SMART DISPLAY MODULE SPECIFICATION

3.5 Inch Smart Display with TOUCH	
Model:	UEDX32480035E-WB-A
Version:	V1.0
Date:	2024-11-08

Customer Confirmation

Approved by	Notes

REVISION HISTORY

Revision	Date	Contents of Revision Change	Remark
V1.0	202401108	Preliminary release	

TABLE of CONTENTS

1. Introduction

1.1 Features

Brief Info:

1) Button control: one is the reset button, the other is the boot button.
2) Backup IO: download ports and multiple IO leads to use on both sides of the periphery.
3) Power: DC 5V, 200mA

System

1) OS: RTOS
2) CPU: ESP32-S3 240Mhz
3) RAM: 8MB
4) Flash: 16MB
5) Interface: UART/USB
6) Support 2.4GHz Wi-Fi、BLE 5、BLE Mesh
7) Support Peripherals:

   GPIO, SPI, LCD interface, Camera interface, UART, I2C, I2S, remote control,
   pulse counter,LED PWM, full-speed USB 2.0 OTG, USB Serial/JTAG controller, MCPWM,
   SDIO host, GDMA, TWAI® controller (compatible with ISO 11898-1), ADC, touch sensor,
   temperature sensor, timers and watchdogs

For more information on ESP32-S3-WROOM-1, please refer to the following link:datasheet_en.pdf

Display

1) Size:3.5 Inch
2) Resolution: 320(W)*3(RGB)*480(H)
3) Mode: MCU 8/16BIT/SPI
4) Display mode：Normally Black
5) Driver IC：ST7365P
6) Touch IC：CHSC6540
7) Pixel Density：166 PPI
8) Backlight Type：White LED
9) Brightness: 250 cd/m2

More information about Display can be found here:Display Specification.pdf

Other

1) Operation Temperature：-20~70℃
2) Storage Temperature：-30~80℃

1.2 Appearance picture
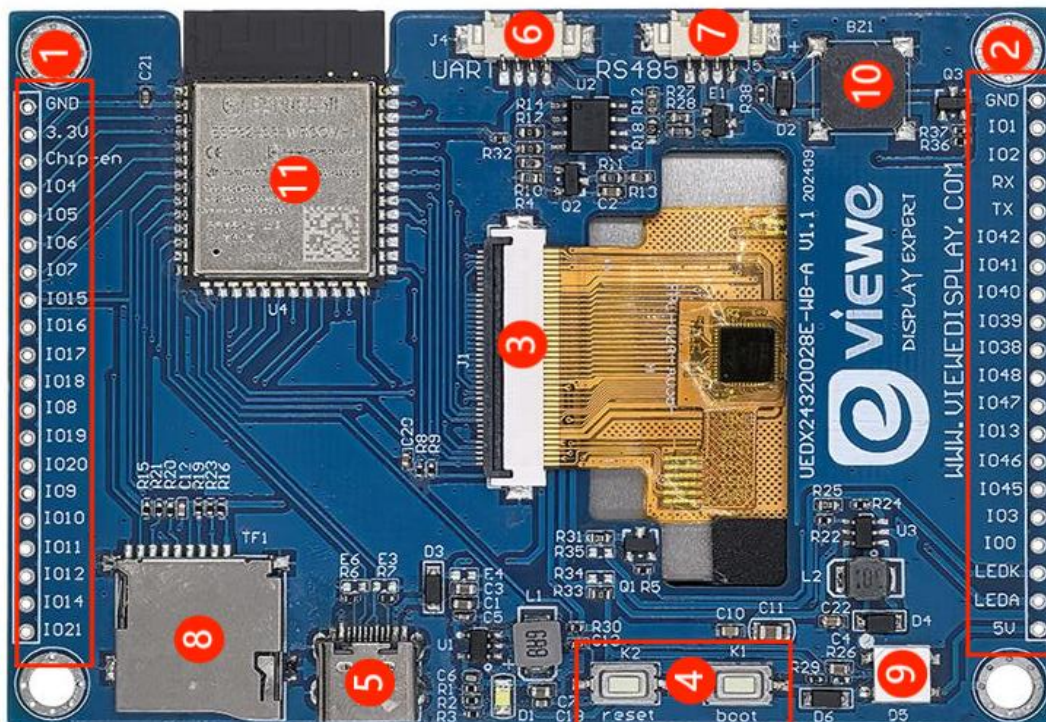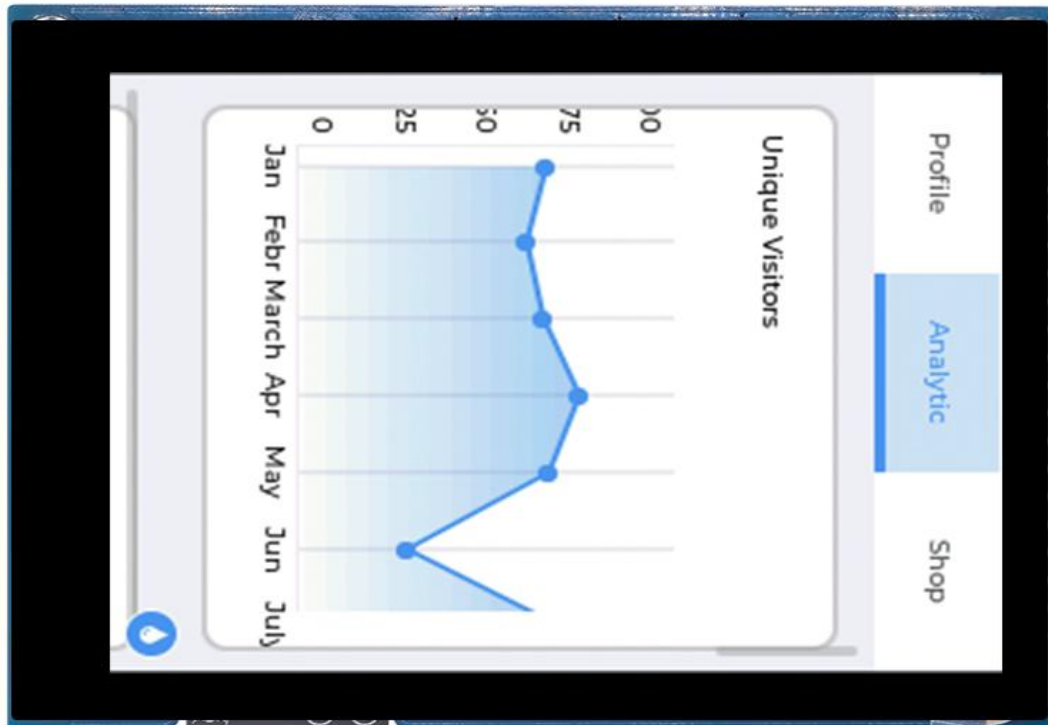
The hardware marked above will be described one by one by number in 2

2. Product information

2.1 Interface Description

(1) External GPIO:J3

Symbol	Description	Current Usage
GND	Grounds	GND
3.3V	Power 3.3V	Power 3.3V
CHIP-EN	High: on, enables the chip Low: off, the chip powers off. Note: Do not leave the EN pin floating	Note: Do not leave the EN pin floating
GPIO4	RTC_GPIO4, GPIO4, TOUCH4, ADC1_CH3	TP-INT
GPIO5	RTC_GPIO5, GPIO5, TOUCH5, ADC1_CH4	Not used
GPIO6	RTC_GPIO6, GPIO6, TOUCH6, ADC1_CH5	Not used
GPIO7	RTC_GPIO7, GPIO7, TOUCH7, ADC1_CH6	Not used
GPIO15	RTC_GPIO15, GPIO15, U0RTS, ADC2_CH4, XTAL_32K_P	SD-D2
GPIO16	RTC_GPIO16, GPIO16, U0CTS, ADC2_CH5, XTAL_32K_N	SD-MISO
GPIO17	RTC_GPIO17, GPIO17, U1TXD, ADC2_CH6	SD-MOSI
GPIO18	RTC_GPIO18, GPIO18, U1RXD, ADC2_CH7, CLK_OUT3	SD-D1
GPIO8	RTC_GPIO8, GPIO8, TOUCH8, ADC1_CH7, SUBSPICS1	Not used
GPIO19	RTC_GPIO19, GPIO19, U1RTS, ADC2_CH8, CLK_OUT2, USB_D-	USB-DN
GPIO20	RTC_GPIO20, GPIO20, U1CTS, ADC2_CH9, CLK_OUT1, USB_D+	USB-DP
GPIO9	RTC_GPIO9, GPIO9, TOUCH9, ADC1_CH8, FSPIHD, SUBSPIHD	Not used
GPIO10	RTC_GPIO10, GPIO10, TOUCH10, ADC1_CH9, FSPICS0, FSPIIO4, SUBSPICS0	Not used
GPIO11	RTC_GPIO11, GPIO11, TOUCH11, ADC2_CH0, FSPID, FSPIIO5, SUBSPID	Not used
GPIO12	RTC_GPIO12, GPIO12, TOUCH12, ADC2_CH1, FSPICLK, FSPIIO6, SUBSPICLK	Not used
GPIO14	RTC_GPIO14, GPIO14, TOUCH14, ADC2_CH3, FSPIWP, FSPIDQS, SUBSPIWP	SD-SCLK
GPIO21	RTC_GPIO21, GPIO21	SD-D3

(2) External GPIO:J2

Symbol	Description	Current Usage
GND	Grounds	Grounds
GPIO1	RTC_GPIO1, GPIO1, TOUCH1, ADC1_CH0	TP-SDA
GPIO2	RTC_GPIO2, GPIO2, TOUCH2, ADC1_CH1	TP-RST
GPIO44	U0RXD, GPIO44, CLK_OUT2	UARTRX
GPIO43	U0TXD, GPIO43, CLK_OUT1	UARTTX
GPIO42	MTMS, GPIO42	LCD-CSB
GPIO41	MTDI, GPIO41, CLK_OUT1	LCD-RS
GPIO40	MTDO, GPIO40, CLK_OUT2	LCD-SCL
GPIO39	MTCK, GPIO39, CLK_OUT3, SUBSPICS1	LCD-RST
GPIO38	GPIO38, FSPIWP, SUBSPIWP	Not used
GPIO48	SPICLK_N_DIFF,GPIO48, SUBSPICLK_N_DIFF	LCD-M1
GPIO47	SPICLK_P_DIFF,GPIO47, SUBSPICLK_P_DIFF	LCD-M0
GPIO13	RTC_GPIO13, GPIO13, TOUCH13, ADC2_CH2, FSPIQ, FSPIIO7, SUBSPIQ	BL-EN
GPIO46	GPIO46	SDO
GPIO45	GPIO45	SDI
GPIO3	RTC_GPIO3, GPIO3, TOUCH3, ADC1_CH2	TP-SCL
GPIO0	RTC_GPIO0, GPIO0	RGB-LED
LEDK	BL-	BL-
LEDA	BL+	BL+
5V	Power 5V	5V

Note: (1) (2)
- A pin can be used for other purposes when it is not used at the same time
- You can also use an external gpio to drive other interface types, such as SPI interface, MCU interface, MIPI interface, etc., without using the Display interface provided by us
- If the Display interface is used but the SD is not used, then the SD pins are freely usable.

(3) Display Interface

Pin No.	Symbol	I/O	Description
1	XL/CTP-SCL	I	I2C clock signals for CTP; Option XL for RTP
2	YU/CTP-SDA	I	I2C data signal for CTP,Option YU for RTP

3	XR/CTP-RST	I	The signal will reset the CTP,Signal is active low, Option XR for RTP
4	YD/CTP-INT	I/O	Interrupt signals for CTP,Option YD for RTP
5	GND	P	Power Ground
6	IOVCC	P	Power supply for I/O system
7	VCI	P	Power supply for analog circuits
8	TE	O	Tearing effect signal is used to synchronize MCU to frame memory
9	SPI_CS/MCU_CS	I	Chip selection pin. Low-active
10	SPI_SCL/MCU_RS	I	Display data/command selection pin in MCU interface In SPI mode, this pin is used as SCL
11	SPI_RS/MCU_WR	I	Write enable in MCU parallel interface RS=1 display data or parameter;RS=0 register index / command
12	MCU_RD	I	Read enable in 8080 MCU parallel interface. Low-active.
13	SPI_SDA	I/O	Serial communication data input and output, internal pull low.
14	SPI_SDO	O	SPI interface output pin
15	RESET	I	The signal will reset the LCM, Signal is active low.
16	GND	P	Power Ground
17-32	DB0-DB15	I/O	data bus for MCU
33	LED-A	P	Power supply for backlight anode
34-36	LED-K	P	Power supply for backlight cathode
37	GND	P	Power Ground
38	IM0	I	The MCU interface mode select.
39	IM1	I	The MCU interface mode select.
40	IM2	I	The MCU interface mode select.

The choice of IM is as follows: 1 means pull up, and 0 means pull down
IM2IM1IM0:   110-2.4inch   111-3.5inch   101-3.9/4.3inch

(4) button:

The following picture shows the boot button on the left and the reset button on the right.

(5) Power:

The following figure is the schematic diagram of USB. USB is used for power and download.

(6) UART:

a hardware communication protocol that enables asynchronous serial communication between devices, allowing for the transmission and reception of data using two wires: one for transmitting (TX) and one for receiving (RX).

(7) RS485:

a standard for serial communication that allows for long-distance data transmission and multi-point connections, using differential signaling to improve noise immunity and enabling communication over distances up to 1,200 meters at speeds of up to 10 Mbps.

(8) SD:

A port or slot for inserting an SD card, usually for storing data

(9) RGB LCD:

Can emit red, green, blue three colors of light, and through different combinations to produce a variety of colors

(10) Passive Buzzer:

An external circuit is required to generate a driving signal (usually a square wave) to sound

(11) Main Control Chip: ESP32S3

Dual-core processor, up to 240MHz operating frequency

2.2 Display Information

Item	Specification	Unit	Remark
Pixel Driving element	IPS TFT	-	-
Screen Size	3.5	Inch	Diagonal
Resolution	320(W)*3(RGB)*480(H)	Dots	-
Interface	MCU 8/16BIT/SPI	-	40PIN
Module Power Consumption	0.584	Watt	Typ.
Active Area	48.96(W)*73.44(H)	mm	-
Pixel pitch (W*H)	0.153(W)*0.153(H)	mm	-
Module Size (W*H*D)	61.5(W)*89.97(H)*3.48(D)	mm	-
Luminance	250	cd/m^2	Typ.
Viewing Direction	ALL	O'clock	-
Display Color	262K	Colors	18bits

2.3 Voltage & Current

Item	Conditions	Min	Typ	Max	Unit
Power Voltage	DC	4. 0	5.0	5.5	V
Operation Current	VCC= +5V, Maximum backlight current	-	200	-	mA
	VCC= +5V,backlight off	-	100	-	mA
Recommended power supply:5V 1A DC					

2.4 Reliability Test

| Item	Conditions	Min	Typ	Max	Unit
Working Temperature	60%RH at 5V voltage	-20	25	70	C
Storage Temperature	---	-30	25	80	C
Working Humidity	25°C	10%	60%	90%	RH
ESD	---	Contact: ±4KV Air: ±8KV			KV

2.5 Related software

Software name	Version	Software associated configuration	Development environment configuration link
Arduino IDE	esp32 (3.0.0-3.0.4)	1. Board: ESP32S3 Dev Module 2. CPU Frequency: 240MHz (WiFi) 3. Flash Frequency: NO 4. Flash Mode: QIO 80MHz 5. Flash Size: 16MB (128Mb) 6. Partition Scheme: Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS) 7. PSRAM: OPI PSRAM 8. Programmer: Esptool	ESP32-Arduino config (github.com)
ESP-IDF	5.1.1 5.2.2 5.3	Once configured, no configuration is required（If you have any problem with the configuration, please contact us, we will help you）	ESP-IDF config (github.com)

3. MECHANICAL DRAWING

3.1 Display

4. Schematic
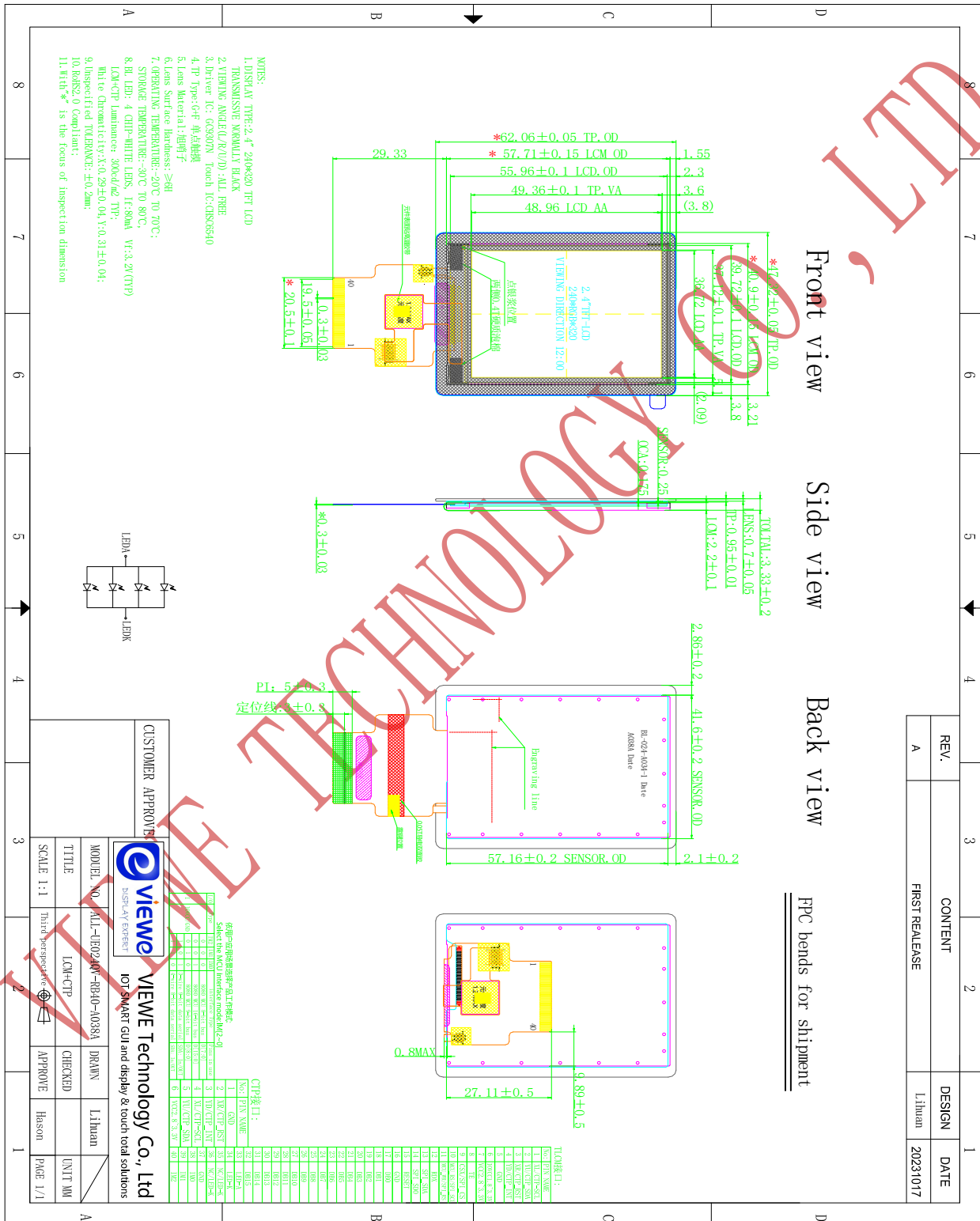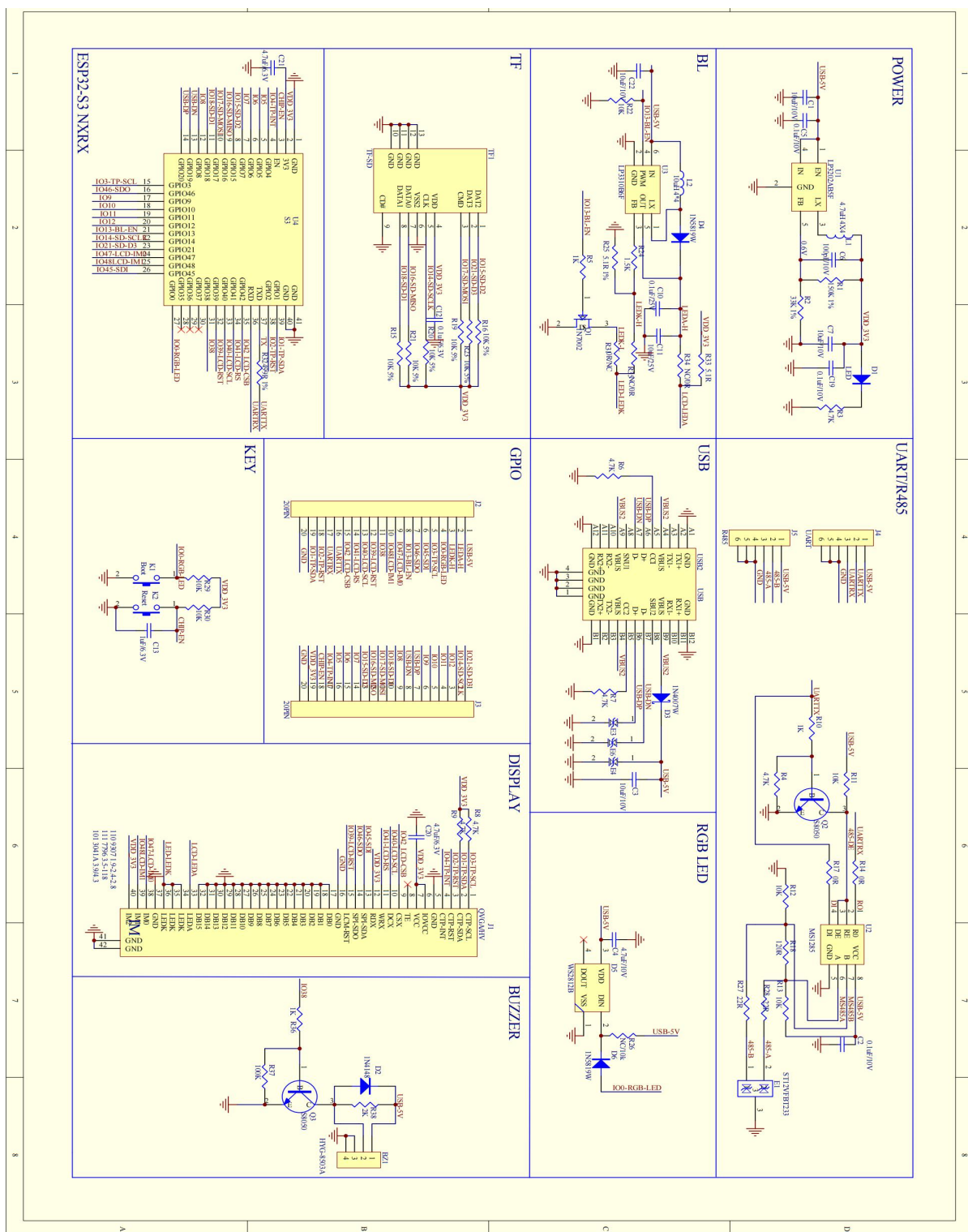
5. Related downloads

5.1 Arduino relevant information

Download the ESP32_Display_Panel library directly from Arduino to use the use case

5.2 Libraries required for Arduino

Download ESP32_Display_Panel library and dependency library directly in Arduino to use

5.3 IDF relevant information

https://github.com/VIEWESMART/ESP32-IDF/tree/main/examples/3.5inch/High-resolution_240320/UEDX24_28_35E-WB-A-General-SDK