

T5UIC1 DWIN OS Program Development Guide

T5UIC1 is a simplified serial command screen designed for applications that do not require a touch screen, simple UI functions, and demanding cost requirements based on the T5 CPU.

Its main features include:

(1) 65K color TFT display.

(2) 384Kbytes font space.

Stored 6*12-32*64 dot matrix ASCII and 12*12-64*64 dot matrix GB2312 Chinese character library (Chinese characters are scaled based on 16*16 dot matrix).

(3) 512Kbytes image and icon storage space, divided into 16 storage spaces according to 32KB.

It can store up to 16 JPEG full-screen pictures.

Or store 0-16 JPEG icon library files (a single icon library file can exceed 32KB and occupy multiple memory spaces).

(4) 32KBytes SRAM data memory that can be read and written by serial port, the data is lost when power is off, and it is initialized to 0x00 when power on.

Mainly used for online pictures, icon library data update, or real-time JPEG icon, picture display.

(5) 16Kbytes Flash data memory that can be read and written by serial port, the data will not be lost when power off, and the write life is 100,000 times.

Mainly used for data storage such as user configuration parameters.

(6) SD/SDHC interface configuration parameters and update fonts and pictures.

(7) The CPU can be configured to run at 250MHz or 400MHz.

2.3 Instruction set

(1) Configuration and interface instructions

Features	instruction	Description
shake hands	0x00	Example: Tx: AA 00 CC 33 C3 3C Rx: AA 00 4F 4B CC 33 C3 3C
Backlight brightness adjustment	0x30	DIM_Set: backlight brightness value, 0x00-0xFF. 0x00 backlight is off, 0xFF backlight is the brightest, and 0x01-0x1F setting value backlight

		<p>may flash.</p> <p>The power-on default value is 0xFF.</p> <p>Example: AA 30 80 CC 33 C3 3C adjust the brightness to 50%.</p>
Write data memory	0x31	<p>Processing time, SRAM can be ignored; Flash takes up to 1 second.</p> <p>Type: Write memory selection, 0x5A=32KB SRAM, 0xA5=16KB Flash.</p> <p>Address: write data memory address, 0x0000-0x7FFF or 0x3FFF.</p> <p>Datas: The data string to be written.</p> <p>Example: AA 31 5A 00 00 31 32 33 34 CC 33 C3 3C write SRAM</p>
Read data memory	0x32	<p>Processing time, SRAM can be ignored, and Flash delay is about 0.1mS.</p> <p>Type: Read memory selection, 0x5A=32KB SRAM, 0xA5=16KB Flash.</p> <p>Address: write data memory address, 0x0000-0x7FFF or 0x3FFF.</p> <p>Length: The length of the read data byte, 0x01-0xF0.</p> <p>Datas: The read data string.</p> <p>Example:</p> <p>Tx: AA 32 5A 00 00 04 CC 33 C3 3C read SRAM</p> <p>Rx: AA 32 5A 00 00 04 31 32 33 34 CC 33 C3 3C data response</p>
Write picture memory	0x33	<p>The processing time can take up to 2 seconds.</p> <p>Write the contents of the 32KB SRAM data memory into the designated image memory space.</p> <p>PIC_ID: Picture memory space location, 0x00-0x0F, each space is 32Kbytes.</p> <p>Example: Tx: AA 33 5A A5 00 CC 33 C3 3C</p>
Expansion serial port configuration	0x38	<p>Bode_Set: Set the baud rate of the extended serial port, 0x0001-0x03FF.</p> <p>Bode_Set=15667200/baud rate, the lowest baud rate is 15300.</p> <p>The power-on default value is 0x0088, which corresponds to a baud rate of 115200bps.</p> <p>Example:</p> <p>AA 38 03 30 CC 33 C3 3C</p> <p>sets the baud rate of the extended serial port to 19200bps.</p>
Expansion serial port data transmission	0x39	<p>Send the Datas packet from the extended serial port.</p> <p>Example:</p> <p>AA 39 31 32 33 34 35 36 37 38 39 CC 33 C3 3C</p> <p>sends the character string "123456789" from the extended serial port.</p>
Expansion serial port data receiving	0x3A	<p>The screen actively uploads the data received by the extended serial port.</p> <p>Len_Data: The length of the data uploaded this time.</p>

Datas: The data uploaded this time.

Example:

Assuming that the extended serial port receives a byte of data 0x55, the screen will automatically upload
AA 3A 01 55 CC 33 C3 3C.

(2) Drawing related instructions

instruction	Description
0x01	<p>Clear the screen; processing time 1.5mS (corresponding to 400MHz main frequency, the same below).</p> <p>Color: Clear screen color.</p> <p>Example: AA 01 00 1F CC 33 C3 3C</p>
0x02	<p>Set points; processing time=$0.4 \times N_x \times N_y \times \text{number of set points}$ uS.</p> <p>Color: Set point color.</p> <p>N_x: Actual pixel size in X direction, 0x01-0x0F.</p> <p>N_y: actual pixel size in Y direction, 0x01-0x0F.</p> <p>(X_n, Y_n): Set point coordinate sequence.</p> <p>Example: AA 02 F8 00 04 04 00 08 00 08 01 00 01 00 CC 33 C3 3C</p>
0x03	<p>End point connection; processing time=$0.5 \times \text{Max}(\text{length of line segment in X direction, length of line segment in Y direction})$ uS.</p> <p>Color: Connection color, 2Bytes.</p> <p>(X_n, Y_n): End point coordinates of the line segment.</p> <p>Example: AA 03 FF FF 00 40 00 40 01 00 01 00 CC 33 C3 3C</p>
0x05	<p>Rectangular area display; processing time=$0.14 \times \text{number of pixels}$ uS.</p> <p>Mode:</p> <p>0x00=Color color display rectangular frame.</p> <p>0x01=Color fills the rectangular area with color.</p> <p>0x02=Color XOR rectangle area data, mostly used for menu selection/unselection coloring.</p> <p>Color: color.</p> <p>(X_s, Y_s), (X_e, Y_e): The coordinates of the upper left and lower right corners of the rectangle.</p> <p>Example: AA 05 02 07 E0 00 40 00 40 01 00 01 00 CC 33 C3 3C</p>
0x09	<p>The screen area moves; processing time=$0.20 \times \text{the number of pixels in the moving area}$ uS .</p> <p>Mode: Movement mode</p> <p>. 7: Movement mode, 0=circular movement. 1=Translation, the vacant area</p>

	<p>is filled with color.</p> <p>.6-.4: Write 0.</p> <p>.3-.0: moving direction, 0x00=left. 0x01=To the right. 0x02=Up. 0x03=Down.</p> <p>DIS: moving distance, number of pixel dots, 0x0000-horizontal resolution/2, 2Bytes.</p> <p>Color: Fill color, only valid when DIR.7=1.</p> <p>(Xs, Ys): The coordinates of the upper left corner of the selected area.</p> <p>(Xe, Ye): The coordinates of the lower right corner of the selected area.</p> <p>Example: AA 09 00 00 08 FF FF 00 40 00 40 01 00 01 00 CC 33 C3 3C</p>
--	---

(3) Text related instructions

instruction	Description
0x11	<p>Character string display; 1 16*16 dot matrix Chinese character processing time is 76uS, and the rest are converted according to the ratio of dot matrix .</p> <p>Mode: Display mode.</p> <p>.7 Character width adjustment setting 1=adjust 0=no adjustment.</p> <p>.6 Background color display setting 1=display 0=not display.</p> <p>.5-.4 Write 0.</p> <p>.3-.0: Font size, 0x00-0x09, the corresponding font size is as follows: 0x00=6*12 0x01=8*16 0x02=10*20 0x03=12*24 0x04=14*28 0x05=16*32 0x06 =20*40 0x07=24*48 0x08=28*56 0x09=32*64</p> <p>Color: character display color.</p> <p>Bcolor: The color of the character background display.</p> <p>(X, y): The coordinates of the upper left corner of the string display.</p> <p>Strings: Strings to be displayed, non-ASCII characters are displayed in GB2312 encoding format Chinese characters.</p> <p>Example: AA 11 41 FF FF 00 00 00 20 00 80 44 57 49 4E 20 B5 CF CE C4 CC 33 C3 3C</p>
0x14	<p>Data variable display; processing time is the same as 0x11 instruction calculation.</p> <p>Mode: Display mode.</p> <p>.7 Background color display setting 1=display 0=not display.</p> <p>.6 1=signed number 0=unsigned number.</p> <p>.5 1=invalid 0 display 0=invalid 0 not display.</p> <p>.4 1=Invalid 0 is displayed as 0 0=Invalid 0 is displayed as a space.</p> <p>.3-.0: Font size, 0x00-0x09, same as 0x11 command.</p> <p>Color: Character display color.</p> <p>Bcolor: The color of the character background display.</p>

Num_I: The number of integer digits displayed, 0x01-0x14.
 Num_F: The number of decimal places displayed, 0x00-0x14, the sum of Num_I+Num_F cannot exceed 20.
 (X, y): The coordinates of the upper left corner of the variable display.
 Datas: Data variables, up to 8 bytes.
 Example: AA 14 85 FF FF 00 00 0A 02 00 00 00 00 49 96 02 D2 CC 33 C3 3C

(4) Instructions related to pictures and icons

instruction	Description
0x21	<p>QR code display; QR_Pixel=4 QR code processing time is 7.5mS. (X, y) is the coordinate position of the two-dimensional code display; QR_Pixel: the size of pixels occupied by each point of the two-dimensional code , 0x01-0x0F; DATA: display data, up to 154 bytes. The size of the QR code is (46*QR_Pixel)*(46*QR_Pixel) dot matrix. Example: AA 21 00 08 00 08 04 68 74 74 70 3A 2F 2F 77 77 77 2E 64 77 69 6E 2E 63 6F 6D 2E 63 6E CC 33 C3 3C</p>
0x22	<p>JPEG picture display; 480*272 resolution 4:1:1 format compression processing time is 250mS. Display JPEG pictures saved in 512Kbytes picture memory. The picture is also cached in the 0# virtual display area (0x27 command operation can be used). JPEG_ID: 0x00-0x0F, corresponding to the starting ID of the picture stored in JPEG. Example: AA 22 00 00 CC 33 C3 3C</p>
0x23	<p>Icon library icon display; 1 28*45 icon, background display mode, processing time 3.2mS . (X, y): the starting position of the first icon, corresponding to the upper left corner of the icon. Mode: Icon display mode. .7 Icon background display settings: 0=Background filtering is not displayed, 1=Background display. When setting the background filter to not display, the background must be pure black. .6 Background picture restoration setting (only valid when .7=0): 0=Background picture is not restored, 1=Automatically use 0# virtual display area picture for background restoration. .5 Background filtering strength selection (only valid when .7=0) 0=normal, 1=enhanced. 4 undefined, write 0. .3-.0 Icon library storage location, 0x00-0x0F. Icon_IDs: Icon IDs that need to be displayed, each ID is represented by 1 Byte, 0x00-0xFF. Example: AA 23 00 10 00 10 08 00 01 02 03 CC 33 C3 3C</p>
0x24	<p>SRAM memory icon display; 1 28*45 icon, background display mode, processing time 3.1mS . (X, y): The display position of the icon, corresponding to the upper left corner of the icon. Mode: Icon display mode.</p>

	<p>.7 Icon background display settings: 0=Background filtering is not displayed, 1=Background display.</p> <p>When setting the background filter to not display, the background must be pure black.</p> <p>.6 Undefined, write 0.</p> <p>.5 Background filtering intensity selection (only valid when .7=0) 0=normal, 1=enhanced.</p> <p>4-.0 undefined, write 0.</p> <p>Address: The starting address of SRAM memory to store JPEG icon data, 0x0000-0x7FFF.</p> <p>Example: AA 24 00 10 00 10 00 00 00 CC 33 C3 3C</p>
0x25	<p>The JPEG picture is decompressed to 1# virtual display area.</p> <p>480*272 resolution 4:1:1 format compression processing time is 240mS .</p> <p>Decompress the JPEG pictures saved in the 512Kbytes picture memory to the 1# virtual display area, which is convenient for operations such as icon copy and paste.</p> <p>JPEG_ID: 0x00-0x0F, corresponding to the starting ID of the picture stored in JPEG.</p> <p>Example: AA 25 01 01 CC 33 C3 3C</p>
0x26	<p>1# Copy and paste the designated area of the virtual display area to the current display interface.</p> <p>256*256 pixel area processing time is 12.5mS (0.2uS per pixel).</p> <p>(Xs, Ys): 1# The coordinates of the upper left corner of the selected area of the virtual display area icon.</p> <p>(Xe, Ye): 1# The coordinates of the lower right corner of the area specified by the icon in the virtual display area.</p> <p>(X, y): When pasting to the current display area, the coordinate position of the upper left corner.</p> <p>Example: AA 26 00 40 00 40 01 00 01 00 00 20 00 20 CC 33 C3 3C</p>
0x27	<p>Copy and paste from the designated area of the virtual display area to the current display interface.</p> <p>256*256 pixel area processing time is 12.5mS (0.2uS per pixel).</p> <p>Mode: Display mode.</p> <p>.7 Background display setting 0=Background filter is not displayed, 1=Background display.</p> <p>When setting the background filter to not display, the background must be pure black.</p> <p>.6 Background picture restoration setting (only valid when .7=0 and .1=1):</p> <p>0=Background picture is not restored, 1=Automatically use 0# virtual display area picture for background restoration.</p> <p>.5 Background filtering strength selection (only valid when .7=0) 0=normal, 1=enhanced.</p> <p>4-.1 reserved, write 0.</p> <p>.0 Virtual display area selection 0=0#virtual display area, 1=1#virtual display area.</p> <p>(Xs, Ys): The coordinates of the upper left corner of the selected area of the icon in the virtual display area.</p> <p>(Xe, Ye): The coordinates of the lower right corner of the icon area in the virtual display area.</p> <p>(X, y): When pasting to the current display area, the coordinate position of the upper left corner.</p> <p>Example: AA 27 01 00 40 00 40 01 00 01 00 00 40 00 40 CC 33 C3 3C</p>

0x28	<p>The icon animation automatically displays the command settings.</p> <p>(X, y): The starting position of the animation icon, corresponding to the upper left corner of the icon.</p> <p>Mode: Animation icon display mode.</p> <p>.7 Switch control 1=This group of animation is on 0=This group of animation is off; it can be controlled by 0x29 command.</p> <p>.6 Start mode 1=start from the start icon 0=start from the last stop position.</p> <p>.5-.4 Undefined, write 0.</p> <p>.3-.0 The command position of this group of animation icons, 0x00-0x0F, there are a total of 16 groups of animation commands.</p> <p>Icon_lib: icon library storage location, 0x00-0x0F.</p> <p>Icon_IDs: the starting icon position of the animation, 0x00-0xFF.</p> <p>Icon_IDe: the position of the animation termination icon, 0x00-0xFF.</p> <p>Delay_time: The display time interval of the animation icon, 0x00-0xFF, the unit is 10mS.</p> <p>Example: AA 28 00 10 00 10 80 09 00 09 0A CC 33 C3 3C Set the 0th group of animation</p>
0x29	<p>Icon animation automatically displays command control.</p> <p>Cartoon_Set: ICON animation command switch control; each bit corresponds to a group of commands, 1=on, 0=off;</p> <p>.15 corresponds to the 15th group of animation commands, and .0 corresponds to the 0th group of animation commands.</p> <p>Example: AA 29 00 05 CC 33 C3 3C Turn on group 0 and group 2 animation commands.</p>