	FileName	XFD/SFD Application Documentation		
Document number	SP004	Edition	I	Page number	No 1 Pages in total 131 页

Version	Publisher	Modify the content	Date
A	Liangyq	First edition	2022/10/18
B	Liangyq	Add report _ pos and report _ data user functions	2023/04/26
C	Liangyq	Change the buzzer function to set _ buzzer, remove the enable bits of set _ sleep and set _ buzzer, and add the get _ sleep _ bl ()/get _ sleep _ time () function	2023/04/29
D	Liangyq	Appendix Added Control Type Coding Table	2023/05/26
		Optimize document content	2023/10/24
E	Liangyq	Contents of functions and instructions added to set _ hidden _ current _ page and get	2024/01/04

			_ ver	
F	Liangyq	Add 3 user function interfaces to animate		2024/03/20
G	Liangyq	Add METER instruction, optimize and add user function of curve part, modify button control to key control, enable slide block, arc progress bar, code table and curve content; Merge the operation instructions of qrcode into set _ text/get _ text		2024/07/31
		Added the user function of auto-increment and the description of SFD part		2024/11/12
H	Liangyq	Added instructions and functions related to touch enable and setting touch state		2024/11/16

Document number	SP004	FileName		XFD/SFD Application Documentation	
		Edition	I	Page number	No 3 Pages in total 131 页
	I	Liangyq	Add some user functions and directives		2025/01/11

Preface:

XF series/SF series serial display panel (XFD/SFD for short) is a serial display module series product developed by our company for customers. The design tenet of the product is simplicity and high efficiency. During the design process, a lot of optimization work has been done on its appearance and software. Its main features are: two-wire UART TTL and 485 are used as communication ports to provide universal plug-in connectors. The design software and the bottom firmware are highly integrated, and complex control functions can be completed through simple drag and drop operations. The appearance size is kept consistent with a color display screen on the market, and four assembly positioning holes are provided at the same time, so that the assembly of a customer is convenient, and the structural design of a finished product of the customer is not affected; a user and the module can perform human-computer interaction through the capacitive touch screen, so that the user experience is greatly improved; The embedded image processor deeply strips the complex underlying operations of the bare display screen and the touch screen (reading and writing of the data register/command register, calling of

the initialization parameters, modification of the screen rotation register,

adjustment of the backlight resistance, initialization and correction of the

capacitive touch screen, etc.), and extracts a set of efficient serial port screen user

interfaces, so that the interaction between the main control and the serial port

screen is simple and worry-free. Based on the above characteristics of software and

hardware, XFD is suitable for many applications and occasions such as instruments,

meters, intelligent terminals and so on.

In this document, XF series/SF series serial port screens are abbreviated as

XFD/SFD or collectively referred to as modules. The contents of this document

are applicable to XFD/SFD series developed by our company. Its size includes

but is not limited to products such as 4.3-inch (XF043), 5-inch XF050, 7-inch

(XF070) and 3.5-inch SF035.

1、 Resource description:

⑴.XFD/SFD supports UTF-8 encoding and can display Chinese, English and

multiple languages within the encoding range, but does not support GB2312

encoding. If GB2312 encoding is required, JC and HF series products of our

company can be considered.

(2).XFD/SFD supports images in PNG format and can realize transparent

controls and image effects.

(3). The user storage space of XFD/SFD is 14 M, which can be used to store fonts, images, scripts and user variable units (IVAR and SVAR). At present, user resources exceeding 14 M are not supported. If more image resources need to be displayed but XFD/SFD cannot meet the requirements, please contact customer service. Our company can develop customized solutions for large-capacity FLASH or SD card storage for customers.

Font file
Picture file
Script
Other

FLASH Storage Space Distribution Map

(4).The number of XFD/SFD pages and controls depends on the size of internal memory resources and storage space, but there is no limit to the total number of pages and the number of controls on the pages. The specific resources will be calculated and processed by the UI design software sHMI. If XFD/SFD can not meet

the need for more resources in the design process, please contact customer service, our company can make customized solutions for customers.

(5).XFD/SFD has a maximum limit of 256/128 bytes for the number of characters displayed in text-related controls (such as textbox, btn, etc.). Characters beyond this range cannot be displayed normally. If you have more text character display requirements, please contact customer service, our company can make customized programs for customers.

(6). The instruction buffer of XFD/SFD is 1024/128. If the total length of the sent instruction exceeds this range, the instruction will be truncated and cannot be parsed and executed.

(7)XFD/SFD supports the standard 485 protocol, but not the standard modbus protocol. If you need to communicate directly with PLC and other modbus controllers, please contact customer service. Our company can make customized solutions for customers.

(8)XFD/SFD is not provided with a real-time clock module. If a clock is required, the user's main control chip shall provide the clock information, and the XFD/SFD shall be responsible for displaying it.

(9). The interface editing software used for XFD/SFD matching is sHMI. If Sunstudio

and sGUI are used to develop the interface for XFD/SFD, it will not work properly.

⑽.The XFD/SFD bottom layer supports 11 dynamic controls, 2 keyboards, and 9

timers (8 general-purpose timers and 1 sleep timer).

11 dynamic controls

Control type	Control type number
Page control	0X01
Label control	0X02
Text control	0X03
Key controls	0X06
Progress bar control	0X07
Image control	0X08
Meta control	0X0A
Code Table Control	0X0B
Curve controls	0X0C
Slider control	0X0D

Scroll wheel control	0X11
QR Code control	0X12
Table control	0X13
Radio box control	0X14
Toggle controls	0X15
Arc progress control	0X16
Turntable progress control	0X17

2 keyboards

Numeric keypad and English keyboard

8 timers

The number is 0 ~ 7

(11)Configuration information comparison table (subject to the specification)

| Model	Resolution	Size	Maximum pages	Maximum widgets one page	Maximum pictures	Maximum fonts	Maximum characters	Instruction buffer
XF043	480*272	4.3 inch	32	64	255	32	256	1024
XF050	480*854	5 inch	32	64	255	32	256	1024
XF050	800*480	5 inch	32	64	255	32	256	1024
XF070	800*480	7 inch	32	64	255	32	256	1024
XF070	1024*600	7 inch	32	64	255	32	256	1024
XF101	1024*600	10.1 inch	32	64	255	32	256	1024
SF035	320*480	3.5 inch	32	36	128	4	128	128
SF024	240*320	2.4 inch	32	36	128	4	128	128

2. Hardware description

(1) XFD/SFD supports UART TTL serial data port (TX and RX lines) or UART 485 data

port for serial communication with the master control (hexadecimal format). If TTL

is used for communication with XFD/SFD, TX is the data sending end of the master

control. RX is the data receiving end of the master control (note that TX and RX do

not need cross wiring). If 485 is used to communicate with the master control, A +

and B- can be directly connected to A + and B- of the master control.
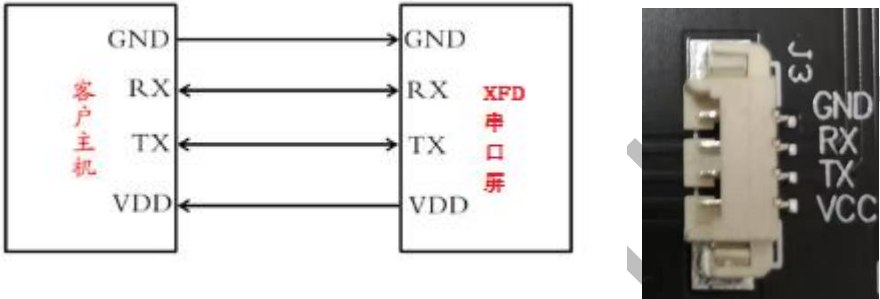
Schematic diagram of TTL wiring logic
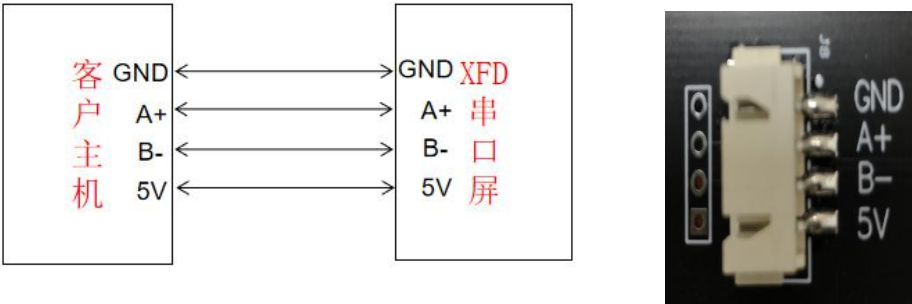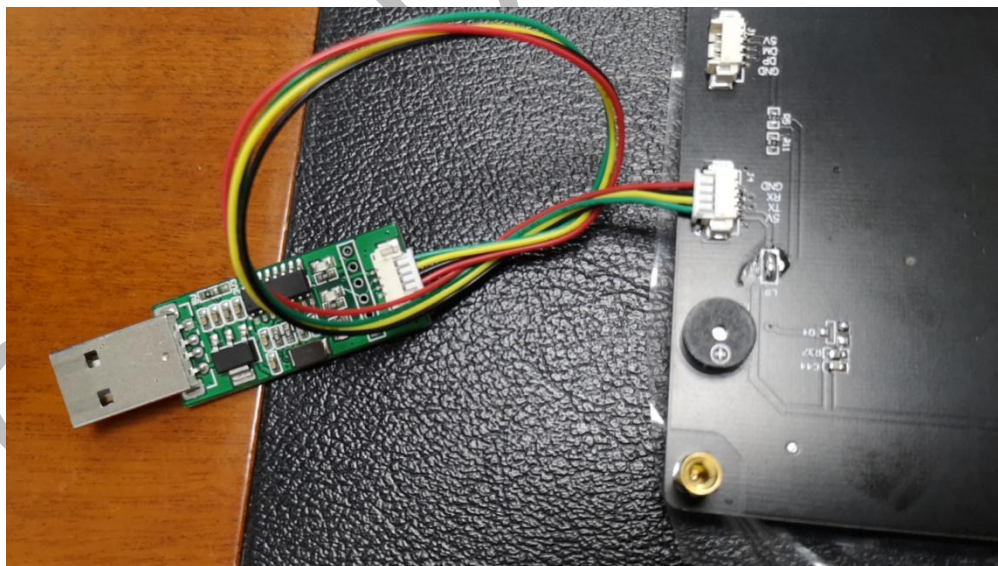

485 wiring logic diagram

(2) The XFD serial port panel only supports 5V power supply voltage, and the

communication level logic is 3.3V;

	FileName	XFD/SFD Application Documentation		
Document number	SP004	Edition	I	Page number	No 11 Pages in total 131 页

(3) When downloading the project, it is necessary to use another serial port to communicate with the sHMI upper computer. Use the following TTL to USB tool to connect with the computer, update the driver of the TTL tool, and then directly insert the cable into the XFD/SFD connector.

USB to UART TTL Tool XFD/SFD Project Download Serial Port

Connection method

2. Software description

(1) Preliminary knowledge before programming

①. Video memory of display screen:

DDRAM is the video memory, and the video memory of the module

corresponds to the resolution of the screen, such as the XF043 module.

Video memory: 480 * 272 * 4 = 522240 bytes, where 4 indicates that one video

memory unit occupies 4 bytes (i.e. 32 bits), and the corresponding relationship is:

4 bytes (32 bits (1 pixel (8 bits (red) 8 bits (green) 8 bits (blue) 8 bits

(transparency)

②Video memory address:

The address of DDRAM is the position of DDRAM, which is actually

broken down into two positions in the X direction and the Y direction.

Therefore, the address of DDRAM is the coordinates of X and Y. Starting

from the upper left corner when XF043 is placed, X increases from left to

right (for example, XF043 starts from 0 to 271, and then returns to 0). Y

increases from top to bottom (for example, XF070 starts from 0 to 479

and then returns to 0). The black box in the figure below is the display

area (i.e. DDRAM), and the blue line in the box represents the scanning

trace.

	FileName	**XFD/SFD Application Documentation**		
Document number	SP004	Edition	I	Page number	No 13 Pages in total 131 页
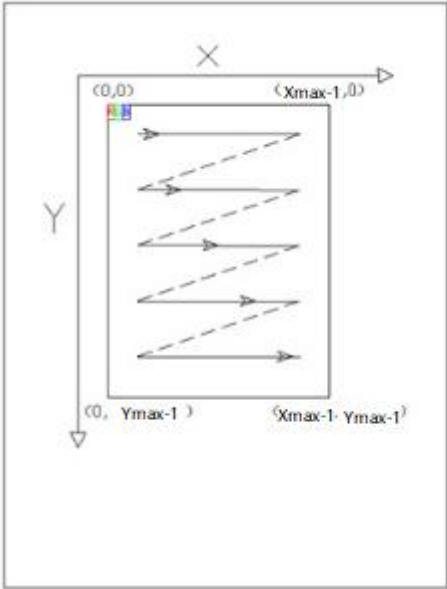
DDRAM address layout of XFD

③. Relationship between display content and video memory address:
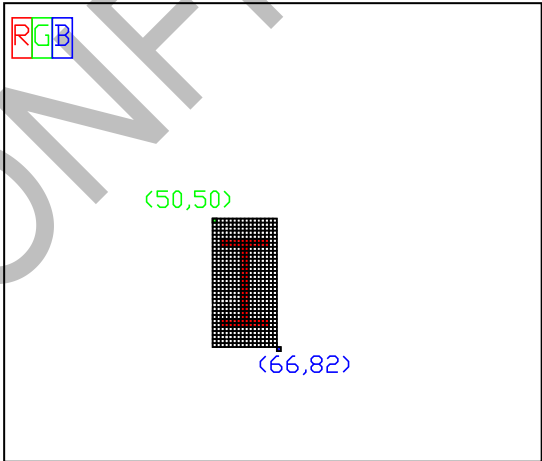
As can be seen from the above figure, write 'I' to the start position of (50, 50) in the video memory, and the content of the display screen corresponds to the content of the corresponding address one by one. Similarly, the address of the video memory is consistent with the location of
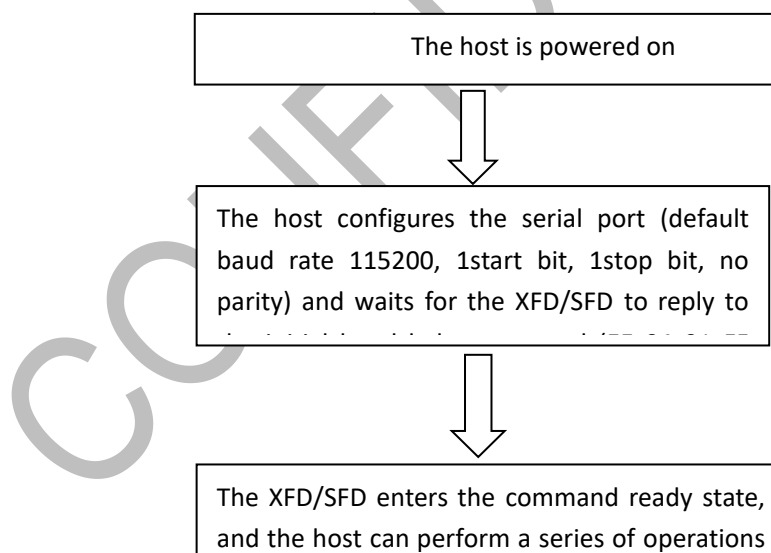
the screen.

The integrated serial port screen has shielded many details of the operation of the display screen, and the user only needs to provide the corresponding instructions with the X and y positions of the display screen to display the corresponding contents at the corresponding positions on the screen.

(2) Startup instructions

After the XFD/SFD serial port screen is powered on, it will automatically complete the initialization of each functional component (LCD, FLASH, etc.). After the XFD/SFD completes the initialization, it will send the start completion command to the master control. After receiving the completion command, the master control can interact with the XFD/SFD. Therefore, the power-on process must meet the following conditions:

```
┌─────────────────────────────────────┐
│      The host is powered on          │
└─────────────────────────────────────┘
                  ⇓
┌─────────────────────────────────────┐
│ The host configures the serial port │
│ (default baud rate 115200, 1start    │
│ bit, 1stop bit, no parity) and waits │
│ for the XFD/SFD to reply to          │
└─────────────────────────────────────┘
                  ⇓
┌─────────────────────────────────────┐
│ The XFD/SFD enters the command ready │
│ state, and the host can perform a    │
│ series of operations                 │
└─────────────────────────────────────┘
```

Note that when the address or CRC check function is enabled, the reset handshake command will be:

EE addrH addrL 04 01 CRCH CRCL FF FC FF FF

(3) Application interface

①XFD/SFD provides users with the interface to control and access its internal resources. There are two kinds of user interfaces: hexadecimal serial instruction set and user function.

②. The baud rate of the main control serial port of the user must be consistent with that of the XFD/SFD. The baud rate of the XFD/SFD can be modified through the "Display Settings" menu in the sHMI or through the application interface. The baud rates that can be modified are 4800/9600/19200/38400/115200. The protocol format is: 1stop bit, 8 data bits, no parity.

③Serial port instruction set of XFD/SFD:

3.1 Instruction description:

3.1.1 The instruction of XFD/SFD has a fixed frame header (0xEE) and frame trailer (0xFF 0 XFC 0xFF 0xFF). If the frame header or frame trailer is incorrect, XFD/SFD will not be able to parse the instruction correctly.

3.1.2 XFD/SFD When the instruction parameter is greater than 1 byte, the high byte precedes the low byte. The maximum length allowed for an instruction is 1024 bytes/128 bytes (including the header and trailer).

3.1.3 The command can be checked with address (485 mode) and CRC _ 16. Whether to enable 485 mode and CRC16 will be set by sHMI before engineering download.

3.1.4 The instruction can have two instruction codes, the first one is the master instruction code, the second one is the slave instruction code, and some control instructions only have the master instruction code but not the slave instruction code.

3.1.5 There are 8 possible formats according to different instruction types and addresses and the requirements of CRC16:

Instruction format with no address, no CRC _ 16 check, no instruction code			
Frame header	Master Instruction Code	Command parameter	Trailer (4 bytes)
0xEE	0xXX	0xXXX~0xYYY	**0xFF 0XFC 0xFF 0xFF**

Instruction format with no address, no CRC _ 16 check, and slave instruction code

Frame header	Master Instruction Code	Slave instruction code	Command parameter	Trailer (4 bytes)
0xEE	0xXX	0xYY	0xXXX~0xYYY	**0xFF 0XFC 0xFF 0xFF**

Instruction format with no address, CRC _ 16 check, and no instruction code

Frame header	Master Instruction Code	Command parameter	CRC16 (2 bytes)	Trailer (4 bytes)
0xEE	0xXX	0xXXX~0xYYY	0xCCCC	0xFF 0XFC 0xFF 0xFF

Instruction format without address, with CRC _ 16 check, with slave instruction code

Frame header	Master Instruction Code	Slave instruction	Command parameter	CRC16 (2 bytes)	Trailer (4 bytes)

		code			
0xEE	0xXX	0xYY	0xXXX~0xYYY	0xCCCC	0xFF 0XFC 0xFF 0xFF

Instruction format with address, no CRC _ 16 check, no instruction code					
Frame header	High address	Low address	Master Instruction Code	Command parameter	Trailer (4 bytes)
0xEE	0xHH	0xLL	0xXX	0xXXX~0xYYY	**0xFF 0XFC 0xFF 0xFF**

Instruction format with address, without CRC _ 16 check, with slave instruction code						
Frame header	High address	Low address	Master Instruction Code	Slave instruction code	Command parameter	Trailer (4 bytes)
0xEE	0xHH	0xLL	0xXX	0xYY	0xXXX~0xYYY	**0xFF 0XFC 0xFF 0xFF**

Instruction format with address, CRC _ 16 check, and no instruction code						
Frame header	High address	Low address	Master Instruction Code	Command parameter	CRC16 (2 bytes)	Trailer (4 bytes)
0xEE	0xHH	0xLL	0xXX	0xXXX~0xYYY	0xCCCC	0xFF 0XFC 0xFF 0xFF

			Instruction format with address, CRC _ 16 check and slave instruction code				
Frame header	High address	Low address	Master Instruction Code	Slave instruction code	Command parameter	CRC16 (2 bytes)	Trailer (4 bytes)

	FileName	XFD/SFD Application Documentation					
Document number	SP004	Edition	I	Page number	No 20 Pages in total 131 页		
0xEE	0xHH	0xLL	0xXX	0xYY	0xXXX~0xYYY	0xCCCC	0xFF 0XFC 0xFF 0xFF

3.2 Instruction details

3.2.1 Reset command

This command is used to restart the XFD/SFD. If the host is restarted in an unexpected situation, the XFD/SFD can be synchronized with the host through the reset command.

After receiving this command, XFD/SFD restarts the XFD/SFD program and sends the EE 55 FF FC FF FF command to the user master.

Type of instruction	Reset command
Instruction content	**EE 03 A5 5A 53 35 FF FC FF FF**
An example of a directive	**EE** 03 A5 5A 53 35 **FF FC FF FF**
Instruction example description	EE represents the frame header
	03 is the command code
	A5 5A 53 35 is the content of the reset command
	FF FC FF FF indicates the end of frame

XFD/SFD Feedback	**EE** 04 01 **FF FC FF FF**
Precautions	If there is an address (assuming 0x0001), the data format is EE 00 01 03 A5 5A 53 35 FF FC FF FF If there is an address and the address is set to 0 × 0000, the data is broadcast data. The broadcast data will be executed by all XFDs attached to the 485 bus, but the XFD will not feed back any content after execution.
	If there is an address and CRC check is performed, the data format is EE 00 01 03 A5 5A 53 35 38 9E FF FC FF FF, where 38 9E is the value of CRC16 (CRC-16/MODBUS x16 + x15 + x2 + 1). The frame header, frame trailer and 16-bit address are not included, and the counting method of CRC16 is shown in the appendix.

3.2.2 Handshake instruction

This command is used to synchronize with the XFD/SFD. It can be used to confirm whether the XFD/SFD is online or in normal working state.

After receiving this command, the XFD/SFD will reply the EE 55 FF FC FF FF command to the user's master control, indicating that the XFD/SFD is online and

working normally.

Type of instruction	Handshake instruction
Instruction content	**EE 04 FF FC FF FF**
An example of a directive	**EE** 04 **FF FC FF FF**
Instruction example description	EE represents the frame header
	04 is the command code
	No parameters after the command code
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**EE** 04 01 **FF FC FF FF**
Precautions	If there is an address (assuming 0x0001), the data format is EE 00 01 04 FF FC FF FF
	If there is an address and the address is set to 0 × 0000, the data is broadcast data. The broadcast data will be executed by all XFDs attached to the 485 bus, but the XFD will not feed back any content after execution.
	If there is an address and CRC check is performed, the data format is EE 00 01 04 83 BE FF FC FF FF, where 83 BE is the value of CRC16. When counting CRC16, the frame

	header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.3 Jump Page Command

This command is used for page switching. When the page ID to be switched to is within the designed UI ID range, the user will exit the current page and switch to the specified page.

Type of instruction	Jump page instruction
Instruction content	**EE + 05 + page ID (16-bit) + jump time setting value (16-bit) + FF FC FF FF**
An example of a directive	**EE** 05 00 01 00 64 **FF FC FF FF**
Instruction example description	EE represents the frame header
	05 is the command code
	00 01 indicates a jump to page 1
	0064 indicates that it jumps in after a delay of 100ms
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**None**

Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 05 00 01 00 64 FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 05 00 01 00 64 2B B8 FF FC FF FF, where 2B B8 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.4 Command to obtain the current page ID

This directive is used to read the ID value of the currently displayed UI page from the XFD/SFD

Type of instruction	Get current page ID instruction
Instruction content	**EE 06 FF FC FF FF**
An example of a directive	**EE** 06 **FF FC FF FF**

		EE represents the frame header
	Instruction example description	06 is the command code
		No parameters
		FF FC FF FF indicates the end of frame
	XFD/SFD Feedback	If the ID of the current page is 3, it is: EE 06 00 03 FF FC FF FF
	Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 06 FF FC FF FF Get directive does not support broadcast operation
		If there is an address and CRC check is performed, the data format is EE 00 01 06 423F FF FC FF FF, where 423F is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.5 Set backlight brightness command

This command is used to dynamically set the backlight brightness value of the

XFD. The range of the value is 0 ~ 255. 0 is the darkest and 255 is the brightest.

Type of instruction	Set backlight brightness command

Instruction content	**EE + 07 + backlight brightness value (16-bit) + FF FC FF FF**
An example of a directive	**EE 07 00 64 FF FC FF FF**
Instruction example description	EE represents the frame header
	07 is the command code
	00 64 indicates that the set brightness is 100 (the value range of backlight brightness is 0 ~ 255, where 0 is the darkest and 255 is the brightest)
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	None
Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 07 00 64 FF FC FF FF
	If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 07 00 64 EA C1 FF FC FF FF, where EA C1 is the value of CRC16. When counting CRC16, the frame header,

frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.6 Obtain backlight brightness command

This command is used to obtain the backlight brightness value of the current XFD. The range of the value is 0 ~ 255. 0 is the darkest and 255 is the brightest.

Type of instruction	Obtain the current backlight brightness command
Instruction content	**EE 08 FF FC FF FF**
An example of a directive	**EE** 08 **FF FC FF FF**
Instruction example description	EE represents the frame header
	08 is the command code
	No parameters
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	If the current backlight brightness value is 45, then: EE 08 00 2D FF FC FF FF
	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 08 FF FC FF FF

Precautions	Get directive does not support broadcast operation
	If there is an address and CRC check is performed, the data format is EE 00 01 08 86BE FF FC FF FF, where 86BE is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.7 Set Baud Rate Command

This command is used to modify the communication baud rate of XFD/SFD. The baud rate is set by the index value. The range of the value is: 0/1/2/3 respectively representing: 115200/38400/19200/9600/4800. 0 is the default 115200 baud rate.

Type of instruction	Set baud rate command
Instruction content	**EE + 09 + encoded value of baud rate + FF FC FF FF**
An example of a directive	**EE** 09 00 01 **FF FC FF FF**
Instruction example description	EE represents the frame header
	09 is the command code
	00 01 indicates that the number of the baud rate is 1 (that is, the baud rate set to the 38400)

	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	None
Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 09 00 01 FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 09 00 01 02 60 FF FC FF FF, where 02 60 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.8 Command to obtain the current baud rate

This instruction is used to obtain the baud rate used by the current XFD/SFD.

The value range is 0 ~ 3, which represents 115200/38400/19200/9600 respectively.

0 is the default 115200 baud rate.

Type of instruction	Command to get the current baud rate
Instruction content	**EE 0A FF FC FF FF**

An example of a directive	**EE** 0A **FF FC FF FF**
Instruction example description	EE represents the frame header
	0 A is the instruction code
	No parameters
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	Assuming that the current baud rate is 115200, then: EE 0 A 00 00 FF FC FF FF
Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 0A FF FC FF FF
	Get directive does not support broadcast operation
	If there is an address and CRC check is performed, the data format is EE 00 01 0A 47 3F FF FC FF FF, where 47 3F is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.9 Enable General-Purpose Timer Command

XFD/SFD supports 8 general-purpose timers, whose ids are 0 ~ 7 respectively.

The timer will execute the event preset by the user after the timer time is up. To enable the timer, you need to specify the timer ID, single trigger option, trigger interval time and set the enable switch.

Type of instruction	Enable the general-purpose timer
Instruction content	**EE + 0B + timer ID + one-shot option + trigger interval + enable switch**
An example of a directive	**EE** 0B 00 00 00 01 00 64 00 01 **FF FC FF FF**
Instruction example description	EE represents the frame header
	0 B is the script code
	00 00 is the timer ID
	00 01 indicates that the timer is a one-shot timer
	00 64 indicates that the timing time is 100ms
	The last 00 01 indicates that the timer is enabled
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	None
Precautions	If there is an address (assuming 0x0001), the data format is: EE 00 01 0B 00 00 00 01 00 64 00 01 FF FC FF FF
	If there is an address and the address is set to 0x0000,

the data is broadcast data, and the broadcast data will be executed by all XFDs attached to the 485 bus

If there is an address and CRC check is performed, the data format is EE 00 01 0B 00 00 00 01 00 64 00 01 1F C4 FF FC FF, where 1F C4 is the value of CRC16. When counting CRC16, the frame header, frame trailer, and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.10 Save Control Properties Command

XFD/SFD can modify the attributes of controls (such as set _ X/set _ y/set _ text, etc.) online during operation, but the modified contents are only saved in RAM and will be lost after power failure. Therefore, if the modified attributes of controls need to be saved, they can be saved through this command.

Type of instruction	Save Control Properties Instruction
Instruction content	**EE** 0C **FF FC FF FF**
An example of a directive	**EE** 0C **FF FC FF FF**
Instruction example	EE represents the frame header
	0 C is the script code

description	No parameters
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	None
Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 0C FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 0C 45 BF FF FC FF FF, where 45 BF is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.11 Set delay command

The delay instruction is used to wait for a period of time in ms

Type of instruction	Set the delay command
Instruction content	**EE + 0F + delay time + FF FC FF FF**
An example of a directive	**EE** 0F 00 64 **FF FC FF FF**

		EE represents the frame header
	Instruction example description	0 F is the instruction code
		00 64 indicates a delay of 100ms
		FF FC FF FF indicates the end of frame
	XFD/SFD Feedback	None
	Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 0F 00 64 FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
		If there is an address and CRC check is performed, the data format is EE 00 01 0F 00 64 28 40 FF FC FF FF, where 28 40 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.12 Enable Sleep Timer Command

XFD/SFD supports one dormancy timer. When the set timer time is up, the dormancy timer will enter the dormancy state and execute the event preset by the user. To enable the dormancy timer, you need to specify two parameters: the trigger dormancy interval time (the time to exit dormancy) in seconds and the backlight brightness during dormancy.

Type of instruction	Enable Sleep Timer Instruction
Instruction content	**EE + 10 + sleep time + backlight brightness during sleep**
An example of a directive	**EE 10** 00 01 00 0A **FF FC FF FF**
Instruction example description	EE represents the frame header
	10 is the instruction code
	00 01 is the delay time of 1 second
	00 0 A Go to Sleep Backlight brightness adjusted to 10
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	None
Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 10 00 01 00 0A FF FC FF FF If there is an address and the address is set to 0x0000,

	FileName	XFD/SFD Application Documentation		
Document number	SP004	Edition	I	Page number
				No 36 Pages in total 131 页

the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus

If there is an address and CRC check is performed, the data format is EE 00 01 10 00 01 00 0A 04 34 FF FC FF FF, where 04 34 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.13 Command for Obtaining Sleep Timer Parameters

XFD/SFD supports one sleep timer, which can be used to obtain the values of sleep interval time (time to exit sleep) and backlight brightness during sleep.

Type of instruction	Command for acquiring sleep timer parameter
Instruction content	**EE 11 FF FC FF FF**
An example of a directive	**EE** 11 **FF FC FF FF**
Instruction example description	EE represents the frame header
	11 is the command code
	No parameters

		FF FC FF FF indicates the end of frame
	XFD/SFD Feedback	If the current sleep timer time is 5S and the sleep backlight brightness is 10: EE 10 00 05 00 0A FF FC FF FF
	Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 11 FF FC FF FF Get directive does not support broadcast
		If there is an address and CRC check is performed, the data format is EE 00 01 11 4C 7F FF FC FF FF, where 4C 7F is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.14 Command for setting time parameter of buzzer

The beep time of the XFD/SFD buzzer can be set by this command

Type of instruction	Command for setting time parameter of buzzer
Instruction content	**EE + 12 + beep time + FF + FC + FF + FF**

An example of a directive	**EE** 12 00 01 **FF FC FF FF**
Instruction example description	EE represents the frame header
	12 is the instruction code
	00 01 indicates that the beep time is 1ms
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**None**
Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 12 00 01 FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 12 00 01 05 10 FF FC FF FF, where 05 10 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix

for the counting method of CRC16.

3.2.15 Command for obtaining buzzer parameters

The XFD/SFD can read the setting parameters of the buzzer, and its return is the beep time.

Type of instruction	Command for obtaining buzzer parameter
Instruction content	**EE 13 FF FC FF FF**
An example of a directive	**EE** 13 **FF FC FF FF**
Instruction example description	EE represents the frame header
	13 is the instruction code
	No parameters
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	If the current beep duration is 1ms, EE 13 00 01 FF FC FF FF
Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 13 00 01 FF FC FF FF
	Get directive does not support broadcast

If there is an address and CRC check is performed, the data format is EE 00 01 13 8D FE FF FC FF FF, where 8D FE is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.16 Set Integer Variable Instruction

There are at most 32 integer variables (IVAR [0] ~ IVAR [31]) in XFD/SFD. IVAR is used to store user parameters, and its value is 32 bits. This instruction can assign a value to IVAR

Type of instruction	Set Integer Variable Instruction
Instruction content	EE + 14 + IVAR ID + value of IVAR (32-bit) + FF FC FF FF
An example of a directive	**EE** 14 00 02 00 00 00 38 **FF FC FF FF**
	EE represents the frame header

		14 is the instruction code
Instruction example description		00 02 for IVAR [2]
		00 00 00 38 means 56
		FF FC FF FF indicates the end of frame
XFD/SFD Feedback		**None**
Precautions		If there is an address (assuming 0x0001), the data format is EE 00 01 14 00 02 00 00 38 FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
		If there is an address and CRC check is performed, the data format is EE 00 01 14 00 02 00 00 38 13 37 FF FC FF FF, where 13 37 is the value of CRC16. When counting CRC16, the frame header, frame trailer, and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.17 Get Value of Integer Variable Instruction

The user can read the value of any one of the built-in 32 integer variables

through this instruction, and the ID of IVAR must be specified when reading.

Type of instruction	Get the value of an integer variable instruction
Instruction content	**EE+15+**IVAR id**+FF FC FF FF**
An example of a directive	**EE 1**5 00 02 **FF FC FF FF**
Instruction example description	EE represents the frame header
	15 is the command code
	00 02 means to read the value of IVAR [2] with ID 2
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	If the current value of IVAR [2] is -56, then: EE 15 00 02 FF FF FF C8 FF FC FF FF
Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 15 00 02 FF FC FF FF Get directive does not support broadcast
	If there is an address and CRC check is performed, the data format is EE 00 01 15 00 02 C5 E1 FF FC FF FF,

where C5 E1 is the value of CRC16. When counting

CRC16, the frame header, frame trailer and 16-bit

address are not included. See the appendix for the

counting method of CRC16.

3.2.18 Set String Variable Instruction

There are at most 32 string variables (SVAR [0] ~ SVAR [31]) in XFD/SFD. SVAR is

used to store the string of the user. The maximum number of strings is 256. This

instruction can be used to assign values to SVAR

Type of instruction	Set String Variable Instruction
Instruction content	**EE + 16 + string variable SVAR ID + content of character + terminator + FF FC FF FF**
An example of a directive	**EE** 16 00 01 31 31 31 33 34 00 **FF FC FF FF**
Instruction description	EE represents the frame header
	16 is the instruction code
	00 01 for SVAR [1]
	31 31 31 33 34 represents the string '11134'.
	The last 00 is the end of the string.

	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**None**
Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 16 00 01 31 31 31 33 34 00 FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 16 00 01 31 31 31 33 34 00 86 6D FF FC FF, where 86 6D is the value of CRC16. When counting CRC16, the frame header, frame trailer, and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.19 Get String Variable Instruction

The user can read the value of any one of the built-in 32 string variables through

this command, and the ID of the SVAR must be specified when reading.

Type of instruction	Get String Variable Instruction
Instruction content	**EE + 17 + string variable ID + FF FC FF FF**
An example of a directive	**EE** 17 00 01 **FF FC FF FF**
Instruction example description	EE represents the frame header
	17 is the instruction code
	00 01 indicates that the value of SVAR [1] is read
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	If the current value of SVAR [1] is 11134, EE 17 00 01 31 31 31 33 34 00 FF FC FF FF.
Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 17 00 01 FF FC FF FF
	Get directive does not support broadcast
	If there is an address and CRC check is performed, the data format is EE 00 01 17 00 01 04 00 FF FC FF FF, where 04 00 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the

counting method of CRC16.

3.2.20 Obtain the version number

The user can read the version information of the firmware through this

command

Type of instruction	Get version number directive
Instruction content	**EE** +18+ **FF FC FF FF**
An example of a directive	**EE** 18 **FF FC FF FF**
Instruction example description	EE represents the frame header
	18 is the instruction code
	/
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	If the current version number is' FWxx _ SHMI _ ST _ 01 _ V01 ', then: EE 18 46 57 78 78 5F 53 48 4D 49 5F 53 54 5F 30 31 5F 56 30 31 00 FF FC FF FF

Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 18 FF FC FF FF Get directive does not support broadcast
	If there is an address and CRC check is performed, the data format is EE 00 01 18 4A BF FF FC FF FF , where 4A BF is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.21 Calibrate the touch screen

The user can recalibrate the resistive touch screen with this command

Type of instruction	Calibrate the touch screen command
Instruction content	**EE** +1B+ **FF FC FF FF**
An example of a directive	**EE** 1B **FF FC FF FF**
	EE represents the frame header

Instruction example description	1B is the instruction code
	/
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**EE** 1B 01 **FF FC FF FF**
Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 1B 01 FF FC FF FF Get directive does not support broadcast
	If there is an address and a CRC check is performed, the data format is EE 00 01 1B 01 80 CA FF FC FF FF , where 80 CA is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.22 Set Control X Coordinate Command

This command is used to change the X coordinate of the control. The ID of the control must be specified when using this command. Note that the set coordinates will be lost after power failure. If you want to save the changes to the control, you can use the Save Control Properties command or the User Function user. Save _ gui (1) to save them.

Type of instruction	Set control X coordinate instruction
Instruction content	**EE + 81 + 01 + control ID + X coordinate value set + FF FC FF FF**
An example of a directive	**EE** 81 01 00 06 00 0A **FF FC FF FF**
Instruction description	EE represents the frame header
	81 The master instruction code 01 is the slave instruction code
	00 06 represents the ID of the control 00 0 A of 10 modifies the X coordinate of control 6 to 10
	FF FC FF FF indicates the end of frame

XFD Feedback Content	**None**
Precautions	If there is an address (assuming 0x0001), the data format is EE 00 01 81 01 00 06 00 0A FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 81 01 00 06 00 0A CC 43 FF FC FF FF, where CC 43 is the value of CRC16. When counting CRC16, the frame header, frame trailer, and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.23 Set control Y coordinate command

This command is used to change the y coordinate of the control. The ID of the control needs to be specified when using this command. Note that the set coordinates will be lost after power failure. If you want to save the changes to the control, you can use the Save Control Properties command or the User Function

user. Save _ gui (1) to save them.

Type of instruction	Set control Y coordinate instruction
Instruction content	**EE + 81 + 02 + control ID + set y coordinate value + FF FC FF FF**
An example of a directive	**EE** 81 02 00 06 00 32 **FF FC FF FF**
Instruction example description	EE represents the frame header
	81 Master code 02 is a slave code
	00 06 represents the ID of the control
	00 32 = 50 means that the y coordinate of control 6 is modified to 50
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**None**
Precautions	If there is an address (assuming 0x0001), the data format is EE 00 01 81 02 00 06 00 32 FF FC FF FF
	If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the

data format is EE 00 01 81 02 00 06 00 32 1E 06 FF FC FF FF, where 1E 06 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.24 Set Control W Width Command

This command is used to change the width of the control. The ID of the control must be specified when using this command. Note that the set width will be lost after power failure. If you want to save the changes to the control, you can use the Save Control Properties command or the User Function user. Save _ gui (1) to save.

Type of instruction	Set Control W Width Command
Instruction content	**EE + 81 + 03 + control ID + set width value + FF FC FF FF**
An example of a directive	**EE** 81 03 00 06 00 32 **FF FC FF FF**
Instruction example	EE represents the frame header
	81 Master code 03 is a slave code

description	00 06 represents the ID of the control
	00 32 = 50 means that the width of control 6 is modified to 50
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**None**
Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 81 03 00 06 00 32 FF FC FF FF
	If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 81 03 00 06 00 32 DE 3B FF FC FF FF, where DE 3B is the value of CRC16. When counting CRC16, the frame header, frame trailer, and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.25 Set Control H Height Command

This command is used to change the width of the control. The ID of the control must be specified when using this command. Note that the height setting will be lost after power failure. If you want to save the changes to the control, you can use the Save Control Properties command or the User Function user. Save _ gui (1) to save.

Type of instruction	Set Control W Width Command
Instruction content	**EE + 81 + 04 + control ID + set width value + FF FC FF FF**
An example of a directive	**EE** 81 04 00 06 00 32 **FF FC FF FF**
Instruction description	EE represents the frame header
	81 Master code 04 is a slave code
	00 06 indicates that the ID of the control 00 32 is 50 indicates that the width of the control 6 is modified to 50
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**None**
	If there is an address (assuming 0x0001), the data

Precautions	format is EE 00 01 81 04 00 06 00 32 FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 81 04 00 06 00 32 1E 8E FF FC FF FF, where 1E 8E is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.26 Set Text Instructions for Controls

This command is used to change the text content of the control. The ID of the control needs to be specified when it is used. Note that the content modified by this command will be lost after power failure. If you want to save the modifications to the control, you can use the Save Control Properties command or the user function. The user. Save _ gui (1) is stored.

Type of instruction	Sets the text instruction for the control

Instruction content	EE + 81 + 05 + control ID + text string + 00 + FF FC FF FF
An example of a directive	**EE** 81 05 00 0C 64 69 73 70 6C 61 79 00 **FF FC FF FF**
Instruction example description	EE represents the frame header
	81 Master code 05 is a slave code
	00 0C indicates that the ID of the control is 12
	64 69 73 70 6C 61 79 is the string 'display'.
	The last 00 is the end of the string
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**None**
Precautions	If there is an address (assuming 0x0001), the data format is: EE 00 01 81 05 00 0C 64 69 73 70 6C 61 79 00 FF FC FF FF
	If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 81 05 00 0C 64 69 73 70 6C 61 79 00 00 9B FF FC FF FF, where 00 9B is the value of CRC16. The frame header,

frame trailer and 16-bit address are not included, and the

counting method of CRC16 is shown in the appendix.

3.2.27 Set Control Background Color 1 Command

This command is used to change the background color 1 of the control. The ID of the control must be specified when using this command. Note that the content modified by this command will be lost after power failure. If you want to save the modifications to the control, you can use the Save Control Properties command or the user function. The user. Save _ gui (1) is stored.

Type of instruction	Set the background color of the control 1 command
Instruction content	**EE 81 + 06 + control ID + status + color (24 bits) + FF FC FF FF**
An example of a directive	**EE** 81 06 00 08 00 00 FF FF FF FF **FF FC FF FF**
Instruction example description	EE represents the frame header
	81 Master code 06 is a slave code
	00 08 indicates that the ID of the control is 8 00 00 indicates the default state FF 24-bit background

	color
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**None**
Precautions	If there is an address (assuming the address is 0x0001), the data format is: EE 00 01 81 06 00 08 00 00 FF FC FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 81 06 00 08 00 00 FF DE 85 FF FC FF, where DE 85 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.28 Set Control Background Color 2 Command

This command is used to change the background color of the control 2. The ID of the control must be specified when using this command. Note that the content modified by this command will be lost after power failure. If you want to save the modifications to the control, you can use the Save Control Properties command or the user function. The user. Save _ gui (1) is stored.

Type of instruction	Set the background color of the control 2 command
Instruction content	EE + 81 + 07 + control ID + status + color (24 bits) + FF FC FF FF
An example of a directive	**EE** 81 07 00 09 00 01 FF FF FF FF **FF FC FF FF**
Instruction description	EE represents the frame header
	81 Master code 07 is a slave code
	00 09 indicates that the ID of the control is 9 00 01 indicates the pressed state FF 24-bit background color
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**None**
	If there is an address (assuming the address is 0x0001), the data format is: EE 00 01 81 07 00 09 00 01 FF FC FF

Precautions	If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 81 07 00 09 00 01 FF 4E A5 FF FC FF, where 4E A5 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.29 Command to set the foreground color of the control

This command is used to change the foreground color (the color of the font) of the control. The ID of the control must be specified when using this command.

Note: The content modified by this command will be lost after power failure. If you want to save the changes to the control, you can use the Save Control Properties command or the user function; user. Save _ gui (1) to save.

Type of instruction	Set the foreground color command for the control
Instruction content	EE + 81 + 08 + control ID + color (24 bit) + FF FC FF FF
An example of a directive	**EE** 81 08 00 0A 00 00 FF FF **FF FC FF FF**

		EE represents the frame header
	Instruction example description	81 Master code 08 is a slave code
		00 0 A indicates that the control has an ID of 10
		00 00 FF FF 24-bit background color
		FF FC FF FF indicates the end of frame
XFD/SFD Feedback		**None**
Precautions		If there is an address (assuming 0x0001), the data format is: EE 00 01 81 08 00 0A 00 00 FF FC FF
		If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
		If there is an address and CRC check is performed, the data format is EE 00 01 81 08 00 0A 00 00 FF D699 FF FC FF, where D699 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.30 Set Value Command for Control

	FileName	**XFD/SFD Application Documentation**	
Document number	SP004	Edition	I	Page number	No 62 Pages in total 131 页

This command is used to change the value of the control (usually prog/slider/arc/IMG), and the ID of the control needs to be specified when it is used. Note: The content modified by this command will be lost after power failure. If you want to save the changes to the control, you can use the Save Control Properties command or the user function; user. Save _ gui (1) to save.

Type of instruction	Set Control's Value Instruction
Instruction content	EE + 81 + 09 + control ID + value of control + FF FC FF FF
An example of a directive	**EE** 81 09 00 09 00 0A **FF FC FF FF**
Instruction example description	EE represents the frame header
	Master instruction code 09 is a slave instruction code
	00 09 indicates that the ID of the control is 9
	0 A means to modify the value to 10
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**None**
Precautions	If there is an address (assuming 0x0001), the data format is: EE 00 01 81 09 00 09 00 0A FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be

executed by all XFDs/SFDs attached to the 485 bus

If there is an address and CRC check is performed, the data format is EE 00 01 81 09 00 09 00 0A 0E 92 FF FC FF FF, where 0E 92 is the value of CRC16. When counting CRC16, the frame header, frame trailer, and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.31 Set Hidden Control Command

This command is used to hide all the controls on a certain page or a certain page. When using it, you need to specify the ID of the control and all the flags. Note that the contents modified by this command will be lost after power failure. If you want to save the modifications to the control, you can use the Save Control Attribute command or the user function. The user. Save _ gui (1) is stored.

Type of instruction	Set Control Hide Instruction
Instruction content	EE + 81 + 0A + control ID + hide all option + hide switch + FF FC FF FF
An example of a directive	**EE** 81 0A 00 01 00 00 00 01 **FF FC FF FF**

		EE represents the frame header
	Instruction example description	81 Master code 0 A is the slave code
		00 01 indicates that the ID of the control is 1
		00 00 means not all need to be hidden
		00 01 indicates that hide is enabled
		FF FC FF FF indicates the end of frame
	XFD/SFD Feedback	**None**
	Precautions	If there is an address (assuming 0x0001), the data format is: EE 00 01 81 0A 00 01 00 00 00 01 FF FC FF FF
		If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
		If there is an address and CRC check is performed, the data format is EE 00 01 81 0A 00 01 00 00 00 01 A7 DF FF FC FF, where A7 DF is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.32 Set Hidden Curve Command

This command is used to hide the curve. When using it, you need to specify the ID of the control and all the flags. Note that the content modified by this command will be lost after power failure. If you want to save the modification of the control, you can use the Save Control Attribute command or the user function. The user. Save _ gui (1) is stored.

Type of instruction	Set Hidden Curve Command
Instruction content	**EE + 81 + 0B + control ID + hide switch + FF FC FF FF**
An example of a directive	**EE** 81 0B 00 02 00 01 00 01 **FF FC FF FF**
Instruction example description	EE represents the frame header
	81 Master code 0 B is a slave code
	00 02 indicates that the ID of the control is 2 00 01 indicates that the curve with ID 1 is hidden 00 01 indicates that hiding is enabled
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**None**

Precautions	If there is an address (assuming 0x0001), the data format is: EE 00 01 81 0B 00 02 00 01 00 01 FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 81 0B 00 02 00 01 00 01 A7 DA FF FC FF FF, where A7 DA is the value of CRC16. When counting CRC16, the frame header, frame trailer, and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.33 Set Value Command for Curve

This command is used to set the corresponding data point inside the curve. Note that the content modified by this command will be lost after power failure. If you want to save the modifications to the control, you can use the Save Control Properties command or the user function. The user. Save _ gui (1) is stored.

Type of instruction	Set Value Command for Curve

Instruction content	**EE + 81 + 0 C + Update curve ID + Update curve series ID + Update point ID + Number of updates n + Data 0 + Data 1 ... + data n + FF FC FF FF**
An example of a directive	**EE** 81 0C 00 04 00 01 00 01 00 02 00 0A 00 14 **FF FC FF FF**
Instruction example description	EE represents the frame header
	81 Master code 0 C is a slave code
	00 04 indicates that the ID of the curve is 4 00 01 indicates that the series inside the curve is specified 1 00 01 indicates that the point with ID 1 is started 00 02 indicates that 2 points are updated 00 0 A 00 14 is 2 data 10 and 20
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**None**
Precautions	If there is an address (assuming 0x0001), the data format is: EE 00 01 81 0C 00 04 00 01 00 01 00 02 00 0A 00 14 FF FC FF FF
	If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs attached to the 485 bus

If there is an address and CRC check is performed, the data format is EE 00 01 81 0C 00 04 00 01 00 01 00 02 00 0A 00 14 E1 6A FF FC FF FF, where E1 6A is the value of CRC16. The frame header, frame trailer and 16-bit address are not included, and the counting method of CRC16 is shown in the appendix.

3.2.34 Hide Current Page Command

This directive hides the page that is currently being displayed

Type of instruction	Set hide current page directive
Instruction content	**EE + 81 + 0D + enable hidden switch + FF FC FF FF**
An example of a directive	**EE** 81 0D 00 01 **FF FC FF FF**
Instruction example description	EE represents the frame header
	81 Master code 0 D is a slave code
	00 01 Enable Hide
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**None**
	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 81 0D 00 01 FF FC FF FF

Precautions	If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 81 0D 00 01 1B 78 FF FC FF FF, where 1B 78 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.35 Set pointer value instruction of METER

This instruction is used to set the pointer value of the code table.

Type of instruction	Instruction for setting pointer value of code table
Instruction content	**EE + 81 + 0E + update code table ID + Val + state + FF FC FF FF**
An example of a directive	**EE** 81 0E 00 04 00 0A 00 01 **FF FC FF FF**
	EE represents the frame header

Instruction example description	81 Master code 0E is a slave code
	00 04 indicates that the ID of the code table is 4 00 0a is the value to be updated 00 01 indicates which pointer, and the number of the pointer starts from 0
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**None**
Precautions	If there is an address (assuming 0x0001), the data format is: EE 00 01 81 0E 00 04 00 01 00 0A FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 81 0E 00 04 00 01 00 0A 60 46 FF FC FF FF, where 60 46 is the value of CRC16. When counting CRC16, the frame header, frame trailer, and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.36 Set the touch enable command of the control

This command is used to set whether the control can receive the touch action

of the user.

Type of instruction	Set the touch enable command for the control
Instruction content	**EE + 81 + 0F + control ID + enable byte + FF FC FF FF**
An example of a directive	**EE** 81 0F 00 04 00 01 **FF FC FF FF**
Instruction example description	EE represents the frame header
	81 Master code 0 F is a slave code
	00 04 indicates that the control ID is 4 00 01 indicates that the touch function of the control is enabled
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**None**
Precautions	If there is an address (assuming 0x0001), the data format is EE 00 01 81 0F 00 04 00 01 FF FC FF FF
	If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will

	be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 81 0F 00 04 00 01 0A CA FF FC FF FF, where 0A CA is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.37 Set Touch Control Status Command

This command is used to simulate the touch control state when there is no touch screen. It is used to simulate the touch screen pressing and lifting action.

Type of instruction	Set touch control state command
Instruction content	**EE + 81 + 10 + control ID + status byte + FF FC FF FF**
An example of a directive	**EE** 81 10 00 04 00 01 **FF FC FF FF**
Instruction example description	EE represents the frame header
	Master instruction code 10 is a slave instruction code
	00 04 indicates that the control ID is 4 00 01 causes the touch control to press 00 00 to indicate lifting.

	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**None**
Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 81 10 00 04 00 01 FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 81 10 00 04 00 01 C8 5F FF FC FF FF, where C8 5F is the value of CRC16. When counting CRC16, the frame header, frame trailer, and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.38 Self-increasing menu cursor command

When there is no external touch screen, this command will increase the value of the cursor of the menu control of the screen, which is used to move the current cursor position.

Type of instruction	Set cursor increment command
Instruction content	**EE+81+11+FF FC FF FF**
An example of a directive	**EE** 81 11 **FF FC FF FF**
Instruction example description	EE represents the frame header
	Master instruction code 11 is a slave instruction code
	No parameters
	FF FC FF FF indicates the end of frame
XFD Feedback Content	**None**
Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 81 11 FF FC FF FF
	If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 81 11 EC A1 FF FC FF FF, where EC A1 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting

method of CRC16.

3.2.39 Auto Subtract Menu Cursor Command

When there is no external touch screen, this command will reduce the value of

the cursor of the menu control of the screen to move the position of the current

cursor.

Type of instruction	Set Cursor Decrement Command
Instruction content	**EE+81+12+FF FC FF FF**
An example of a directive	**EE** 81 12 **FF FC FF FF**
Instruction example description	EE represents the frame header
	Master instruction code 12 is a slave instruction code
	No parameters
	FF FC FF FF indicates the end of frame
XFD Feedback Content	**None**
Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 81 12 FF FC FF FF
	If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will

	be executed by all XFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 81 12 ED E1 FF FC FF FF, where ED E1 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.40 Set Next Value Command for Curve

This command is used to set the next value of the curve, continuously assigning values to the curve, allowing the value of the curve to move from right to left.

Type of instruction	Set the next value of the curve command
Instruction content	**EE + 81 + 13 + update curve ID + update curve family ID + updated data H + updated data L + FF FC FF FF**
An example of a directive	**EE** 81 13 00 04 00 01 00 02 **FF FC FF FF**
Instruction example	EE represents the frame header
	Master instruction code 13 is a slave instruction code

description	00 04 indicates that the ID of the curve is 4 00 01 indicates that the series 1 000 02 in the specified curve is the data to be updated
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**None**
Precautions	If there is an address (assuming 0x0001), the data format is EE 00 01 81 13 00 04 00 01 00 02 FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 81 13 00 04 00 01 00 02 A7 8A FF FC FF FF, where A7 8A is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.41 Get the X coordinate command of the control

This directive is used to get the X coordinate of the control dynamically

Type of instruction	Gets the X coordinate instruction for the control
Instruction content	**EE + 82 + 01 + Control ID + FF FC FF FF**
An example of a directive	**EE** 82 01 00 06 **FF FC FF FF**
Instruction example description	EE represents the frame header
	82 Master code 01 is a slave code
	00 06 indicates that the ID of the acquisition control is 6.
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	EE 82 01 00 06 00 0a FF FC FF FF 00 0a means that the x-coordinate of control 6 is 10
Precautions	If there is an address (assuming the address is 0x0001), the data format is: EE 00 01 82 01 00 06 FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus

	If there is an address and CRC check is performed, the data format is EE 00 01 82 01 00 06 9E F9 FF FC FF FF, where 9E F9 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.42 Get the Y coordinate command of the control

This directive is used to get the Y coordinate of the control dynamically

Type of instruction	Gets the Y coordinate instruction for the control
Instruction content	EE + 82 + 02 + Control ID + FF FC FF FF
An example of a directive	**EE** 82 02 00 07 **FF FC FF FF**
Instruction example description	EE represents the frame header
	82 Master instruction code 02 is a slave instruction code
	00 07 indicates that the ID of the acquisition control is 7.
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	EE 82 02 00 07 01 3F FF FC FF FF 01 3F indicates that the y-coordinate of control 7 is 319

Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 82 02 00 07 FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 82 02 00 07 5E C8 FF FC FF FF, where 5E C8 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.43 Get the width W command of the control

This instruction is used to dynamically get the width W of the control

Type of instruction	Get the width of the control W instruction
Instruction content	**EE + 82 + 04 + Control ID + FF FC FF FF**
An example of a directive	**EE** 82 04 00 00 **FF FC FF FF**

	FileName	**XFD/SFD Application Documentation**		
Document number	SP004	Edition	I	Page number

Instruction example description	EE represents the frame header	
	Master instruction code 04 is a slave instruction code	
	00 00 indicates that the ID of the acquisition control is 0.	
	FF FC FF FF indicates the end of frame	
XFD/SFD Feedback	EE 82 04 00 10 FF FC FF 00 means control 0, 00 10 means width 16	
Precautions	If there is an address (assuming 0x0001), the data format is EE 00 01 82 03 00 00 FF FC FF FF	
	If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus	
	If there is an address and CRC check is performed, the data format is EE 00 01 82 03 00 00 5C D8 FF FC FF FF, where 5C D8 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.	

	FileName	XFD/SFD Application Documentation		
Document number	SP004	Edition	I	Page number	No 82 Pages in total 131 页

3.2.44 Get Height H Command of Control

This instruction is used to dynamically get the height H of the control

Type of instruction	Gets the height H instruction of the control.
Instruction content	**EE + 82 + 04 + Control ID + FF FC FF FF**
An example of a directive	**EE** 82 04 00 07 **FF FC FF FF**
Instruction example description	EE represents the frame header
	Master instruction code 04 is a slave instruction code
	00 07 indicates that the ID of the acquisition control is 7.
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	EE 82 04 00 07 00 96 FF FC FF FF 00 00 means control 7, 00 96 means height 150
Precautions	If there is an address (assuming the address is 0x0001), the data format is EE 00 01 82 04 00 07 FF FC FF FF
	If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data

format is EE 00 01 82 04 00 07 5F 28 FF FC FF FF, where 5F

28 is the value of CRC16. When counting CRC16, the frame

header, frame trailer and 16-bit address are not included.

See the appendix for the counting method of CRC16.

3.2.45 Get Text Content Instruction for Control

This instruction is used to dynamically obtain the text content of the control,

and the ID must be specified when obtaining.

Type of instruction	Gets the text content directive for the control
Instruction content	**EE + 82 + 05 + Control ID + FF FC FF FF**
An example of a directive	**EE** 82 05 00 0C **FF FC FF FF**
Instruction example description	EE represents the frame header
	Master instruction code 05 is a slave instruction code
	00 0C indicates that the ID of the acquisition control is 12
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	EE 82 05 00 0C E8 8E B7 E5 8F 96 E5 A4 B1 E8 B4 A5 00 FF FC FF FF 00 0C represents control 12, E8 8E B7 E5 8F

	96 E5 A4 B1 E8 B4 A5 indicates that the string 'Fail to get' 00 is the end character
Precautions	If there is an address (assuming 0x0001), the data format is EE 00 01 82 05 00 0C FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 82 05 00 0C 58 38 FF FC FF FF, where 58 38 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.46 Get Control Background Color 1 Instruction

This instruction is used to dynamically obtain the background color 1 of the control. When obtaining, you need to specify two parameters: ID and state.

Type of instruction	Gets the background color 1 instruction for the

	control.
Instruction content	EE + 82 + 06 + Control ID + Status + FF FC FF FF
An example of a directive	**EE** 82 06 00 07 00 00 **FF FC FF FF**
Instruction example description	EE represents the frame header
	Master instruction code 06 is a slave instruction code
	00 07 indicates that the ID of the acquisition control is 7.
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	EE 82 06 00 07 00 00 FF FF FF FF FF FC FF FF 00 07 for control 7, 00 00 for background color 1 in default state FF FF FF FF for 24-bit background color is all white
Precautions	If there is an address (assuming 0x0001), the data format is EE 00 01 82 06 00 07 00 00 FF FC FF FF
	If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the

data format is EE 00 01 82 06 00 07 00 00 F8 27 FF FC FF FF, where F8 27 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.47 Get control's background color 2 instruction

This instruction is used to dynamically obtain the background color 2 of the control. Two parameters, ID and state, need to be specified when obtaining.

Type of instruction	Get the background color of the control 2 instruction
Instruction content	EE + 82 + 07 + Control ID + Status + FF FC FF FF
An example of a directive	**EE** 82 07 00 07 00 00 **FF FC FF FF**
Instruction example description	EE represents the frame header
	82 Master code 07 is a slave code
	00 07 indicates that the ID of the acquisition control is 7. 00 00 indicates the default state
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	EE 82 07 00 07 00 FF FC FF 00 07 indicates the control 7,

	00 indicates the background color in the default state 1 00 indicates the background color of the 24-bit is all black
Precautions	If there is an address (assuming 0x0001), the data format is EE 00 01 82 07 00 07 00 00 FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 82 07 00 07 00 00 38 1A FF FC FF FF, where 38 1A is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.48 Get control's foreground color command

This instruction is used to dynamically obtain the foreground color of the

control, and the ID needs to be specified when obtaining.

Type of instruction	Gets the foreground color instruction for the control.
Instruction content	**EE + 82 + 08 + Control ID + FF FC FF FF**
An example of a directive	**EE** 82 08 00 07 **FF FC FF FF**
Instruction example description	EE represents the frame header
	Master instruction code 08 is a slave instruction code
	00 07 indicates that the ID of the acquisition control is 7.
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	EE 82 08 00 07 00 FF FC FF 00 07 indicates that the control 7 00 indicates that the background color of the 24-bit is all black
Precautions	If there is an address (assuming the address is 0x0001), the data format is: EE 00 01 82 08 00 07 FF FC FF FF
	If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 82 08 00 07 5C E8 FF FC FF FF, where 5C E8 is the value of CRC16. When counting

	CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.49 Get Value Command for Control

This instruction is used to dynamically obtain the value of the control (generally refers to prog/slider/arc/IMG), and the ID must be specified when obtaining.

Type of instruction	Gets the value instruction for the control.
Instruction content	**EE + 82 + 09 + Control ID + FF FC FF FF**
An example of a directive	**EE** 82 09 00 01 **FF FC FF FF**
Instruction example description	EE represents the frame header
	Master instruction code 09 is a slave instruction code
	00 01 indicates that the ID of the acquisition control is 1.
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	EE 82 09 00 01 00 02 FF FC FF FF 00 01 indicates control 1, and 00 02 indicates that the obtained control value is 2
	If there is an address (assuming the address is 0x0001),

Precautions	the data format is EE 00 01 82 09 00 01 FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 82 09 00 01 9E 39 FF FC FF FF, where 9E 39 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

3.2.50 Command to get the value of the curve

This instruction is used to dynamically obtain the value of the curve by specifying the ID and quantity

Type of instruction	Command to get the value of a curve
Instruction content	EE + 82 + 0a + curve ID + series of curves + ID of points + number of reads + FF FC FF FF

An example of a directive	**EE** 82 0A 00 02 00 01 00 02 00 02 **FF FC FF FF**
Instruction example description	EE represents the frame header
	82 Master code 0 A is the slave code
	00 02 means to get the ID of the curve is 2 00 01 means to specify this 01 series 00 02 means to start from the point of 02 00 02 means to take 2 points
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	EE 82 0A 00 02 00 01 00 02 00 02 00 10 00 20 FF FC FF FF
	00 02 indicates that the control curve ID is 2, 00 01 indicates that this 01 series is specified
	00 02 means starting from the point of 02 00 02 means taking 2 points
	00 10 00 20 means that the values taken are 16 and 32, respectively
Precautions	If there is an address (assuming 0x0001), the data format is: EE 00 01 82 0A 00 02 00 01 00 02 00 02 FF FC FF FF
	If there is an address and the address is set to 0x0000,

	the data is broadcast data, and the broadcast data will be executed by all XFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 82 0A 00 02 00 01 00 02 00 02 84 96 FF FC FF FF, where 84 96 is the value of CRC16. When counting CRC16, the frame header, frame trailer, and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.51 Get METER pointer value instruction

This instruction is used to dynamically obtain the value of the METER pointer by specifying the ID and quantity

Type of instruction	Command to get the value of a curve
Instruction content	EE + 82 + 0B + code table ID + pointer ID + FF FC FF FF
An example of a directive	**EE** 82 0B 00 02 00 01 **FF FC FF FF**
Instruction example description	EE represents the frame header
	82 Master code 0 B is a slave code
	00 02 indicates that the ID for acquiring the code table is 2 00 01 indicates that the pointer with the number of

		1 is to be acquired, and the number of the pointer starts from 0
		FF FC FF FF indicates the end of frame
	XFD/SFD Feedback	**EE** 82 0B 00 02 00 01 00 20 **FF FC FF FF** 00 02 indicates that the control ID of the code table is 2, and 00 01 indicates that the pointer with the number 1 is specified. 00 20 means that the value obtained is 32
	Precautions	If there is an address (assuming 0x0001), the data format is: EE 00 82 0B 00 02 00 01 00 20 FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
		If there is an address and CRC check is performed, the data format is EE 00 01 82 0B 00 02 00 01 00 20 AA 5A FF FC FF FF, where AA 5A is the value of CRC16. When counting CRC16, the frame header, frame trailer, and 16-bit address are not included. The counting method

of CRC16 is shown in the appendix

3.2.52 Obtain control touch enable command

This instruction is used to dynamically obtain the touch-enabled state of the control

Type of instruction	Acquire control touch enable instruction
Instruction content	**EE + 82 + 0C + Control ID + FF FC FF FF**
An example of a directive	**EE** 82 0C 00 02 **FF FC FF FF**
Instruction example description	EE represents the frame header
	82 Master code 0 C is a slave code
	00 02 indicates to acquire the touch enabled state in which the ID of the control is 2
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**EE** 82 0C 00 02 00 01 **FF FC FF FF** 00 02 indicates that the control ID is 2, and 00 01 indicates that the current control is in the touch-enabled state, and the user can operate the control through the touch screen

Precautions	If there is an address (assuming 0x0001), the data format is EE 00 01 82 0C 00 02 00 01 FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 82 0C 00 02 00 01 38 6E FF FC FF FF, where 38 6E is the value of CRC16. When counting CRC16, the frame header, frame trailer, and 16-bit address are not included. The counting method of CRC16 is shown in the appendix

3.2.53 Acquire Touch Control Status Command

This instruction is used to dynamically get the current state of the touch control

Type of instruction	Acquire touch control state instruction
Instruction content	**EE + 82 + 0D + Control ID + FF FC FF FF**
An example of a directive	**EE** 82 0D 00 02 **FF FC FF FF**

		EE represents the frame header
Instruction example description		82 Master code 0 D is a slave code
		00 02 indicates to obtain the state of the touch control whose control ID is 2
		FF FC FF FF indicates the end of frame
XFD/SFD Feedback		**EE** 82 0D 00 02 00 01 **FF FC FF FF**
		00 02 indicates that the control ID is 2, 00 01 indicates the state of the current control, 00 00 indicates the lifted state, and 00 01 indicates pressed
Precautions		If there is an address (assuming 0x0001), the data format is EE 00 01 82 0D 00 02 00 01 FF FC FF FF
		If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
		If there is an address and CRC check is performed, the data format is EE 00 01 82 0D 00 02 00 01 F8 53 FF FC FF FF, where F8 53 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address

are not included. The counting method of CRC16 is shown in the appendix

3.2.54 Command for obtaining the current menu cursor value

When there is no external touch screen, this command is used to obtain the menu cursor value on the current interface.

Type of instruction	Get Current Menu Cursor Value Command
Instruction content	**EE+**82+0E+**FF FC FF FF**
An example of a directive	**EE** 82 0E **FF FC FF FF**
Instruction example description	EE represents the frame header
	The master code 0 E is the slave code
	No parameters
	FF FC FF FF indicates the end of frame
XFD/SFD Feedback	**EE** 82 0E 00 01 **FF FC FF FF** 00 01 indicates that the current cursor is 1, that is, the position of the cursor is in the second control, and the position of the cursor starts from 0
	If there is an address (assuming 0x0001), the data

Precautions	format is EE 00 01 82 0E 00 01 FF FC FF FF If there is an address and the address is set to 0x0000, the data is broadcast data, and the broadcast data will be executed by all XFDs/SFDs attached to the 485 bus
	If there is an address and CRC check is performed, the data format is EE 00 01 82 0E 00 01 5F 88 FF FC FF FF, where 5F 88 is the value of CRC16. When counting CRC16, the frame header, frame trailer and 16-bit address are not included. See the appendix for the counting method of CRC16.

④User functions for XFD/SFD:

4.1 User function description:

4.1.1 User functions are the interface to control and access XFD/SFD. User functions are designed as python statements and can be executed in python scripts.

4.1.2 The use of user functions needs to be inserted into the user library

4.2 Detailed explanation of user function

Function name	Function content	Function description	Sample code	Illustration of examples
Reboot Instruction	user.restart()	This instruction is used to restart the XFD/SFD, The XFD/SFD can be synchronized with the	user.restart()	Restart the module

		host through a reset command when the host is restarted in an unexpected situation.		
Set the baud rate	user.set_bps(rate)	Set the communication baud rate of the module Rate: 115200/38400/19200/9600	user.set_bps(9600)	Set the baud rate of the module to 9600
Get Baud rate	user.get_bps()	The return values are: 115200/38400/19200/9600	rate=user.get_bps()	The value of the baud rate that will be obtained Save to rate
Set the backlight brightness	user.set_bl(n)	And n is the brightness of the backlight, ranging from 0 to 255, with 0 being the darkest	user.set_bl(100)	Set the brightness of the backlight to 100

Obtaining backlight brightness	user.get_bl()	The return value is the brightness value of the backlight	val=user.get_bl()	The obtained brightness value of the backlight Save to Val
Sets the X coordinate	user.set_x(id,val)	ID is the ID of the control and Val is the x-coordinate of the control	user.set_x(1,20)	Set the X coordinate of control 1 to 20
Get the x-coordinate	user.get_x(id)	Specifies the control ID to get and returns the coordinate value of X	val=user. get_x(1)	Save the obtained X coordinate of control 1 to Val
Sets the y coordinate	user.set_y(id,val)	ID is the ID of the control and Val is the y coordinate of the control	user.set_y(1,20)	Set the y coordinate of control 1 to 20
Get the y coordinate	val=user.get_y(id)	Specifies the control ID to get and returns the coordinate value of X	val=user. get_y(1)	Save the acquired y coordinate of

				control 1 to Val
Set Control W Width	user.set_width(id,val)	ID is the ID of the control and Val is the width of the control	user. set_width (1,20)	Set the width of control 1 to 20
Get the W width	user.get_width(id)	Specifies the control ID to get. The return value is the width value	val=user.get_width(1)	Save width of control 1 into Val
Set the H height	user.set_height(id,val)	ID is the ID of the control and Val is the height of the control	user. set_height (1,20)	Set the height of control 1 to 20
Gets the height of the control H	val=user.get_height(id)	Specifies the control ID to get, and the return value is the height value	val=user. get_height(1)	Save the height of control 1 into Val
Sets the text value	user.set_text(id,*str)	Specify the control ID and the text content str to be modified, which are generally specified as label/textbox/qrcode.	User. Set _ text (1, 'OK')	Change the text of control 1 to the word'Normal '.

Gets the text of the control	user.get_text(id)	Specifying the control ID to obtain the text content is generally specified as label/textbox/qrcode.	text=user. get_text(1)	Saves the contents of the text of control 1 to the text pointer
Sets the value of the control	user.set_val(id,val)	Specify the ID to modify the value of the control, which is generally specified for the prog/slider/arc/IMG controls	user. set_val(3,100)	Change the value of control 3 to 100
Gets the value of the control	user.get_val(id)	Gets the value of the control by specifying the control ID	val=user. get_val(1)	Save the obtained value of control 1 to Val
Set the value of the buzzer	user.set_buzzer(time)	Specifies the time of the buzzer time in milliseconds. When the value is 0, the buzzer is disabled	user. set_buzzer(100)	Set the beep time of the beeper to 100 milliseconds
Get the value	user.get_buzzer()	Read the beep time		Read the beep

of the buzzer		parameter of the buzzer	val=user. get_buzzer()	time of the beeper into the Val temporary variable
Set the timer	user.set_timer (id, one_time, time, en)	XFD/SFD has 8 general-purpose timers with timer ID from 0 to 7. To start the timer, you need to specify timer ID, timer type option, timer time and enable switch.	user. set_timer(0,1,100,1)	Enabling Timer 0 is a one-time trigger type, and its trigger time is 100ms.
~~Update qrcode~~	~~user.qrcode_update(id,str)~~	~~To dynamically update the QR Code, specify the QR Code ID and the updated content str~~	~~— User. Qrcode_update (0, "I am Chinese")~~	~~Update the display content of QR Code 0 to "I am Chinese"~~
Set Ivar	user.set_ivar(id,val)	Specify ID to set the value of integer variable inside XFD to Val (ID is 0	user.set_ivar(0,20)	Set the value of IVAR [0] to 20

		~ 31)		
Get Ivar	user.get_ivar(id)	Specify ID to get the value of IVAR [ID]	val=user. get_ivar(1)	Get the value of IVAR [1] and save it in the Val temporary variable
Set svar	user.set_svar(id,str)	The value of the text variable set by the specified ID is str	User. Set _ svar (0, "Shenzhen, Guangdong")	Let the content of SVAR [0] be: "Guangdong Shenzhen"
Get svar	user.get_svar(id)	Specify ID to get the text content of SVAR [ID]	str=user. get_svar(9)	Get the contents of SVAR [9] and save it to the str pointer
Jump to the page	user.set_page(id,delay)	Jump to the ID screen, where delay is to wait for delay milliseconds before jumping	user.set_page(1,10)	Jump to page 1 after a delay of 10 ms

Get the current page	user.get_page()	Gets the ID value of the current page	id=user.get_page()	Get the value of the current page and save it to the ID temporary variable
Set the count value of the timer	set_timer_times(id,cnt)	Specify the ID of the timer. This command can be used to set the number of times CNT the timer is called	user. set_timer_times (1,10)	Set the count value of timer 1 to 10
Obtain the count value of the timer	get_timer_times(id)	Specify the ID of the timer. This command can be used to set the number of times CNT the timer is called	cnt=user.get_timer_times (1)	Get the count value of Timer1 and save it to the CNT temporary variable

Address to enable 485	addr_enable(en)	Set the address mode of XFD485	user.addr_enable(1)	Enable the address mode of XFD485 serial port
Set the address of 485	set_addr(addr)	Set the address of XFD 485 serial port	user.set_addr(1)	Set the address of the XFD 485 serial port to 1
Enabling data transmission CRC16 check	set_crc16(en)	Set the CRC16 verification function of XFD 485	user.set_crc16 (1)	Set CRC16 verification of XFD 485 serial port
Hide the control	set_hidden (id,all,en)	Specifies the control ID, all is all hidden, and en is the enable switch	user.set_ hidden (1,0,1)	Hide control 1
Set the background color 1	set_bg_color (id,state,color)	Specifies the control ID, state is the state of the control, and color is the color value	set_bg_color(1,0,255)	Sets the color in control 1state 0 to blue

Get background color 1	get_bg_color (id,state)	Specifies the control ID and state, and returns the value of the background color 1	color=user. get_bg_color (1,0)	Gets the background color 1 of control 1 in state 0
Set the background color 2	set_bg_color2 (id,state,color)	Specifies the control ID, state is the state of the control, and color is the color value	user.set_bg_color2(1, 0,255)	Sets the color in control 1state 0 to blue
Set the background color 2	get_bg_color2 (id,state)	Specifies the control ID and state, and returns the value of the background color 1	color=user.get_bg_color2 (1,0)	Gets the background color 1 of control 1 in state 0
Sets the font color	set_font_color (id,color)	Specifies the control ID, and color is the color value	user.set_font_color(1 ,255)	Set the color of control 1 to blue
Gets the font	get_font_color	Specifies the control ID		Gets the font

	FileName	**XFD/SFD Application Documentation**		
Document number	SP004	Edition	I	Page number

No 109 Pages in total 131 页

color	(id)	and returns the value of the font color	color=user.get_font_color (2)	color of control 2
Sets the opacity	set_opa (id,state,val)	Sets the transparency of the control by specifying the control ID, state, and settings	user.set_opa(2,0,0)	Set control 2 to be fully transparent in 0 state
Gets the opacity	get_opa(id,state)	Returns the transparency of the control by specifying the control ID and state	val=user.get_opa(2,0)	Obtain the transparency of control 2 in state 0 and save it to the temporary variable Val
Set the idle waiting time	set_delay(ms)	A specified wait time ms, For null, etc	user.set_delay(100)	Set wait for 100ms
Set the pointer value of meter	set_mt_val (wid,val,state)	Sets the Val of the state of the pointer of the Meter control wid, where	user.set_mt_val(1,100,0)	Set the value of pointer 0 in meter control 1

		state is used to specify which pointer, and its value starts at 0		to 100
Get the pointer value of meter	get_mt_val (wid,state)	Gets the Val of the state of the pointer of the Meter control wid, where state is used to specify which pointer, and its value starts at 0	val=user.get_mt_val(1,0)	Get the value of the pointer 0 in the meter control 1 and store it in a temporary variable
~~Set the clock hour~~	~~set_clock_hr(id,val)~~	~~Specifies the clock control ID and the value of the settings~~	~~──~~ ~~user. set_clock_hr (1,20)~~	~~Set the clock value in clock control 1 to 20~~
~~Set the clock minute~~	~~set_clock_min(id,val)~~	~~Specifies the clock control ID and the value of the settings~~	~~──~~ ~~user. set_clock_min (1,30)~~	~~Set the minute value in clock control 1 to 30~~
~~Set the clock seconds~~	~~set_clock_sec(id,val)~~	~~Specifies the clock control ID and the value~~	~~──~~ ~~user. set_clock_sec~~	~~Set the seconds value in clock~~

		of the settings	(1, 0)	control 1 to 0
Save the contents of the gui	save_gui(en)	When en = 1, save the changes to the pack	user. save_gui(1)	
The chart content is updated	chart_update (wid,ch,id,val,enable)	Modify the value of the point ID in the ch channel in the chart control of the wid as Val, and enable as the enable switch	— user.chart_update (1,1,1,20,1)	Change the value of point 1 in channel 1 in the chart control 1 to 20
Set the value of the curves	set_ct_val(wid,ch,id,val,en)	Wid is used to specify the control number of the curve, ch refers to which curve (the number starts from 0), ID is a point in the curve, Val is the value of a point to be updated, and when en = 1, it is	user.set_ct_val(1,1,1,20,1)	Update the value of point 1 in curve number 1 of control 1 to 20

				used to enable updating.		
Get the value of the curve	get_ct_val(wid,ch,id)	Wid specifies the control number of the curve, ch which curve (numbering starts at 0), and ID specifies a point in the curve to get		val=get_ct_val(1,0,1)	Obtain the value of point 1 in curve 0 in curve control 1 and store it in the temporary variable Val
Sets the hidden curve	chart_hidden (wid,ch,en)	Hide the ch channel in the curve control of wid (the channel number starts from 0), and en is the enable switch		user. chart_hidden(1,1,1)	Hide channel 1 in chart control 1
Set up hibernation	set_sleep (time,bl_val)	Specify the sleep time (in seconds) and the backlight brightness after sleep. If the time value is 0, the sleep		user. set_sleep(5,10)	Start-up sleep time is 5 seconds, and the backlight brightness after

		mode is not entered.		sleep is 10
Gets the sleep time	get_sleep_time()	Gets the sleep time of the XFD	val=user.get_sleep_time()	Save the sleep time to the temporary variable Val
Obtain the backlight brightness when sleeping	get_sleep_bl()	Obtain the backlight brightness value when the XFD is sleeping	val=user. get_sleep_bl()	Saves the specified backlight brightness at sleep time to a temporary variable Val
Animate	set_gif (wid, start_id, end_id, delay, loop_times,	Wid is the IMG control ID that performs the animation Start _ ID is the start ID of the picture End _ ID is the end ID of	user.set_gif (2,0,10,100,10,0,0,3,1)	Start the animation function of page0 control 2. There are 11 specified

	infinite ,page, next_page, en)	the graph Delay is the time delay in milliseconds to execute the animation Loop _ times specifies the number of times to animate Infinite Is an infinite loop Page Specifies the page on which the animation is executed Next _ page The page to jump to after the number of times the loop is executed En is the animation enable switch		pictures from 0 to 10. The animation delay is 100ms. Execute the animation for 10 times and then exit and jump to page 3.
Control	report_pos(en)	The button control and		Enable the

automatic reporting coordinate data switch			panel control can report the position coordinate information when they are touched. This function can be realized by setting en = 1. If you want to turn off the automatic reporting of coordinates, en = 0 is enough.	user.report_pos (1)	coordinate reporting function of the control
Escalate data manually	report_data(header,index,data)		This user function is used to send data to the host computer through the serial port, where header is the data header, which can be any data such as 0 XAA, 0 XBB, etc., index is the index value, which	user.report_data(0xAA,1,0x01020304)	Send: 0XAA + 1 + 01 + 02 + 03 + 04 + FF FC FF FF to the upper computer. Note that if the address and CRC are

		can be any value from 0 to 255, and data is 32-bit integer data, which is any data of four bytes.		included, the data will be sent together with the corresponding address and verification content.

Set to hide the current page	set_hidden_current_ page(en)	This function can be used to close the current page display by setting en = 1	user.set_hidden_curr ent_page(1)	Enables hiding the current page
Obtain the	get_ver()	This command has no		Get the current

firmware version number		parameters, and the returned string "XXX _ YY" XXX is the version content, and YY is the version number. For example, the obtained content is "FWxx _ SHMI _ ST _ 01", the version content is "FWxx _ SHMI _ ST" ", and the version number is" 01 ".	ver=user.get_ver()	version information to the temporary variable ver
Animate horizontally	set_anim_hdir(id,start,end,time,mode)	ID is the corresponding control ID, start is the start coordinate of X, end is the end coordinate of X, time is the duration of the animation in milliseconds, and there	user.set_anim_hdir(1, 10,100,100,5)	Control 1 implements the horizontal movement from 10 to 100, the movement time is 100ms, and

		are 7 modes in total: 0: Same speed from beginning to end 1: Start with low speed 2: End with low speed 3: Start with low speed End with low speed 4: Exceed the end value 5: End bounce some 6 Last step change		the movement mode is rebound after reaching the end point
Animate vertically	set_anim_vdir(id,start ,end,time,mode)	ID is the corresponding control ID, start is the start coordinate of y, end is the end coordinate of y, time is the duration of the animation in milliseconds, and there are 7 modes in total: 0: Same speed from	user.set_anim_vdir(1, 10,100,100,5)	Control 1 realizes the longitudinal movement from 10 to 100, the movement time is 100ms, and the movement mode is

		beginning to end 1: Start with low speed 2: End with low speed 3: Start with low speed End with low speed 4: Exceed the end value 5: End bounce some 6 Last step change		rebounding after reaching the end point
Animate in any direction	set_anim_any(id,startx,endx,starty,endy,time,mode)	ID is the corresponding control ID, startx is the start coordinate of X, endx is the end coordinate of X, starty is the start coordinate of y, Endy is the end coordinate of y, time is the duration of the animation, the unit is milliseconds, and there	user.set_anim_any(4,-10,33,400,210,400,5)	Control 4 implements the movement of the lower left corner of X from -10 to 33 and y from 400 to 210 for 100ms in the form of a bounce at the end

		are 7 modes in total: 0: Same speed from beginning to end 1: Start with low speed 2: End with low speed 3: Start with low speed End with low speed 4: Exceed the end value 5: End bounce some 6 Last step change		
Set the value of user IO port	set_usr_pin(id,val)	XFD provides 5 user-used GPIO ports (SFD has no open GPIO port at present), and the corresponding IO ports (5 ports are set at present, and the ID is from 0 to 4) can be set as 1 or 0 through the set _	user.set_usr_pin(0,1)	Set the value of user gpio port 0 to 1

		usr _ pin.		
Obtain the value of user IO port	get_usr_pin(id)	XFD provides 5 user GPIO ports (SFD has no user GPIO port that can be opened temporarily), and the values of the corresponding IO ports (currently set to 5 ports, with ID from 0 to 4) can be obtained through the get _ usr _ pin.	val=user.get_usr_pin (0)	Obtain the value of user gpio port 0 and save it to the temporary variable Val
Serial port sending character string str	report_str(str)	Send the string str through the communication port	From gb2312 import Gb2312 gcode = Gb2312() R = gcode. STRs ('I love Shenzhen!') user.report_str(r)	Convert 'I love Shenzhen!' Of UTF-8 into Gb2312 code and send it out through serial port

Control value autoincrement	inc_val(id,val)	Increments the value of the control ID by the step size of Val	user.inc_val(1,1)	Increments the value of the control by 1
Control values are decremented	dec_val(id,val)	Subtracts the value of the control ID from the step size of Val	user.dec_val(1,1)	Subtracts 1 from the value of the control
Meter value self-increasing	inc_mt_val(id,val,state)	The pointer value of the corresponding state in the meter numbered ID is automatically incremented by the value of Val	user.inc_mt_val(1,1,0)	Increments the value of pointer 0 in the meter control with ID 1 by 1
Meter Value Subtract	dec_mt_val(id,val,state)	Subtract the value of Val from the pointer value of the corresponding state in the meter numbered ID	user.dec_mt_val(1,1,0)	Subtracts 1 from the value of pointer 0 in the meter control with ID 1
Backlight	inc_bl(val)	Increment the value of	user.inc_bl(1)	Increment the

self-increasing		the backlight by the value of Val		value of the backlight by 1
Backlight subtraction	dec_bl(val)	Subtract the value of Val from the value of the backlight	user.dec_bl(1)	Subtracts 1 from the value of the backlight
The label value is self-increasing	inc_label_val(id,val,limit)	Increment the value of the numeric string set in the label by the value of Val, where limit is the upper limit of the increment	user.inc_label_val(1,1,100)	Increments the value of the numeric string in label control 1 by 1. When it reaches 100, it is no longer incremented
The label value subtracts from itself	dec_label_val(id,val,limit)	Decrement the value of the numeric string set in the tag by the value of Val, where limit is the lower limit of the	user.dec_label_val(1,1,0)	Subtracts 1 from the value of the numeric string in label control 1. When it is

		self-subtraction		incremented to 0, it is no longer decremented
Control touch enable	touch_enable(id,en)	When en is 1, the touch function of the control numbered ID is enabled	user.touch_enable(1, 1)	Enable the touch function of control 1
Set menu cursor information	set_cursor_info(width ,color,opa,num,"map ")	When there is no touch screen, this user function can be used to set the number of highlighted controls in the interface, the width, color, opacity and other information of the cursor. Generally, this information is loaded in the page entry event	user.set_cursor_info(2,16777215,255,2,"1 2/13")	Set the controls 12 and 13 in the page to the range selected by the cursor. When 12 or 13 is selected, the width of the cursor is 2, the color is white, and the opacity is 255

Toggle Page Load Direction Function	swi_page_load_dir(dir)	It is used to dynamically modify the direction of page loading. There are 8 dir directions from 0 to 7. For details, please refer to the direction option in SHMI loading mode.	user.swi_page_load_dir(0)	Change the direction of page loading to left-to-right
Increment Menu Cursor Value	set_cursor_inc()	This function is used to change the value of the menu cursor (the highlighted control in the outer border) by setting the menu cursor information when there is no touch screen	user.set_cursor_inc()	Increases the ID number of the highlighted control on the current pack
Subtract the value of the menu cursor	set_cursor_dec()	This function is used to change the value of the menu cursor (the	user.set_cursor_dec()	Decreases the highlighted control ID

		highlighted control in the outer border) by setting the menu cursor information when there is no touch screen		number on the current page
Gets the value of the menu cursor	get_cursor()	This function gets the value of the current menu cursor.	Val=user.get_cursor()	Get the value of the current menu cursor and save it to the Val temporary variable
Sets the next value of the curves	set_ct_val2(wid,cid,val)	This function is used to set the next value of the curve. By setting this value, you can achieve the effect of rolling the curve cyclically.	user.set_ct_val2(1,0,20)	Set the next value of curve number 0 of curve control 1 to 20

Note: 1. Since SFD does not support the programming mode of custom script, the function of user. Get _ XXX will not be supported.

2. Due to the limitation of SFD resources, the function of set _ anim _ XX is not supported either.

(4). Control return data

When some controls are touched, they will send the touch status information to the user's master control to prompt the user to enter the next operation.

①, key controls

1.1 Composition of key control report data (without address and CRC)

Frame header 0xCC + type number of control (8 bits) + ID value of current page (16 bits) + X coordinate of control (16 bits) + Y coordinate of control (16 bits) + value of control (16 bits)

1.2 Example:

When key 7 is pressed: CC 06 00 01 00 07 00 E3 01 3F 00 00 FF FC FF FF

Where CC is the frame header

06 indicates that the control is a button control

00 01 indicates that the current page ID is 1

00 07 indicates that the control ID is 7

0 0 E3 X coordinate is 227

01 3 F y coordinate is 319

00 00 indicates that the key is pressed for the first time, and its initial value is 0.

Appendix:

1. Type coding table for the control:

```
+----+---------+-----------+
| #  | Widgets | 控件类型  |
+----+---------+-----------+
|  1 |   page  |     1     |
|  2 |    lb   |     2     |
|  3 |    tx   |     3     |
|  4 |   btn   |     6     |
|  5 |   prog  |     7     |
|  6 |    img  |     8     |
|  7 |  panel  |     a     |
|  8 |    mt   |     b     |
|  9 |    ct   |     c     |
| 10 |    sl   |     d     |
| 11 |    rl   |    11     |
| 12 |    qr   |    12     |
| 13 |    tb   |    13     |
| 14 |    cb   |    14     |
| 15 |    sw   |    15     |
| 16 |   arc   |    16     |
| 17 | spinner |    17     |
+----+---------+-----------+
```

2.CRC16 algorithm source code:

Data is the data pointer that participates in the CRC, n is the number of bytes that participate in the algorithm, and pcrc is the result pointer of the algorithm.

```c
void data_crc16(uint8_t *data,uint16_t n,uint16_t *pcrc)
{
    uint16_t i,j,carry_flag,a;
    for (i=0; i<n; i++)
    {
        *pcrc=*pcrc^data[i];
```

```
for (j=0; j<8; j++)

{

    a=*pcrc;

    carry_flag=a&0x0001;

    *pcrc=*pcrc>>1;

    if (carry_flag==1)

        *pcrc=*pcrc^0xa001;

}

}

}
```