

Smart Panlee Rotary Switch Screen

WT32S3-21S
(ZX2D10GE01R-W4848#B)



Features:

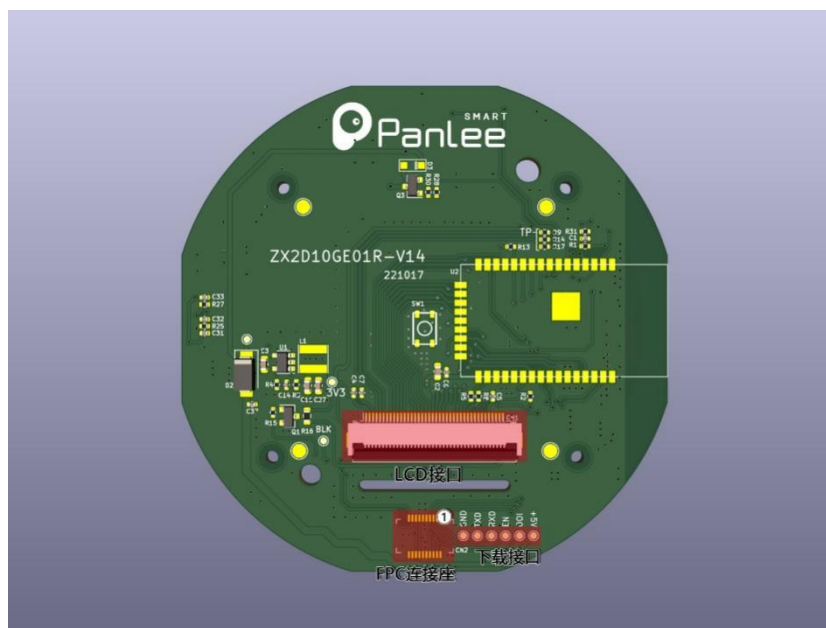
1. Support rapid prototyping

Core Materials (Tab. 0):

No.	Name	Model	Remark
1	ESP32-S3 module	WT32-S3-WROVER-N16R8	
2			
3			

Hardware Interface:

Hardware Interface Diagram:



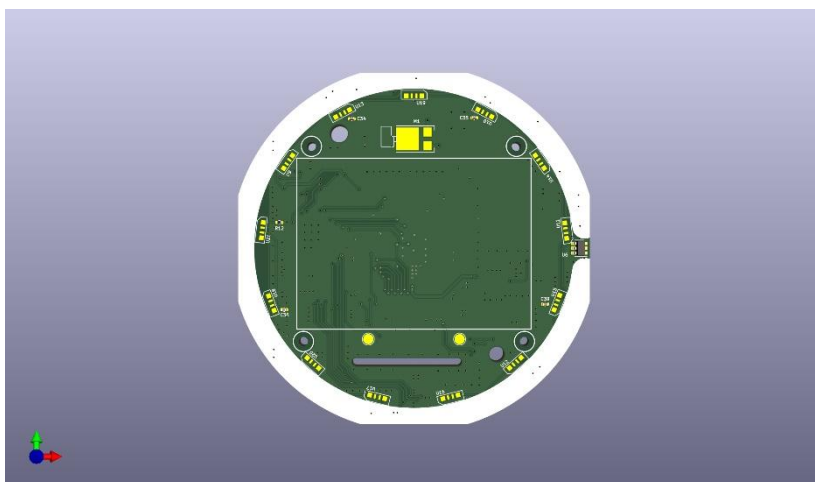


Fig.1 Hardware Interface

Interface Description:

[1] LCD Interface (Tab.1)

Description	Module Pin	Remark
TP_SCL	GPIO 6	<i>Touch IIC bus clock, reused with external interface, no touch at present</i>
TP_SDA	GPIO 5	<i>Touch IIC bus data, reused with external interface, no touch at present</i>
TP_INT	GPIO 7	<i>Touch interrupt, no touch at present</i>
LCD_RST	EN	-
LCD_CS	GPIO 21	LCD SPI bus CS
LCD_SCLK	GPIO 47	LCD SPI bus SCLK
LCD_MOSI	GPIO 41	LCD SPI bus MOSI
RGB_PCLK	GPIO 45	LCD RGB interface PCLK
RGB_DE	GPIO 39	LCD RGB interface DE
RGB_VS	GPIO 48	LCD RGB interface VS
RGB_HS	GPIO 40	LCD RGB interface HS
RGB_D0	GPIO 47	LCD RGB interface D0
RGB_D1	GPIO 41	LCD RGB interface D1

RGB_D2	GPIO 0	LCD RGB interface D2
RGB_D3	GPIO 42	LCD RGB interface D3
RGB_D4	GPIO 14	LCD RGB interface D4
RGB_D5	GPIO 8	LCD RGB interface D5
RGB_D6	GPIO 13	LCD RGB interface D6
RGB_D7	GPIO 18	LCD RGB interface D7
RGB_D8	GPIO 12	LCD RGB interface D8
RGB_D9	GPIO 11	LCD RGB interface D9
RGB_D10	GPIO 17	LCD RGB interface D10
RGB_D11	GPIO 10	LCD RGB interface D11
RGB_D12	GPIO 16	LCD RGB interface D12
RGB_D13	GPIO 9	LCD RGB interface D13
RGB_D14	GPIO 15	LCD RGB interface D14
RGB_D15	GPIO 46	LCD RGB interface D15
LCD_BL	GPIO 38	LCD backlight control, active high

[2] FPC Connector Interface (Tab.2)

Pin	Description	Module Pin	Voltage Range	Remark
1	ESP_TXD	TXD0	3.3V TTL	Download serial port TXD
2	EXT_IO4	GPIO 1	0~3.3V	Output interface 4
3	ESP_RXD	RXD0	3.3V TTL	Download serial port RXD
4	EXT_IO3	GPIO 2	0~3.3V	Output interface 3
5	EN	EN	0~3.3V	Chip enable
6	EXT_IO2	GPIO 20	0~3.3V	Output interface 2, multiplexed with USB
7	BOOT	GPIO 0	0~3.3V	Boot mode select

8	EXT_IO1	GPIO 19	0~3.3V	Output interface 1, multiplexed with USB
9~12	GND	-	0V	Ground
13、14	USB_DP	GPIO 20	0~3.3V	USB pin, which can be used as general IO
15、16	USB_DN	GPIO 19	0~3.3V	USB pin, which can be used as general IO
17、19	ICCIInterface	-	-	Not connected in the current version
18、20	+5V	-	+5V±5%	5V power supply

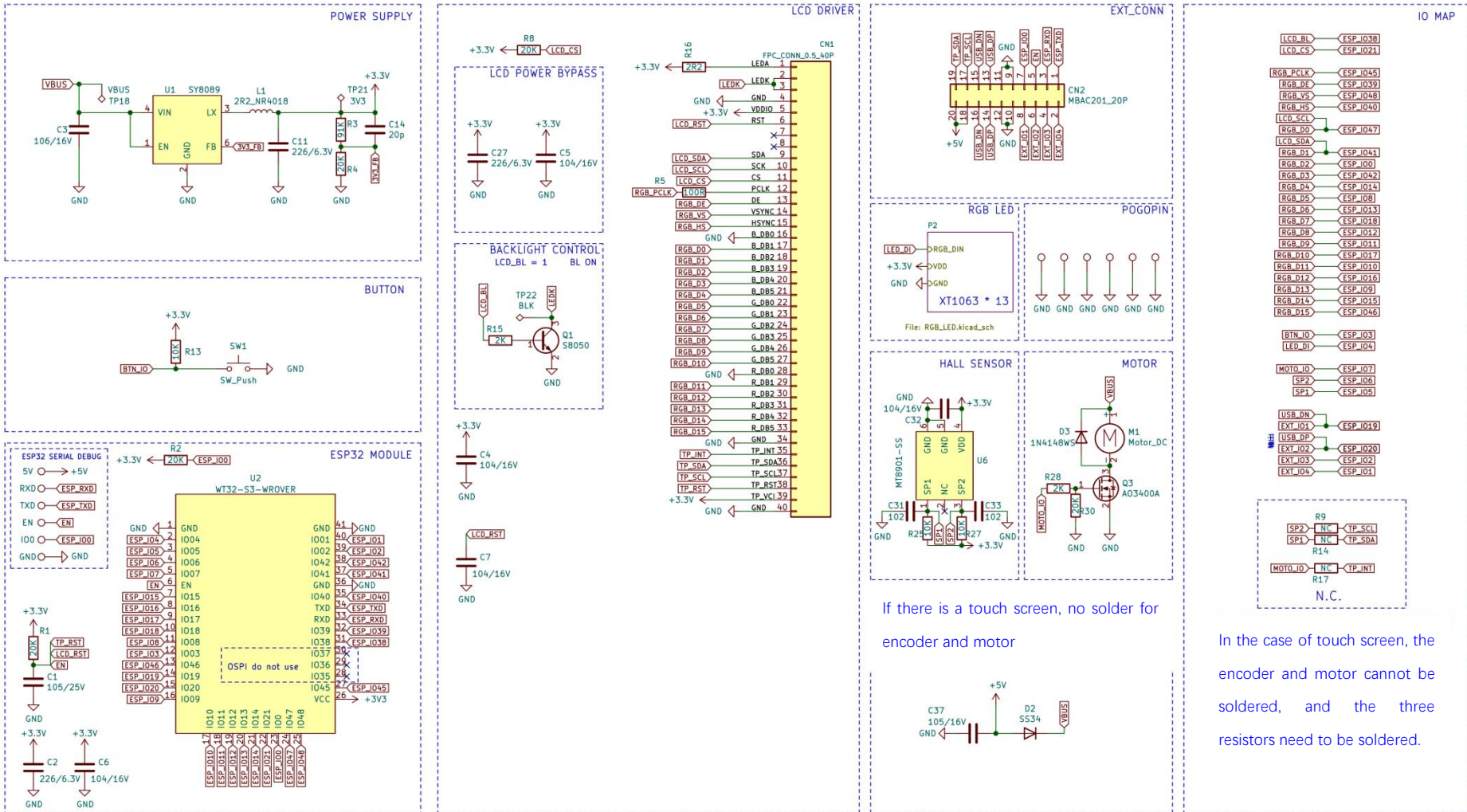
[3] Download Interface (Tab.3)

Pin	Description	Module Pin	Voltage Range	Remark
1	+5V	-	+5V±5%	5V power supply
2	BOOT	GPIO 0	0~3.3V	Boot mode select
3	EN	EN	0~3.3V	Chip enable
4	ESP_RXD	RXD0	3.3V TTL	Download serial port RXD
5	ESP_TXD	TXD0	3.3V TTL	Download serial port TXD
6	GND	-	0V	Ground

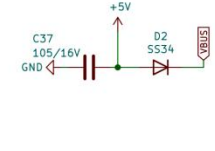
Hardware Peripherals:

Peripheral	Description	Related IO
Button	The button is pulled up, grounded when pressed	GPIO 3
LCD	LCD with RGB interface, no touch, 480 * 480	-
LED	Single bus LED, 13 in total	GPIO 4
Hall encoder	Hall encoder is used for rotating operation in human-computer interaction, and orthogonal encoder is adopted.	GPIO 5、6
Vibrating motor	Feedback for human-computer interaction	GPIO 7

Schematic:

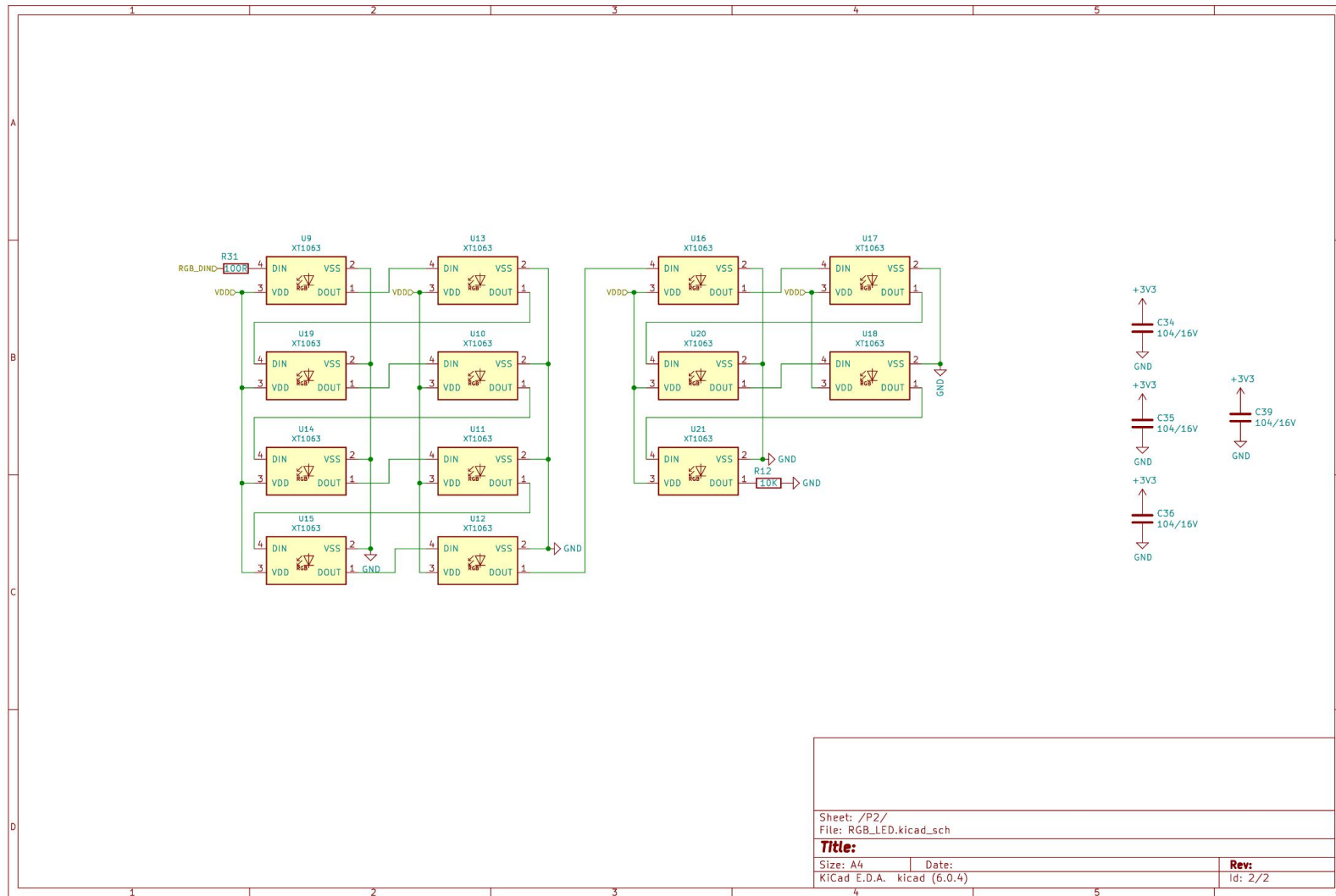


If there is a touch screen, no solder for encoder and motor



In the case of touch screen, the encoder and motor cannot be soldered, and the three resistors need to be soldered.





Specification Parameters:

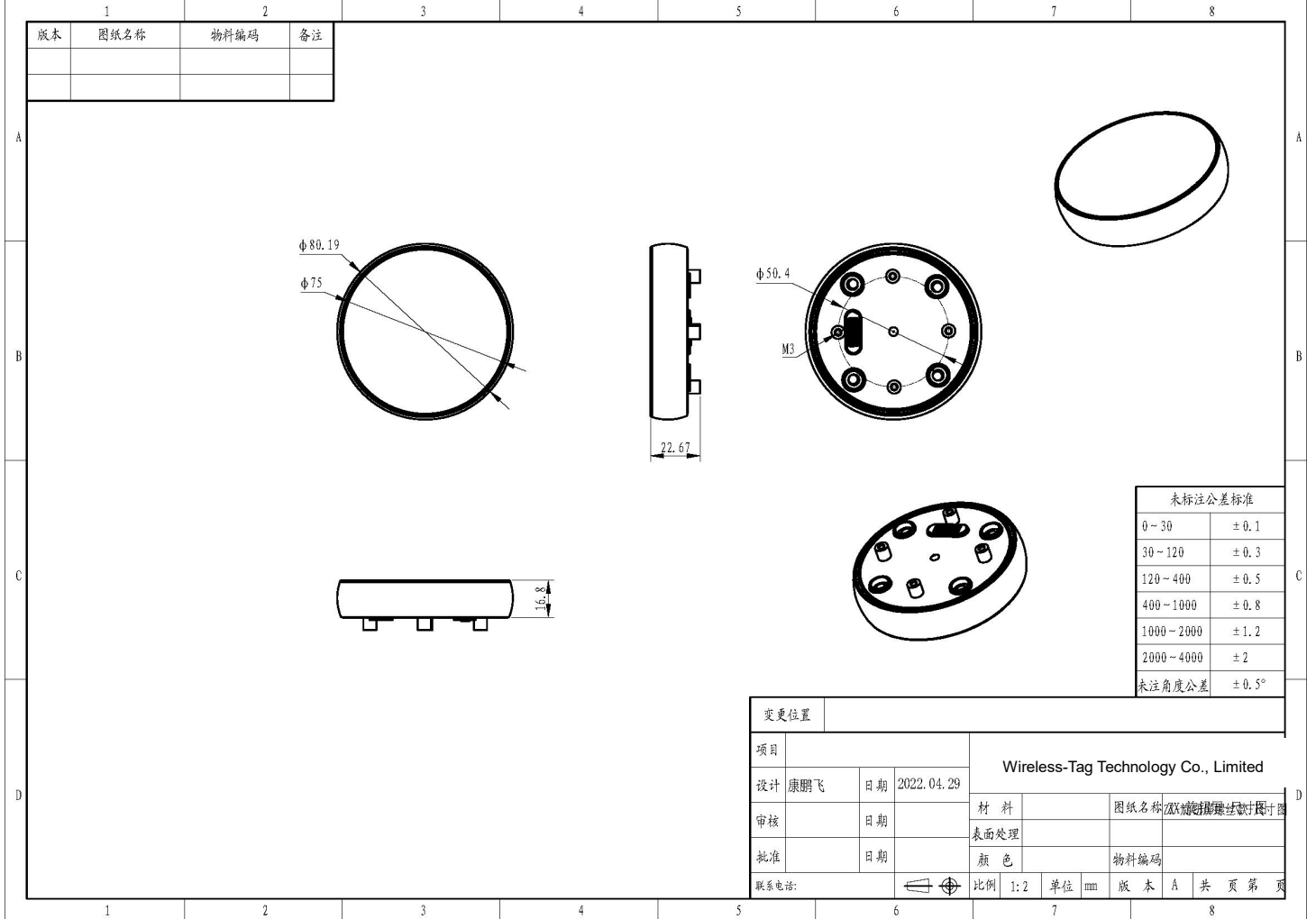
[1] LCD Parameters (Tab.5)

Display Type	IPS
Driver IC Model	ST7701S
Viewing Angle	Full Veiw
Resolution	480*480
Interface	RGB
Color	RGB565

[2] Touch Parameters (Tab.6)

Touchscreen Type	-
Driver IC Model	-
Interface	-
Touchscreen Structure	-
Touch Mode	-
Surface Hardness	-
Light Transmittance	-

Outline Dimensional Drawing (Fig.2)



Firmware Burning:

1. Connect the downloader (ZXACC-ESPDB) via a USB-Type C cable. And then connect the WT32S3-21S board with the downloader (ZXACC-ESPDB) through an MX1.25-7P cable. As the downloader (ZXACC-ESPDB) has automatic data flow processing capabilities, the firmware can be downloaded automatically through the ESP32 Flash Download Tools.

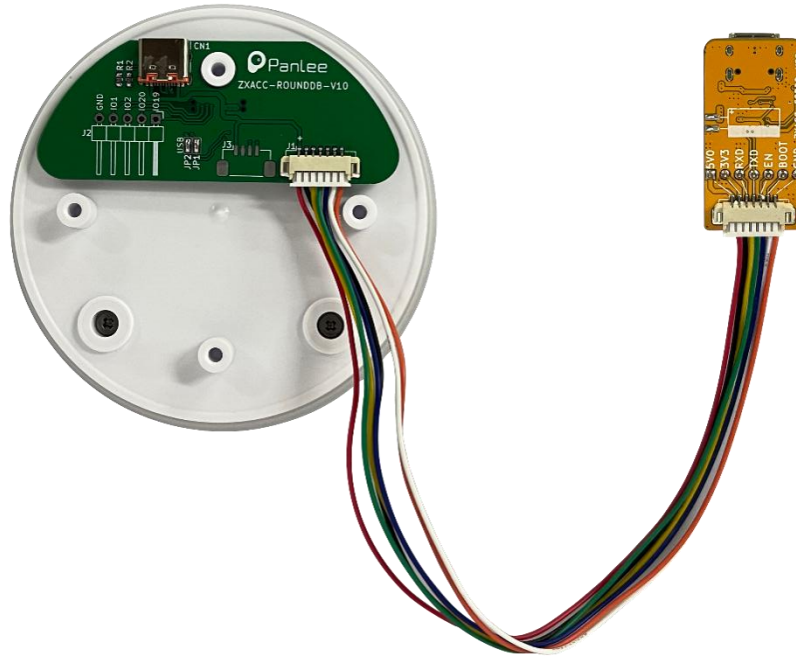


Fig.3

2. As shown in Fig. 4: Select the firmware path at mark 1, and then fill in the burning address, usually 0X00. Note that this checkbox must be checked; Set the crystal frequency to be 40MHz at mark 2; Select 32Mbit for Flash size at mark 3; Select DIO for SPI MODE at mark 4; Select the COM port number recognized by the computer at mark 5; Select the baud rate at mark 6 (the higher the value is, the faster the firmware will be downloaded. Max. 1152000bps).

3. After the previous configuration, click START at mark 7 to start burning the firmware.

4. Complete the above steps, and then press the reset button on the back of the development board to start running the firmware you just burned.

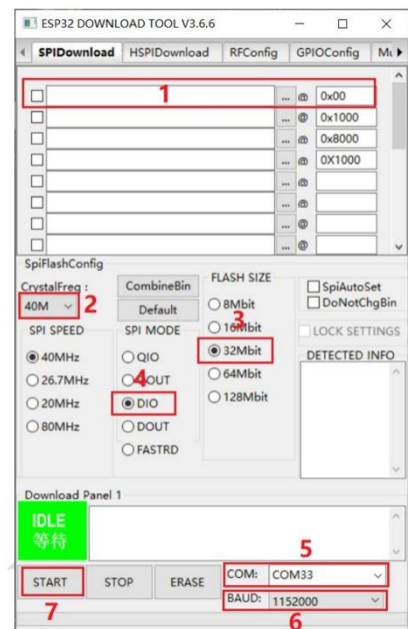


Fig.4

Software design reference:

URL: <https://www.espressif.com.cn/en/support/documents/technical-documents>

Online GUI Designer:

Users can use our online GUI designer platform, which is similar to MIT APP Inventor, to realize the rapid GUI development with building blocks. Currently, the platform has perfected the graphic interface development, and more driver code blocks will be further improved in the future.

Login Page: <http://8ms.xyz/login>

User Manual: <https://doc.panel-tag.com/ESP32-S3/index.html>

For Arduino users, please refer to the link:

https://github.com/smartpanle/PanelAn_esp32_arduino

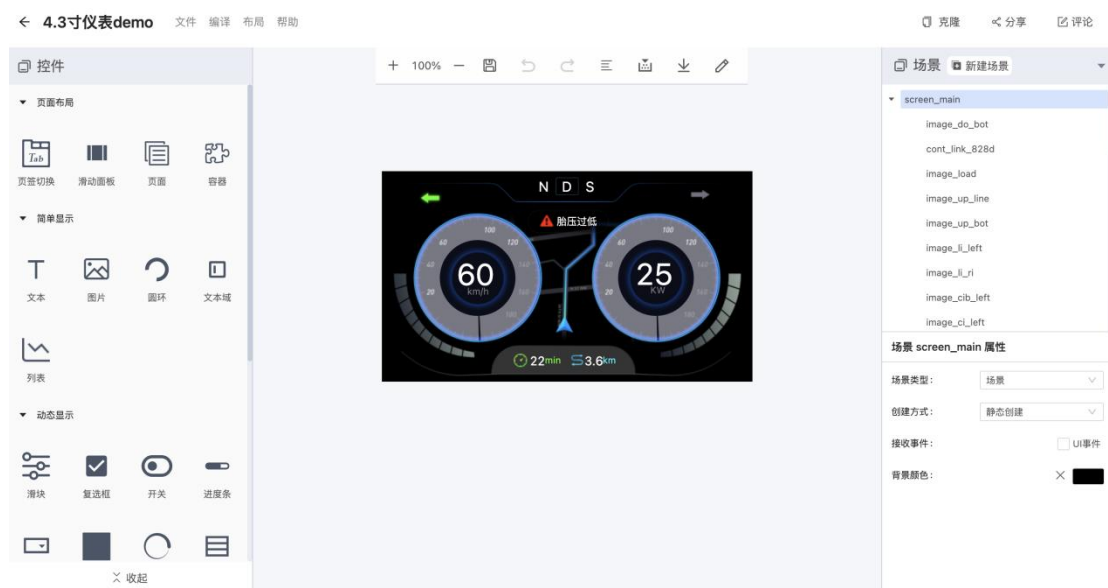


Fig.5

搜索

Blockly Code 返回旧版

```
int left_num = 10; int left_num_sign = 0;

h_arcm1_arc_left_arcnum; h_arcm1_path1_arc_left;

static void h_arcm1_size_obvious var; int32_t;
h_arcm1_arc_angle(arc_left, v);

static void h_arcm1_size_obvious;
h_arcm1_arc_angle(arc_right, v);

static void h_arcm1_size_obvious;
h_arcm1_path1_arc_right;
h_arcm1_path1_arc_right;
h_arcm1_path1_arc_right;
h_arcm1_path1_arc_right;

h_arcm1_size_obvious;
h_arcm1_size_obvious;
h_arcm1_size_obvious;
h_arcm1_size_obvious;

switch (h_arcm1_size_obvious) {
case 1: h_arcm1_size_obvious;
case 2: h_arcm1_size_obvious;
}

switch (h_arcm1_size_obvious) {
case 1: h_arcm1_size_obvious;
case 2: h_arcm1_size_obvious;
}

switch (h_arcm1_size_obvious) {
case 1: h_arcm1_size_obvious;
case 2: h_arcm1_size_obvious;
}

switch (h_arcm1_size_obvious) {
case 1: h_arcm1_size_obvious;
case 2: h_arcm1_size_obvious;
}
```