Command set instruction

Date	Matched firmware version	Update by	
2021/03/04	V1.0	Nigel Tian	Basic version
2021/04/07	V1.2	Nigel Tian	System related, Gauge added
2021/04/26	V1.3	Nigel Tian	Command code define updated; Image animation updated; Text selector updated; Label/edit updated
2021/05/13	V1.4	Nigel Tian	System related updated
2021/05/13		Nigel Tian	Gauge pointer updated, Mistake fixed

Directory

Basic Introduction

Basic command format:

For example:

ST<{"cmd_code":"open_win","widget":"label_value"}>ET
ST<{"cmd_code":"set_value","widget":"label_value",”value”:100}>ET

Important: The text content should be with the quotes(“”), including widgets’ name but the value content shouldn’t be with the quotes, such as value or true/false states.

Frame head: ST<

Command content: { json }

Frame end: >ET

1. Command set code description

1.1 System operation related

sys_reboot System reboot

Example:

ST<{"cmd_code":"sys_reboot","type":"system"}>ET

System reboot

sys_hello Communication test

Example:

ST<{"cmd_code":"sys_hello","type":"system"}>ET

Test if the communication between the MCU and the display works normal. The display will return command 0x0001 if the communication works.

sys_version Get version information

Example:

ST<{"cmd_code":"sys_version","type":"system"}>ET

Get the version information for the current firmware. The display will return the command 0x0002 for the firmware version information.

****

1.2 Common command

Command can be used for all widgets:

set_enable **Set enable/disable for the widgets**

set_visible **Set visible/un-visible for the widgets**

Example:

ST<{"cmd_code":"set_enable","type":"widget","widget":"button1","enable":true}>ET

Set the widget available

ST<{"cmd_code":"set_enable","type":"widget","widget":"button1","enable":false}>ET

Disable the widget

ST<{"cmd_code":"set_visible","type":"widget","widget":"button1","visible":true}>ET

Set the widget visible

ST<{"cmd_code":"set_visible","type":"widget","widget":"button1","visible":false}>ET

Set the widget un-visible

****

1.3 Window related

ST<{"cmd_code":"*command type*","widget":"*window name*"}>ET

open_win Open an existing window

*The previous window will not be closed if open a window. The value you have saved in the widget will be temporarily saved unless it is closed or the display is shut down.

Example :

Open "label_value" window

ST<{"cmd_code":"open_win","widget":"label_value"}>ET

close_win Close all the windows you opened previously (not suggested, be cautious to use)

*The previous window will be closed and the value you have saved in the widget will be cleared.

Example:

Close "lable_value" window

ST<{"cmd_code":"close_win","widget":"label_value"}>ET

back_win Close the current window and go back to the previous window

*The previous window will be closed and the value you have saved in the widget will be cleared.

Example:

ST<{"cmd_code":"back_win"}>ET

back_win_to Close current window and go back to the defined window

*The previous window will be closed and the value you have saved in the widget will be cleared.

Example:

Back to the window named "label_value", close all the window above it. Normally worked for the multiple layers windows

ST<{"cmd_code":"back_win_to","widget":"label_value"}>ET

Back to home page

ST<{"cmd_code":"back_win_to","widget":"home_page"}>ET

back_home Go back to home_page

*The previous window will not be closed and the value you have saved in the widget will be temporarily saved unless it is closed or the display is shut down..

Example:

ST<{"cmd_code":"back_home"}>ET

Notice: The home_page can not be closed

****

1.4  Label/edit related

ST<{"cmd_code":"*command type*","type":"*type name*","widget":"*label/edit name*","text":"*text content*"}>ET

set_text Set the text displayed on the label/edit;

Example:

Set the "label1" content as "Hello Stone"

ST<{"cmd_code":"set_text","type":"label","widget":"label1","text":"Hello Stone"}>ET

Set the “edit1” content as “1234567890”

ST<{"cmd_code":"set_text","type":"edit","widget":"edit1","text":"1234567890"}>ET

get_text Get the text content on the label/edit;

Example:

Get the text content from “label1”

ST<{"cmd_code":"get_text","type":"label","widget":"label1"}>ET

Get the text content from “edit1”

ST<{"cmd_code":"get_text","type":"edit","widget":"edit1"}>ET

get_value Get the value from the label/edit;

Notice: the label widget can be only set/get value as the float type

Example:

Get the value from “label1” (**float type only**)

ST<{"cmd_code":"get_value","type":"label","widget":"label1"}>ET

Get the value from “edit1”

ST<{"cmd_code":"get_value","type":"edit","widget":"edit1"}>ET

****

1.5  Progress_bar

set_max Set the max value of the progress bar

Example:

Set the “progress_bar1” max value as 100

ST<{"cmd_code":"set_max","type":"progress_bar","widget":"progress_bar1","max":100}>ET

show_text If the text displayed on the progress bar

Example:

Show the text value on the “progress_bar2”

ST<{"cmd_code":"show_text","type":"progress_bar","widget":"progress_bar2","show_text":"true"}>ET

set_value Set the value of the progress bar

Example:

Set the current value of "progress_bar3" to 40

ST<{"cmd_code":"set_value","type":"progress_bar","widget":"progress_bar3","value":40}>ET

get_value Get the value of the current progress bar

Example:

ST<{"cmd_code":"get_value","type":"progress_bar","widget":"progress_bar"}>ET

get_percent Get the percentage value of the current progress bar

Example:

ST<{"cmd_code":"get_percent","type":"progress_bar","widget":"progress_bar"}>ET

****

1.6  Slider

set_max Set max value of the slider

Example:

ST<{"cmd_code":"set_max","type":"slider","widget":"slider1","max":200}>ET

set_min Set the min value of the slider

Example:

ST<{"cmd_code":"set_min","type":"slider","widget":"slider1","min":0}>ET

set_step Set step value for slider

Example:

ST<{"cmd_code":"set_step","type":"slider","widget":"slider1","step":1}>ET

set_value Set current point value for slider

Example:

ST<{"cmd_code":"set_value","type":"slider","widget":"slider1","value":0}>ET

get_value Get current point value for slider

Example:

ST<{"cmd_code":"get_value","type":"slider","widget":"slider1"}>ET

****

1.7  Image

set_image Set image name for display

Example:

ST<{"cmd_code":"set_image","type":"image","widget":"image","image":"guage_bg"}>ET

ST<{"cmd_code":"set_image","type":"image","widget":"image","image":"vgus01"}>ET

set_draw_type Set the image draw type

Example:

ST<{"cmd_code":"set_draw_type","type":"image","widget":"image","draw_type":2}>ET

draw_type value define as follow:

>IMAGE_DRAW_DEFAULT = 0: *Default setting. Show the image as the original size on the top-left corner on the target rectangular;*

>IMAGE_DRAW_CENTER = 1: *Center display. Show the image as the original size on the center of the target rectangular;*

>IMAGE_DRAW_ICON = 2: *Icon display mode. The showing method will be same with the center display but the size will be centered and scaled as the size of the target rectangular.*

>IMAGE_DRAW_SCALE = 3: *Scale display mode. Zoom in/out the image as the size of the target rectangular (the original ratio is not guaranteed).*

>IMAGE_DRAW_SCALE_AUTO = 4: *Auto scale mode. Zoom in/out the image as the height or width of the target rectangular (It will choose the smallest ratio) and center displayed.*

>IMAGE_DRAW_SCALE_DOWN = 5: *Zoom in mode. If the image is larger than target rectangular, it will be zoomed in. If not then center display.*

>IMAGE_DRAW_SCALE_W = 6: *Width scale mode. Zoom in/out the image as the width of the target rectangular. The height will be adjusted as the zoomed scale of the width. The oversized part will not be displayed.*

>IMAGE_DRAW_SCALE_H = 7: *Height scale mode. Zoom in/out the image as the height of the target rectangular. The width will be adjusted as the zoomed scale of the height. The oversized part will not be displayed.*

>IMAGE_DRAW_REPEAT = 8: *Tiled display mode.*

>IMAGE_DRAW_REPEAT_X = 9: *Tiled horizontally and scaled vertically.*

>IMAGE_DRAW_REPEAT_Y = 10: *Tiled vertically and scaled horizontally.*

>IMAGE_DRAW_REPEAT_Y_INVERSE = 11: *Tiled vertically and scaled vertically (from bottom to the top);*

set_scale Set the zoom in/out scale for the image

Example:

ST<{"cmd_code":"set_scale","type":"image","widget":"image","scale_x":0.5,"scale_y":0.5}>ET

ST<{"cmd_code":"set_scale","type":"image","widget":"image","scale_x":1,"scale_y":1}>ET

set_rotation Set the image rotation angle

*The image draw type must be center or icon if you need to enable the rotate the image.

Example:

ST<{"cmd_code":"set_rotation","type":"image","widget":"image","rotation":90}>ET

ST<{"cmd_code":"set_rotation","type":"image","widget":"image","rotation":180}>ET

****

1.8  Image_animation related

set_play Start play the image animation

Example:

Start the animation playing for widget "image_animation"

ST<{"cmd_code":"set_play","type":"image_animation","widget":"image_animation"}>ET

set_pause Pause the image animation

Example:

Pause the animation playing for widget "image_animation"

ST<{"cmd_code":"set_pause","type":"image_animation","widget":"image_animation"}>ET

set_stop Stop the image animation

Example:

Stop the animation playing for widget "image_animation"

ST<{"cmd_code":"set_stop","type":"image_animation","widget":"image_animation"}>ET

set_format Set the name format for the animation image value

Example:

Set the name format for "image_animation" as text+digits (for example: num1)

ST<{"cmd_code":"set_format","type":"image_animation","widget":"image_animation","format":"%s%d"}>ET

set_image Set the prefix for the animation image name

Example:

Set the prefix name for "image_animation" as "num"

ST<{"cmd_code":"set_image","type":"image_animation","widget":"image_animation","image":"num"}>ET

set_interval Set the gap between 2 frame of the image animation (ms)

Example:

Set the interval for "image_animation" as 200 ms

ST<{"cmd_code":"set_interval","type":"image_animation","widget":"image_animation","interval":200}>ET

set_loop Set if the animation will start the loop display

Example:

Enable the loop play for "image_animation"

ST<{"cmd_code":"set_loop","type":"image_animation","widget":"image_animation","loop":true}
>ET

set_range Set the start and end value for the animation image

Example:

Set the "image_animation" range from 1-10

ST<{"cmd_code":"set_range","type":"image_animation","widget":"image_animation","start_ind
ex":1,"end_index":10}>ET

****

1.9  Image value

Image value is a special function which you can assign a special name and value to a series images. And you can increase or edit the value, to change the related image display. Mostly used in the art-work digit/text function.

set_image Define the image value's name

Example:

Set "image_value1" name as "num"

ST<{"cmd_code":"set_image","type":"image_value","widget":"image_value1","image":"num"}>
ET

set_format Set image value format

Example:

Set the "image_value2" format as 2 bit int and 2 bit float.

ST<{"cmd_code":"set_format","type":"image_value","widget":"image_value2","format":"%02.2
f"}>ET

set_max Set max value of the image

Example:

ST<{"cmd_code":"set_max","type":"image_value","widget":"image_value","max":200}>ET

set_min Set min value of the image

Example:

ST<{"cmd_code":"set_min","type":"image_value","widget":"image_value","min":0}>ET

set_value Set current image value

Example:

ST<{"cmd_code":"set_value","type":"image_value","widget":"image_value","value":6.66}>ET

get_value Get current image value

Example:

ST<{"cmd_code":"get_value","type":"image_value","widget":"image_value"}>ET

****

1.10 Text selector related

set_text Change the selector to the option with the input text

Example:

Change the current option of "text_selector1" to option "green"

ST<{"cmd_code":"set_text","type":"text_selector","widget":"text_selector1","text":"green"}>ET

set_value Change the selector to the option with the input value(If the option type is text, the value will be the digits in front of each option)

Example:

Change the current option of "text_selector1" to option 2, which is option "green"

ST<{"cmd_code":"set_value","type":"text_selector","widget":"text_selector1","value":2}>ET

set_selected Change the selector to the option with the input selected number(The option number will start from 0)

Example:

Change the current option of "text_selector1" to option No.1, which is option "green"

ST<{"cmd_code":"set_selected","type":"text_selector","widget":"text_selector1","selected_index":1}>ET

get_text Get the current option text content

Example:

ST<{"cmd_code":"get_text","type":"text_selector","widget":"text_selector1"}>ET

get_value Get the current option value

Example:

ST<{"cmd_code":"get_value","type":"text_selector","widget":"text_selector1"}>ET

get_selected Get the current option number

Example:

ST<{"cmd_code":"get_selected","type":"text_selector","widget":"text_selector1"}>ET

****

1.11  Digit clock

set_date Set RTC time and date;

Example:

ST<{"cmd_code":"set_date","type":"digit_clock","widget":"digit_clock","date":"12:23"}>ET

ST<{"cmd_code":"set_date","type":"digit_clock","widget":"digit_clock","date":"12:23:46"}>ET

ST<{"cmd_code":"set_date","type":"digit_clock","widget":"digit_clock","date":"2021-02-26 12:23"}>ET

ST<{"cmd_code":"set_date","type":"digit_clock","widget":"digit_clock","date":"2021-02-26 12:23:46"}>ET

set_format Set RTC format;

Example:

ST<{"cmd_code":"set_format","type":"digit_clock","widget":"digit_clock","format":"hh:mm"}>ET

ST<{"cmd_code":"set_format","type":"digit_clock","widget":"digit_clock","format":"hh:mm:ss"}>ET

ST<{"cmd_code":"set_format","type":"digit_clock","widget":"digit_clock","format":"YYYY-MM-DD hh:mm"}>ET

ST<{"cmd_code":"set_format","type":"digit_clock","widget":"digit_clock","format":"YYYY-MM-DD hh:mm:ss"}>ET

ST<{"cmd_code":"set_format","type":"digit_clock","widget":"digit_clock","format":"YYYY-MM-DD hh:mm:ss w"}>ET

ST<{"cmd_code":"set_format","type":"digit_clock","widget":"digit_clock","format":"YYYY-MM-DD hh:mm:ss W"}>ET

ST<{"cmd_code":"set_format","type":"digit_clock","widget":"digit_clock","format":"YYYY-MM-DD hh:mm:ss MMM"}>ET

ST<{"cmd_code":"set_format","type":"digit_clock","widget":"digit_clock","format":"YYYY-M-D h:m:s"}>ET

ST<{"cmd_code":"set_format","type":"digit_clock","widget":"digit_clock","format":"YYYY/MM/DD hh:mm:ss"}>ET

get_date Get current RTC time and date;

Example:

ST<{"cmd_code":"get_date","type":"digit_clock","widget":"digit_clock"}>ET

****

1.12  Switch

set_value Set the switch value between 0/1 (true or false)

Example:

ST<{"cmd_code":"set_value","type":"switch","widget":"switch","value":true}>ET

ST<{"cmd_code":"set_value","type":"switch","widget":"switch","value":false}>ET

****

1.13  Check box

set_value Set the check box value between 0/1 (true or false)

Example:

ST<{"cmd_code":"set_value","type":"check_button","widget":"check_button","value":true}>ET

ST<{"cmd_code":"set_value","type":"check_button","widget":"check_button","value":false}>ET

****

1.14  Radio button

set_value Set the radio button value between 0/1(true or false);

Example:

ST<{"cmd_code":"set_value","type":"radio_button","widget":"radio_button","value":true}>ET

ST<{"cmd_code":"set_value","type":"radio_button","widget":"radio_button","value":false}>ET

get_checked Get the value of currently checked radio buttons; radio button

Example:

ST<{"cmd_code":"get_checked","type":"radio_button","widget":"radio_button"}>ET

****

1.15  Gauge related

set_image Set image name for gauge (same with the image command)

set_draw_type Set the draw type for the gauge image

Example:

ST<{"cmd_code":"set_image","type":"gauge","widget":"guage1","image":"gauge_bg"}>ET

ST<{"cmd_code":"set_image","type":"gauge","widget":"guage1","image":"gauge_bg1"}>ET

Set the image for gauge

ST<{"cmd_code":"set_draw_type","type":"gauge","widget":"gauge1","draw_type":2}>ET

Set the draw type for the gauge image. The draw type value is same with the image widget

****

1.16  Gauge_pointer

set_image Set the image of the pointer

Example:

ST<{"cmd_code":"set_image","type":"gauge_pointer","widget":"guage_pointer","image":"guag
e_pointer1"}>ET

set_angle Set the rotation angle of the pointer (Datatype: int)

Example:

ST<{"cmd_code":"set_angle","type":"gauge_pointer","widget":"guage_pointer","angle":60}>ET

Return command set instruction

Date	Matched firmware version	Update by	
2021/03/18	V1.0	Nigel Tian	Basic version
2021/03/22	V1.1	Nigel Tian	Label, Edit added
2021/04/27	V1.3	Nigel Tian	Radio button, progress bar update; image value, text selector added
2021/05/13	V1.4	Nigel Tian	System related updated

2.Return command set code description

2.1 Basic introduction

The command returned from display UART will use the hex code, to reduce the MCU analysis difficulty and processing burden. The multiple byte data will use the format as high bit-low bit.

Basic command format:

Frame head + CMD + LEN + DATA + Frame tail + CRC16 CRC

Example: ST< 0x1068 0x0004 0x01 0x02 0x03 0x04 >ET CRC16

****

2.2 System related

0x0000 System states (automatically return 3 times after system boots. The gap between each time is 100 ms)

0x01 System running

0x02 System Standby

0xFF System error

Example:

System working normal, serial port auto return after display boots:

ST<0x00 0x00 0x00 0x01 0x01>ET

Hex code:53 54 3C 00 00 00 01 01 3E 45 54 AB 25

0x0001 System return code for communication test command (sys_hello)

0x01 System running, communication OK

Example:

ST<0x00 0x01 0x00 0x01 0x01>ET

Hex code:53 54 3C 00 01 00 01 01 3E 45 54 6B 35

0x0002 System firmware information return for get_version command

Data: Firmware version information

Example:

ST<0x00 0x02 0x00 0x18 V1.0.01.800x480.210513RC>ET

Version number: V1.0.01.800x480.210513RC

Hex code:53 54 3C 00 02 00 18 56 31 2E 30 2E 30 31 2E 38 30 30 78 34 38 30 2E 32 31 30 35 31 33 52 43 3E 45 54 67 50

****

2.3 Button related

0x1001 button status send

Key value:

0x01 Button pressed

Example:

ST< 0x10 0x01 0x00 0x08 button9 0x01 >ET CRC16

Hex code: 53 54 3C 10 01 00 08 62 75 74 74 6F 6E 39 01 3E 45 54 E7 E0

0x02 Button released (after release the button, the CLICK event function will be trigger)

Example:

ST< 0x10 0x01 0x00 0x08 button9 0x02 >ET

Hex code: 53 54 3C 10 01 00 08 62 75 74 74 6F 6E 39 02 3E 45 54 A3 E0

0x03 long press trigger (Long press will NOT trigger the CLICK event)

Example:

ST< 0x10 0x01 0x00 0x08 button9 0x03 >ET

Hex code: 53 54 3C 10 01 00 08 62 75 74 74 6F 6E 39 03 3E 45 54 5F E1

****

2.4 Button customization related

0x1002 button key value send (will be released soon)

Key value:

2 byte data, defined by user

Example:

The code return while button pressed

ST< 0x10 0x02 0x00 0x08 button1 0x01 0x02 >ET

Example:

The code return while button released

ST< 0x10 0x02 0x00 0x08 button1 0x03 0x02 >ET

****

2.5 Switch customize related

0x1010 switch value changed return

Key value:

Last byte of the data content

0x00 **Switch off**

0x01 **Switch on**

Example

Switch value changed return:

Switch off

ST< 0x10 0x10 0x00 0x07 switch 0x00 >ET

Hex code: 53 54 3C 10 10 00 07 73 77 69 74 63 68 00 3E 45 54 21 F2

Switch on

ST< 0x10 0x10 0x00 0x07 switch 0x01 >ET

Hex code: 53 54 3C 10 10 00 07 73 77 69 74 63 68 01 3E 45 54 DD F3

****

2.6 Check box customization related

0x1020 check button value changed return

Key value:

Last byte of the data content

0x00	**Change to un-check status**
0x01	**Change to checked status**

Example

check button value changed return:

Un-check

ST< 0x10 0x20 0x00 0x0D check_button 0x00 >ET

Hex code: 53 54 3C 10 20 00 0D 63 68 65 63 6B 5F 62 75 74 74 6F 6E 00 3E 45 54 AF 1E

Checked

ST< 0x10 0x20 0x00 0x0D check_button 0x01 >ET

Hex code: 53 54 3C 10 20 00 0D 63 68 65 63 6B 5F 62 75 74 74 6F 6E 01 3E 45 54 53 1F

****

2.7 Radio button customization related(Single check button)

0x1030 radio button value changed return initiative

0x1031 radio button value changed return passively(by get_checked command from MCU)

Key value:

Last byte of the data content

0x00 **Change to un-check status**

0x01 **Change to checked status**

Example:

radio button value changed return initiative:

Manually check the radio_button1

ST< 0x10 0x30 0x00 0x0E radio_button1 0x01 >ET

Hex code: 53 54 3C 10 30 00 0E 72 61 64 69 6F 5F 62 75 74 74 6F 6E 31 01 3E 45 54 34 4E

Radio_button automatically un-check

ST< 0x10 0x30 0x00 0x0D radio_button 0x00 >ET

Hex code: 53 54 3C 10 30 00 0D 72 61 64 69 6F 5F 62 75 74 74 6F 6E 00 3E 45 54 32 36

radio button value changed return passively be getting command from MCU:

radio_button1 states return as checked

ST< 0x10 0x31 0x00 0x0E radio_button1 0x01 >ET

Hex code: 53 54 3C 10 31 00 0E 72 61 64 69 6F 5F 62 75 74 74 6F 6E 31 01 3E 45 54 34 4E

radio_button states return as unchecked

ST< 0x10 0x31 0x00 0x0D radio_button 0x00 >ET

Hex code: 53 54 3C 10 31 00 0D 72 61 64 69 6F 5F 62 75 74 74 6F 6E 00 3E 45 54 32 36

****

2.8 Slider customization related

0x1040 slider value changing return

0x1041 slider value changed return

Key value:

Last 4 bytes of the data content

Example: 0x42400000 current slider's value: 48.000000 (Type: float compliant with IEEE 754 specification)

Example:

Slider value changing return (real-time slider value return from UART. The data return step is based on the slider step setting in the GUI):

Slider1 value changing, value: 48.000000

ST< 0x10 0x40 0x00 0x0B slider1 0x42 0x40 0x00 0x00 >ET

Hex code: 53 54 3C 10 40 00 0B 73 6C 69 64 65 72 31 42 40 00 00 3E 45 54 27 6D

Slider1 value changing, value: 49.000000

ST< 0x10 0x40 0x00 0x0B slider1 0x42 0x44 0x00 0x00 >ET

Hex code: 53 54 3C 10 40 00 0B 73 6C 69 64 65 72 31 42 44 00 00 3E 45 54 A3 6C

Slider1 value changed to 49.000000

ST<0x10 0x41 0x00 0x0B slider1 0x42 0x44 0x00 0x00>ET

Hex code: 53 54 3C 10 41 00 0B 73 6C 69 64 65 72 31 42 44 00 00 3E 45 54 33 3D

****

2.9 Progress bar customization related

0x1050 progress bar value changed return

0x1051 progress bar percentage return passively, by get_percent command from MCU

Key value:

Last 4 bytes of the data content

Example: 0x42400000 current slider's value: 48.000000 (Type: float compliant with IEEE 754 specification)

progress bar value changed return:

progress_bar value changed to 54.978615

ST< 0x10 0x50 0x00 0x10 progress_bar 0x42 0x5B 0xEA 0x1A >ET

Hex code: 53 54 3C 10 50 00 10 70 72 6F 67 72 65 73 73 5F 62 61 72 42 5B EA 1A 3E 45 54 BF 09

progress_bar value changed to 54.999928

ST< 0x10 0x50 0x00 0x10 progress_bar 0x42 0x5B 0xFF 0xED >ET

Hex code: 53 54 3C 10 50 00 10 70 72 6F 67 72 65 73 73 5F 62 61 72 42 5B FF ED 3E 45 54 08 36

progress_bar value changed to 55.000000

ST< 0x10 0x50 0x00 0x10 progress_bar 0x42 0x5C 0x00 0x00 >ET

Hex code: 53 54 3C 10 50 00 10 70 72 6F 67 72 65 73 73 5F 62 61 72 42 5C 00 00 3E 45 54 C7 16

Get progress_bar percentage by get_percent command as 40%

ST< 0x10 0x51 0x00 0x10 progress_bar 0x00 0x00 0x00 0x28 >ET

Hex code: 53 54 3C 10 51 00 10 70 72 6F 67 72 65 73 73 5F 62 61 72 00 00 00 28 3E 45 54 C7 16

****

2.10 Label related

0x1060 label text content return passively (The display will only return the command after received the get_text command from MCU)

0x1062 label value return (float type)(while the label was targeted by the set_value function on the button, the code will return once the value is changed)

Command format:

“target name”: text content

*The data length must be less than 1024 byte

Float value: the last 4 bytes (Type: float compliant with IEEE 754 specification)

Text content (the content after “target name”:)

Current label text content: Hello Stone Desinger

Hello Stone Designer

Float type value content:

Current value: 123.456

0x42 0xF6 0xE9 0x79

Example:

Label text content returned passively:

Get the text content of “label”:

ST< 0x10 0x60 0x00 0x0D "label":Stone >ET

Hex code: 53 54 3C 10 60 00 0D 22 6C 61 62 65 6C 22 3A 53 74 6F 6E 65 3E 45 54 00 CE

ST< 0x10 0x60 0x00 0x0D "label":12345 >ET

53 54 3C 10 60 00 0D 22 6C 61 62 65 6C 22 3A 31 32 33 34 35 3E 45 54 A4 2B

ST< 0x10 0x60 0x00 0x1C "label":Hello Stone Designer >ET

53 54 3C 10 60 00 1C 22 6C 61 62 65 6C 22 3A 48 65 6C 6C 6F 20 53 74 6F 6E 65 20 44 65 73 69 67 6E 65 72 3E 45 54 00 8B

Label float value return code:

Float value: 1.26

ST<0x10 0x62 0x00 0x09 label 0x3F 0xA1 0x47 0xAE>ET

Hex code: 53 54 3C 10 62 00 09 6C 61 62 65 6C 3F A1 47 AE 3E 45 54 6C 8B

Float value: 8

ST<0x10 0x62 0x00 0x0A label 0x41 0x00 0x00 0x00>ET

Hex code: 53 54 3C 10 62 00 0A 6C 61 62 65 6C 32 41 00 00 00 3E 45 54 C2 99

****

2.11 Edit related

Edit input data has multiple data types, includes text, int, uint, float, ufloat, email, password, phone number, IP, date time, etc. All data types except int, uint, float and ufloat are returning the text, and the others are returning the data.

0x1070 edit text return (Initiatively or passively. It can be returned after the edit text changed, or returned by get_text command)

0x1071 edit data return (int type)

0x1072 edit data return (float type)

Command format:

Text: “target name”: text content

*The text length has to be less than 1024 byte (the text length after “target name”:)

Example:

Edit text content return:

ST<0x10 0x70 0x00 0x10 "edit":abcdefg>ET

Hex code: 53 54 3C 10 70 00 0E 22 65 64 69 74 22 3A 61 62 63 64 65 66 67 3E 45 54 CA EB

*edit text content: abcdefg

ST<0x10 0x70 0x00 0x10 "edit":Stone Designer>ET

Hex code: 53 54 3C 10 70 00 15 22 65 64 69 74 22 3A 53 74 6F 6E 65 20 44 65 73 69 67 6E 65 72 3E 45 54 9E 0F

*edit text content: Stone Designer

Int value: The last 4 bytes of the command data

Example:

ST<0x10 0x71 0x00 0x08 edit 0x00 0x00 0x00 0x7B>ET

Hex code: 53 54 3C 10 71 00 08 65 64 69 74 00 00 00 7B 3E 45 54 B6 5C

*edit int type data return: 123

ST<0x10 0x71 0x00 0x08 edit 0xFF 0xFF 0xFF 0x85>ET

Hex code: 53 54 3C 10 71 00 08 65 64 69 74 FF FF FF 85 3E 45 54 4A 62

*edit int type data return: -123

Float value: The last 4 bytes of the command data(Type: float compliant with IEEE 754 specification)

Example:

ST<0x10 0x72 0x00 0x08 edit 0x42 0xF6 0xE9 0x79>ET

Hex code: 53 54 3C 10 72 00 08 65 64 69 74 42 F6 E9 79 3E 45 54 48 75

edit float type data return:123.456

ST<0x10 0x72 0x00 0x08 edit 0xC2 0xF6 0xE9 0x79>ET

Hex code: 53 54 3C 10 72 00 08 65 64 69 74 C2 F6 E9 79 3E 45 54 80 F4

edit float type data return:-123.456

****

2.12 Text selector related

0x1080 text selector text content return passively (by get_text command)

0x1081 text selector value return initiatively (int type value, can be read by get_value command from MCU)

0x1082 text selector index number return passively (int type value by get_value command from MCU)

Command format:

Text: “widget name”: text content (text length can not be longer than 1024 bytes)

Value: last 4 bytes

Index number: last 4 bytes

Example:

text selector text content return:

text_selector1 text content: 2020

ST<0x10 0x80 0x00 0x15 "text_selector1":2020>ET

Hex code: 53 54 3C 10 80 00 15 22 74 65 78 74 5F 73 65 6C 65 63 74 6F 72 31 22 3A 32 30 32 30 3E 45 54 63 40

text_selector1 text content: yellow

ST<0x10 0x80 0x00 0x17 "text_selector2":yellow>ET

Hex code: 53 54 3C 10 80 00 17 22 74 65 78 74 5F 73 65 6C 65 63 74 6F 72 32 22 3A 79 65 6C 6C 6F 77 3E 45 54 06 5E

text selector value return:

text_selector1 value: 2021

ST<0x10 0x81 0x00 0x12 text_selector1 0x00 0x00 0x07 0xE5>ET

Hex code: 53 54 3C 10 81 00 12 74 65 78 74 5F 73 65 6C 65 63 74 6F 72 31 00 00 07 E5 3E 45 54 FE 5A

Text_selector1 value: 4

ST<0x10 0x81 0x00 0x12 text_selector2 0x00 0x00 0x00 0x04>ET

Hex code: 53 54 3C 10 81 00 12 74 65 78 74 5F 73 65 6C 65 63 74 6F 72 32 00 00 00 04 3E 45 54 17 99

*Notice: if the text selector is in text mode, the returned value will be the current option number, NOT index number. For example, the setting is: *1:red;2:blue;3:green;4:yellow;5:grey;* the text right now should be *yellow*

text selector index number return:

text_selector1 current option index number: 50 (the 51st option)

ST<0x10 0x82 0x00 0x12 text_selector1 0x00 0x00 0x00 0x32>ET

Hex code: 53 54 3C 10 82 00 12 74 65 78 74 5F 73 65 6C 65 63 74 6F 72 31 00 00 00 32 3E 45 54 75 32

Text_selector1 current option index number: 0 (the first option)

ST<0x10 0x82 0x00 0x12 text_selector2 0x00 0x00 0x00 0x00>ET

Hex code: 53 54 3C 10 82 00 12 74 65 78 74 5F 73 65 6C 65 63 74 6F 72 32 00 00 00 05 3E 45 54 14 7C

****

2.13 Image_value related

0x1092 image_value value return (float type, can be returned initiatively or passively)

Command format:

image_value value: last 4 bytes

Example:

image_value value return:

ST<0x10 0x92 0x00 0x0F image_value 0x40 0x87 0x5C 0x29>ET

Hex code: 53 54 3C 10 92 00 0F 69 6D 61 67 65 5F 76 61 6C 75 65 40 87 5C 29 3E 45 54 F6 DB

****