < "Y1" < 240.

RETURN:

<ACK> if picture exists, otherwise <NAK>.

4.18 Retrieve Stored Picture in the Device

COMMAND: <0x8A><0x46><0x72>

Retrieve stored picture in the device.

PARAMETERS:

<0x8A><0x46><0x72> is the command head.

RETURN:

<ACK><Picture Count><ID(2 bytes)> <ID(2 bytes)>

5.0 Signature Commands

5.1 Calibrate Device

COMMAND: <0x7A><0x46><0x01>

Calibrate the device.

PARAMETERS:

<0x7A><0x46><0x01> is the command head.

RETURN:

<ACK>

5.2 Set Clip Area

COMMAND: <0x7A><0x46><0x03><Clip area data(8 bytes)><Show Mode><Line Color>

Set new clip area. The max area is (0,0) – (319,239).

Clip area is a rectangle coded as: left (2 bytes) + top (2 bytes) + right (2 bytes) + bottom (2 bytes)

PARAMETERS:

<0x7A><0x46><0x03> is the command head.

<Clip area data (8 bytes)> is the new clip area data.

<Show Mode> is a bitmap for 4 lines. Bit 0 for left line, Bit 1 for right line, Bit 2 for top line and Bit 3 for bottom line. Value 1b means show this line, 0b means don't show this line.

<Line Color> is the rectangle lines (which surround the clip area) color defined as RED (1 byte) GREEN (1 byte) BLUE (1 byte).

RETURN:

<ACK>

5.3 Create Region

COMMAND:

<0x7A><0x46><0x04><ID><Type><State><X0><Y0><X1><Y1><DataLen><Data>.

Create object like picture, button and text showed on LCD when during signature. The object can be notified when touched.

PARAMETERS:

<0x7A><0x46><0x04> is the command head

<ID> specifies the region's ID. 1 byte.

<Type> specifies the region's type. 1 byte. 0x01 means BUTTON, 0x02 means PICTURE and 0x03 means TEXT.

Note: when <Type> is been set to 0x01/0x03, the <DataLen> must be more than 0x0d. It should include font of text(6 bytes) and color of text(3 bytes) and back mode of LCD(1 byte) and back color of text(3 bytes) and text string. If <Type> is 0x01, the width and height of the button must be larger than text string's.

<State> specifies the region's state. 1 byte.

State:: Bit 0 Exists.

Bit 1 Visible.

Bit 2 Enabled.

Bit 3 Notify.

<X0> is X-coordinate of top left point, 2 bytes.

<Y0> is Y-coordinate of top left point, 2 bytes.

<X1> is X-coordinate of bottom right point, 2 bytes.

<Y1> is Y-coordinate of bottom right point, 2 bytes.

<X0 ><Y0 ><X1 ><Y1 > is the LCD coordinate, 0<="X0" < "X1"< 320, 0<="Y0" < "Y1" < 240.

<DataLen> specifies the <Data> length. 2 bytes.

<Data> specifies the object/s data.

For Button, <Data> is arranged as: Font (Height 1 byte, Width 1 byte, Weight 1 byte, Italic 1 byte, Underline 1 byte, CharSet 1 byte - Please refer to section 4.8 Select Font for Text Display) TextColor(RED GREEN BLUE) ButtonBKDisplayMode (1 byte, 00-Button Color displayed by ButtonBkColor, 01-Button Color displayed by change in the ButtonBkColor), ButtonBkColor(RED GREEN BLUE) String showed on the button.

For Picture, <Data> is the picture data. Picture is arranged as top left point first and bottom right end. Each point occupies three bytes defined as: RED GREEN BLUE.

For Text, <Data> is arranged as: Font(Height 1 byte, Width 1 byte, Weight 1 byte, Italic 1 byte, Underline 1 byte, CharSet 1 byte - Please refer to section 4.8 Select Font for Text

Display) TextColor(RED GREEN BLUE) TextBkMode(1 byte) TextBkColor(RED GREEN BLUE) String.

RETURN:

<ACK>

5.4 *Start Capture*

COMMAND:

<0x7A><0x46><0x10><Capture Mode><Point Interval><Signature Color><Background Color>

Start capture using specified parameters

PARAMETERS:

<0x7A><0x46><0x10> is the command head.

<Capture Mode> specifies the capture mode. 1 byte

0x01: Out signature data using FBP format, pen up is 0x8C and pen down is 0x9C.

0x02: Out signature data (the difference of the current point and previous point) using FBP format, pen up is 0x80 and pen down is 0x90.

0x03: Out signature data using CMP format.

0x04: Out signature data using FBP format, pen up is 0x80 and pen down is 0x90.

<Point Interval> specifies the maximum points' interval during signature. If exceeds, the signature will be cleared. 1 byte. The unit is second. When <Capture Mode> is 0x02/0x03, <Point Interval> is ignored.

<Signature Color> specifies signature's color, defined as RED GREEN BLUE.

<Background Color> specifies signature region's background color, defined as RED GREEN BLUE.

RETURN:

<ACK>

During capture, data will be sent out if not buffered.

Notify data for regions: 0x7A ID.

Signature data: FBP format or CMP format.

5.5 *Continue Capture*

COMMAND: <0x7A><0x46><0x11>

Continue capture using specified parameters

PARAMETERS:

<0x7A><0x46><0x11> is the command head.

RETURN:

<ACK>

5.6 *Pause Capture*

COMMAND: <0x7A><0x46><0x12>

Pause capture

PARAMETERS:

<0x7A><0x46><0x12> is the command head.

RETURN:

<ACK><Script point count(4 bytes)>

5.7 *Get Script Point Count*

COMMAND: <0x7A><0x46><0x17>

Get script point count

PARAMETERS:

<0x7A><0x46><0x17> is the command head.

RETURN:

<ACK><Script point count(4 bytes)>

5.8 *Clear Signature*

COMMAND: <0x7A><0x46><0x19>

Clear signature

PARAMETERS:

<0x7A><0x46><0x19> is the command head.

RETURN:

<ACK>

5.9 *Exit Capture*

COMMAND: <0x7A><0x46><0x1F>

Exit capture All regions will be deleted.

PARAMETERS:

<0x7A><0x46><0x1F> is the command head.

RETURN:

<ACK>

5.10 *Get SIG Format Signature*

COMMAND: <0x7A><0x46><0x20>

Get buffered signature SIG format data.

PARAMETERS:

<0x7A><0x46><0x20> is the command head.

RETURN:

<ACK><SIG format signature data>

5.11 *Get CMP Format Signature*

COMMAND: <0x7A><0x46><0x21>

Get buffered signature CMP format data.

PARAMETERS:

<0x7A><0x46><0x21> is the command head.

RETURN:

<ACK><CMP format signature data>

5.12 *Get RAW Format Signature*

COMMAND: <0x7A><0x46><0x22>

Get buffered signature RAW format data.

PARAMETERS:

<0x7A><0x46><0x22> is the command head.

RETURN:

<ACK><RAW format signature data>

RAW data format: Three bytes for one point: xxxxxxxx xxxxyyyy yyyyyyyy

5.13 *Get BMP Format Signature*

COMMAND: <0x7A><0x46><0x23>

Get buffered signature BMP format data.

PARAMETERS:

<0x7A><0x46><0x23> is the command head.

RETURN:

<ACK><BMP format signature data>

BMP data format: three bytes for one point: Red Green Blue

5.14 *Set SIG Format Signature*

COMMAND: <0x7A><0x46><0x30><SIG format signature data>

Send SIG format signature to Sign&Pay.

PARAMETERS:

<0x7A><0x46><0x30> is the command head.

<SIG format signature data>

RETURN:

<ACK>

5.15 *Set CMP Format Signature*

COMMAND: <0x7A><0x46><0x31><CMP format signature data>

Send CMP format signature to using.

PARAMETERS:

<0x7A><0x46><0x31> is the command head.

<CMP format signature data>

RETURN:

<ACK>

5.16 *Set RAW Format Signature*

COMMAND: <0x7A><0x46><0x32><RAW format signature data>

Send RAW format signature to using.

PARAMETERS:

<0x7A><0x46><0x32> is the command head.

<RAW format signature data>

RETURN:

<ACK>

5.17 Set BMP Format Signature

COMMAND: <0x7A><0x46><0x33><BMP format signature data>

Send BMP format signature to using.

PARAMETERS:

<0x7A><0x46><0x33> is the command head.

<BMP format signature data>:BMP data format: three bytes for one point: Red Green Blue

RETURN:

<ACK>

6.0 Error Code List

ERROR_PARAMETER	0xE100
ERROR_LOWOUTBUFFER	0xE200
ERROR_SAVED_DATA_NOT_EXIST	0xE600
ERROR_NO_DATA_AVAILABLE	0xE800
ERROR_COMMAND_UNSUPPORTED	0xEC00
ERROR_COMMAND_PROCESS	0xED00
ERROR_INVALID_COMMAND	0xEE00