

# XFD/SFD User Functions and Instruction Set V05

1. XFD module has built-in python script, which supports script programming. SFD only supports inserting user functions, not python programming
2. XFD supports python's basic logic operations, SFD does not
3. SFD cannot use the user function corresponding to the user. Get \_ XX because it does not support scripting
4. For performance reasons, SFD does not support user functions for set \_ anim \_ XXX.
5. List of user functions for XFD/SFD

Function name	Function content	Function description	Sample code	Illustration of examples
Restart command	<code>user.restart()</code>	This instruction is used to restart the module. If the host is restarted in an unexpected situation, it can be reset through the reset command. XFD and host synchronization	<code>user.restart()</code>	Restart the module
Set the baud rate	<code>user.set_bps(rate)</code>	Set the communication baud rate of the module Rate corresponds to the baud rate: 115200/38400/19200/9600	<code>user.set_bps(9600)</code>	Set the baud rate of the module to 9600
Get the baud rate	<code>user.get_bps()</code>	The return values are: 115200/38400/19200/9600	<code>rate=user.get_bps()</code>	The value of the baud rate that will be obtained To the rate temporary variable
Set the backlight brightness	<code>user.set_bl(n)</code>	Where n is the brightness of the backlight and ranges from 0 255, 0 is the darkest	<code>user.set_bl(100)</code>	Set the brightness of the backlight to a value of 100
Obtaining backlight brightness	<code>user.get_bl()</code>	The return value is the brightness value of the backlight	<code>val=user.get_bl()</code>	The obtained brightness value of the backlight Save to Val temporary variable
Sets the X coordinate	<code>user.set_x(id,val)</code>	ID is the control ID of the current page, Val is the x-coordinate of the control	<code>user.set_x(1,20)</code>	Set the X coordinate of control 1 to 20
Get the x-coordinate	<code>user.get_x(id)</code>	Specifies the control ID to get and returns the value Is the coordinate value of X	<code>val=user.get_x(1)</code>	Save the X coordinate of the obtained control 1 Into the Val temporary variable
Sets the y coordinate	<code>user.set_y(id,val)</code>	The ID is the ID of the control on the current page, Val is the y coordinate of the control	<code>user.set_y(1,20)</code>	Set the y coordinate of control 1 to 20
Get the y coordinate	<code>user.get_y(id)</code>	Specifies the control ID to get and returns the value Is the coordinate value of X	<code>val=user.get_y(1)</code>	Save the acquired y coordinate of control 1 Into Val
Set Control W Width	<code>user.set_width(id,val)</code>	ID is the control ID of the current page, Val is the width of the control	<code>User.set_width(1,20)</code>	Set the width of control 1 to 20
Get the W width	<code>user.get_width(id)</code>	Specifies the control ID to get and returns the value Is the width value	<code>val=user.get_width(1)</code>	Save width of control 1 into Val
Set the H height	<code>user.set_height(id,val)</code>	ID is the control ID of the current page, Val is the height of the control	<code>user.set_height(1,20)</code>	Set the height of control 1 to 20
Gets the height of the control H	<code>val=user.get_height(id)</code>	Specifies the control ID to get and returns the value Is the height value	<code>val=user.get_height(1)</code>	Save the height of control 1 into Val
Sets the text value	<code>user.set_text(id,*str)</code>	Specify the control ID and the text content str to be modified, which is generally specified as Controls such as label/textbox/qrcode	<code>User.set_text(1,normal)</code>	Change the text of control 1 to the two words "Normal"
Gets the control text	<code>user.get_text(id)</code>	Specifies the control ID to get the text content, typically specified as Controls such as label/textbox/qrcode	<code>text=user.get_text(1)</code>	Save the contents of the text of control 1 to the In the text pointer
Sets the value of the control	<code>user.set_val(id,val)</code>	Specifies the ID to modify the value of the control, typically specified for The prog/slider/arc/IMG controls	<code>user.set_val(3,100)</code>	Change the value of control 3 to 100
Gets the value of the control	<code>user.get_val(id)</code>	Gets the value of the control by specifying the control ID	<code>val=user.get_val(1)</code>	Save the obtained value of control 1 to <b>val</b>
Set the buzzer value	<code>user.set_buzzer(time)</code>	Specifies the time of the buzzer time in milliseconds. When the value is 0, the buzzer is disabled	<code>user.set_buzzer(100)</code>	Set the beep time of the beeper to 100 Milliseconds
Get the value of the buzzer	<code>user.get_buzzer()</code>	Read the beep time parameter of the buzzer	<code>val=user.get_buzzer()</code>	Read the beep time of the buzzer to <b>In the Val temporary variable</b>
Set the timer	<code>user.set_timer(id, one_time, time, enable)</code>	The XFD/SFD has 8 general-purpose timers, and the timer ID is from 0 to 7. To start the timer, you need to specify: timer ID, timer type option, timer time, and enable. Off	<code>user.set_timer(0,1,100,1)</code>	Enabling Timer 0 is a one-time trigger type, and its trigger time is 100ms.
Set Ivar	<code>user.set_ivar(id,val)</code>	Specify ID to set the value of the integer variable inside the XFD to Val (ID is 0 31)	<code>user.set_ivar(0,20)</code>	Set the value of IVAR 0 to 20
Get Ivar	<code>user.get_ivar(id)</code>	Specify ID to get the value of IVAR [ID]	<code>val=user.get_ivar(1)</code>	Get the value of IVAR 1 and save it in the Val temporary variable

Set svar	<code>user.set_svar(id,str)</code>	The value of the text variable that specifies the ID setting is <b>str</b>	User. Set _ svar (0, Shenzhen, Guangdong)	Let the content of SVAR 0 be: "Shenzhen, Guangdong"
Get svar	<code>user.get_svar(id)</code>	Specify ID to get the text content of SVAR [ID]	<code>str=user.get_svar(9)</code>	Get the contents of SVAR 9 Save to <b>In the str pointer</b>
Jump to the page	<code>user.set_page(id,delay)</code>	Jump to the ID screen, where delay To wait for delay milliseconds before jumping	<code>user.set_page(1,10)</code>	Jump to page 1 after a delay of 10 ms
Get the current page	<code>user.get_page()</code>	Gets the ID value of the current page	<code>id=user.get_page()</code>	Get the value of the current page and save to ID <b>Temporary variable</b>
Set the count value of the timer	<code>set_timer_times(id,cnt)</code>	Specifies the ID of the timer. This command sets the number of times the timer is called <b>cnt</b>	<code>user.set_timer_times(1,10)</code>	Set the count value of timer 1 to 10
Obtain the count value of the timer	<code>get_timer_times(id)</code>	Specifies the ID of the timer. This command sets the number of times the timer is called	<code>cnt=user.get_timer_times(1)</code>	Get the count value of Timer1 and save it to <b>In the CNT temporary variable</b>
<b>Function name</b>	<b>Function content</b>	<b>Function description</b>	<b>Sample code</b>	<b>Illustration of examples</b>
Address to enable 485	<code>addr_enable(en)</code>	Set the address mode of XFD/SFD 485	<code>user.addr_enable(1)</code>	Enables the address mode of the XFD/SFD 485 serial port
Set the address of 485	<code>set_addr(addr)</code>	Set the address of XFD/SFD 485 serial port	<code>user.set_addr(1)</code>	Set the address of the XFD/SFD 485 serial port to 1
Enabling data transmission <b>CRC16 check</b>	<code>set_crc16(en)</code>	Set CRC16 of XFD/SFD 485 Verification function	<code>user.set_crc16(1)</code>	Set up the XFD/SFD 485 serial port <b>CRC16 check</b>
Hide the control	<code>set_hidden (id,all,enable)</code>	Specifies the control ID, all is all hidden Hide, enable is the enable switch	<code>user.set_hidden(1,0,1)</code>	Hide control 1
Set background color 1	<code>set_bg_color (id,state,color)</code>	Specifies the control ID, state is the state of the control, and color is the color value	<code>user.set_bg_color(1,0,255)</code>	Sets the color in control 1 state 0 to Blue
Get background color 1	<code>get_bg_color (id,state)</code>	Specifies the control ID and state, and returns the value of the background color 1	<code>color=user.get_bg_color(1,0)</code>	Gets the background color 1 of control 1 in state 0
Set the background color 2	<code>set_bg_color2 (id,state,color)</code>	Specifies the control ID, state is the state of the control, and color is the color value	<code>user.set_bg_color2(1,0,255)</code>	Sets the color in control 1 state 0 to Blue
Set the background color 2	<code>get_bg_color2 (id,state)</code>	Specifies the control ID and state, and returns the value of the background color 1	<code>color=user.get_bg_color2(1,0)</code>	Gets the background color 1 of control 1 in state 0
Sets the font color	<code>set_font_color (id,color)</code>	Specifies the control ID, and color is the color value	<code>user.set_font_color(1,255)</code>	Set the color of control 1 to blue
Gets the font color	<code>get_font_color (id)</code>	Specifies the control ID, which is returned as the font color <b>The value of</b>	<code>color=user.get_font_color(2)</code>	Gets the font color of control 2
Set the transmittance	<code>set_opa (id,state,val)</code>	Specify the control ID, state, and settings, Set the transparency of the control	<code>user.set_opa(2,0,0)</code>	Set control 2 to be fully transparent in 0 state <b>Ming</b>
Obtain the transmittance	<code>get_opa(id,state)</code>	Specifies the control ID and state, and returns the control <b>Transparency</b>	<code>val=user.get_opa(2,0)</code>	Gets the transparency of control 2 in state 0 <b>The degree is saved in the temporary variable Val.</b>
Set the idle waiting time	<code>set_delay(ms)</code>	A specified wait time ms, <b>For null, etc</b>	<code>user.set_delay(100)</code>	Set wait for 100ms
Set the value of meter	<code>set_mt_val (wid,val,state)</code>	Set the pointer number of the Meter control wid to the content of Val	<code>user.set_mt_val(1,100,0)</code>	Set the value of the pointer 0 in the meter control to 100
Get the value of meter	<code>get_mt_val (wid,state)</code>	Gets the pointer to the Meter control wid. <b>The value of the number state</b>	<code>val=user.get_mt_val(1,0)</code>	Read the value of the pointer 0 in the meter control 1 into the temporary variable Val
Save the contents of the gui	<code>save_gui(en)</code>	When en = 1, save the changes to the pack	<code>user.save_gui(1)</code>	
Hide curves	<code>chart_hidden (wid,ch,enable)</code>	Hide ch in wid's chart control Channel, enable is the enable switch	<code>user.chart_hidden(1,1,1)</code>	Hide channel 1 in chart control 1
Set up hibernation	<code>set_sleep (time,bl_val)</code>	Specifies the sleep time in seconds and the backlight brightness after sleep, if <b>When the time value is 0, the sleep mode is not entered</b>	<code>user.set_sleep(5,10)</code>	Start-up sleep time is 5 seconds, and the backlight brightness after sleep is 10
Gets the sleep time	<code>get_sleep_time()</code>	Gets the sleep time of the XFD	<code>val=user.get_sleep_time()</code>	Save the sleep time to a temporary variable <b>In Val</b>
Obtain the backlight brightness when sleeping	<code>get_sleep_bl()</code>	Obtain the backlight brightness value when the XFD is sleeping	<code>user.set_sleep_bl()</code>	Saves the specified backlight brightness at the time of sleep <b>Into the temporary variable Val</b>

Animate	set_gif(wid,start_id,end_id,delay,loop_times,infinite,page,next_page,en)	Wid is the IMG control that executes the animation, the ID start _ ID is the start of the picture, and the ID end _ ID is the end ID of the picture. 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 Face next _ page indicates to jump to the corresponding page after executing the animation. En is the animation enable switch	user.set_gif (2,0,10,100,10,0,0,3,1)	Start the animation function of page0 control 2. The specified pictures are from 0 to 10, totaling 11. The delay of the animation is 100ms, and the page jumps to page 3 after executing the animation 10 times.
Control automatic reporting coordinate switch	report_pos(en)	The button control and panel control can report the position coordinate information when touched. This function can be set by En = 1, if you want to close the Move to report coordinates, en = 0	user.report_pos(1)	Enable the 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 0 XAA, 0 XBB and other arbitrary data, Index is the index value, which can be Any value of 0 255, data is 32 Bit integer data, four bytes of Arbitrary data	user.report_data(0xAA,1,0x01020304)	Send: 0XAA+1+01+02+03+04+FF FC FF FF to the upper computer. Note that if there is an address and CRC, the data will be sent together with the corresponding address and verification content.
Obtain the firmware version number	get_ver()	This command has no parameters, and the returned string "XXX _ YY YXXXX" is the content of the version, and YY is the number of the version, such as the content obtained. Is: FWxx _ SHMI _ ST _ 01 01 The contents of the version Prepared for FWxx _ SHMI _ ST ST version Number is 01.	ver=user.get_ver()	Get the current version information to the temporary variable ver
Set to hide the current page 面	set_hidden_current_page(en)	This function can be used to close the current page display by setting en = 1	user.set_hidden_current_page(1)	Enables hiding the current page
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 are 7 modes in total: 0: speed from beginning to end Same 1: Start at low speed 2: End at low speed 3: Open at low speed Start Low End 4: Exceed End Value 5: End Rebound Some 6 Last Step to Change	user.set_anim_hdir(1,10,100,100,5)	Control 1 implements a lateral movement from 10 to 100 with a movement time of 100ms, the way to move is to rebound after reaching the end point
<b>Function name</b>	<b>Function content</b>	<b>Function description</b>	<b>Sample code</b>	<b>Illustration of examples</b>
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: speed from beginning to end Same 1: Start at low speed 2: End with low speed 3: End with Low Start Low End 4: Over End Value 5: End Bounce Some 6 The last step to change	user.set_anim_vdir(1,10,100,100,5)	Control 1 implements a longitudinal movement from 10 to 100 with a movement time of 100ms, the way to move is to rebound 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, and endy is the end coordinate of y Time is the duration of the animation in milliseconds, and mode is 7 in total Species: 0: Same speed from beginning to end 1: Start at low speed 2: End with low speed 3: Start with low speed End with low speed 4: Exceed end value 5: End bounce some 6 Last step change	user.set_anim_any(4,-10,33,400,210,400,5)	Control 4 implements the movement of X from 10 to 33 and y from 400 to the lower left corner of 210 for 100ms. The way is to rebound after the end.
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 when Enable update when en = 1	user.set_ct_val(1,1,1,20,1)	Set the value of point 1 in curve number 1 of control 1 to 20
Get the value of the curve	get_ct_val(wid,ch,id)	Wid is used to specify the control number of the curve, which curve does ch refer to (numbering starts from 0), and ID specifies a point in the curve to get	val=get_ct_val(1,0,1)	Get the value of point 1 in curve 0 in curve control 1 and save it to a temporary variable In Val
Set the value of user IO port	set_usr_pin(id,val)	XFD provides 5 user GPIO ports, the corresponding IO ports can be set to 1 or 0 through the set _ usr _ pin (5 ports are set at present, and the ID is from 0 to 4)	user.set_usr_pin(0,1)	Set the value of user gpio port 0 to 1
Obtain the value of user IO port	get_usr_pin(id)	XFD provides 5 user GPIO port, the corresponding IO port (currently set) can be obtained through get _ usr _ pin. Is the value of 5 ports, ID from 0 (4)	val=user.get_usr_pin(0)	Obtain the value of user gpio port 0 and save it to the temporary variable Val

Serial port sends string str	report_str(str)	Send the string str through the serial port	From gb2312 import Gb2312 gcode = Gb2312 R = gcode. STRs ( 'I love Shenzhen!') uesr. Report _str (r)	Will UTF 8 I love Shenzhen! Turn into Gb2312 is sent through serial port after encoding Get out
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)	Change the value of the control from the original Minus 1
Meter value self-increasing	inc_mt_val(id,val,state)	Corresponding in the meter with the number ID The pointer value of state increments the value of Val	user.inc_mt_val(1,1,0)	Number 0 in the meter control with ID 1 Pointer value increments by 1
Meter Value Subtract	dec_mt_val(id,val,state)	Corresponding in the meter with the number ID The pointer value of state subtracts the value of Val	user.dec_mt_val(1,1,0)	Number 0 in the meter control with ID 1 The pointer value is decremented by 1
Backlight self-increasing	inc_bl(val)	Increment the value of the backlight by the value of Val	user.inc_bl(1)	Increment the 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)	The value of the character string of the numeric type set in the label is automatically increased by the value of Val, Where limit is the self-increasing upper limit value.	user.inc_label_val(1,1,100)	Increments the value of the numeric type string in label control 1 by 1. When incremented to 100, no further increase
The label value subtracts from itself	dec_label_val(id,val,limit)	The value of the character string of the numeric type set in the label is automatically reduced by the value of Val, Where limit is the lower limit of the self-subtraction	user.dec_label_val(1,1,0)	Subtracts 1 from the value of the numeric string in label control 1. To 0, no further reduction
Control touch enable	touch_enable(id,en)	When en is 1, the touch function of the control numbered ID is enabled	touch_enable(1,1)	Enable the touch function of control 1
Set menu cursor information	set_cursor_info(width,color,opa,num,"map")	Without the touch screen, through this The user function can set information such as the number of highlighted controls, the width, color, and opacity of the cursor in the interface, which is generally loaded in the page entry event	user.set_cursor_info(2,16777215,255,2,"12/13")	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 07. For details, please refer to the method in SHMI loading method Direction option	user.swi_page_load_dir(0)	Change the direction of page loading to left-to-right
Increment Menu Cursor Value	set_cursor_inc()	When there is no touch screen, this function is used to change the menu cursor (outer border highlight) after setting the menu cursor information The displayed control)	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()	When there is no touch screen, this function is used to change the menu cursor (outer border highlight) after setting the menu cursor information The displayed control)	user.set_cursor_dec()	Decreases the highlighted control ID number on the current page
Gets the value of the menu cursor	get_cursor()	This function gets the current menu. The value of the cursor	Val=user.get_cursor()	Get the value of the current menu cursor and save it. Into 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. Achieve the effect of a circular scrolling curve	user.set_ct_val2(1,0,20)	Set the next value of curve number 0 of curve control 1 to 20
<b>Function name</b>	<b>Function content</b>	<b>Function description</b>	<b>Sample code</b>	<b>Illustration of examples</b>

- Syntax of python (refer to for details:<https://www.runoob.com/python/python-basic-syntax.html>)

Entries	Explain	Sample code	Illustration of examples
Basic Syntax I	The biggest difference between learning Python and other languages is that Python's code blocks do not use curly braces { } to control classes, functions, and other logical decisions. The most distinctive feature of python is the use of indentation to write modules.  The number of indented blanks is variable, but all code block statements must contain the same number of indented blanks, and this must be strictly enforced.	if True: print("True") else: print("False")	
Basic Syntax II	Comments in python use # instead of//	#!/usr/bin/python #-*-coding: UTF 8 *-# filename: test. Py  if True: print("Answer") print ("True") else: print("Answer") # Not strictly indented, in execution An error will be reported print("False")	The contents below are all notes. Partial, not involved in running: #!/usr/bin/python #-*-coding: UTF 8 *-# filename: test. Py # Not strictly contracted # Not strictly contracted Shrink

Operation	Variables in python do not need to be declared to be able to participate directly in operations	a= 10 b = 20 c = serial port d = "display module print (a + B) print (C + d)"	Serial port output after running python 30 and "serial display module"
Conditional statement	The if judgment condition: execute the statement else: Execute statement	num =10 if num <0 or num >10 print ('hello') else: print ('undefine')	Serial port output after running python undefine
The while loop statement	While condition: execute statements	count =0 while (count <9); print 'The count is:', count count = count +1 print "Good bye!"	Output: The count is: 0 The count is: 1 The count is: 2 The count is: 3 The count is: 4 The count is: 5 The count is: 6 The count is: 7 The count is: 8 Good bye!

### XFD/SFD instruction set

- The instruction set must be sent to the module as a hexadecimal data stream via serial port
- System Control Command

Instruction name	Description	Instruction Code 1	Instruction Code 2	Parameter 1	Parameter 2	Parameter 3	Parameter 4	Example 1 (with address 0002, with CRC16, the value of parameter 1 is 1. If parameter 2 is valid, its value and parameter 2 are equal to the values on are all 2 for example)	Example 2 (with no address, without CRC16, parameter 1 has a value of 1 For parameter 2 and above, the values are 2 as an example)
restart_device	Reset the system	0x03	-	42330	21301	-	-	EE {00 02} 03 A5 5A 53 35 {38 9E} FF FC FF FF	EE 00 02 03 A5 5A 53 35 FF FC FF FF
shake_hand	Shake hands	0x04	-	-	-	-	-	EE {00 02} 04 {83 BE} FF FC FF FF	EE 04 FF FC FF FF
set_page	Jump to a specific page	0x05	-	pid	times(ms)	-	-	EE {00 02} 05 00 01 00 02 {01 38} FF FC FF FF	EE 05 00 01 00 02 FF FC FF FF
get_page	Get the current page	0x06	-	-	-	-	-	EE {00 02} 06 {42 3F} FF FC FF FF	EE 06 FF FC FF FF
set_bl	Set the backlight brightness	0x07	-	bl	-	-	-	EE {00 02} 07 00 01 {C1 01} FF FC FF FF	EE 07 00 01 FF FC FF FF
get_bl	Obtaining backlight brightness	0x08	-	-	-	-	-	EE {00 02} 08 {86 BE} FF FC FF FF	EE 08 FF FC FF FF
set_bps	Set the baud rate	0x09	-	bps	-	-	-	EE { 00 02 } 09 00 01 02 60 FF FC FF FF	EE 09 00 01 FF FC FF FF
get_bps	Get the baud rate	0x0a	-	-	-	-	-	EE {00 02} 0A {47 3F} FF FC FF FF	EE 0A FF FC FF FF
set_timer	Switch timer	0x0b	-	tid	one_shot	time(ms)	enable	EE {00 02} 0B 00 01 00 02 00 02 00 02 {C1 30} FF FC FF FF	EE 0B 00 01 00 02 00 02 00 02 FF FC FF FF
save_gui	Save the contents of the control	0x0c	-	-	-	-	-	EE {00 02} 0C {45 BF} FF FC FF FF	EE 0C FF FC FF FF
save_sys	Save the system parameters	0x0d	-	-	-	-	-	EE {00 02} 0D {85 7E} FF FC FF FF	EE 0D FF FC FF FF
save_xvar	Save the XVAR variable	0x0e	-	-	-	-	-	EE {00 02} 0E {84 3E} FF FC FF FF	EE 0E FF FC FF FF
set_delay	Set the delay time	0x0f	-	time(ms)	-	-	-	EE {00 02} 0F 00 01 {03 80} FF FC FF FF	EE 0F 00 01 FF FC FF FF
set_sleep	Set up hibernation	0x10	-	time(s)	bl_val	-	-	EE {00 02} 10 00 01 00 02 {C2 35} FF FC FF FF	EE 10 00 01 00 02 FF FC FF FF
get_sleep	Gets the hibernation parameters	0x11	-	-	-	-	-	EE {00 02} 11 {4C 7F} FF FC FF FF	EE 11 FF FC FF FF
set_buzzer	Set the buzzer parameter	0x12	-	time(s)	-	-	-	EE {00 02} 12 00 01 {05 10} FF FC FF FF	EE 12 00 01 FF FC FF FF
Instruction name	Description	Instruction Code 1	Instruction Code 2	Parameter 1	Parameter 2	Parameter 3	Parameter 4	Example 1 (with address 0002, with CRC16, the value of parameter 1 is 1. If parameter 2 is valid, its value and parameter 2 are equal to the values on are all 2 for example)	Example 2 (with no address, without CRC16, parameter 1 has a value of 1 For parameter 2 and above, the values are 2 as an example)
get_buzzer	Gets the buzzer parameter	0x13	-	-	-	-	-	EE {00 02} 13 {8D FE} FF FC FF FF	EE 13 FF FC FF FF
set_ivar	Set the value of Ivar	0x14	-	xid	valH	valL	-	EE {00 02} 14 00 01 00 02 00 02 {C0 52} FF FC FF FF	EE 14 00 01 00 02 00 02 FF FC FF FF
get_ivar	Get the value of Ivar	0x15	-	xid	-	-	-	EE {00 02} 15 00 01 {C4 A1} FF FC FF FF	EE 15 00 01 FF FC FF FF

set_svar	Set the value of svar	0x16	-	xid	str	-	-	EE {00 02} 16 00 01 32 00 {63 29} FF FC FF FF	EE 16 00 01 32 00FF FC FF FF
get_svar	Get the value of svar	0x17	-	xid	-	-	-	EE {00 02} 17 00 01 {04 00} FF FC FF FF	EE 17 00 01 FF FC FF FF
get_ver	Gets the version number	0x18	-	-	-	-	-	EE {00 02} 18 {4A BF} FF FC FF FF	EE18 FF FC FF FF
set_cali	Calibrate the resistive touch screen	0x1B	-	-	-	-	-	EE {00 02} 1B {4B FF} FF FC FF FF	EE 1B FF FC FF FF

• UI-related setting instruction

Instruction name	Description	Instruction Code 1	Instruction Code 2	Parameter 1	Parameter 2	Parameter 3	Parameter 4	Example 1 (with an address of 0002, with CRC16, the value of parameter 1 is 1. If parameter 2 is valid, its value and the values above parameter 2 are all 2. 子)	Example 2 (without address, without CRC16, the value of parameter 1 is 1, and the value of parameter 2 and above is 2 子)
set_obj_x	Sets the X coordinate of the control	0x81	0x01	wid	x_val	-	-	EE{00 02}81 01 00 01 00 02 {CB F3} FF FC FF FF	EE 81 01 00 01 00 02 FF FC FF FF
set_obj_y	Sets the Y coordinate of the control	0x81	0x02	wid	y_val	-	-	EE{ 00 02} 81 02 00 01 00 02 {CB B7} FF FC FF FF	EE 81 02 00 01 00 02 FF FC FF FF
set_obj_width	Sets the W width of the control	0x81	0x03	wid	width	-	-	EE{00 02} 81 03 00 01 00 02 {0B 8A} FF FC FF FF	EE 81 03 00 01 00 02 FF FC FF FF
set_obj_height	Sets the H height of the control	0x81	0x04	wid	height	-	-	EE {00 02} 81 04 00 01 00 02 {CB 3F} FF FC FF FF	EE 81 04 00 01 00 02 FF FC FF FF
set_obj_text	Set control text content	0x81	0x05	wid	str	-	-	EE{00 02} 81 05 00 01 32 00 {AA 96} FF FC FF FF	EE 81 05 00 01 32 00 FF FC FF FF
set_obj_bgcolor1	Set the background of the control 色1	0x81	0x06	wid	state	bcolor1H	bcolor1L	EE {00 02} 81 06 00 01 00 02 00 02 00 02 {8B 44} FF FC FF FF	EE 81 06 00 01 00 02 00 02 00 02 FF FC FF FF
set_obj_bgcolor2	Set the background of the control 色2	0x81	0x07	wid	state	bcolor2H	bcolor2L	EE {00 02} 81 07 00 01 00 02 00 02 00 02 {1B 49} FF FC FF FF	EE 81 07 00 01 00 02 00 02 00 02 FF FC FF FF
set_obj_font_color	Set the foreground of the control 色	0x81	0x08	wid	fcolorH	fcolorL	-	EE {00 02} 81 08 00 01 00 02 00 02 {A6 1D} FF FC FF FF	EE 81 08 00 01 00 02 00 02 FF FC FF FF
set_obj_val	Set the value of the control	0x81	0x09	wid	val	-	-	EE {00 02} 81 09 00 01 00 02 {0A 12} FF FC FF FF	EE 81 09 00 01 00 02 FF FC FF FF
set_obj_hidden	Sets the hidden control	0x81	0x0a	wid	all	enable	-	EE {00 02} 81 0A 00 01 00 02 00 02 {66 3E} FF FC FF FF	EE 81 0A 00 01 00 02 00 02 FF FC FF FF
set_chart_hidden	Sets the hidden curve	0x81	0x0b	wid	ch	enable	-	EE {00 02} 81 0B 00 01 00 02 00 02 {A6 2E} FF FC FF FF	EE 81 0B 00 01 00 02 00 02 FF FC FF FF
set_chart_val	Set the value of the curves	0x81	0x0c	wid	ch	pid	num	EE {00 02} 81 0C 00 01 00 02 00 02 00 02 00 02 00 02 {D1 5C} FF FC FF FF	EE 81 0C 00 01 00 02 00 02 00 02 00 02 00 02 FF FC FF FF
set_hidden_current_page	Set to hide the current page 画面	0x81	0x0d	en	-	-	-	EE {00 02} 81 0D 00 01 {0B 78} FF FC FF FF	EE 81 0D 00 01 FF FC FF FF
set_meter_val	Set the pointer value of meter	0x81	0x0e	wid	val	state	-	EE { 00 02} 81 0E 00 01 00 02 00 02 {A6 7B} FF FC FF FF	EE 81 0E 00 01 00 02 00 02 FF FC FF FF
set_touch_enable	Set control touch enable	0x81	0x0f	wid	en	-	-	EE {00 02} 81 0F 00 01 00 01 {0B DA} FF FC FF FF	EE 81 0F 00 01 01 01 FF FC FF FF
set_widget_status	Sets the state of the touch control	0x81	0x10	wid	status	-	-	EE {00 02} 81 10 00 01 00 01 {C9 4F} FF FC FF FF	EE 81 10 00 01 01 01 FF FC FF FF
set_cursor_increment	Increment Menu Cursor Value	0x81	0x11	-	-	-	-	EE {00 02}81 11 {EC A1} FF FC FF FF	EE 81 11 FF FC FF FF

Instruction name	Description	Instruction Code 1	Instruction Code 2	Parameter 1	Parameter 2	Parameter 3	Parameter 4	Example 1 (with an address of <b>0002</b> , with <b>CRC16</b> , the value of <b>parameter 1 is 1</b> . If <b>parameter 2 is valid</b> , its value and the values above <b>parameter 2</b> are all 2. 子)	Example 2 (without address, without CRC16, the value of parameter 1 is 1, and the value of parameter 2 and above is 2 子)
set_cursor_de c	Subtract Menu Cursor Value	0x81	0x12	-	-	-	-	EE {00 02} 81 12 {EDE1} FF FC FF FF	EE 81 12 FF FC FF FF
set_ct_val2	Sets the next value of the curves	0x81	0x13	wid	cid	val	-	EE {00 02} 81 13 00 01 00 00 00 02 {67 17} FF FC FF FF	EE 81 13 00 01 00 00 00 02 FF FC FF FF

• UI-related read instruction

Instruction name	Description	Instruction Code 1	Instruction Code 2	Parameter 1	Parameter 2	Parameter 3	Parameter 4	Example 1 (with address 0002, with <b>CRC16</b> , the value of <b>parameter 1 is 1</b> , and if <b>parameter 2 is valid</b> , the value of <b>parameter 2</b> and the values above <b>parameter 2</b> are all 2.)	Example 2 (with no address, without <b>CRC16</b> , <b>parameter 1</b> The value for parameter 2 and above is 1 <b>2 as an example</b> )
get_obj_x	Gets the X coordinate of the control	0x82	0x01	wid	-	-	-	EE {00 02} 82 01 00 01 {5C B8} FF FC FF FF	EE 82 01 00 01 FF FC FF FF
get_obj_y	Gets the Y coordinate of the control	0x82	0x02	wid	-	-	-	EE {00 02} 82 02 00 01 {5C 48} FF FC FF FF	EE 82 02 00 01 FF FC FF FF
get_obj_width	Gets the width of the control	0x82	0x03	wid	-	-	-	EE {00 02} 82 03 00 01 {9C 19} FF FC FF FF	EE 82 03 00 01 FF FC FF FF
get_obj_height	Gets the height of the control	0x82	0x04	wid	-	-	-	EE {00 02} 82 04 00 01 {5D A8} FF FC FF FF	EE 82 04 00 01 FF FC FF FF
get_obj_text	Gets the text of the control	0x82	0x05	wid	-	-	-	EE {00 02} 82 05 00 01 {9D F9} FF FC FF FF	EE 82 05 00 01 FF FC FF FF
get_obj_bgcolor or1	Gets the of the control bgcolor1	0x82	0x06	wid	state	-	-	EE {00 02} 82 06 00 01 00 02 {38 46} FF FC FF FF	EE 82 06 00 01 00 02 FF FC FF FF
get_obj_bgcolor or2	Gets the of the control bgcolor2	0x82	0x07	wid	state	-	-	EE {00 02} 82 07 00 01 00 02 {F8 7B} FF FC FF FF	EE 82 07 00 01 00 02 FF FC FF FF
get_obj_font_c olor	Gets the of the control font_color	0x82	0x08	wid	-	-	-	EE {00 02} 82 08 00 01 {5E 68} FF FC FF FF	EE 82 08 00 01 FF FC FF FF
get_obj_val	Gets the value of the control	0x82	0x09	wid	-	-	-	EE {00 02} 82 09 00 01 {9E 39} FF FC FF FF	EE 82 09 00 01 FF FC FF FF
get_chart_val	Get the value of the curve	0x82	0x0a	wid	sid	pid	num	EE {00 02} 82 0A 00 01 00 02 00 02 00 02 {84 E1} FF FC FF FF	EE 82 0A 00 01 00 02 00 02 00 02 FF FC FF FF
get_meter_val	Get the pointer value of meter	0x82	0x0b	wid	state	-	-	EE {00 02} 82 0B 00 01 00 02 {F9 6B} FF FC FF FF	EE 82 0B 00 01 00 02 FF FC FF FF
get_touch_ena ble	Acquire that touch enable state of the control	0x82	0x0c	wid	-	-	-	EE {00 02} 82 0C 00 01 {9F 29} FF FC FF FF	EE 82 0C 00 01 FF FC FF FF
get_widget_st atus	Gets the state of the touch control	0x82	0x0D	wid	-	-	-	EE {00 02} 82 0D 00 01 {5F 78} FF FC FF FF	EE 82 0D 00 01 FF FC FF FF
get_cursor	Get the current menu cursor value	0x82	0x0E	-	-	-	-	EE {00 02} 82 0E 00 01 {D4 E0} FF FC FF FF	EE 82 0E FF FC FF FF