T5L_DGUSII Application Development Guide

Version 1.0

DWIN Technologies

2020/12/30

Contents

1 Introduction

This section contains important information on how to read this document.

1.1 Overview

This document provides a general overview of T5L firmware for DWIN smart LCMs, its features and utilities, instructions on how to use it, and descriptions of all functionalities. It assumes the user has basic prior knowledge about microcontroller or computer programming and binary and hexadecimal numeric representations.

If you are a new user of DWIN smart LCMs, we recommend reading this document sequentially, from start to end. However, this document's sections are structured to be used as a reference guide, where you can look for specific information on each subject quickly.

1.2 Naming Rule

Knowing this rule will make it easier to learn about DWIN smart LCMs.

Product line code	DM	DWIN smart LCM product line
Color code	1 character	B=single color D=256 color T=65K color(16bit) G=16.7M color(24bit)
Resolution	5 digits	32240=320*240 48270=480*272 64480=640*480 80480=800*480 85480=854*480 80600=800*600 10600=1024*600 10768=1024*768 12720=1280*720 12800=1280*800 13768=1364*768 or 1366*768 19108=1920*1080
Classification	1 character	L=simple application grade C=commerce grade T=industrial grade K=medical grade Q=automotive grade S=military grade
Size	3 digits	022=2.2inch 024=2.4inch 028=2.8inch 035=3.5inch 040=4.0inch 043=4.3inch 047=4.7inch 050=5.0inch 056=5.6inch 057=5.7inch 064=6.4inch 070=7.0inch 080=8.0inch 084=8.4inch 090=9.0inch 097=9.7inch 101=10.1inch 104=10.4inch 121=12.1inch 150=15.0inch 156=15.6inch 185=18.5inch
_	_	Separator
Attribute code	1 digit or character	0=basic type 1=basic type with a shell 2=Simulate video processing platform products 3=Android platform products 4=Digital video processing platform products A=DGUSII kernel product

		B=DGUSIII kernel product
ROM ID	1 digit	0-9 to distinguish between different hardware versions
LCD temperature grade	1 character	N=normal temperature W=wide temperature
TP category	1 or 2 character	N=without touch panel TR=resistive touch panel TC=capacitive touch panel
Custom tag	Z+ number	Z01-Z99，Standard is blank
Extended memory mark	F+1 digit	F0=512MB F1=1GB F2=2GB F3=3GB，Standard is blank

For example, DMG48270C043_03WTR is a 4.3 inch, 480*272 resolution, commercial grade T5L smart LCM with resistive touch panel.

1.3 T5L_DGUSII Development System

DGUS development system is composed of DGUS screen and DGUS development software. DGUS is the abbreviation of DWIN Graphic Utilized Software. DGUS screen is based on configuration file to work, so the whole development process is the process that users complete variable configuration file with the help of PC DGUS development software.

(1) Variable programming

Users can make a table before the project, frame and plan the required variable address to facilitate the modification and maintenance of subsequent projects.

(2) Interface design

The simple interface can be made by yourself; the complex or beautiful interface can be made by professional artists. Icon image making is the same as image.

(3) Interface configuration

The interface is configured by T5L_DGUS software on the PC side. After completing the project, click "save" and "generate" in the "file" option in the upper Ⅱ left corner of the software to generate 13. Bin touch configuration file, 14. Bin display configuration file and 22. Bin variable initialization file.

(4) Debugging

Put the required files into the DWIN_SET folder and download them to the screen through SD card. The sequence is: power off – insert SD card - power on - blue screen reading SD card content, download complete display "SD card process END! " ——Power off, exit SD card - power on.

(5) Version setting

After finalizing the version, put the configuration files, picture files, icons, fonts, etc. into the DWIN_SET folder, and then mass production can be downloaded through SD card.

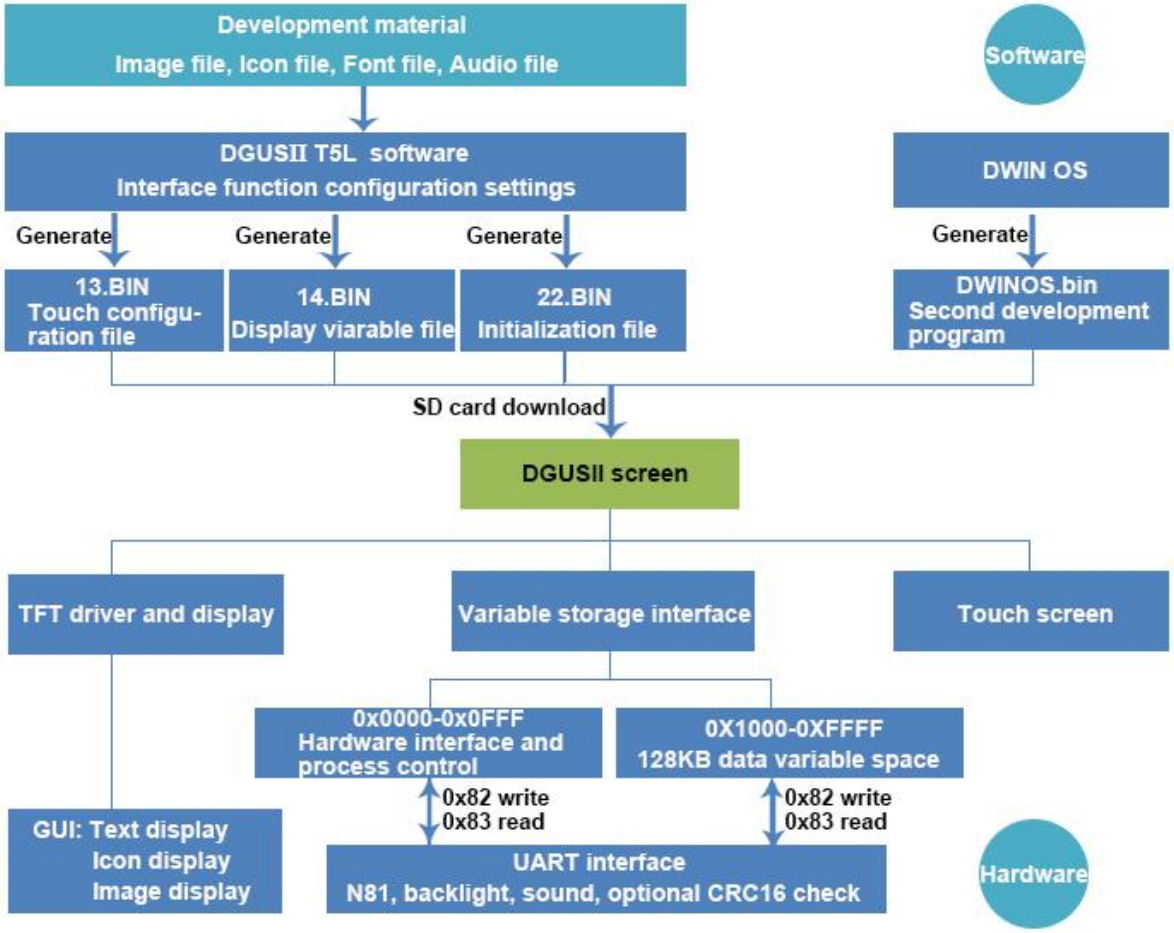


Figure 1.1 T5L_DGUSII development system diagram

1.4 Information and Caution Statements

This document may contain information and caution statements.

	Info	This is an information statement. It draws attention to certain key aspects about the current topic.
	Caution	This is a caution statement. It describes a situation that could potentially damage your software, equipment or cause data loss.

The information in caution and warning statements is provided for your protection. Read each caution and warning statement carefully.

2 Hardware

This section describes the hardware (physical) components of a DWIN Liquid Crystal Module (DWIN LCM), and the hardware connection tools used in production development.

2.1 Processing Core

As shown in the figure 2.1 The T5L series ASIC is a single-chip and dual-core ASIC IC designed by DWIN technology co., Ltd for AIoT applications with low power consumption, high cost performance, GUI and highly integrated application, including T5L1(low resolution) and T5L2(high resolution). Its main features are as follow:

(1) Using 8051 core which is the most widely used, mature and stable, the maximum operating frequency of T5L is up to 250MHz, 1T (single instruction cycle) high speed operation.

(2) Separated GUI CPU Core running DGUSII System:

- High-speed display memory, 2.4 GB/S bandwidth, 24bit color display resolution supporting to 800*600(T5L1) or 1366*768(T5L2).
- 2D hardware acceleration, the decompression speed of JPEG is up to 200fps@1280*800, the UI interface with animation and icons as its main feature is extremely cool and smooth.
- JPEG stores images and icons in compressed mode, external memory greatly reduced to low-cost 16Mbytes SPI Flash.
- With maximum 400 Hz touch point speed, touch panel supports resistance or capacitance, and its sensitivity can be adjusted.
- High-quality speech compression storage and playback.
- 128KBytes variable storage space, exchanging data with OS CPU Core using memory interface, extremely simple to apply.
- 2-way 10bit, 800KHz, DC/DC controller, simplify LED backlight, analog power supply design and save cost and space.
- 1-way 15bit 32Ksps PWM digital power amplifier driver loudspeaker, save power amplifier cost and achieve high signal-to-noise ratio and sound quality restoration.
- Supporting PC configuration development and simulation, supporting background remote upgrade.

(3) Separated OS CPU core runs user 8051 code or DWIN OS system, user CPU is omitted in practical

application.

- Standard 8051 architecture and instruction set, 64Kbytes code space, 32Kbytes on-chip RAM.
- 64 bit integer mathematical operation unit (MDU), including 64 bit MAC and 64 bit divider.
- Built-in software WDT, three 16 bit Timers, 12 interrupt signals with the highest four interrupt nesting.
- 22 IO, 4-channel UARTS,1-channel CAN interface, up to 8-channel 12-bit A/D, 1-channle 16-bit resolution adjustable PWM.
- Support IAP on-line simulation and debugging, unlimited number of breakpoints.
- Upgrade code online through DGUS system.

(4) 1Mbytes on-chip Flash with DWIN patent encryption technology ensure code and data security, eliminate copycat and cloning.

(5) Reduces crystal requirements and PCB design challenges for a variety of inexpensive wide-range tuned impedance crystal oscillators and PLL.

(6) 3.3V IO voltage, can adapt to 1.8/2.5/3.3 various levels.

(7) Supporting SD interface download and configuration, supporting SD card file reading and rewriting.

(8) Supporting DWIN WiFi module to access to DWIN cloud directly, and easily developing various cloud platform applications.

(9) Working temperature ranges from - 40℃ to +85℃ (Customizable IC for -55℃ to 105℃ operating temperature range)

(10) With low power consumption and strong anti-interference ability, it can work steadily on the double-sided PCB design, and easy to pass EMC/EMI test.

(11) Using 0.4 mm ELQFP128 packaging, low manufacturing difficulty and low cost.

(12) For industry customers to provide T5L IC+LCD+touch panel cost-effective supporting scheme and comprehensive technical service support.

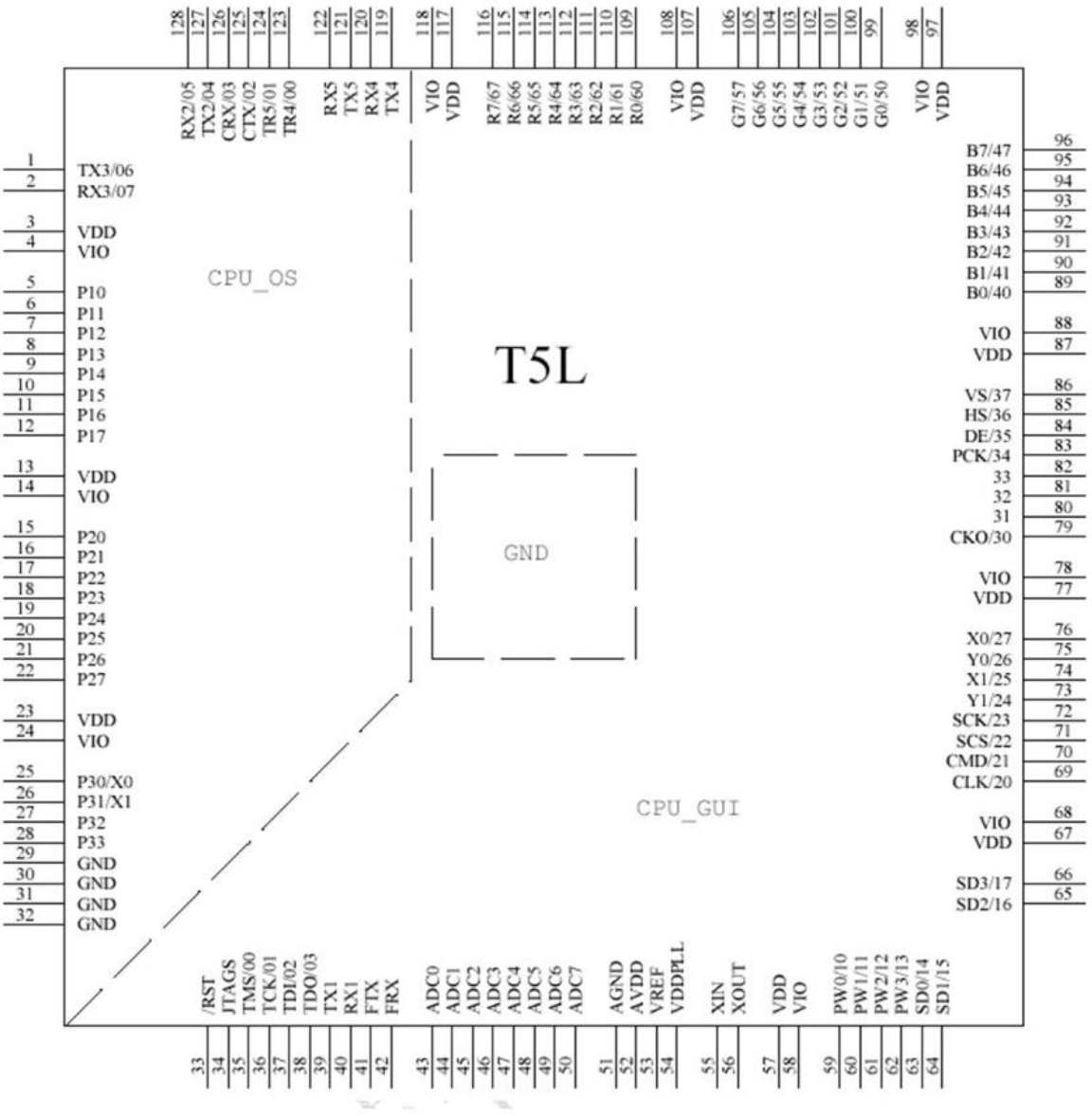


Figure 2.1 The arrangement of T5L pin

2.2 Power

Confirm the screen voltage and power consumption according to the selection manual or data manual, lit up the screen by a 5v/12v/24v DC regulated supply. The power supply plays a very important role in the normal display of the screen. Too-low voltage, instable current or too-low power may lead to flashing screen, black screen and other abnormal display phenomenon.

2.3 Interface Definition and Wiring

PIN name	PIN type	PIN instruction	Socket type		
			CN1 PIN NUM.	CN2 PIN NUM.	CN3 PIN NUM.
VCC	P	Power supply input	1,2	1,2,3	1
TX4	O	Serial 4 output	3	4	2
TX2	O	Serial 2 output	4	5	3
RX2	I	Serial 2 input	5	6	4
RX4	I	Serial 4 input	6	7	5
GND	P	Ground	7,8	8,9,10	6

Info

①I: INPUT，O: OUTPUT，P: POWER

②PCB pins of the same defined pins have been connected in parallel.

③CN1: Use 8 Pin 2.0 mm spacing patch socket;

CN2: Use 10 Pin 1.0 mm spacing FPC mount;

CN3: Additional 6 Pin 2.54 mm spacing through hole pad is provided.

④Serial port 2 is the communication protocol developed by DGUSII.

T5L DGUS products are equipped with extended serial port, and it is UART4, and its configuration and communication need to be completed through DWIN OS program. In the serial instruction set mode, only serial 2 communication is available. Other non-serial 2 communication has no open function for the time being.

⑤In order to improve baud rate and facilitate connection to PC debugging, Some models of DWIN adopt TTL/RS232 compatible interface. Users can use 0Ωresistor or solder to directly short circuit. R232=0 (short circuit) selects TTL level input; R232=1(disconnect), selects RS232 level. As shown in figure 1.0, the place is marked by white silk screen. By default, the factory disconnects the 232 level, and selects the TTL level. That is OFF=232, ON=TTL.

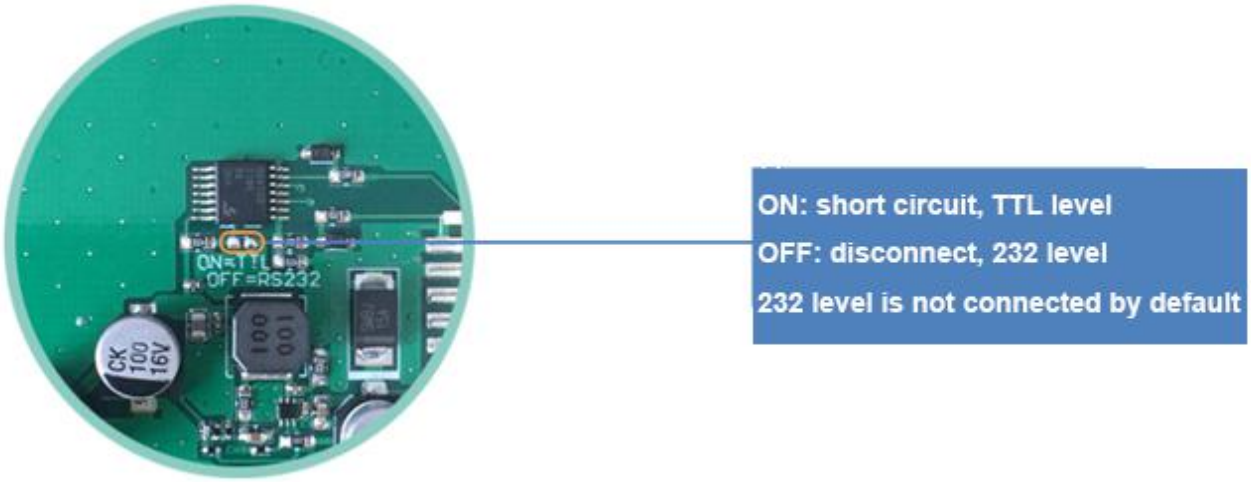


Figure 2.2 Level selection

(1) 10 pin interface

As shown in the figure 2.3. One-end of the different side FCC row line is connected to terminal block of the screen, and the other end is connected to the DWIN HDL662B adapter board, both blue side face up. One-end of the double-male USB cable is connected to the adapter board, and the other end is connected to the computer for communication.

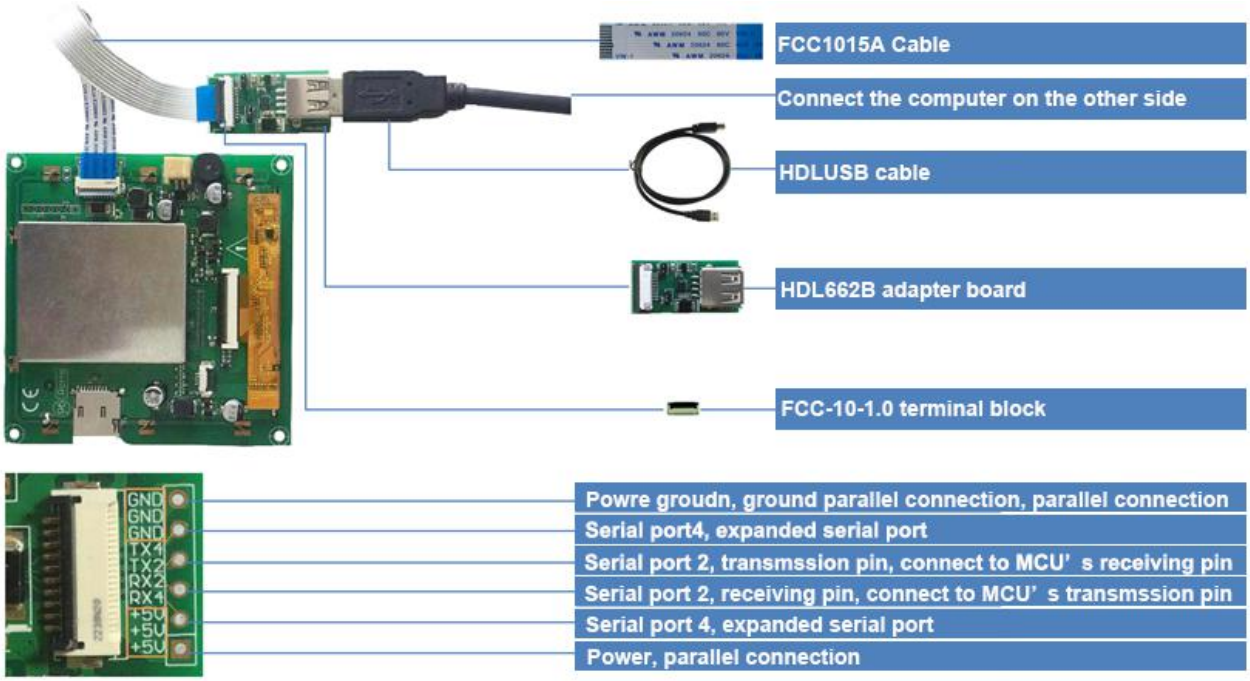


Figure 2.3 10 PIN interface definition and wiring description

(2) 8 pin interface

8 PIN interface of 2.0 mm space is as follow.

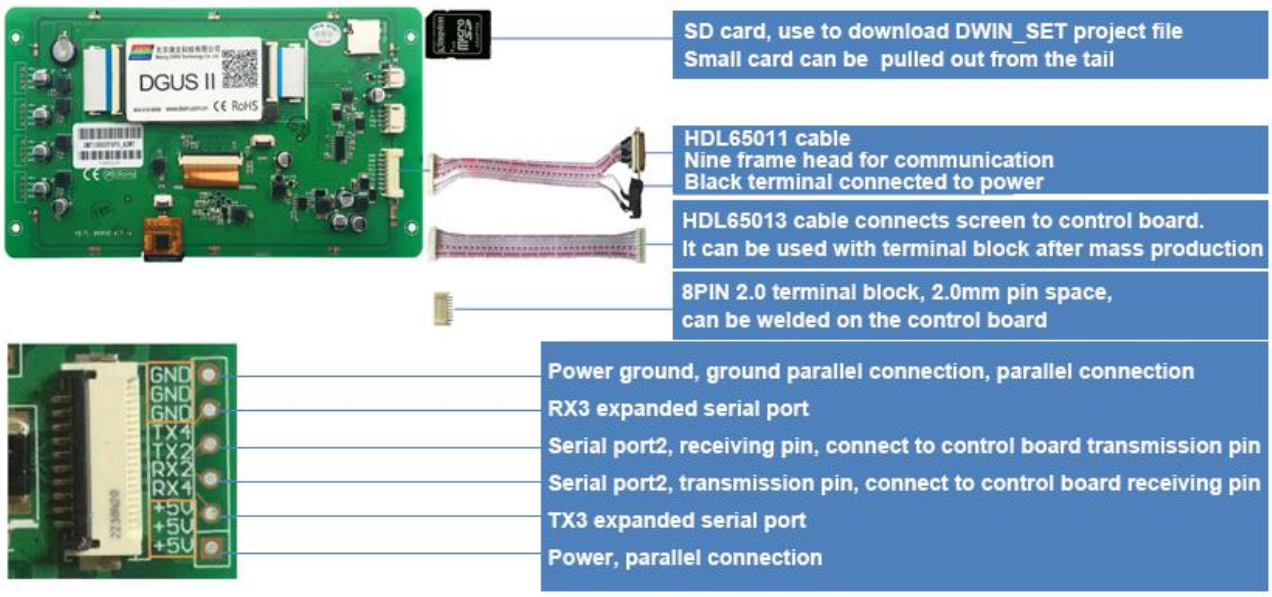


Figure 2.4 8 pin 2.0 interface definition and wiring description

2.4 SD Interface

All hardware parameter settings and data of T5L display terminal can be completed through SD/TF card interface on the screen, and the file must be in FAT32 format.

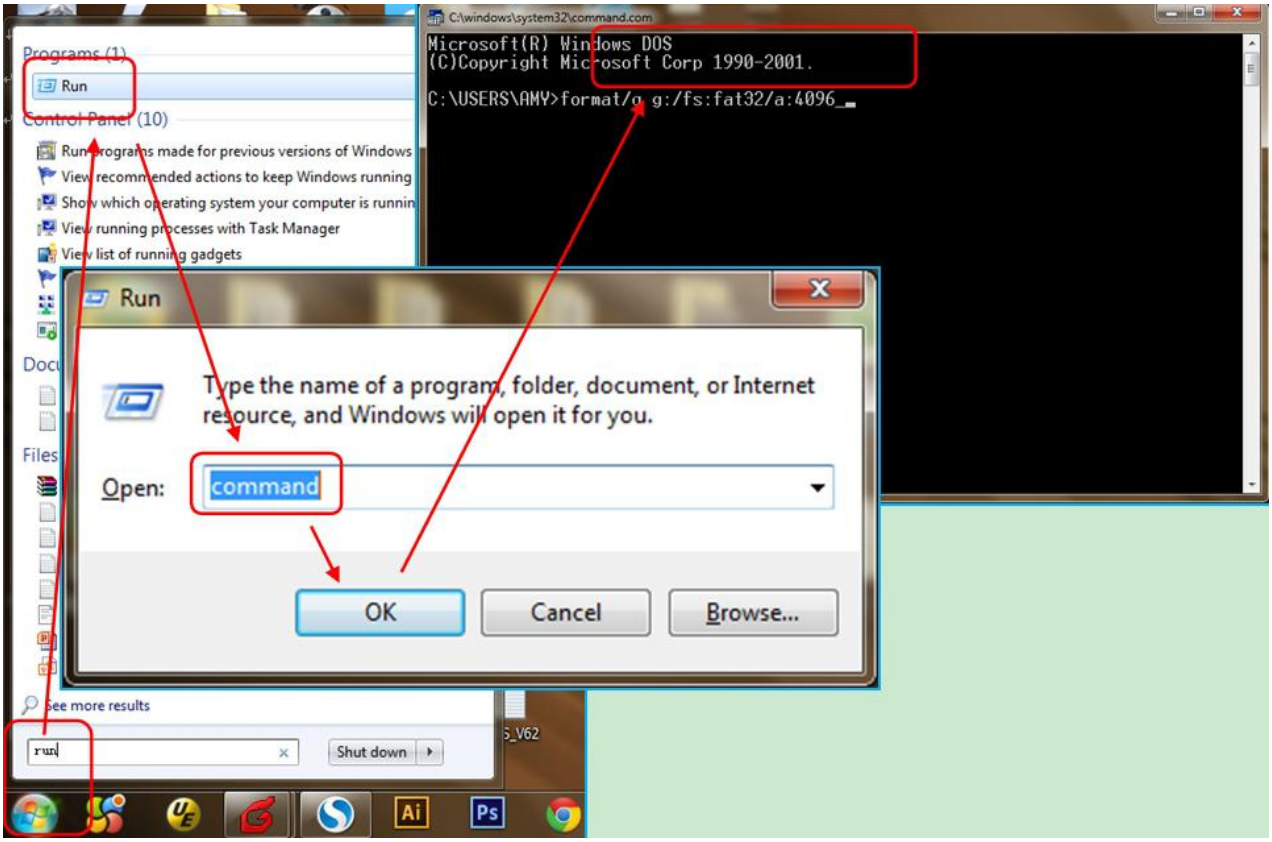
SD cards sold by non DWIN factories often have to be formatted under DOS system. Otherwise, the download phenomenon is usually just that the number of downloaded files is 0 after the blue screen, or the display terminal fails to recognize that the card cannot enter the download interface normally. Format as follows:

Step 1: start 》 run 》 enter command (enter CMD for win7 system) to enter DOS system;

Step 2: input the command: format/q g:/fs:fat32/a:4096 (Note: q is followed by a space). Click enter after input.

Where g is the disk number of SD card displayed on the user's computer, and the corresponding disk number of different users is not fixed (for example, h, i can be replaced)

Info Note: the format operation after right clicking SD card cannot be completely formatted into FAT32 format. Generally, only SD card with size range of 1-16g is supported.



2.5 Buzzer

The built-in buzzer provides audible feedback for the user when they touch a button on the Screen.

The automatic sound feedback can be turned off.

The buzzer can also be activated by Serial Communication.

2.6 Audio Output

Some LCM models have an audio output connector for speakers, to play audio files stored in memory

2.7 Serial Port Tool

All DWIN LCMs have one or more connectors for Serial Communication with external controllers.

There are XR21V1410 and CP2102 chips in the USB to UART chip of DWIN serial port adapter board. According to the chip type, you can download the corresponding driver on DWIN official website or consult 400 technical

support to obtain and install it for DGUS screen communication.

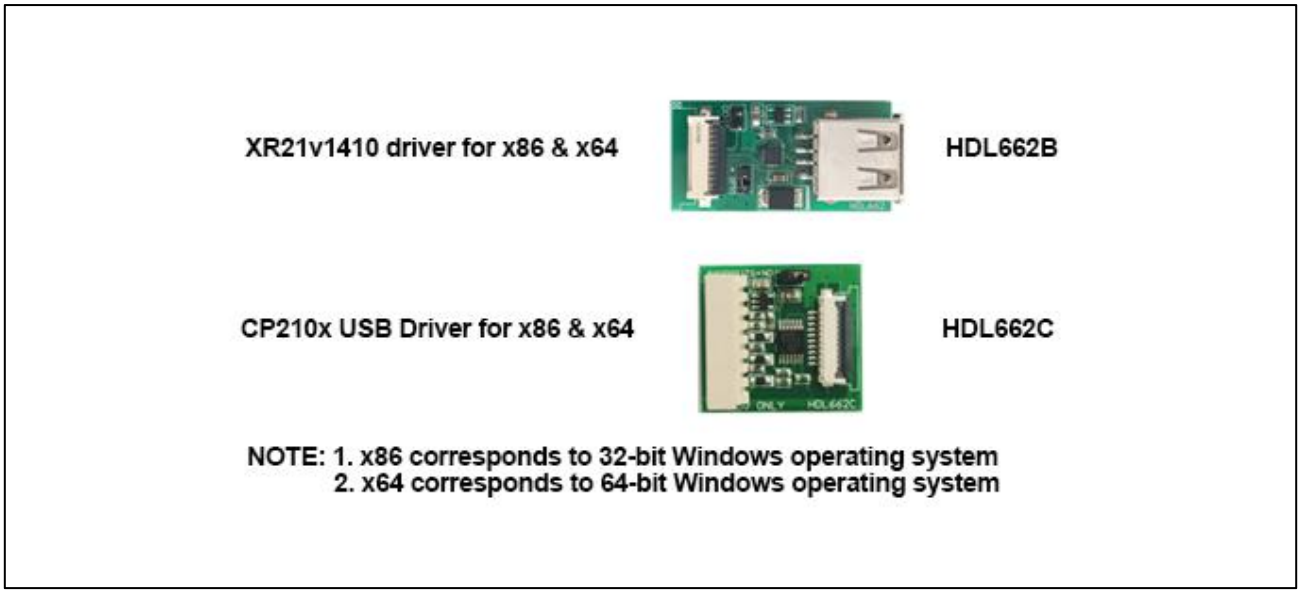


Figure 2.5 Serial port driver

3 Firmware Structure

3.1 Software Processing Flow

In the T5L_DGUS II development platform, the configuration of touch operation of touch panel is also stored in 13. Bin file in the same way. The characteristic attributes of variables (characters, data), various animations and other functions to be displayed are represented by codes and stored in 14. Bin file according to the address. The operation in the development software of DGUS II is to configure the control of display function or touch function. When the system is powered to run, the system will call 13. Bin file and 14. Bin file, so that the human-computer interface can run normally.

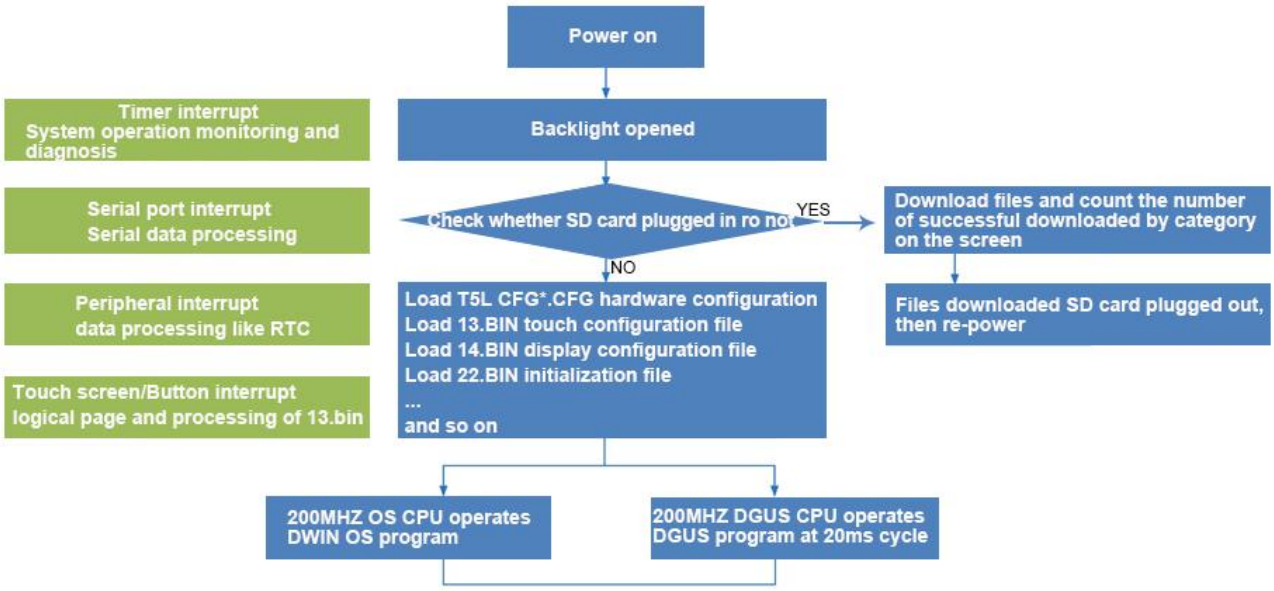


Figure 3.1 DGUS Processing Flow

3.2 Memory Spaces

3.2.1 FLASH Space

The 16MB Flash memory is divided into 64 fixed 256KB subspaces, and the file ID ranges from 0 to 63. According to the different contents of the stored files, the memory is mainly divided into two parts:

(1) 4-12MB font space can save BIN, HZK, DZK format files, the file ID ranges from 00 to 47.

(2) 4-12MB picture space can save ICL files of background picture library and ICL files of icon library. The file ID ranges from 16 to 63.

Info	Note that there are overlaps between the font space and the image space, therefore you should avoid conflicts when naming ID.

For the T5L1 CPU platform, the size of a single picture file in the packed ICL file should not exceed 252KB, and that in the T5L2 CPU platform should not exceed 764KB.The download file must be placed in the DWIN_SET folder of the SD card root directory, which must be a 4KB sector, FAT32 format SD or SDHC card.

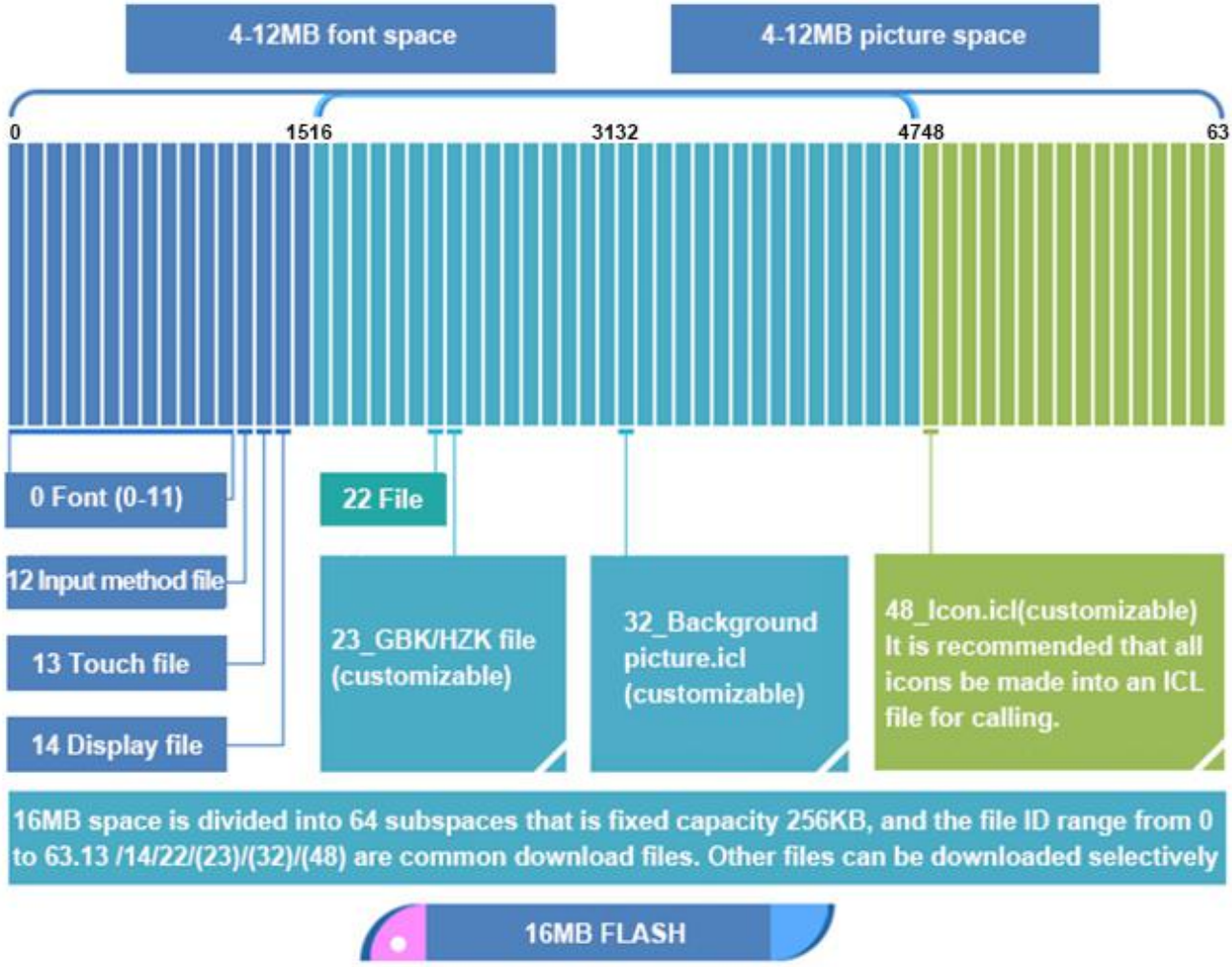


Figure 3.2 16MB FLASH space

3.2.2 RAM Space

The RAM space is fixed at 128KB, which is divided into the range of 0x0000-0xFFFF subspace, and the size of

each subspace is 1 word.

It is divided into 8KB system variable interface address space and 120KB user-defined address space.

Variable storage space	Size	Space scope	Description
System variable interface	8KB	0x0000-0x0FFF	Used for system variable interface definition
User variable interface	32KB	0x1000-0x4FFF	User curve buffer space
	88KB	0x5000-0xFFFF	User defined variable address(*VP) and description pointer (* SP) spaces

If the user does not use the curve function, the 0x1000-0x4FFFF space range is used as the user-defined variable address (* VP) and description pointer (* SP) spaces.

Info 1Word=2Byte=16bit

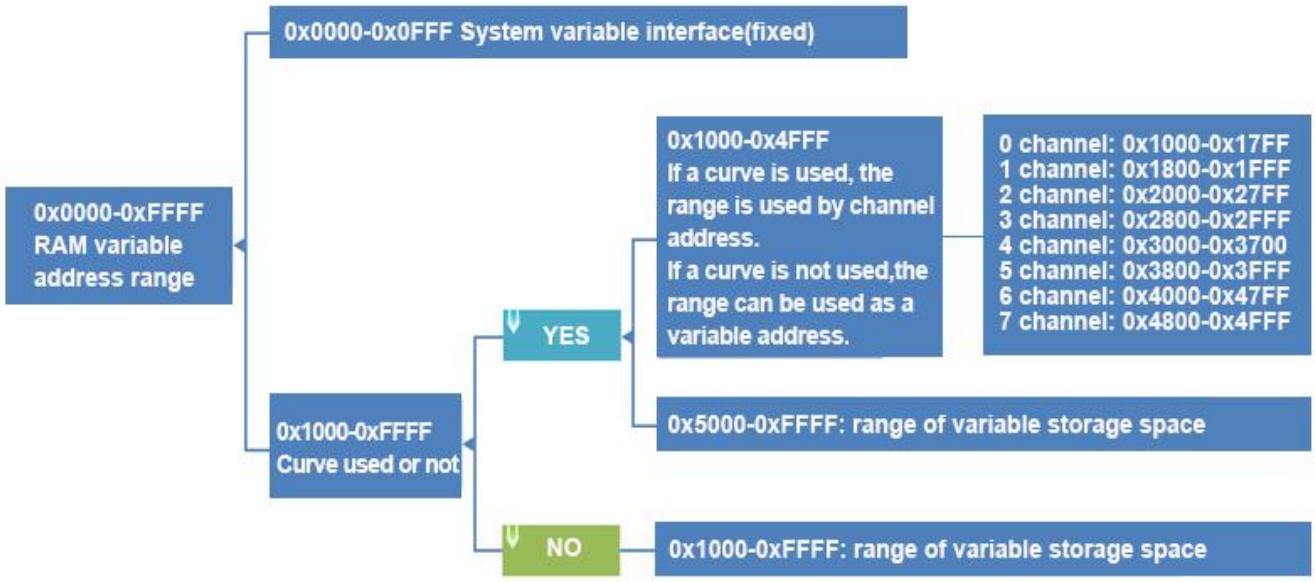


Figure3.3 Partition of RAM storage space

3.3 File Structure

The following table describes the file types and naming rules accepted by DGUSII platform.

Caution	The serial screen SD/SDHC interface based on T5L supports downloading and updating of the following files. SD card upgrade does not support online hot plug-in update. You must first power off the screen, insert SD card, and then power on to download.

Table3.1 download file type

File type	Naming rule	Description
Program file	T5L_UI*.BIN T5L_OS*.BIN	Underlying kernel firmware application
DWIN OS program	DWINOS*.BIN	DWIN OS program, the code must start from 0x1000.
OS CPU 8051 program	T5L51*.BIN	The user develops the application program based on the standard 8051 platform.
NOR Flash data base	ID+（optional）filename.LIB	Each ID corresponds to 2KWords memory with ID range from 0 to 79. The database is located in on-chip NOR Flash, with a size of 160KWords. It can be used to save user data or DWIN OS program library files.
Font file（4~12MB）	Font ID+（optional）filename. BIN/DZK/HZK	Font ID: 00-47; ASCII font uses DGUS 0# font.
DGUS input method file	12*.BIN	Fixed storage in 12 font location
DGUS touch control file	13*.BIN	Fixed storage in 13font location, cannot over 256KB
DGUS variable file	14*.BIN	Fixed storage in 14 font location, cannot over 256KB, must be in DGUS II format
DGUS variables initialization file	22*.BIN	Fixed storage in 22 font location. Load the variable space of 0x2000-0x1FFFF address content initialization 0x1000-0xFFFF.
JPEG, ICON file	Font ID+ (optional) filename. ICL	Must be a JPEG ICL file format in DGUSIII format. When multi Flash expansion, ensure that one picture data is saved in one Flash.
Wave file	Font ID+ (optional) filename. WAE	Must be in DGUSII format, generated using DWIN special tools.
UI assembly file	Configuration module number ID+(optional) filename. UIC	The assembly module number is from 0x0000 to 0xFFFF, and each ID corresponds to 32KB memory space. In case of multi Flash expansion, ensure that the data of one assembly module is saved in one flash.
Hardware configuration file	T5LCFG*.CFG	Configure CRC on, baud rate modification, touch panel sound control, touch upload mode, display direction, etc.

Info

Only need to update or download the files according to the needs, without downloading all the files. For example, if a user uses nor flash to save user data (such as 00.lib, 01.lib), then download 00.lib and 01.lib.

4 System Configuration

4.1 Configuration List

T5L CFG *. CFG hardware configuration file is in binary data format and can be edited by software such as UltraEdit. The description is shown in the following table:

Table 4.1 CFG configuration file

Category	Address	Length	Definition	Description
Recognition configuration	0x00	5	0x54 0x35 0x4C 0x43 0x31	Fixed content
System configuration	0x05	1	Parameter configuration	.7: Serial port CRC check 0=close 1=open
				.6: Buzzer / music playback selection， 0=Buzzer 1=music playback
				.5: 22 File initialization variable space 1= load 0= no load
				.4: variable automatic upload setting 1= on, 0= off
				.3: touch panel audio control 1= open 0= close
				.2: touch panel backlight standby control 1= open 0= close
				.1-.0: display direction 00 = 0 °, 01 = 90 °, 10 =180 ° ,11 = 270 °
	0x06	1	Parameter configuration 1	.7: PWM0 output 0=user control, 1= control for PMW brightness of backlight, 1KHz frequency. .6: Expanded SPI NAND Flash , 0=not expanded 1=expanded. .5: Set 1 to start SPI NAND Flash format once (data will be cleared). .4: SPI NAND Flash Memory：0=1Gbits 1=4Gbits. .3-.0: Reserved. Write 0.
	0x07	1	Music WAE file save position	0x00-0x3F(00-63)
	0x08	1	Background picture ICL file save position	0x10-0x3F(16-63)，correspond to 12MB-4MB background picture space
	0x09	1	Touch panel reporting point rate setting	Set range 0x01-0xFF， by default 0x28， reporting point rate =400Hz/set value
	0x0A	2	Serial port baud rate setting	Baud rate setting value=3225600/baud rate 115200bps，set value=0x001C, maximum value 0x03FF
Backlight standby configuration	0x0C	1	Normal operation and startup brightness	0x00-0x64，unit 1%
	0x0D	1	Standby brightness	0x00-0x64，unit 1%
	0x0E	2	Wake up time after standby	0x0001-0xFFFF，unit 10 ms

LCD configuration	0x10	2	Display_Config_En	0x5AA5= Enable the configuration. It has been configured at the factory. Users do not need to configure it again.
	0x12	1	PCLK_PHS	Data latch phase setting：0x00=PCLK falling edge 0x01=PCLK rising edge
	0x13	1	PCLK_DIV	PCLK frequency setting: PCLK frequency (MHz) =206.4/PCLK_DIV
	0x14	1	H_W	
	0x15	1	H_S	
	0x16	2	H_D	Horizontal (X direction) resolution
	0x18	1	H_E	
	0x19	1	V_W	
	0x1A	1	V_S	
	0x1B	2	V_D	Vertical (Y) resolution
	0x1D	1	V_E	
	0x1E	1	TCON_SEL	0x00= do not need to configure TCON
TP configuration	0x1F		Reserved	Write 0x00
	0x20		TP_Set_En	0x5A= Enable the configuration. It has been configured at the factory. Users do not need to configure it again
	0x21	1	TP_Mode	Touch panel mode configuration .7-.4(high 4bit), select touch panel type: 0x0*= 4-wire resistive touch panel 0x1*= Driving IC capacitive touch panel such as GT911, GT9271 or GT9110 IC 0x2*=ILI9881H Incell CTP 0x3*=ILI driving IC capacitive touch panel such as ILI2117 0x4*=CHIP NE driving IC capacitive touch panel such as ICNT8952 0x5*=Betterlife driving IC capacitive touch panel such as BL8825 0xF*= 5-wire resistive touch panel .3 Resistive touch panel calibration: 0=off; 1=on, enable when SD card downloading is in process .2-.0 (low 3bit) Coordinate setting: .2 X coordinate selection: 0= from 0 to Xmax; 1=from Xmax to 0； .1 Y coordinate selection: 0= from 0 to Ymax; 1=from Ymax to 0； .0 X,Y exchange: 0=XY; 1=YX
	0x22	1	TP_Sense	TP sensitivity setting: 0x00-0x1F, 0x00 lowest sensitivity, 0x1F highest sensitivity. Default sensitivity is 0x14, which is higher sensitivity. (ILI9881 is 0x01-0x06).
	0x23	1	TP_Freq	TP frequency selection, suitable for ILI9881H. Fixed frequency ranges from 0x01 to 0x14, and 0x00 represents frequency hopping.
Clock output configuration	0x24	1	CKO_Set_En	0x5A= Enable the configuration.
	0x25	1	CKO_En	0x5A = Enable the output function of CKO (P3.0), others

				represents closing the output.
BUZZ configuration	0x26	1	CKO_DIV	CKO output clock setting. Frequency = 825.7536/CKO_DIV MHz.
	0x27	1	BUZZ_Set_En	0x5A= Enable the configuration.
	0x28	1	BUZZ_Freq_DIV1	Buzzer frequency = 825753.6 / (BUZZ_Freq_DIV1 * BUZZ_Freq_DIV1) KHz.
	0x29	2	BUZZ_Freq_DIV2	Factory settings: DIV1=0x6E，DIV2=0x0BB8，correspond to the 2.5KHz frequency.
	0x2B	2	BUZZ_Freq_Duty	Buzzer duty ratio settings: High level duty ratio=BUZZ_Freq_Duty / BUZZ_Freq_DIV2. Factory settings: 0x00F0 corresponds to 8% high level duty ratio.
	0x2D	1	BUZZ_Time	Buzzing time after touch action, unit 10 ms；Factory setting: 0x0A
Reserved	0x2E	18	Reserved	Write 0x00.
Setting for name of download folder	0x40	2	SD_Set_En	0x5AA5 means setting the name of SD download folder once, which are saved in the device Flash. It will not be lost after power off.
	0x42	1	Character length of download folder name	0x01-0x08
	0x43	8	Folder name	Up to 8 ASCII characters (only 0-9, a-z, A-Z, -, _). Invalid character setting will use "DWIN_SET" as the default value, which is effective after power off and restart.
	0x4B	37	Reserved	Write 0x00.

Info　　Note: parameters of green background part must be configured.

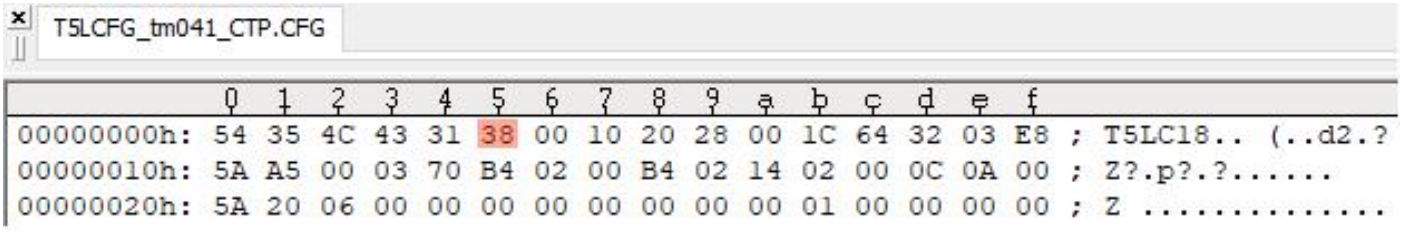
4.2 Hardware Parameter

For example, CFG file 0x05 parameter configuration:

①For example: set the initial value of power on display, automatically upload the touch data, turn on the touch panel sound, turn off the backlight, and configure the display direction at 0 °.

Bit	Parameter	Value	Description
.7	CRC check	0	0= close 1= open
.6	Buzzer	0	0=buzzer 1=music playback
.5	Initial value	1	0= close 1= open
.4	Data upload	1	0= close 1= open
.3	Sound	1	0= close 1= open
.2	Backlight	0	0= close 1= open
.1	Display direction	0	00=0°01=90°10=180°11=270°
.0		0	

Here, 1 byte corresponds to 8 bits, and each bit corresponds to two states: on and off. The corresponding bit is enabled to write 1 and off to write 0. Convert the binary 0011 1000 to hexadecimal 0x38, write the address of the CFG file 0x05 to 0x38, and download the CFG file to complete the corresponding configuration.



②Enable CRC check

The 7th bit of address 0x05 is configured as 1, which means CRC verification is enabled.

③Different display direction

Setting value of DMG10600C070-03W (Horizontal display screen series) in CFG file at different display direction

Mode	Picture resolution	ICL	Configuration value
0 ° lateral	1024 * 600 picture	1024*600 made.ICL file	0x38
90 ° vertical	600 * 1024 picture	600 * 1024 made.ICL file	0x39
180° lateral	1024 * 600 picture	1024*600 made.ICL file	0x3A
270 ° vertical	600 * 1024 picture	600 * 1024 made.ICL file	0x3B

Setting value of DMG85480C050-03W (Vertical display screen series) in CFG file at different display direction

Mode	Picture resolution	ICL	Configuration value
0 ° lateral	854 * 480 picture	854 * 480 made.ICL file	0x38
90 ° vertical	480 * 854 picture	480 * 854 made.ICL file	0x39
180° lateral	854 * 480 picture	854 * 480 made.ICL file	0x3A
270 ° vertical	480 * 854 picture	480 * 854 made.ICL file	0x3B

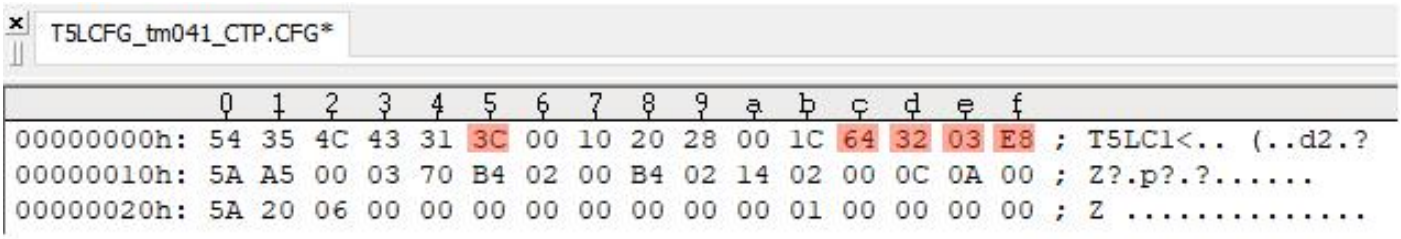
④Audio playback

The address 0x07 is used to set the music saving position. The name of the audio file should be the same as the setting value.

4.3 Backlight

CFG address	Description	Configuration value
0x05	Turn on backlight	0x3C
0x0C	Brightness value	Range: 0x00-0x64
0x0D	Standby brightness value	Range: 0x00-0x64
0x0E 0x0F	Screen on time setting without touching	Range: 0x0001-0xFFFF, unit 10 ms.

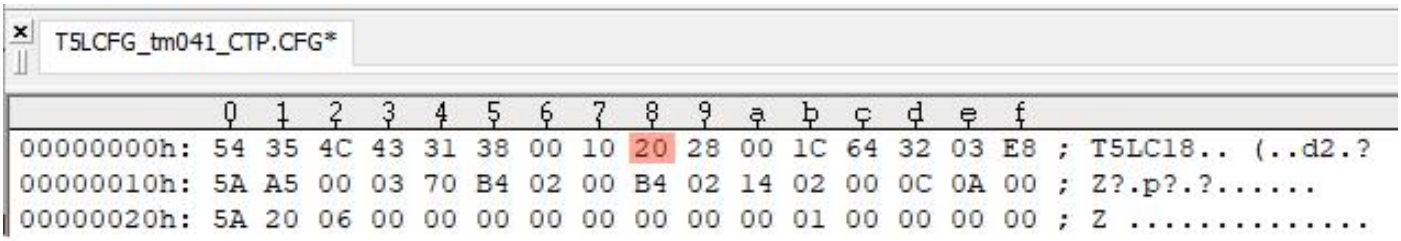
Info Note: After the backlight standby screen protection is turned on, the first click on the touch panel will wake up the backlight, and the second click will trigger the touch control.



4.4 ICL File Save Position

If the picture background file is set to 32_picture. ICL file, 0x20 needs to be written in the address of CFG file 0x08 for bottom reading and display.

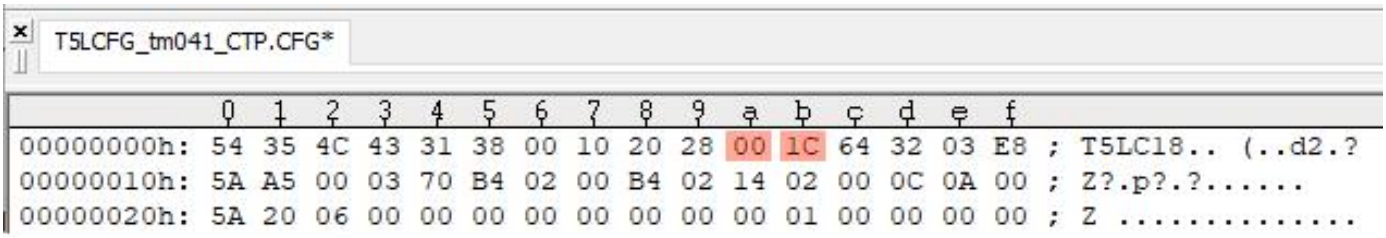
If it is a named other value, 0x08 needs to be set to the corresponding value, such as 33 _background picture, 0x08 needs to write 0x21.



4.5 Baud Rate

Formula: Baud rate setting value =3225600/ setted baud rate. The maximum value is 0x03FF.

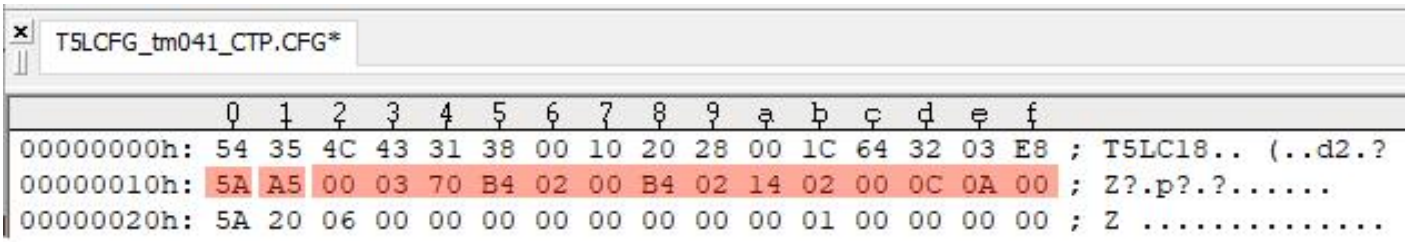
E.g.: set baud rate:115200, baud rate setting value=3225600/115200=28, the hexadecimal number which 28 is converted to is 0x1C. As the baud rate address is 2 bytes, CFG file 0x0A address can start writing 0x001C. The 9600 baud rate is 0x0150.



4.6 Display

Display configuration mode. Touch configuration mode is factory configured. Under normal circumstances, users do not need to configure. In case of scrolling, reverse touch or white screen during configuration, you can try to download the factory CFG file.

0x10, 0x11 write 0x5A A5 to indicate that 0x12-0x1F parameter will take effect, and 0x00 00 indicate that it will not take effect.



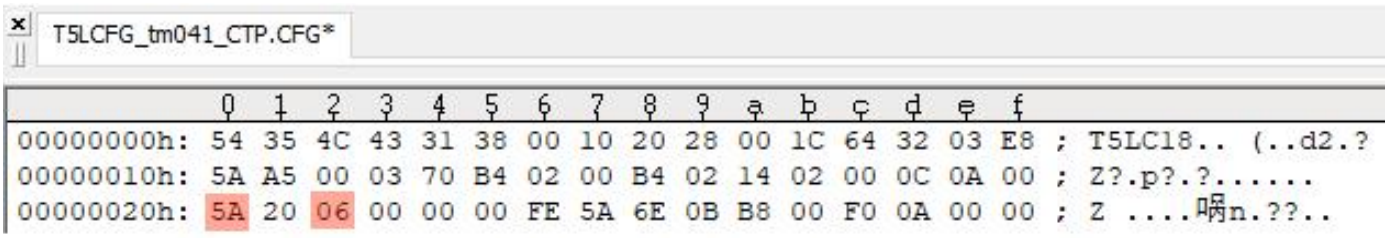
0x12-0x1F position of the corresponding CFG file

Table 3.4 reference list of display configuration

Size_ resolution	Display screen configuration value （HEX format）													
	0x12	0x13	0x14	0x15	0x16	0x17	0x18	0x19	0x1A	0x1B	0x1C	0x1D	0x1E	0x1F
2.4_240*320（IPS）	01	26	0A	14	00	F0	0A	02	02	01	40	02	05	00
2.8_240*320A	01	26	0A	14	00	F0	0A	02	02	01	40	02	03	00
2.8_240*320B	01	20	10	20	00	F0	20	02	0E	01	40	08	01	00
3.5_320*240	01	1C	1E	14	01	40	40	03	0F	00	F0	10	02	00
3.5_320*480	01	14	0A	04	01	40	0A	02	02	01	E0	02	04	00
3.5_320*480（IPS）	01	14	0A	04	01	40	0A	02	02	01	E0	02	06	00
3.5_480*640	01	0A	10	20	01	E0	20	02	03	02	80	02	07	00
4.0_480*480（IPS）	00	0E	08	08	01	E0	08	02	0C	01	E0	06	08	00
4.0_480*800（IPS）	00	08	08	08	01	E0	08	04	0A	03	20	0A	09	00
4.0_720*720（IPS Incell）	00	03	70	B4	02	D0	B4	02	14	02	D0	DC	0A	00
5.0_720*1280（IPS Incell）	00	03	04	14	02	D0	14	02	12	05	00	C8	0A	00
5.0_480*854（IPS）	00	08	08	08	01	E0	08	02	0C	03	56	06	0C	00
480*272	01	16	29	02	01	E0	02	0A	02	01	10	02	00	00
640*480	01	08	1E	72	02	58	10	03	20	01	E0	0A	00	00
800*480	01	06	1E	10	03	20	D2	03	14	01	E0	0C	00	00
800*600	01	05	1E	10	03	20	D2	03	14	02	58	0C	00	00
1024*600	01	04	A0	88	04	00	18	06	1D	02	58	03	00	00
1024*768	01	04	10	40	04	00	20	04	08	03	00	04	00	00
1280*720	01	03	10	40	05	00	20	08	20	02	D0	20	00	00
1280*800	01	03	10	1C	05	00	10	08	10	03	20	10	00	00
1366*768	01	03	10	20	05	54	20	06	10	03	00	08	00	00
1366*768 eDP	00	03	10	20	05	54	20	06	10	03	00	08	10	00
1024*768 VGA	00	03	88	A0	04	00	18	06	1D	03	00	03	00	00
1280*800 VGA	00	03	80	C8	05	00	48	06	16	03	20	03	00	00

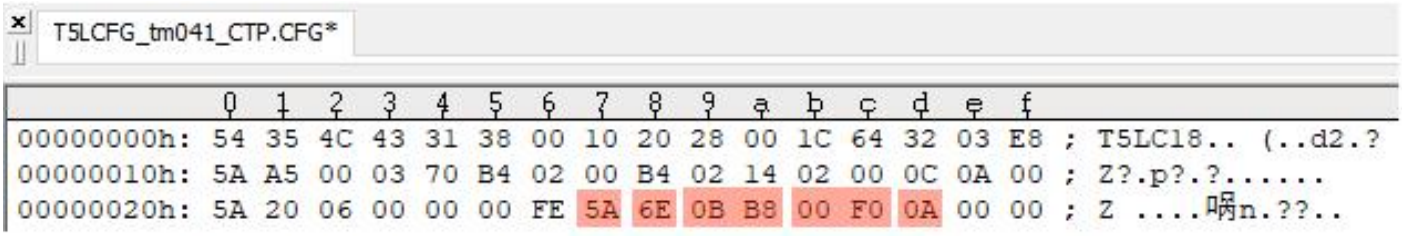
4.7 Sensitivity

Sensitivity address is 0x22, by default the value is set to 0x14, the range is from 0x0000 to 0x1F. Address 0x20 is set to 0x5A.



4.8 Buzzer

The buzzer can be configured through the CFG file. Adjusting the duty ratio of the position 0x2B of the CFG file can change the size of the buzzer's sound. The larger the duty ratio is, the louder the sound is. Adjust the 0x2D position to change the buzzer's beeping time.



4.9 Touch Calibration

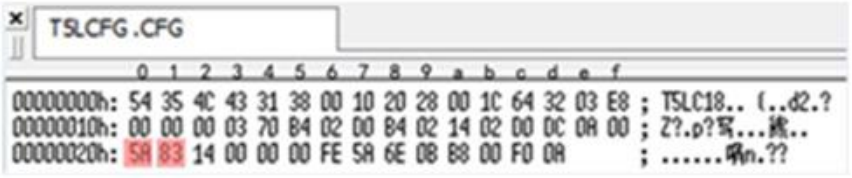
The product calibration methods of T5L TA and DGUS are the same. During the switch between TA instruction set development mode and DGUS development mode, touch drift may occur occasionally. At this time, it can also be calibrated as follows.

Step 1: download the CFG file

Here, take the DGUSII screen DMG48480C040_03WTR as an example, edit the address data of the CFG file 0x21 as 0x8B, and turn on the touch panel calibration.

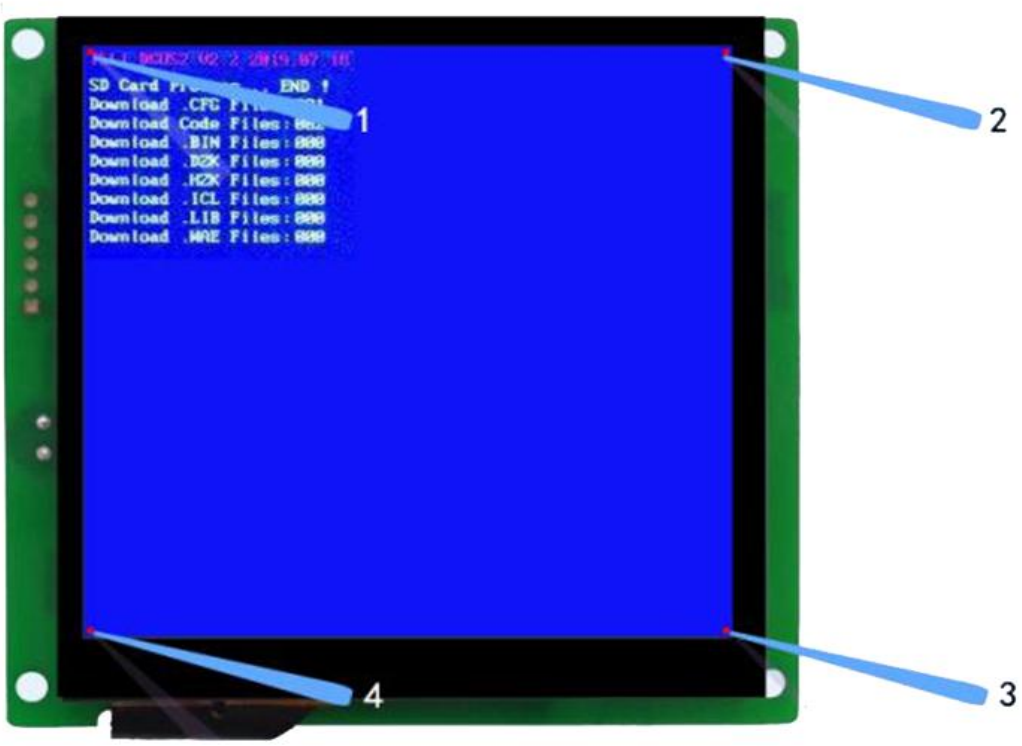
	Caution	Set 0x21.7 and 0x21.3 to 1 and download it to T5L screen to achieve touch calibration. Then download the CFG file to the screen, and carry out step 2 under the condition of continuous power supply.

0x21	1	TP_Mode	Touch screen mode configuration .7-.4 （high 4bit)type selection. .7=1,indicate that start touch calibration 0x0*= Resistance touch screen 0x1*= Capacitance touch screen derived by GT911, GT9271 or GT9110 IC 0x2*=ILI9881H Incell CTP .3 Resistance touch screen calibration: 0=off 1=on, enable when SD card downloading is in process .2-.0 (low 3bit) Coordinate setting: .2 X coordinate selection：0=from 0 to Xmax, 1=from Xmax to 0； .1 Y coordinate selection：0=from 0 to Ymax, 1=from Ymax to 0； .0 XY exchange：0=XY 1=YX 。 (3inch RTP is 0x06; 7inch RTP is 0x07; 8inch RTP is 0x05; 10.4inch RTP is 0x03)



Step 2: touch calibrate

Click the "top left corner, top right corner, bottom right corner and bottom left corner" of the touch panel successively with sharp objects, and some screens will display red pixel points to indicate that the calibration is completed.



T5L touch panel standard configuration reference DMG80480L070_01 WTR and DMG80480L080_01WTR only have resistance touch, the factory kernel is the TA instruction set development mode, and the other models are all DGUSII development mode.

DMG12720C050-03WTC and DMG72720C041-03WTC are Incell capacitive touch, LCD IPS all-in one screen.

Table 3.3 parameter configuration of T5L touch panel 0x21

Size	Resolution	model	0x21 standard configuration	
			Resistive screen (RTP)	Capacitance screen (CTP)
3.5	480*320	DMG48320C035_03W	03	10
4	800*480	DMG80480C040-03W	03	16
4	480*480	DMG48480C040-03W	05	16
4.1	720*720	DMG72720C041-03WTC	20 (Incell touch panel)	
4.3	480*272	DMG48270C043-03W	03	10
4.3	800*480	DMG80480C043-03W	05	16
5	854*480	DMG85480C050-03W	06	18
5	800*480	DMG80480C050-03W	03	10
5	1280*720	DMG12720C050-03WTC	20 (Incell touch panel)	
5.6	640*480	DMG64480C056-03W	00	12
7	800*480	DMG80480C070-03W	06	10
7	800*480	DMG80480L070_01WTR	07	-
7	1024*600	DMG10600C070-03W	06	10

8	800*600	DMG80600L080_01WTR	05	-
8	800*600	DMG80600C080-03W	05	10
8	1024*768	DMG10768C080-03W	05	10
9.7	1024*768	DMG10768C097-03W	06	19
10.1	1024*600	DMG10600C101-03W	06	10
10.4	800*600	DMG80600C104-03W	03	10

4.10 CFG Configuration File Generation

Users generally only need to set system configuration and backlight standby, which can meet general setting requirements.

4.11 File Configuration for Expanded NAND Flash

Can be expanded up to 64MB for 16MB Flash memory.

The 16MB Flash memory (up to 64mbytes Nor Flash or 48MB Nor+512MB NAND Flash) is divided into two parts:

(1)4-12MB font library space (single font library 256Kbytes) which can save font library, icon library, configuration file.

(2)4-12MB of storage space (.ICL file).

For the T5L1 CPU platform, the size of a single JPG image file should not exceed 252KB, and the size of a single file on the T5L2 CPU platform should not exceed 764KB.

It must be soldered to the position of Flash expansion 3 when using expanded NAND Flash, corresponding to font ID=0xC0-0xFF. Also, the size of each font is 8MB.

4.12 Process of Download File

DWIN_SET is unique folder name which can be identified by devices, which contains all files that will be downloaded to devices. It includes configuration files such as files of "13 touch configuration file.bin","14 variable configuration file.bin", "22_Config.bin" and other configuration files. When creating a new project, the software will automatically generate a series of files, among which "DWprj.hmi" is the only editing program that the DGUS software can recognize. This file cannot be renamed or deleted.

Since the T5L device cannot read JPEG pictures directly, it is necessary to make pictures and icons into the format of ICL file before downloading in practical applications. The generation process of DWIN_SET folder is as follows.

5 System Variable Interface

5.1 System Variable Interface

System variable address range: 0x0000-0x0FFF.

VP	Definition	R/W	Length (word)	Description
0x00	Reserved		4	Undefined
0x04	System_Reset	W	2	Writing 0x55AA 5AA5 resets the T5L CPU once. E.g.:5A A5 07 82 00 04 55 AA 5A A5 Reset T5L chip, clear all data, that is equivalent to power off for restarting.
0x06	OS_Update_CMD	W	2	D3: write 0x5A to enable DWIN OS once (write into 1MB Nor Flash), clear after operation. D2: Fixed at 0x10. DWIN OS program must start from 0x1000. D1:0: Start address of SRAM to save program to update, it must be even. E.g: This instruction is used for downloading and updating the user OS program. "DWIN OS Build" PC software has integrated the function of downloading the os.bin file into the screen through serial port 2 after compilation.
0x08	NOR_FLASH_RW_CMD	W	4	D7: Mode: 0x5A=read 0xA5=write, clear after operation. D6:4: Start address of Nor Flash. Must be even. 0x000000-0x03:FFFE, 256KWords. D3:2: Start address of data variable space. Must be even. D1:0: Data length to read/write. Must be even. E.g.: (1) Write the data 0x12345678 to the variable storage space 0x1002: 5A A5 07 82 10 02 12 34 56 78 (2)Store 2-word data of variable storage space 0x1002 address into Nor Flash database header address 0x000002: 5A A5 0B 82 0008 A5 000002 1002 0002 (3) Read data from Nor Flash database 0x000002 to variable storage space address 0x2000: 5A A5 0B 82 0008 5A 000002 2000 0002. After reading and storing the database with OS, it is necessary to judge whether the D7 address has been automatically cleared. If the D7 address has not been cleared, no operation will be carried out until it is cleared for other operations. Otherwise, logic disorder will occur. If the serial port is used for instruction reading and storage operation, other instructions can be sent after appropriate delay.
0x0C	Reserved		3	
0x0F	Ver	R	1	Application software version. D1 stands for GUI version, D0 stands for DWIN OS version. E.g: 5A A5 04 83 000F 01 return: 5A A5 06 83 000F 01 1110 (11 is GUI software

				version, 10 is DWIN OS software version) Application : (1)When using a display terminal, due to the control system, display terminal start time inconsistent (depending on the user's power supply capacity and electricity rate), to ensure that the control system of a start sending data display terminal received correctly and carry out the corresponding functions and commands, so need to confirm whether the display terminal has the normal operation, the user can be identified by this directive; (2)In the process of debugging need to know whether the system version, configuration and so on is correct, you can also send this instruction to read. (3)The transmission and response of this instruction is often used to verify that the communication between the computer serial port and the serial screen is normal.
0x10	RTC	R/W	4	D7=Year (0-0x63) D6=month(0-0x0C), D5=day(0-0x1F), D4=week(0-0x6), D3=hour(0-0x17), D2=minute(0-0x3B), D1=second(0-0x3B), D0 undefined. Data format is HEX. If there is no RTC on hardware, user can write RTC data. E.g.: Write: 5A A5 0B 82 0010 13 0A 01 00 0B 0C 0D 00 Text RTC display 2019-10-0111:12:13 SUN, The first two digits of the year are automatically identified; The week is displayed in English, and the system will convert automatically. Read: 5A A5 04 83 00 10 04 Answer: 5A A5 0C 83 00 10 04 13 0A 01 00 0B 0C 0D 00
0x14	PIC_Now	R	1	Display current page ID. Read only. E.g.: Read: 5A A5 04 83 0014 01 Answer: 5A A5 06 83 00 14 01 0007 (0007 is page 07)
0x15	GUI_Status	R	1	GUI status feedback: 0x0000=free, 0x0001=processing 13.bin and 14.bin. The operation state feedback is in millisecond unit level, which is generally applied in special cases. The user can judge whether the GUI kernel is occupied by the DWIN-OS program.
0x16	TP_Status		4	D7: 0x5A=touch panel data is updated. Others=touch panel coordinates not updated After reading the data, the user can write ≠ 0x5A as a mark. As long as the user does not touch, the mark will not be updated D6: touch panel status. 0x00=release, 0x01=first press, 0x02=lift, 0x03=pressing D5:D4=X coordinate D3:D2=Y coordinate D1:D0=0x0000. E.g.: Read: 5A A5 04 83 0016 03 Answer: 5A A5 0A 83 00 16 03 5A 02 02 55 00 E5

				Read the coordinates of the touch Write: 5A A5 05 82 0016 0000 reset touch panel data update mark. Usually, DWIN OS program can be used to deal with the judgment of whether to touch, long press, short press and other operations.
0x1A-0x30	Reserved		23	Undefined
0x31	LED_Now	R	1	D1: 0x5A : backlight brightness value, AD0-AD1 instant value has updated. D0: Current backlight brightness value, 0x00-0x64.
				E.g.: Read: 5A A5 04 83 0031 01 Answer: 5A A5 06 83 00 31 01 5A 64
0x32	AD0-AD7 Instantaneous value	R	8	Instantaneous value of AD0-AD1, 1 word per AD. Voltage=AD value*3300/4095 mV.
0x3A-0x79	Reserved		64	Undefined
0x7A	LCD_HOR	R	1	Horizontal resolution of the screen
				E.g.: Read: 5AA5 04 83 007A 01 Answer: 5A A5 06 83 00 7A 01 01 E0 (0x01E0 is horizontal resolution)
0x7B	LCD_VER	R	1	Vertical resolution of the screen
				E.g.: Read:5AA5 04 83 007B 01 Answer:5A A5 06 83 00 7B 01 01 10 (0x0110 is vertical resolution)
0x7C-0x7F	Reserved		4	Undefined
0x80	System_Config	R/W	2	D3: 0x5A = enable the parameter configuration once, clear after operation. D2: touch panel sensitivity configuration value, read only. D1: touch panel mode configuration value, read only. D0: system status setting. .7: Serial port CRC check 0=off 1=on, read only. .6: Reserved, write 0. .5: Power on load 22 file to initialize variable space. 1= load 0= no load, read only. .4: Variable automatic upload setting 1= on, 0= off, read and write. .3: Touch panel audio control 1= on 0= off, read and write. .2: Touch panel backlight standby control 1= open 0= close, read and write. .1. 0: display direction 00 = 0 °, 01 = 90 °, 10 =180 ° ,11 = 270 °, read and write.
				E.g.: Read: 5A A5 04 83 0080 02 Answer: 5A A5 08 83 00 80 02 00 14 10 38 Read commands are typically used to view the screen .CFG configuration . D0 can change the configuration. CFG file 0x05 address, for example, the

				next two instructions are on the corresponding.4= 1.5 =1, touch panel sound control off or on, can be applied to the touch button of the buzzer "mute" function. Note: write instruction is not saved after power off. Write: 5A A5 07 82 0080 5A 00 00 30 Write: 5A A5 07 82 0080 5A 00 00 38
0x82	LED_Config		2	Set standby backlight. D3= Turn on brightness, 0x00-0x64; When backlight standby control is off, D3 can be used for brightness adjustment interface. D2= Turn off brightness, 0x00-0x64; D1:0=wait time /10 ms.
				E.g.: 1. When CFG address 0x05 is configured to turn on backlight standby control of touch panel: 5A A5 07 82 0082 6432 03E8, 10 seconds after the screen backlight automatically reduced to 0x32 brightness. 5A A5 07 82 0082 2020 03E8, direct instruction control backlight brightness 0x20 brightness, and backlight screen saver brightness is consistent with standby. Note: the first physical touch on the touch panel after a certain period of time is to wake up the backlight function of the screen saver. Even if the brightness is the same as the setting value of standby brightness, "additional first awakening touch and click operation" is still needed after a certain period of time. 2. When CFG address 0x05 is configured with backlight standby control over touch panel: 5A A5 05 82 0082 0A, direct instruction control backlight brightness is 0x0A brightness.
0x84	PIC_Set	R/W	2	D3: 0x5A = enable page operation once, clear after operation. D2: Mode. 0x01=page switch (Display the image specified in the image storage area to the current background page). D1:D0: picture ID.
				E.g.: 5A A5 07 82 0084 5A01 0001, Display the 01 serial number picture under the background picture ICL file (If the serial number picture ID of the sending instruction does not exist, the instruction operation judgment will not take effect.)
0x86	PWM0_Set	R/W	2	D3=write 0x5A to enable PWM0 setting once, clear after operation. D2= frequency division coefficient D1:D0=PWM0 precision PWM0 carrier frequency = 825.7536MHz /(frequency division coefficient *PWM0 precision).
				E.g.: 1.100KHz configuration The division factor is set to 1，upper limit value=825.7536MHz/0.1MHz=8258 (0x2042).

				Write data 5A 01 20 42 to the variable of 0x0086, output 13 bit resolution, 100 KHz carrier PWM. 2.Write 0x0092 variable to control the output duty ratio of PWM0 Write value = high level ratio * upper limit value For example, 10%, write value is 10% * 8258 = 826 (0x033A)
0x88-0x91	Reserved		10	Undefined
0x92	PWM0_Out	R/W	1	D1:D0=PWM0 output high level width, 0x0000-PWM0 precision.
0x93-0x9B	Reserved		9	Undefined
0x9C	RTC_Set	W	4	D7:D6= write 0x5AA5 to enable RTC setting once; D5:D0=year, month, day, hour, minute, second, all in HEX format. Need hardware RTC support. Method1: instruction: 5A A5 0B 82 009C5AA512 06 1B 15 15 15 Method2: Input through the keyboard of the interface: T5L RTC input: set address 009C, 009D, 009E, 009F. Write 5AA5 to 009C to enable RTC setting, 009D high and low bytes correspond to month and year, 009E high and low bytes correspond to day and hour, 009F high and low bytes correspond to minute and second. ①Return key value: the address is set to 009C and the key value is set to 5AA5. Press "Save Settings". ②Variable data input: Address 009D, variable type is set to VP high byte for change year. Variable data input: Address 009D, variable type is set to VP low byte for change month. ③Variable data input: Address 009E, variable type is set to VP high byte for change day.Variable data input: Address 009E, variable type is set to VP low byte for change hour. ④Variable data input: Address 009F, variable type is set to VP high byte for change minute.Variable data input: Address 009F, variable type is set to VP low byte for change second.
0xA0	WAE Music_Play_Set	R/W	2	WAE music playback setting: D3: Segment ID of this playback, 0x00-0xFF. WAE file location is specified by CFG configuration file. D2: Number of segments, fixed 0x01. Clear after DGUS operation. Under buzzer mode, it is buzz time, unit: 8 ms. D1: Volume, unit: 1/64; Initial value is 0x40(100%). D0: Playback status feedback, 0x00=stop, 0x01=pause, 0x02=playing. E.g.: WAE file save address ①Through the 0x07 address of CFG file , set the saving location of. WAE file. For example: 26.WAE writes 0x1A to the 0x07 address. ②Configure the parameters of the 0x05 address of CFG file , for example, the position of 0x05 is written to 0x7C, where. 6 bit enable music playing. (.6: buzzer / music play selection, 0 = buzzer, 1 = music play.)

				③Command format: 5A A5 07 82 00 A0 03 01 40 00 play music with ID of 03 serial number in WAE file, play 1 segment, play volume with 100%. D0: play status feedback is read-only, write command is not defined, write 00.
0xA2-0xA9	Reserved		8	Undefined
0xAA	External 16Mbytes FLASH write operation	R/W	6	Update the external memory contents with 32Kbyes block as the benchmark. D11: 0x5A= enable external memory(16Mbytes-64Mbytes) read/write operation, clear after operation. D10: Operation mode, 0x01=read data 0x02=write 32Kybtes data block For D10=0x01 read data D9:font library ID, 0x10-0x1F, 256Kbytes per font library, maximum 4Mbytes D8:D6: the starting address of data in the font library is defined according to the Word, 0x0000-0x01FFFF D5:D4: The first address of the read data variable space, which must be even. D3:D2: The length of the data read, defined according to the Word, must be even. D1:D0: undefined, write 0x00. For D10=0x02 write 32Kbytes data block. D9:D8: 32Kybtes memory block address, 0x0000-0x01FF, correspond to 16Mbytes memory. D7:D6: The first address of the update data stored in the data variable space must be even. D5:D4: After the completion of this operation, the time of delay waiting for the next write operation, unit: 1ms. DGUS refresh will stop during the delay wait to prevent errors caused by incomplete updates. D3:D0: Undefined, write 0x00.
				E.g.: The address operation can realize the update function of serial port 2, such as picture library, word library, etc.The 16Mbytes memory can be seen as either 512 32Kbytes space for writing operation data update or 64 single font ID space with fixed capacity of 256KB. The following instruction 00B0 can also be seen as representing update to 22 font ID. 5A A5 0F 82 00 AA 5A 02 00 B0 80 00 00 14 00 00 00 00
0xB0	Touch instruction access interface	W	36	0xB0: 0x5AA5= enable accessing touch control interface once. Clear after CPU operation. 0xB1: Page ID of touch control. 0xB2: High byte: touch control ID (set in DGUS II development software), 0x01-0xFF; Low byte: touch control code, 0x00-0x7F. 0xB3: Access mode 0xB4-0xD3: data to modify of mode 0x02, 0x03. Mode 0x0000: turn off this touch control.

				Mode 0x0001: turn on this touch control. Mode 0x0002: Read this touch control and write it to SRAM that 0xB4 pointing to. Mode 0x0003: update current touch control with data that 0xB4 pointing to, the format and data length must be the same.
				E.g.: 5AA5 0B 82 00B0 5AA5 0001 0905 0000 (close the 9th touch file on page 0001. 05 represents the command code, 0000 represents to turn off the touch control.)Basic touch control cannot be turned on/off. If it is ranked in PC software, it does not need to be counted as effective touch number. If you need to change the specific properties of the touch, you need to use mode 03 to replace the 0xb4 prewritten touch file in the property configuration of the 13bin file. Search for the key words of 0xB0 in the DWIN forum, with detailed post description.
0xD4	TP operation simulation	W	5	0xD4: 0x5AA5=enable the operation once, clear after operation. 0xD5: press mode. 0x0001=press, 0x0002=release, 0x0003=keep pressing, 0x0004=touch (press + release) 0xD6: X coordinate of press position. 0xD7: Y coordinate of press position. After simulating mode 0x0001 and 0x0003, must simulate 0x0002. When the assembly touch function is running, x= 0xAA:KH y=0xA5:KL will directly return the key values KH and KL to the assembly touch. For example, when variables are input in assembly mode, the coordinates (0xAAF0, 0xA5F0) will cause the input to end immediately. When the 13 touch file is designed with keying function, X coordinate = 0xFF: key code y coordinate = 0x0001 will trigger the corresponding keying function.
				E.g.: 5AA5 0B 82 00D4 5AA5 0004 00EE 008F (0004 is click, press + lift, 00EE 008F (283,143) coordinates, Press mode 0x0001= press; 0x0002= release; 0x0003= continue pressing 0x0004=click After applying the simulated lift modes 0x0001 and 0x0003, there must be a simulated lift mode of 0x0002. (after "touch panel sound control" is enabled, touch panel operation simulation will also trigger the touch panel buzzer) (after “touch panel backlight standby control” is enabled, touch panel operation simulation will also wake up backlight)
0xD8	Pointer icon overlay display		4	0xD8_H: 0x5A = enable the overlay display. 0xD8_L: the position of ICL file which the pointer icon is saved at. 0xD9: the pointer icon ID. 0xDA: the X coordinates of pointer icon. 0xDB: the Y coordinates of pointer icon.

				The pointer icon is always displayed in the background filtering mode, and the background filtering intensity is fixed at 0x08.
				E.g.: 5A A5 0B 82 00D8 5A 2D 0001 0064 0064 call the 1 Icon of 45.icl icon to display directly on the screen page position (100100). Note: After switching page, icon overlay will not disappear. write 00 to 0xD8_ H or call the blank position icon to make the icon disappear. The ID of the icon can be selected by the user to achieve the image display effect of "click" and "wait" effect.
0xDC	Reserved		4	Undefined
0xE0	Memory CRC check	R/W	2	D3: 0x5AA5=enable the operation once, clear after operation. D2: memory type selection 0x00= font space (16Mbytes memory) 0x02=DWIN OS code 0x03=Nor flash database (LIB file). D1:D0: data interface. ▶ Enable the CRC Font check mode: D1= start font ID (256KB for each font); D0= the number of 4KB blocks checked, 0x00-0xFF. OS code check mode: D1:D0= starting at 0x1000, the byte length of OS code to be checked is 0x0001-0x7000. Nor flash database mode: D1:D0=Nor flash database ID, fixed validation of 4KB data each time. ▶After checking Return CRC value.
0xE2-0xEF	Reserved		14	Undefined
0xF0	Interface of playing music flow data	W	4	D7: 0x5A = enable the music playback operation, clear after operation. D6: mode, 0x00 = stop(clear the buffer), 0x01 = suspend(reserve the buffer), 0x02 = start. D5:D4: undefined, write 0x00. D3:D2: Variable memory address for storing music data, even. D1:D0: Music data word length, maximum 8KWords, even; data is 16bit integer format.
				This instruction is used to play online music files. First, send the music flow data to a variable address between 0x1000 and 0xFFFF. For example, the buffer address is 0x8000, Instruction sending format: 5A A5 0B 82 00F0 5A 02 00 00 8000 0400 Play music saved start from 0x8000, instruction with 1K music data.
0xF4	Painting interface	W	8	D15: 0x5A = enable the drawing touch window. D14: operation mode, 0x00 = normal mode, 0x01 = initialization (clear after initialization).

				D13: parameter configuration. .7-2 reserved, write 0. .1-0 painting buff processing mode after page changing. 0x00 = close, 0x01 = re-initialization, others = unchanged. D12: reserved, write 0x00. D11: line width, from 0x01 - 0x0F. D10:D8: painting color, D6=RED D5=GREEN D4=BLUE D7:D4: the coordinate position of the window upper left corner(x,y). D3:D2: the width of the pixel points window, must be divisible by 4 D1:D0: the height of the pixel points window, must be divisible by 4 The variable memory occupied by drawing touch window is 128KB(Double word address range 0x00:8000-0x00: FFFF) which is corresponding to 208*208 pixel.
				This command is used after touching screen. Display the track of coordinates in the designated area of the screen, which is generally used in "handwriting drawing function". E.g.: 0x00 mode=normal mode 1.Send drawing window command on a page: Tx:5A A5 13 82 00 F4 5A 00 00 00 02 FF 00 00 00 2C 00 BC 00 D0 00 D0 2.Hand touch panel drawing graphics 3.Switch page 4.Switch back to the previous page, send the drawing window command again, and the characters will be displayed again. In the upper left corner of the window (44,188), the coordinate point corresponds to the largest 208 * 208 pixel drawing window.
0xFC	Reserved		2	0x01 mode: initialization mode 1.Send drawing window command on a page: Tx:5A A5 13 82 00 F4 5A 01 00 00 02 FF 00 00 00 2C 00 BC 00 D0 00 D0 2.Hand touch panel drawing graphics 3.Switch page 4.Switch back to the previous page, send the drawing window command again, the characters are no longer displayed, and the blank space is convenient for redrawing graphics.
0xFE	UART1 high speed download		2	0x5AA5=enable the high-speed download operation through UART1
0x100-0x2FF	Reserved		512	Undefined
0x300-0x37F	Dynamic curve interface	R/W	128	0x300-0x30F: state feedback for 8 channel curve buffers (read only advised), 2 words per channel, high word is the storage pointer location (0x0000-0x07FF) where curve data is stored, and low word is the effective data length of curve buffer (0x0000-0x0800). Writing 0x0000 to the effective data length of the curve buffer will cause the curve unable display. 0x310-0x311: start writing curve buffer data D3: D2:0x5AA5 enable the writing curve buffer data operation once, and

				clear after operation. D1: the number of data blocks, 0x01-0x08. D0: undefined, write 0x00. 0x312-0x37F: data block written to the curve buffer, which is 16 bits unsigned. Single data block: data channel ID (0x00-0x07) + data word length (0x01-0x6E) + data. With dynamic curve display enabled, start at 0x1000 and create a data buffer for each curve according to 2Kwords per channel. The CH0 buffer is 0x1000-0x17FF, the CH1 buffer is 0x1800-0x1FFF, and so on, the unused curve buffer zones can be used as user variables. Also users can directly overwrite the curve buffer data and then modify 0x300-0x30F corresponding storage pointer position and data length to ensure the correct display of the curve.
0x380-0x3FF	Reserved		128	Undefined, user can't use
0x400-0x4FF	Network communication interface	R/W	256	WiFi and other Internet communication equipment application control interface.
0x500 -0x5BF	Multi-Media interface	R/W	192	Multi-Media application interface, 0x500-0x57F digital Multi-Media interface, 0x580-0x5BF analog Multi-Media interface
0x5C0-0x5FF	External memory interface	R/W	64	External memory interface (such as U disk) read or write interface
0x600-0xEFF	Reserved		2404	Undefined
0x0F00	Variable change indication	R	2	After setting variables to change the automatic upload function, this function is enabled. D3=5A means variable change, D2:D1= variable memory pointer, D0= variable length (word).

				E.g.:
				It is used to judge whether the touch data is uploaded to the serial port, or whether there is a touch key operation change for uploading.
				Because the time of this state is very short, it can't be read with the serial port 0x83 instruction. It is recommended to read the value of D3 with DWIN OS and then judge.
				LDWR R0,0F00H
				MOVXR R10,1,1
				IJNE R10,5AH,TEXT; Judge the value of R10，if it is 0x5A, process will excute in sequence, if not process will jump to TEXT
				....
				TEXT:

				E.g.: It is used to judge whether different address data are changed or not, and it can also be judged by OS.

				Send: 5A A5 04 83 0F 00 02 Response: 5A A5 08 83 0F 00 02 00 11 0001 indicates that the variable address 0x1100 has uploaded 1 word of data. Send:5A A5 04 83 0F 00 02 Response: 5A A5 08 83 0F 00 02 0010 00 02 indicates that the variable address 0x1100 has uploaded 2 word of data.
0xF02-0xFFF	Reserved		254	Undefined

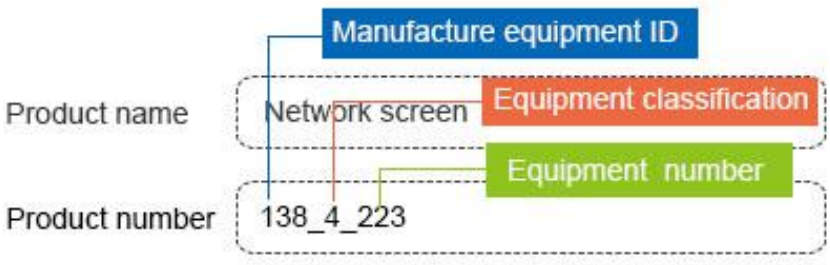
Info User variable occupancy address range: 0x1000-0xFFFF.

5.2 Network Interface

DWIN has developed WiFi module “WiFi-10” and Cloud platform “DWIN Cloud” for AIoT applications. WiFi -10 is specially designed for DGUS development platform. The data interface has been defined, which significantly reduce the development difficulty of customers and shorten development time. DWIN DGUSII platform (including T5 CPU series and T5L CPU series) has opened the network interface, with the WiFi module, it can be connected to DWIN Cloud through simple DGUS development. Based on the Cloud platform of DWIN, customers can realize remote APP control, data analysis, equipment operation and other functions.

Info The green filling part of the table indicates that the user needs to modify it selectively according to the actual situation. For other parts, it is recommended to refer to the default recommended value of DWIN and configure the double byte address corresponding to the 22. Bin file.

Definition	Address	Length (word)	Recommended value(hex)	Instructions
Network switch interface	0x400	1	5AA5	0x5AA5 indicates that the network communication interface is enabled Recommended value description: Generally 0x5AA5 is written as a fixed value to 22 file. It is also possible to use the configuration button to return or incremental adjustment and other touch controls to write the trigger key value 0x5AA5 to address 0x400 to achieve the opening of the network interface.
RAM ALARM	0x401	3	0000 0000 0000	D5-D4:0x5AA5 enable the RMA spatial data uploading to the server, clear after operation. D3-D2: RMA variable memory address to be uploaded. D1-D0: the word length of the RMA variable memory to be uploaded.

				(Currently up to 4KByte)
				Recommended value description: It is used to transmit the data of the variable address on the screen to the Cloud server, which is usually applied to the Cloud call view of the alarm history information and other parameters. If you do not need to use this function, you can fill in 0000 by default in 22 file.
Reserve	0x404	12	All are 00	Reserve.
Device description	0x410	1	5A45	High byte: 0x5A indicates that the device description is valid. Low byte: the encoding method and length of the device description text. .7-.6: encoding 0x00=UNICODE 0x01=GBK, GBK is recommended. .5-.0: describes the text length 0x00-0x34.
				Recommended value description: Write the configuration at a fixed value of the 22 file corresponding address, the encoding method GBK written here, the text length is 05
	0x411	2		Device manufacturer ID, assigned by DWIN factory, 0xFFFF: 0000 - 0xFFFF: FFFF segment is reserved. After the device is added to the DWIN cloud platform, it is automatically generated by the platform.
	0x413	1		The classification of each manufacturer's equipment is according to the DWIN classification standard equipment classification. After the device is added to the DWIN cloud platform, it is automatically generated by the platform.
	0x414	2		The individual device number of each type of device. After the device is added to the DWIN cloud platform, it is automatically generated by the platform.
	0x416	26	44475553325F543555494431(behind all 00)	Device description text, up to 52Bytes.
				Recommended value description: 44475553325F543555494431, corresponding to the ASCII code character DGUSII_T5UID1, has been written as a fixed value written in 22 file. The user can also make a text variable address association for display.
	Example of Cloud platform interface:  			
Device description (RMA mapping)	0x430	1	012C	RMA automatically refreshes the server interval 0x0000-0xFFFF in 0.1 seconds.
				Recommended value description: 0x0000 indicates that automatic refresh is not required. 0x012C indicates that the automatic refresh interval is 30 seconds.
	0x431	1	1000	RMA (mapped to the server's variable memory) read space start

				address, word address, out of bounds can not be read.
				Recommended value description: 0x1000 indicates that start reading space from 0x1000 address.
	0x432	1	0004	RMA read space size in 128Words with a maximum size of 2KBytes and 0x0000 indicates forbidden reading.
				Recommended value description: 0x0004 indicates that the read space size is 512Words, which can be set to 0x01F4 at most.
	0x433	1	2000	RMA write space start address, can overlap with the read space, can not write out of bounds.
				Recommended value description: 0x2000 indicates that the space is read from the address of 0x2000.
	0x434	1	0004	RMA write space size in 128Words with a maximum size of 2KWords, 0x0000 indicates write forbidden.
				Recommended value description: 0x0004 indicates that the read space size is 512Words, which can be set to 0x01F4 at most.
	0x435	3	All are 00	reserved
Device description (remote upgrade)	0x438	1	5AA5	The remote upgrade interface is enabled. 0x5AA5 indicates that the device is enabled with the remote upgrade interface. The system will automatically detect the Buffer.
				Recommended value description: Generally 0x5AA5 is written as a fixed value to 22 file. It is also possible to use the configuration button return or incremental regulation and other touch controls to write the trigger key value 0x5AA5 to address 0x438 to achieve the switch.
	0x439	1	0064	Remote upgrade packet timeout timer configuration in 0.1 second.
	0x43A	4	5A 00010 00000 0F00	The first remote upgrade space for the device (aligned to 4KB) is defined: D7:0x5A indicates that the remote upgrade space is enabled. D6-D3: 32-bit start address of the upgradeable space (lower 12 bits is 0), up to 4 GB. D2-D0: The size of the upgradeable space, in 4KB, up to 4GB.
	0x43E	4	5A00 0000 0000 8000	The second remote upgradeable space definition for the device
	0x442	4	All are 00	The third remote upgradeable space definition for the device
	0x446	4	All are 00	The fourth remotable upgrade space definition for the device.
	0x44A	2	5A 00 08 B0 00	Remote upgrade Buffer interface definition: D3:0x5A indicates that the remote upgrade Buffer is valid. D2: upgrade mode 0x00= the communication side is responsible for verifying the data CRC, and the error frame informs the host to resend. D1: the number of buffers available for remote upgrades, 0x01-0x10, up to 16. D0: the starting address of Buffer0 is high byte (word address) and low address is 8bit 0x00. Each Buffer is fixed in 2304 words (0x900) space, which is arranged

				in the back row. A single Buffer definition (the first 512 bytes is the control interface, the next 4KB is the data): D0:0x5A means to start the remote upgrade of this buffer once, and clear it after CPU processing. D1: Remote upgrade space selection, 0x00-0x03, one of 4 remote upgrade spaces. D2-D5: Write the destination address of the remote upgrade space. The lower 12 bits are 0 (aligned to 4KB). D6-D7: Data byte length, 0x0001-0x0FFF. D8-D9: Data CRC checksum. D10-D511: Reserved. D512: Data starts, up to 4096 bytes.
	0x44C	4	All are 00	reserve
Device description QR code	0x450	48		Device QR code. The QR code is automatically generated according to the device description data of 0x411, 0x413, and 0x414, and the function of downloading the APP, binding the device, and paying attention to the public number can be implemented with the mobile phone.
Communication device description	0x480	16	User-defined	D31: 0x5A indicates that the communication device description data is valid. D30: The encoding method and length of the device description text. .7-.6 encode mode 0x00=UNICODE 0x01=GBK, GBK is recommended. .5- .0 Description: The length of the text is 0x00-0x14. D29: Communication device category 0x01=WiFi D28: Communication device status, bit definition .7 Equipment working status 0=Configuration 1=Normal; .6-.3 undefined, write 0; .2- .0 signal quality, 0x00-0x07 a total of 8 levels, 0x00 means the worst, 0x07 means the best. D27-D20: 8Bytes communication device MAC address, low (D20) alignment. D19-D0: Description of the communication device in text format, up to 20Bytes. Use hexadecimal to describe the information as the corresponding ASCII character.
				The user can refer to the DWIN example's configuration at 22 to set 0x480 to write 0x5A 47 01 00 0x0482 can display the 8Bytes communication device MAC address value through the HEX variable display control
Communication device	0x490	8	User-defined	Restart WiFi module is valid D15-D14: The baud rate setting is reserved. Currently fixed at 921600bps, it cannot be set. D13-D9: Reserved. D8: WiFi module switching Debug firmware 5A means valid.

				D7: D0: WiFi module type identification information Default "DWD100".
WiFi_Config	0x498	4		D7: Writing 5A means starting the WiFi configure network, and the configure network is completed. The value is cleared to 0. D6: 0x5A means to start the network name and password to connect to the router (the network name and password are saved at 0x4B0) D7-D6, as long as it is started once, D7-D6 will be cleared at the same time after completion. The two Internet access modes cannot be started at the same time. If they are all 5A, the one-click configure network takes precedence. D5: Network time 5A means the screen comes with RTC (0x9C); 5B means RTC library RTC (0xF430) D4-D2: Reserved. D1: The WiFi module is automatically upgraded. 5A means enable. D0: Soft reset. 5A initiates reset and reset is cleared to 0.
				The user can design the configure button in the UI. Start manual configure network touch button (press to return 0x498 address write 0x005A) Start one-key configure network touch button (press to return 0x498 address write 0x5A00)
Reserve	0x49C	4	All are 00	reserved
Network status	0x4A0	8		D15-D14: WiFi version number. D13-D12: configure network status feedback 00: unmatched network 01: start configure network 02: in the configure network 03: configure network success 04: configure network failure D11-D10: network connection status 00:user name and password not obtained 01:WiFi router connection is successful 02: WiFi module self-upgrade 03: connect to the server 04: logged in to the server 05: connected to the Cloud
				The user can directly display the interface variable, use the 0x4A0 address to display the WiFi version number with the data variable display control, and the 0x4A1 address uses the icon variable to make 5 small icons for displaying the communication status. The 0x4A2 address uses 6 small icons to display the network connection status.
State machine	0x4A8	4		D7: state machine. D6: UART state machine. D3-D0: remaining stack space.
RTC	0x4AC	4		D7:5A means the time is valid.

				D6-D0:Year Month Day Week (0-6) Hours Minutes Seconds
Network information	0x4B0	32		0x4B0: SSID, the end must end 0xFF. 0x4C0: SN, the end must end at 0xFF.
				Text ASCII input and display controls can be used for input and display.

Description:

- WiFi-10 actively reads 0x490 and other registers every 0.1s-0.2s (the register that DGUS issues commands to WiFi-10).
- WiFi-10 will write the status of WiFi-10 to the DGUS register every 3-5 seconds (the status of the WiFi module: 0x480, 0x4A0, etc.).
- After completing the configure operation on the DGUS screen or mobile app, WiFi-10 will automatically access the Internet and start running data synchronization.
- The device description information is corresponding to the 0x0800-0x09FF byte address of the 22 initialization file, and the corresponding content configuration (regardless of whether the 22 file initialization variable buffer function is enabled, the underlying layer will automatically handle the loading).
- Cloud platform website: http://merchant.dwinhmi.com.cn/
- Server mapped memory debug interface website: http://tools.dwinhmi.com.cn/

6 Serial Communication Protocol

6.1 Introduction

Info

This section uses the following notation:

< >: One byte.

[ ]: Optional fields.

Numbers in Serial Commands are in hexadecimal format.

The native DWIN Protocol is composed of 5 commands:

• 0x80: Write Control Registers

• 0x81: Read Control Registers

• 0x82: Write VPs (RAM)

• 0x83: Read VPs (RAM)

A Frame (or packet) structure follows this format:

<Frame Header H> <Frame Header L> <Byte Count> <Command> [<Data>...] [<CRC H> <CRC L>]

Or, in abbreviated notation:

<FHH> <FHL> <BC> <CMD> [<DATA>...] [<CRCH> <CRCL>]

Frame Header: Identifies the start of a new DWIN Protocol packet. Can be used to uniquely identify a LCM on a communication bus. Default value = 0x5AA5.

• Byte Count: Counts the number of bytes in the packet, excluding the Frame Header and this byte, i.e., counts all the bytes starting from the Command byte.

• Command: Defines the Command to be executed.

• Data: Includes addresses, lengths and values.

• CRC: Optional error detection value

6.2 Control Register Commands

6.2.1 Write Register(0x80)

This Command writes one or more Control Registers. You can write multiple Registers at once, if they are sequential.

• Format

<FHH> <FHL> <BC> 80 <ID> <RG> <VL1> [<VL2> <VL3> ...]

<ID>: Register page ID 0x00-0x08).

<RG>: Register Address (0x00-0xFF).

<VL#>: Value(s) to write.

• Answer from LCM:

<FHH> <FHL> <BC> 80 <4F> <4B>

• Examples

Write the value 3 in Register 0x01 (same as setting the backlight level):

5AA5 0480 0001 03

00: Register page 0

01: Register Address 01

03: Data written to 01 register

Answer from LCM:

5AA5 0380 4F4B

6.2.2 Read Registers(0x81)

This Command reads one or more Control Registers. You can read multiple Registers at once, if they are sequential.

• Format

<FHH> <FHL> <BC> 81 <ID> <RG> <LEN>

<ID>: Register page ID (0x00-0x08)

<RG>: Register Address (0x00-0xFF).

<LEN>: Number of Registers (bytes) to read

• Answer from LCM:

<FHH> <FHL> <BC> 81 <ID> <RG> <LEN> <VL1> [<VL2> <VL3> ...]

<ID>: Register page ID (0x00-0x08)

<VL#>: Value(s) read;

<LEN>: Number of data (bytes)

• Examples

Read the value in Register 0x01 (same as reading the backlight level):

5AA5 0481 0001 01

00: Register page 0

01: Register Address 01

01: Number of Registers (bytes) to read

Answer from LCM:

5AA5 0581 0001 01 3F

00: Register page 0

01: Register Address 01

01: Number of data (bytes)

3F: Data

6.3 VP (RAM) Commands

6.3.1 Write VPs (0x82)

This Command writes one or more VPs. You can write multiple VPS at once, if they are sequential.

• Format

<FHH> <FHL> <BC> 82 <VPH><VPL> <VL1><VL1> [<VL2><VL2> <VL3><VL3> ...]

<VPH><VPL>: RAM Address.

<VL#><VL#>: Value(s) written.

• Answer from LCM:

<FHH> <FHL> <BC> 82 <4F> <4B>

• Examples

Write the value 1234 in VP 0x1000:

5AA5 0582 1000 04D2

1000: RAM Address

04D2: Data

Answer from LCM:

5AA5 0382 4F 4B

6.3.2 Read VPs (0x83)

This Command writes one or more VPs. You can write multiple VPS at once, if they are sequential.

• Format

<FHH> <FHL> <BC> 83 <VPH><VPL> <LEN>

<VPH><VPL>: RAM Address.

<LEN>: Number of VPs (words) to read.

• Answer from LCM:

<FHH> <FHL> <BC> 83 <VPH><VPL> <LEN> <VL1><VL1> [<VL2><VL2> <VL3><VL3> ...]

<VL#><VL#>: Value(s) read.

• Examples

Read the value in VP 0x1000:

5AA5 0483 1000 01

1000: RAM Address

01: Number of VPs (words) to read.

Answer from LCM:

5AA5 0683 1000 01 0002

0002: Value in RMA address 1000

6.4 CRC

DGUSII platform uses Cyclic Redundancy Check (CRC) to verify data integrity during communication. The specific variation used is CRC-16 Modbus.

The following pseudo-code explains how to calculate the CRC (already swapped).

```
start
   CRC ← 0xFFFF

   for each byte, do:
   {
         CRC ← CRC xor byte
         repeat
         {
                  If CRC.bit0 = 1, then
                  {
                        shift CRC right once
                        CRC ← CRC xor 0xA001
                  }
                  else
                        shift CRC right once
         }
         until 8 right shifts have been performed
   }

   Swap CRC
end
```

Some real code examples (already swapped):

• C Language:

```
// Compute the MODBUS RTU CRC
UInt16 ModRTU_CRC(byte[] buf, int len )
{
   UInt16 crc = 0xffff;

   for (int pos = 0; pos< len; pos++)
   {
        // XOR byte into least sig. byte of crc
        crc    ^= (UInt16)buf[pos];

        for (int i = 8; i != 0; i--) // Loop over each bit
        {
             if ((crc & 0x0001) !=0) // If the LSB is set
             {
                  crc >>= 1; // shift right and XOR 0xA001
                  crc ^= 0xA001;
             }
             else // Else LSB is not set
                  crc >>= 1; // Just shift right
        }
   }

// Swap high and low bytes
crc = ((crc & 0Xff00) >> 8) | ((crc & 0x00FF) << 8);

   return crc;
}
```

C# Language:

```
/// <summary>
/// Calculates CRC16 for a input byte array.
/// </summary>
/// <param name=”dataArray”>Input data.</param>
/// <returns>The calculated CRC16.<returns>
Public static ushort CalculateCrc16(byte[] dataArray)
{
    // Compute the MODBUS RTU CRC
    ushort crc = 0xFFFF;

    foreach (var data in dataArray)
    {
        crc ∧= data; // XOR byte into least significative byte of CRC
        for (var i= 8; I > 0; i--)
        {
            // Loop over each bit
            if ((crc & 0x0001) != 0)
            {
                // If the LSB is set
                crc >>= 1; // Shift right and XOR 0xA001
                crc ∧=0xA001;
            }
            else // Else LSB is not set
                crc >> 1; // Just shift right
        }
    }
    var highByte = (crc & 0xFF00);
    var lowByte = crc & 0x00FF;

    crc = (ushort) ((highByte >> 8) | (lowByte << 8));
    return crc;
}
```

7 Interface Objects

7.1 VP and SP Distribution

Most Interface Objects must be assigned to a Variable Pointer (VP). A VP is an address on the RAM space. Each VP points to a 2-byte (1 word) value.

For example, if an Incremental Input is assigned to VP 0x1000, when it is activated, it will increment the value stored at this VP. Assuming the initial value is 0, after two activations (touches on the touch panel), the new value stored on VP x1000 is 2.

VP	Contents	
	Decimal	Hexadecimal
0x1000	0	0x0000
0x1001	0	0x0000
0x1002	0	0x0000
0x1003	0	0x0000
0x1004	0	0x0000
0x1005	0	0x0000
0x1006	0	0x0000
0x1007	0	0x0000
0x1008	0	0x0000
0x1009	0	0x0000
0x100A	0	0x0000
0x100B	0	0x0000
0x100C	0	0x0000
0x100D	0	0x0000
...	...	...
0x6FFF	0	0x0000

VP	Contents	
	Decimal	Hexadecimal
0x1000	2	0x0002
0x1001	0	0x0000
0x1002	0	0x0000
0x1003	0	0x0000
0x1004	0	0x0000
0x1005	0	0x0000
0x1006	0	0x0000
0x1007	0	0x0000
0x1008	0	0x0000
0x1009	0	0x0000
0x100A	0	0x0000
0x100B	0	0x0000
0x100C	0	0x0000
0x100D	0	0x0000
...	...	...
0x6FFF	0	0x0000

A Description Pointer (SP) is a VP used to store the parameters of a Display Variable, so that they can be modified by the user at run-time.

Since there are no restrictions on address assignment, conflicts may show up on a project. For example, usually, you should not assign two Text Displays with Text Lengths greater than 2 to subsequent VPs, because their data will overwrite each other:

VP	Contents	
	Decimal	Hexadecimal
0x1000	0	0x0000
0x1001	0	0x0000
0x1002	0	0x0000
0x1003	0	0x0000
0x1004	0	0x0000
0x1005	0	0x0000
0x1006	0	0x0000
0x1007	0	0x0000
0x1008	0	0x0000
0x1009	0	0x0000
0x100A	0	0x0000
0x100B	0	0x0000
0x100C	0	0x0000
0x100D	0	0x0000
…	…	…
0x6FFF	0	0x0000

Display control VP can not overlap with others.

To avoid this problem, always keep in mind how many VPs an Interface Object takes, space them properly:

VP	Contents		
	Decimal	Hexadecimal	Description
0x1000	0	0x0000	Numeric Display 1
0x1001	0	0x0000	Numeric Display 2
0x1002	0	0x0000	Numeric Display 3
0x1003	0	0x0000	Text Display
0x1004	0	0x0000	
0x1005	0	0x0000	
0x1006	0	0x0000	
0x1007	0	0x0000	
0x1008	0	0x0000	Unassigned (Free) Space
0x1009	0	0x0000	
0x100A	0	0x0000	
0x100B	0	0x0000	
0x100C	0	0x0000	
0x100D	0	0x0000	
…	…	…	
0x6FFF	0	0x0000	

To improve scalability and avoid unwanted data overlapping, we recommend planning your project's address allocation beforehand, so that you can reserve addresses between Objects, and give them plenty of space to expand:

VP	Contents		
	Decimal	Hexadecimal	Description
0x1000	0	0x0000	Numeric Display1
0x1001	0	0x0000	Unassigned or Reserved Space
0x1002	0	0x0000	Numeric Display3
0x1003	0	0x0000	Unassigned or Reserved Space
0x1004	0	0x0000	Text Display
0x1005	0	0x0000	Text Display
0x1006	0	0x0000	Text Display
0x1007	0	0x0000	Text Display
0x1008	0	0x0000	Text Display
0x1009	0	0x0000	Unassigned or Reserved Space
0x100A	0	0x0000	Unassigned or Reserved Space
0x100B	0	0x0000	Unassigned or Reserved Space
0x100C	0	0x0000	Numeric Display 2
0x100D	0	0x0000	Unassigned or Reserved Space
…	…	…	Unassigned or Reserved Space
0x6FFF	0	0x0000	Unassigned or Reserved Space

7.2 Controls

Controls provide direct user input interaction. They can be considered as buttons. All Controls can be activated by a physical touch from the user, and most of them can be activated via Serial Communication. They are usually employed to modify the contents of the RAM space, although they may be used purely as Serial Communication triggers.

	Caution	Controls can't have overlapping areas.

All Controls have at least 5 parameters, detailed in the following table:

Definition	Length(bytes)	Description
Pic_ID	2	Picture ID, high 4bit is button audio ID selection , 0x00 indicates no touch audio.
TP_Area	8	Upper left corner (Xs, Ys) and lower right corner (Xe, Ye) of control button area. If the position of corresponding area is set out of the LCM resolution, user can achieve analog trigger touch button with specific key value in the system variable interface.
Pic_Next	2	Target switch page ID after button press operation, 0xFF** means no page switch.
Pic_On	2	Button press effect page, 0xFF** means no button press effect.
TP_Code	2	Operation Code for the Control. High byte defines the operation mode: 0xFF** = Invalid. 0xFE** = Normal Control. Auto-Send Data enabled for this Control. 0xFD** = Normal Control. Auto-Send Data disabled for this Control. Other Values = Basic Touch Control. High byte and Low byte are ASCII Code characters. Low byte defines the Control Type. If high byte is neither 0xFF, 0xFE or 0xFD, low byte is ASCII Code.
TP_FUN	0,16,32	When TP_Code = 0xFE**, it is used to describe key control function.

7.2.1 Variable Data Input

Address	Definition	Length(byte)	Description
0x00	Pic_ID	2	Page ID
0x02	TP_Area	8	Area of the Control: upper left corner (Xs, Ys), lower right corner(Xe, Ye).
0x0A	Pic_Next	2	Target switch page, 0xFF** means no page switch
0x0C	Pic_On	2	Button press effect page, 0xFF** means no button press effect.
0x0E	TP_Code	2	0xFE00, variable data input key code.
0x10	0xFE	1	0xFE
0x11	*VP	2	Variable Pointer
0x13	V_Type	1	Return variable type： 0x00=2bytes: Integer: -32768 to 32767; Unsigned integer: 0 to 65535 0x01=4bytes: Long integer: -2147483648 to 2147483647 Unsigned long integer: 0 to 4294967295 0x02=*VP high byte, unsigned number: 0 to 255 0x03=*VP low byte, unsigned number: 0 to 255 0x04=8bytes, extra long integer: -9223372036854775808 to 9223372036854775807
0x14	N_Int	1	Integer digits. For example, enter 1234.56, N_ Int=0x04
0x15	N_Dot	1	Decimal digits. For example, enter 1234.56, N_ Int=0x02
0x16	（x，y）	4	Display location: right alignment, （x，y）is the upper right coordinate of the string input cursor. Undefined when use assembly keyboard（KB_Source=0x0F）
0x1A	Color	2	Display color Undefined when use assembly keyboard（KB_Source=0x0F）
0x1C	Lib_ID	1	ASCII Font position, default 0x00 Undefined when use assembly keyboard（KB_Source=0x0F）
0x1D	Font_Hor	1	Font size .0x00 means black, otherwise it is white Undefined when use assembly keyboard（KB_Source=0x0F）
0x1E	Cusor_Color	1	Cursor color. Undefined when use assembly keyboard（KB_Source=0x0F）
0x1F	Hide_En	1	0x00:The entered text is displayed as *; other values are displayed according to the entered content.
0x20	0xFE	1	0xFE
0x21	KB_Source	1	0x00=Current page; 0x01=Other page; 0x0F=Assembly keyboard
0x22	PIC_KB	2	The page ID where the keyboard is located is valid only if KB_Source is not equal to 0x00. Assembly keyboard（KB_Source=0x0F）= assembly function file

			number
0x24	AREA_KB	8	Keyboard area coordinates: upper left coordinate (Xs, Ys), lower right coordinate (Xe,Ye) Valid only if the keyboard is not on the current page, KB_Source does not equal 0x00.
0x2C	AREA_KB_Position	4	The upper-left coordinate of the position the keyboard displays on the current page, only if the keyboard is not on the current page or the assembly keyboard.
0x30	0xFE	1	0xFE
0x31	Limits_En	1	0xFF:indicates that input range restriction is enabled and out of the range is invalid (equivalent to cancellation); Other values indicates that the input has no range limit.
0x32	V_min	4	Enter lower limit, 4 bytes (long integer or unsigned long integer).
0x36	V_max	4	Input upper limit, 4 bytes (long integer or unsigned long integer).
0x3A	Return_Set	1	0x5A: During entry, Return_VP address is loaded with Return_Data and automatically restored after completion. 0x00: data is not loaded during entry Function of loading data: it is mainly used in combination with SP (description pointer) modification of variable display to realize automatic marking of multi-parameter input process, such as modifying font color, size, launching a (bit) variable icon or area anti-color. It can also be used as the mark bit of input process to realize special requirements with DWIN_OS development.
0x3B	Return_VP	2	Load data VP address during entry
0x3D	Return_Data	2	Data is loaded into Return_VP during entry
0x3F	Layer_Gama	1	The transparency of the keyboard when it is not on the current page or the assembly keyboard. Range: 0x00~0xFF.

Info Effective code of data input are 0x0030-0x0039, 0x002E(.), 0x002D(+/-), 0x00F0(ESC), 0x00F1(confirm), 0x00F2(backspace).

7.2.2 Popup Window

Opens a keyboard in a popup window-style.

Address	Definition	Length(byte)	Description
0x00	Pic_ID	2	Page ID
0x02	TP_Area	8	Area of the Control: upper left corner (Xs,Ys), lower right corner (Xe,Ye).
0x0A	Pic-Next	2	Target switch page ID after button press operation, 0xFF** means no page switch.
0x0C	Pic_On	2	Button press effect page, 0xFF** means no button press effect.
0x0E	TP_Code	2	0xFE01, popup window key code.
0x10	0xFE	1	0xFE
0x11	*VP	2	Variable Pointer
0x13	VP_Mode	1	0x00: Write 0x00** key code to the VP (integer) 0x01: Write 0x01**key code to high byte of the VP(VP_H) 0x02: Write 0x02**key code to low byte of the VP(VP_L) 0x10-0x1F: Write the lowest bit of ** (1bit) to specific bit of VP(0x10 corresponds to VP.0, 0X1F corresponds to VP.F)
0x14	Pic_Menu	2	Page ID of the menu
0x16	Area_Menu	8	Menu area: upper left coordinate (Xs, Ys), lower right coordinate (Xe, Ye).
0x1E	Menu_Position_x	2	Upper left corner x-coordinate of the current page
0x20	0xFE	1	0xFE
0x21	Menu_Position_Y	2	Upper left corner y-coordinate of the current page
0x23	Translucent	1	The transparency setting. Range:0x00~0xFF.
0x24	NULL	12	Write 0x00

7.2.3 Incremental Adjustment

Used to implement a button that increments the content of a VP.

Address	Definition	Length(byte)	Description
0x00	Pic_ID	2	Page ID
0x02	TP_Area	8	Area of the Control: upper left corner (Xs,Ys), lower right corner (Xe,Ye).
0x0A	Pic_Next	2	Target switch page ID after button press operation, 0xFF** means no page switch.
0x0C	Pic_On	2	Button press effect page, 0xFF** means no button press effect.
0x0E	TP_Code	2	0xFE02, incremental adjustment key code
0x10	0xFE	1	0xFE
0x11	*VP	2	Variable Pointer, returned data is defined by VP_ Mode
0x13	VP_Mode	1	0x00: Adjust the VP(integer) 0x01: Adjust high byte of the VP(1-byte unsigned integer, VP_H) 0x02: Adjust low byte of the VP(1-byte unsigned integer, VP_L) 0x10-0x1F: Adjust the specific bit of VP(0x10 corresponds to VP.0, 0x1F corresponds to VP.F), range:0-1.
0x14	Adj_Mode	1	0x00= - - others=++
0x15	Return_Mode	1	0x00: Disabled. The value stops changing when max. or min. values are reached. Other Values: Enabled. The value loops around the range when it reaches max. or min. values.
0x16	Adj_Step	2	0x0000-0x7FFF
0x18	V_Min	2	Minimum value accepted by the control: 2-bytes integer(When VP_Mode is 0x01 or 0x02, only low byte is effective)
0x1A	V_Max	2	Maximum value accepted by the control: 2-bytes integer(When VP_Mode is 0x01 or 0x02, only low byte is effective)
0x1C	Key_Mode	1	0x00: Continuous. The value is changed while the user holds it. 0x01: One-Step. The value is changed once per touch.
0x1D	NULL	3	Write 0x00

7.2.4 Slide Adjustment

Used to implement a sliding button that dynamically changes the content of a VP.

Address	Definition	Length(byte)	Description
0x00	Pic_ID	2	Page ID
0x02	TP_Area	8	Area of the Control: upper left corner (Xs,Ys), lower right corner (Xe,Ye).
0x0A	Pic_Next	2	Target switch page ID after button press operation, 0xFF** means no page switch.
0x0C	Pic_On	2	Button press effect page, 0xFF** means no button press effect.
0x0E	TP_Code	2	0xFE03, slide adjustment key code
0x10	0xFE	1	0xFE
0x11	*VP	2	Variable Pointer
0x13	Adj_Mode	1	High 4bit defines the VP_Mode: 0x0*: Adjust the VP(integer) 0x1*: Adjust high byte of the VP(1-byte unsigned integer, VP_H) 0x2*: Adjust low byte of the VP(1-byte unsigned integer, VP_L) Low 4bit defines the slider mode: 0x*0: Horizontal. 0x*1: Vertical.
0x14	Area_Adj	8	Sliding Area: (Xs,Ys) (Xe,Ye) .Should be equal to “Touch Area”.
0x1C	V_Begin	2	Minimum value accepted by the control.
0x1E	V_End	2	Maximum value accepted by the control.

7.2.5 Return Key Value

Return key value is return the setting value to corresponding address.

Address	Definition	Length(byte)	Description
0x00	Pic_ID	2	Page ID
0x02	TP_Area	8	Area of the Control: upper left corner (Xs,Ys), lower right corner (Xe,Ye).
0x0A	Pic_Next	2	Target switch page ID after button press operation, 0xFF** means no page switch.
0x0C	Pic_On	2	Button press effect page, 0xFF** means no button press effect.
0x0E	TP_Code	2	0xFE05
0x10	0xFE	1	0xFE
0x11	*VP	2	Variable Pointer
0x13	TP_Mode	1	0x00: Write 0x00** key code to the VP (integer) 0x01: Write 0x01**key code to high byte of the VP(VP_H) 0x02: Write 0x02**key code to low byte of the VP(VP_L) 0x10-0x1F: Write the lowest bit of ** (1bit) to specific bit of VP(0x10 corresponds to VP.0, 0X1F corresponds to VP.F)
0x14	Key_Code	2	Return value
0x16	Hold_Time	1	Unit 0.1s, response when the press time exceed the Hold_Time, 0x00 indicate that response immediately.
0x17	NULL	10	Write 0x00

7.2.6 Text Input

Opens a keyboard for alphanumeric (text) values input. It uses terminator characters (0xFF) to signal end of text.

7.2.6.1 ASCII Input

Address	Definition	Length(byte)	Description
0x00	Pic_ID	2	Page ID
0x02	TP_Area	8	Area of the Control: upper left corner (Xs,Ys), lower right corner (Xe,Ye).
0x0A	Pic_Next	2	Target switch page ID after button press operation, 0xFF** means no page switch.
0x0C	Pic_On	2	Button press effect page, 0xFF** means no button press effect.
0x0E	TP_Code	2	0xFE06
0x10	0xFE	1	0xFE
0x11	*VP	2	Variable Pointer
0x13	VP_Len_Max	1	Maximum text length, in words (two characters for each word). Range: [0x01,0x7B]. When the text is saved in the specific VP, it will add the 0xFFFF as terminator automatically at the ending of the text. Thus the variable space is actually VP_Len_Max+1
0x14	Scan_Mode	1	Input mode. 0x00: re-input, 0x01: modify existing text.
0x15	Lib_ID	1	Index in the FLASH memory of the ASCII Font to use. 0x00=default.
0x16	Font_Hor	1	Font width, in pixels.
0x17	Font_Ver	1	Font height, in pixels. When using “Lib_ID” = 0x00, it must be twice the Width.
0x18	Cursor_Color	1	Color of the input cursor. 0x00: Black Other Values: White.
0x19	Color	2	Text color.
0x1B	Scan_Area_Start	4	Upper-left coordinates of the area where the text will be displayed: (Xs, Ys).
0x1F	Scan_Return_Mode	1	0x55= saves the input end tag and valid data length at the * (vp-1) position. * (vp-1) high byte, input end mark: 0x5A means input end, 0x00 means idle or input state. * (vp-1) low byte, effective input data length, byte unit. 0x00= does not return the input end tag and data length.
0x20	0xFE	1	0xFE
0x21	Scan_Area_End	4	Lower-right coordinates of the area where the text will be displayed: (Xe, Ye).
0x25	KB_Source	1	Indicates if the keyboard image is in the same screen as this control. 0x00: Keyboard on current page. Other Values: Keyboard on another page.

0x26	PIC_KB	2	Page ID of the Screen used as image source for this control. It's the Page ID where the keyboard image is. Not used if "External_Keyboard" = 0x00.
0x28	Area_KB	8	Keyboard area coordinate: upper left corner (Xs,Ys), lower right corner (Xe,Ye).
0x30	0xFE	1	0xFE
0x31	AREA_KB_Position	4	Upper-left coordinates of the pasting area of the keyboard. Not used if "External_Keyboard" = 0x00.
0x35	Display_EN	1	0x00=Normal display; 0x01=Characters displayed as asterisks(*)
0x36	Layer_Gama	1	The transparency of the keyboard when it is not on the current page. Range: 0x00~0xFF.
0x37	NULL	9	Write 0x00

7.2.6.2 GBK Input

Address	Definition	Length(byte)	Description
0x00	Pic_ID	2	Page ID
0x02	TP_Area	8	Area of the Control: upper left corner (Xs,Ys), lower right corner (Xe,Ye).
0x0A	Pic_Next	2	Target switch page ID after button press operation, 0xFF** means no page switch.
0x0C	Pic_On	2	Button press effect page, 0xFF** means no button press effect.
0x0E	TP_Code	2	0xFE06, GBK input key code.
0x10	0xFE	1	0xFE
0x11	*VP	2	Variable Pointer
0x13	VP_Len_Max	1	Maximum text length, in words (two characters for each word). Range: [0x01,0x7B]. When the text is saved in the specific VP, it will add the 0xFFFF as terminator automatically at the ending of the text. Thus the variable space is actually VP_Len_Max+1
0x14	Scan_Mode	1	Input mode. 0x00: re-input, 0x01: modify existing text.
0x15	Lib_GBK1	1	Index in the FLASH memory of the GBK Font to use after entry. 0x00=default.
0x16	Lib_GBK2	1	Index in the FLASH memory of the GBK Font to use before entry.
0x17	Font_Scale1	1	Lib_GBK1 font size，lattice number
0x18	Font_Scale2	1	Lib_GBK2 font size, lattice number
0x19	Cursor_Color	1	Color of the input cursor. 0x00: Black Other Values: White.
0x1A	Color0	2	Text color after entry
0x1C	Color1	2	Text color during entry
0x1E	PY_Disp_Mode	1	Input process, pinyin prompt and corresponding Chinese characters display: * 0x00= the pinyin prompt is displayed on the top, and the corresponding Chinese character is displayed on the bottom;

			The pinyin prompt and Chinese characters are displayed left-aligned, with Scan_Dis between the lines. * 0x01= pinyin prompt is displayed on the left, and corresponding Chinese characters are displayed on the right; Chinese prompt x position is Scan1_Area_Start+ x Font_Scale2+Scan_Dis.
0x1F	Scan_Return_Mode	1	0xAA= saves the input end tag and valid data length at the * (vp-1) position. * (vp-1) high byte, input end mark: 0x5A means input end, 0x00 means still in input. * (vp-1) low byte, effective input data length, byte unit. 0xFF= does not return the input end tag and data length.
0x20	0xFE	1	0xFE
0x21	Scan0_Area_Start	4	Upper-left coordinates of the area where the text will be displayed: (Xs, Ys).
0x25	Scan0_Area_End	4	Lower-right coordinates of the area where the text will be displayed: (Xe, Ye).
0x29	Scan1_Area_Start	4	Input process, Pinyin prompt text display area of the upper left coordinates.
0x2D	Scan_Dis	1	During the input process, the spacing of each character is displayed. Each line displays up to 8 characters.
0x2E	0x00	1	0x00
0x2F	KB_Source	1	Indicates if the keyboard image is in the same Screen as this Control. 0x00: Keyboard on current Screen. Other Values: Keyboard on another Screen.
0x30	0xFE	1	0xFE
0x31	PIC_KB	2	Page ID of the Screen used as image source for this control. It's the Page ID where the keyboard image is. Not used if "External_Keyboard" = 0x00.
0x33	Area_KB	8	Keyboard area coordinate: upper left corner (Xs,Ys), lower right corner (Xe,Ye).
0x3B	Area_KB_Position	4	Upper-left coordinates of the pasting area of the keyboard.
0x3F	Scan_Mode	1	0x02= pinyin input method 0x03= phonetic input method.

The keys on the keyboard must be designed with Basic Touch Controls. Each Basic Touch's "TP_Code" must be assigned to a code. For digit keys, use ASCII codes:

Code	Lower	Upper	Code	Lower	Upper	Code	Lower	Upper	Code	Lower	Upper
0x7E60	`	~	0x5171	q	Q	0x4161	a	A	0x5A7A	z	Z
0x2131	1	!	0x5777	w	W	0x5373	s	S	0x5878	x	x
0x4032	2	@	0x4565	e	E	0x4464	d	D	0x4363	c	C
0x2333	3	#	0x5272	r	R	0x4666	f	F	0x5676	v	V
0x2434	4	$	0x5474	t	T	0x4767	g	G	0x4262	b	B
0x2535	5	%	0x5979	y	Y	0x4868	h	H	0x4E6E	n	N
0x5E36	6	^	0x5575	u	U	0x4A6A	j	J	0x4D6D	m	M
0x2637	7	&	0x4969	i	I	0x4B6B	k	K	0x3C2C	,	<
0x2A38	8	*	0x4F6F	o	O	0x4C6C	l	L	0x3E2E	.	>
0x2839	9	(	0x5070	p	P	0x3A3B	;	:	0x3F2F	/	?

0x2930	0	)	0x7B5B	[	{	0x2227	‘	“	0x2020	SP	SP
0x5F2D	-	_	0x7D5D	]	}	0x0D0D	Enter	Enter			
0x2B30	=	+	0x7C5C	\	|						

For control keys, use these codes:

Code	Definition	Description
0x00F0	Cancel	Cancels the input, doesn’t change any data.
0x00F1	Return	Completes the input, writing the value to the VP.
0x00F2	Backspace	Deletes the character to the left of the cursor.
0x00F3	Delete	Deletes the character to the right of the cursor.
0x00F4	Caps Lock	Caps Lock. Button Effect must be enabled for this function.
0x00F7	Left	Moves the cursor left. It is used to turn page during the GBK input.
0x00F8	Right	Moves the cursor right. It is used to turn page during the GBK input.

7.2.7 Synchrodata Return

Synchrodata return includes three status: first press, continuous pressing and loosen pressing.

Address	Definition	Length(byte)	Description
0x00	Pic_ID	2	Page ID
0x02	TP_Area	8	Area of the Control: upper left corner (Xs,Ys), lower right corner (Xe,Ye).
0x0A	Pic_Next	2	Target switch page ID after button press operation, 0xFF** means no page switch.
0x0C	Pic_On	2	Button press effect page, 0xFF** means no button press effect.
0x0E	TP_Code	2	0xFE08, synchrodata return key code.
0x10	0xFE	1	0xFE
0x11	TP_On_Mode	1	When the touch panel is pressed down for the first time, data return mode: 0x00= no data is returned 0x01= read *VP2S pointing to LEN2 length data sent in DGUS serial protocol format to serial 2.
0x12	VP1S	2	Read the data in the address when first pressing.
0x14	VP1T	2	Write the data to the address when first pressing.
0x16	0x00	1	0x00
0x17	LEN1	1	Return data length in bytes. When TP_On_Mode=0x01, LEN1 must be even.
0x18	0xFE	1	0xFE
0x19	TP_On_Continue_Mode	1	When the touch panel is pressed down during continuous pressing, data return mode: 0x00= no data is returned 0x01= read *VP2S pointing to LEN2 length data sent in DGUS serial protocol format to serial 2.
0x1A	VP2S	2	Read the data in the address when continuous pressing.
0x1C	VP2T	2	Write the data in the address when continuous pressing.
0x1E	0x00	1	0x00
0x1F	LEN2	1	Return data length in bytes. When TP_On_Continuous_Mode=0x01, LEN2 must be even.
0x20	0xFE	1	0xFE
0x21	TP_OFF_Mode	1	When the touch panel is loosen pressing, data return mode: 0x00= no data is returned 0x01= read *VP2S pointing to LEN2 length data sent in DGUS serial protocol format to serial 2.
0x22	VP3S	2	Read the data in the address after loosen pressing.
0x24	VP3T	2	Write the data in the address after loosen pressing.
0x26	0x00	1	0x00
0x27	LEN3	1	Return data length in bytes. When TP_OFF_Mode=0x01, LEN3 must be even.
0x28	0x00	8	Reserved，write 0x00

7.2.8 Rotation Adjustment

Rotation adjustment function can realize variable data input through rotation button.

Address	Definition	Length(byte)	Description
0x00	Pic_ID	2	Page ID
0x02	TP_Area	8	Area of the Control: upper left corner (Xs,Ys), lower right corner (Xe,Ye).
0x0A	Pic_Next	2	Target switch page ID after button press operation, 0xFF** means no page switch.
0x0C	Pic_On	2	Button press effect page, 0xFF** means no button press effect.
0x0E	TP_Code	2	0xFE09, rotation adjustment key code.
0x10	0xFE	1	0xFE
0x11	*VP	2	Variable Pointer
0x13	Data_Format	1	0x00: Adjust the VP(integer) 0x01: Adjust high byte of the VP(1-byte unsigned integer, VP_H) 0x02: Adjust low byte of the VP(1-byte unsigned integer, VP_L)
0x14	（X，Y）	4	Center coordinate
0x18	R0	2	Inner diameter
0x1A	R1	2	Outer diameter
0x1C	A0	2	Start angle, range from 0 to 719,unit 0.5°
0x1E	V_Begin	2	Value for start angle, integer.
0x20	0xFE	1	0xFE
0x21	A1	2	End angle, range from 0 to 719,unit 0.5°
0x23	V_End	2	Value for end angle, integer

7.2.9 Sliding Adjustment

Sliding adjustment can slide on the screen according to specific direction and return the corresponding value in real time.

Address	Definition	Length(byte)	Description
0x00	Pic_ID	2	Page ID
0x02	TP_Area	8	Area of the Control: upper left corner (Xs,Ys), lower right corner (Xe,Ye).
0x0A	Pic_Next	2	Target switch page ID after button press operation, 0xFF** means no page switch.
0x0C	Pic_On	2	Button press effect page, 0xFF** means no button press effect.
0x0E	TP_Code	2	0xFE0A, sliding adjustment.
0x10	0xFE	1	0xFE
0x11	*VP	2	Variable pointer, return adjustment data. *VP reserved； *VP+1 return data： High byte: adjustment direction, 0x00=increase,0xFF= decrease； Low byte: adjustment size.
0x13	Adj_Mode	1	0x00 horizontal sliding；0x01vertical sliding
0x14	Step_Dis	1	Sensitivity，0x01-0xFF

7.2.10 Page Sliding

Setting the target and area of page switching, use to achieve page turning through sliding.

Address	Definition	Length(byte)	Description
0x00	Pic_ID	2	Page ID
0x02	TP_Area	8	Area of the Control: upper left corner (Xs,Ys), lower right corner (Xe,Ye). Trigger only.
0x0A	Pic_Next	2	Target switch page ID after button press operation, 0xFF** means no page switch.
0x0C	Pic_On	2	Button press effect page, 0xFF** means no button press effect.
0x0E	TP_Code	2	0xFE0B, page sliding key code.
0x10	0xFE	1	0xFE
0x11	Pic_Front	2	Previous page, 0xFF**=none.
0x13	Pic_Next	2	Next page, 0xFF**=none.
0x15	Pic_Area	4	Slide the Y coordinates of the upper right and lower right corners of the area.
0x19	Push_Speed_Set	1	Response time, range :0x01-0xFF，unit 40 ms。
0x1A	Push_Dis_Set	2	Response sliding minimum space in X coordinates.
0x1C	FB_En	1	0x00=no upload； Others=upload page ID
0x1D	End_Carton_Speed	1	0x00=end without animation； Others=unit of pixels divided by DGUS period, recommended value is horizontal resolution divided by 20.

7.2.11 Sliding Icon Selection

Used to realize ICON sliding on the page.

Address	Definition	Length(byte)	Description
0x00	Pic_ID	2	Page ID
0x02	TP_Area	8	Area of the Control: upper left corner (Xs,Ys), lower right corner (Xe,Ye). Trigger only. It must be consistent with icon display area of the 0x07 display variable.
0x0A	Pic_Next	2	Undefined, write FFFF.
0x0C	Pic_On	2	Undefined, write FFFF.
0x0E	TP_Code	2	0xFE0C, sliding icon selection key code.
0x10	0xFE	1	0xFE
0x11	*VP	2	Variables pointer
0x13	Adj_Mode	1	0x00 horizontal sliding；0x01vertical sliding
0x14	TP_Page_ID _ICON	2	Icon ID 0x0000= undefined
0x16	reserved	10	Write 0x00

7.3 Display Variables

Display Variables provide visual interaction to the users. They function as numeric, textual and graphic indicators. Display Variables are always associated to a memory address, and show the contents of such address in some human-readable form.

Caution Make sure to reserve enough VPs for each Display Variable. Consider their data sizes and their SP lengths (when SPs are used).

7.3.1 Variables Icon

Used to show an Icon from an Icon Library. The current Icon is determined by the value of the VP.

Address	SP Address	Definition	Length (bytes)	Description
0x00		0x5A00	2	
0x02		*SP	2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x04		0x000A	2	
0x06	0x00	*VP	2	Variable Pointer, integer.
0x08	0x01	（x,y）	4	Upper-left coordinates of the Icons to display
0x0C	0x03	V_Min	2	Minimum value. Values smaller than “V_Min” will show no Icons.
0x0E	0x04	V_Max	2	Maximum value. Values greater than “V_Max” will show no Icons.
0x10	0x05	Icon_Min	2	Icon associated to the V_Min.
0x12	0x06	Icon_Max	2	Icon associated to the V_Max.
0x14	0x07:H	Icon_Lib	1	Index in the FLASH memory of the Icon Library to use.
0x15	0x07:L	Mode	1	Display mode: 0x00=transparency mode Others= background display
0x16	0x08:H	Layer_Mode	1	0x00：overlay 0x01：overlay mode 1 0x02：overlay mode 2
0x17	0x08:L	ICON_Gamma	1	ICON brightness in overlay mode 2, range 0x00-0xFF, unit 1/256.
0x18	0x09:H	PIC_Gamma	1	Background brightness in overlay mode 2, range 0x00-0xFF, unit 1/256.
0x19	0x09:L	Filter_Set	1	Filter set value in transparency mode, range 0x00-0x3F.

7.3.2 Animation Icon

Used to show a loop animation of Icons from an Icon Library. The animation state is determined by the value of the VP.

Address	SP Address	Definition	Length (bytes)	Description
0x00		0x5A01	2	
0x02		*SP	2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x04		0x000D	2	
0x06	0x00	*VP	2	Variable Pointer. High word: Unsigned Integer(0x0000-0xFFFF). stores animation status Low word: Reserved
0x08	0x01	（x,y）	4	Upper-left coordinates of the Icons to display
0x0C	0x03	Reset_ICON_En	2	Indicates whether the animation should always start from the first frame when restarting. 0x0000: Animation continues from the last shown frame when reset. 0x0001: Animation starts from the first frame (“Icon_Start”) when reset.
0x0E	0x04	V_Stop	2	Value that stops the animation.
0x10	0x05	V_Start	2	Value that starts the animation.
0x12	0x06	ICON_Stop	2	Icon displayed when the animation is stopped.Range:0x0000-0x00FF
0x14	0x07	ICON_Start	2	Icon displayed at the first frame of the animation.
0x16	0x08	ICON_End	2	Icon displayed at the last frame of the animation.
0x18	0x09:H	ICON_Lib	1	Index in the FLASH memory of the Icon Library to use.
0x19	0x09:L	Mode	1	ICON display mode: 0x00: Transparent background. Other Values: Opaque background
0x1A	0x0A:H	Layer_Mode	1	0x00: overlay 0x01: overlay mode 1 0x02: overlay mode 2
0x1B	0x0A:L	ICON_Gamma	1	ICON brightness in overlay mode 2, range 0x00-0xFF, unit 1/256.
0x1C	0x0B:H	PIC_Gamma	1	Background brightness in overlay mode 2, range 0x00-0xFF, 1/256.
0x1D	0x0B:L	Time	1	Time of single ICON, unit DGUS cycle, range 0x01-0xFF.
0x1E	0x0C:H	Display mode	1	0x00: loop mode. 0x01: single mode. When the variable is VP_Stop, it play an animation once from Icon_End to Icon_Start.

				When the variable is VP_Start, it play an animation once from Icon_Start to Icon_End. When the variable are other values, it display stop icon.
0x1F	0x0C:L	Filter_Set	1	Filter set value in transparency mode, range 0x00-0x3F.

7.3.3 Slider Display

Used to show an Icon that moves along a given axis (horizontal or vertical) based on the value of the VP. Typically used in linear graphs, or in conjunction with Slider Inputs.

Address	SP Address	Definition	Length (bytes)	Description
0x00		0x5A02	2	
0x02		*SP	2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x04		0x000C	2	
0x06	0x00	*VP	2	Variable Pointer.
0x08	0x01	V_Begin	2	Minimum value accepted by the control.
0x0A	0x02	V_End	2	Maximum value accepted by the control.
0x0C	0x03	x_Begin	2	Minimum position of the slider, when its value equals “V_Begin”. X coordinate for horizontal sliders, Y coordinate for vertical sliders.
0x0E	0x04	x_End	2	Maximum position of the slider, when its value equals “V_Begin”. X coordinate for horizontal sliders, Y coordinate for vertical sliders.
0x10	0x05	ICON_ID	2	Index of the Icon in the Icon Library that will be used as the Slider handle.
0x12	0x06	Y	2	Position of slider icon in the secondary axis. Y coordinate for horizontal sliders, X coordinate for vertical sliders.
0x14	0x07:H	x_adj	1	Icon position offset adjustment on the primary axis, in pixels. Offset in the X axis for horizontal sliders, and in the Y axis for vertical sliders.
0x15	0x07:L	Mode	1	Slider mode. 0x00: Horizontal 0x01: Vertical.
0x16	0x08:H	ICON_Lib	1	Index in the FLASH memory of the Icon Library to use.
0x17	0x08:L	ICON_Mode	1	ICON display mode: 0x00: Transparent background. Other Values: Opaque background
0x18	0x09:H	VP_Data_Mode	1	Value Memory Size. 0x00: *VP points to the VP(integer). 0x01: *VP points to the High Byte of the VP(VP_H). 0x02: *VP points to the Low Byte of the VP(VP_L).
0x19	0x09:L	Layer_Mode	1	0x00：overlay 0x01：overlay mode 1 0x02：overlay mode 2
0x1A	0x0A:H	ICON_Gamma	1	ICON brightness in overlay mode 2, range 0x00-0xFF, unit 1/256.

0x1B	0x0A:L	PIC_Gamma	1	Background brightness in overlay mode 2, range 0x00-0xFF, 1/256.
0x1C	0x0B:H	Filter_Set	1	Filter set value in transparency mode, range 0x00-0x3F.

7.3.4 Artistic Variables

Works like a Numeric Display, using Icons instead of Fonts. Typically used when you need to display numeric information that needs an anti-aliased look. It uses fixed-point integer values.

Address	SP Address	Definition	Length (bytes)	Description
0x00		0x5A03	2	
0x02		*SP	2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x04		0x0009	2	
0x06	0x00	*VP	2	Variable Pointer.
0x08	0x01	（X，Y）	4	Start display position: Upper-left coordinates of the Icons to display in Left- alignment mode or Upper-right coordinates of the Icons to display in Right-alignment mode.
0x0C	0x03	ICON0	2	Icon corresponding to digit 0. The Icon Library must follow this indexing order: [0123456789-.].
0x0E	0x04:H	ICON_Lib	1	Index in the FLASH memory of the Icon Library to use.
0x0F	0x04:L	ICON_Mode	1	0x00: Transparent background. Other Values: Opaque background
0x10	0x05:H	Integer_Digits	1	Number of digits to the left of the decimal separator.
0x11	0x05:L	Decimal_Digits	1	Number of digits to the right of the decimal separator.
0x12	0x06:H	VP_Mode	1	Value Memory Size. 0x00: 16-bit Integer(2 bytes). Range: -32768 ~ 32767 0x01: 32-bit Integer(4 bytes). Range: -2147483648 ~ 2147483647 0x02: 8-bit Unsigned Integer in High Byte of the VP. Range: 0 ~ 255 0x03: 8-bit Unsigned Integer in Low Byte of the VP. Range: 0 ~ 255 0x04: 64-bit Integer(8 bytes). Range: -9223372036854775808 ~ 9223372036854775807 0x05: 16-bit Unsigned Integer(2 bytes). Range: 0 ~ 65535 0x06: 32-bit Unsigned Integer(4 bytes). Range: 0 ~ 4294967295 0x07: float point number(4 bytes) 0x08:double float point number(8 bytes)
0x13	0x06:L	Alignment	1	0x00: Left- alignment. 0x01: Right- alignment.
0x14	0x07:H	Layer_Mode	1	0x00: overlay 0x01: overlay mode 1 0x02: overlay mode 2
0x15	0x07:L	ICON_Gamma	1	ICON brightness in overlay mode 2, range 0x00-0xFF, unit 1/256.
0x16	0x08:H	PIC_Gamma	1	Background brightness in overlay mode 2, range 0x00-0xFF, unit 1/256.
0x17	0x08:L	Filter_Set	1	Filter set value in transparency mode, range 0x00-0x3F.

7.3.5 Image Animation

Used to create an animation of Screens. Can be implemented via serial communication as a series of Screen jumps.

Address	SP Address	Definition	Length (bytes)	Description
0x00		0x5A04	2	
0x02		*SP	2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x04		0x0005	2	
0x06	0x00	0x0000	2	0x0000
0x08	0x01	Pic_Begin	2	Page ID of the first animation frame.
0x0A	0x02	Pic_End	2	Page ID of the last animation frame.
0x0C	0x03:H	Frame_Time	1	Time spent in each animation Screen. Range: 0x00-0xFF, in 8 ms steps.
0x0D	0x03:L	ICL_LIB_ID	1	Image ID in the FLASH memory, 0x00~0xFF. 0x00: configured by SD card.
0x0E	0x04	Pic_End_Exp	2	Return page after playing animation saved in NANA Flash
0x10	0x05	Reserved	16	0x00

7.3.6 Icon Rotation

Used to show an Icon that pivots around a given rotation center, based on the value of the VP. Typically used in radial graphs, like speedometers and dials.

SP Address	Definition	Length (bytes)	Description	SP Address
0x00		0x5A05	2	0x5A05.
0x02		*SP	2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x04		0x000C	2	0x000C.
0x06	0x00	*VP	2	Variable Pointer.
0x08	0x01	ICON_ID	2	Index of the Icon in the Icon Library that will be rotated (usually a dial needle).
0x0A	0x02	ICON_xc	2	Center of rotation on the Icon. X coordinate.
0x0C	0x03	ICON_Yc	2	Center of rotation on the Icon. Y coordinate.
0x0E	0x04	xc	2	Center of rotation on the Screen. The center of rotation of the Icon is placed in this point, and pivots around it. X coordinate.
0x10	0x05	Yc	2	Center of rotation on the Screen. The center of rotation of the Icon is placed in this point, and pivots around it. Y coordinate.
0x12	0x06	V_Begin	2	Minimum value.
0x14	0x07	V_End	2	Maximum value.
0x16	0x08	AL_Begin	2	Minimum angle, associated to “V_Begin”. Given in 0.5° steps. Range: 0-720(0x000-0x2D0), which is equivalent to 0° to 360°.
0x18	0x09	AL_End	2	Maximum angle, associated to “V_End”. Given in 0.5° steps. Range: 0-720(0x000-0x2D0), which is equivalent to 0° to 360°.
0x1A	0x0A:H	VP_Mode	1	Value Memory Size. 0x00: *VP points to the VP (integer). 0x01: *VP points to the High Byte of the VP (VP_H). 0x02: *VP points to the Low Byte of the VP (VP_L).
0x1B	0x0A:L	Lib_ID	1	Index in the FLASH memory of the Icon Library to use.
0x1C	0x0B	Mode	1	0x00: Transparent background. Other Values: Opaque background

7.3.7 Bit Variable Icon

Used to show fixed and/or animated Icons, according to a bit flag value on the VP. The value of each bit represents the state of a single Icon, and many Icons can be shown, in different states, based on the VPC.

Typically used to display several alarms at once, or to implement bar graphs.

<table>
<tr><th>Address</th><th>SP Address</th><th>Definition</th><th>Length (bytes)</th><th>Description</th></tr>
<tr><td>0x00</td><td></td><td>0x5A06</td><td>2</td><td>0x5A06</td></tr>
<tr><td>0x02</td><td></td><td>*SP</td><td>2</td><td>Parameter Pointer.
0xFFFF: Disables SP (no run-time modification).</td></tr>
<tr><td>0x04</td><td></td><td>0x000C</td><td>2</td><td>0x000C</td></tr>
<tr><td>0x06</td><td>0x00</td><td>*VP</td><td>2</td><td>Variable Pointer.</td></tr>
<tr><td>0x08</td><td>0x01</td><td>*VP_AUx</td><td>2</td><td>Auxiliary Variable Pointer. 2 words. User can not use them, should be allocated right after VP.</td></tr>
<tr><td>0x0A</td><td>0x02</td><td>Act_Bit_Set</td><td>2</td><td>Indicates which bits are displayed.
0b1: Active bit.
0b0: Inactive bit.</td></tr>
<tr><td>0x0C</td><td>0x03:H</td><td>Display_Mode</td><td>1</td><td>The following table describes what icons are shown when each bit value is either 0 or 1.
<table>
<tr><td rowspan="2">Mode</td><td colspan="2">Bit Value</td></tr>
<tr><td>0</td><td>1</td></tr>
<tr><td>0x00</td><td>ICON0S</td><td>ICON1S</td></tr>
<tr><td>0x01</td><td>ICON0S</td><td>None</td></tr>
<tr><td>0x02</td><td>ICON0S</td><td>Animation:ICON1S-ICON1E</td></tr>
<tr><td>0x03</td><td>None</td><td>ICON1S</td></tr>
<tr><td>0x04</td><td>None</td><td>Animation:ICON1S-ICON1E</td></tr>
<tr><td>0x05</td><td>Animation:ICON0S-ICON0E</td><td>ICON1S</td></tr>
<tr><td>0x06</td><td>Animation:ICON0S-ICON0E</td><td>None</td></tr>
<tr><td>0x07</td><td>Animation:ICON0S-ICON0E</td><td>Animation:ICON1S-ICON1E</td></tr>
</table></td></tr>
<tr><td>0x0D</td><td>0x03:L</td><td>Move_Mode</td><td>1</td><td>Bit icons arranged mode.
0x00: Horizontal, no space reserved for inactive bits.
0x01: Vertical, no space reserved for inactive bits.
0x02: Horizontal, space reserved for inactive bits.
0x03: Vertical, space reserved for inactive bits.</td></tr>
<tr><td>0x0E</td><td>0x04:H</td><td>Icon_Mode</td><td>1</td><td>ICON display mode:
0x00: Transparent background.
Other Values: Opaque background.</td></tr>
<tr><td>0x0F</td><td>0x04:L</td><td>Icon_Lib</td><td>1</td><td>Index in the FLASH memory of the Icon Library to use.</td></tr>
<tr><td>0x10</td><td>0x05</td><td>ICON0S</td><td>2</td><td>Modes 0, 1, 2; Bit value = 0; Icon shown.
Modes 5, 6, 7; Bit value = 0: First icon in animation mode.</td></tr>
<tr><td>0x12</td><td>0x06</td><td>ICON0E</td><td>2</td><td>Modes 5, 6, 7; Bit value = 0: Last icon in animation mode.</td></tr>
<tr><td>0x14</td><td>0x07</td><td>ICON1S</td><td>2</td><td>Modes 0, 3, 5; Bit value = 1; Icon shown.
Modes 2, 4, 7; Bit value = 1: First icon in animation mode.</td></tr>
<tr><td>0x16</td><td>0x08</td><td>ICON1E</td><td>2</td><td>Modes 2, 4, 7; Bit value = 1: Last icon in animation mode.</td></tr>
</table>

0x18	0x09	（x，y）	4	Upper-left coordinates of the Icons to display
0x1C	0x0B	DIS_MOV	2	The size reserved for each Icon, in pixels.
0x1E	0x0C	Reserved	2	0x00

7.3.8 Batch Icon Quick Copy and Paste

Used to display a batch of icon in very short time, for example, develop a small game in screen.

SP Address	Definition	Length (bytes)	Description
0x00		2	
0x02		2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x06	0x00	4	Variable Pointer, must be even. D3: 0x5A = enable the display, others = disable the display. D2: displayed ICON ID, range 0-N. D1: display mode, only works for the display of background copy . .7 filter switch 0=open filter,1=display background. .6 reserved,0. .5-.0 filter set value 0x01-0x3F。 D0: brightness of ICON, only works for the display of background copy. If the brightness of ICON is not set to 0xFF, the ICON will display together with background, and the display speed will be slower about 30% .
0x08	0x01	4	The position of the ICON to display, must be even. D3:D2: X coordinate of the upper-left ICON corner. D1:D0: Y coordinate of the upper-left ICON corner. The data is described through the number sequence of the ICON ID(0-N).
0x0A	*VP2	8	Parameter pointer of the source ICON, must be even. Every ICON occupy 4 bytes. D7:D6: X coordinate of the upper-left ICON corner. D5:D4: Y coordinate of the upper-left ICON corner. D3:D2: width pixels of the ICON. D1:D0: Height pixels of the ICON. The data is described through the number sequence of the ICON ID(0-N).
0x0C	Data_Num	2	Number of the source ICON, range 0-4096.
0x0E	Icon_Source	1	0x00: current page. Others: ICON picture.
0x0F	Icon_Lib	1	Index in the FLASH memory of the Icon Library to use. Only works in ICON picture mode.
0x10	Icon_ID	2	ID of the ICON. Only works in ICON picture mode.
0x12	Reserved	14	0x00

7.3.9 Data Variables

Used to display numeric information. It uses fixed-point integer values.

Address	SP Address	Definition	Length (bytes)	Description
0x00		0x5A10	2	0x5A10
0x02		*SP	2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x04		0x000D	2	0x000D
0x06	0x00	*VP	2	Variable Pointer
0x08	0x01	X,Y	4	Upper-left coordinates of the text to display.
0x0C	0x03	COLOR	2	Text color.
0x0E	0x04:H	Lib_ID	1	Index in the FLASH memory of the ASCII Font to use.
0x0F	0x04:L	Font_Width	1	Font width, in pixels.
0x10	0x05:H	Alignment	1	0x00: Left-alignment. 0x01: Right-alignment. 0x02: Center- alignment.
0x11	0x05:L	Integer_Digits	1	Number of digits to the left of the decimal separator.
0x12	0x06:H	Decimal_Digits	1	Number of digits to the right of the decimal separator.
0x13	0x06:L	VP_Mode	1	Value Memory Size. 0x00: 16-bit Integer (2 bytes). Range: -32768 ~ 32767 0x01: 32-bit Integer (4 bytes). Range: -2147483648 ~ 2147483647 0x02: 8-bit Unsigned Integer in High Byte of the VP. Range: 0 ~ 255 0x03: 0x03: 8-bit Unsigned Integer in Low Byte of the VP. Range: 0 ~ 255 0x04: 64-bit Integer (8 bytes). Range: -9223372036854775808 ~ 9223372036854775807 0x05: 16-bit Unsigned Integer (2 bytes). Range: 0 ~ 65535 0x06: 32-bit Unsigned Integer (4 bytes). Range: 0 ~ 4294967295
0x14	0x07:H	Len_unit	1	Length of text to append after the digits, in characters. range 0-11.
0x15	0x07:L	String_Unit	Max 11	Text to append after the digits. ASCII code.

7.3.10 Text Display

Used to display textual information.

Address	SP Address	Definition	Length (bytes)	Description
0x00		0x5A11	2	0x5A11
0x02		*SP	2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x04		0x000D	2	0x000D
0x06	0x00	*VP	2	Variable Pointer
0x08	0x01	（X，Y）	4	Upper-left coordinates of the text to display.
0x0C	0x03	Color	2	Text color.
0x0E	0x04	(Xs,Ys)(Xe,Ye)	8	Upper- left coordinates and lower-right of the area where the text will be displayed.
0x16	0x08	Text_Length	2	Maximum text length, in bytes (characters), stop display when data is 0x0000 or 0xFFFF.
0x18	0x09:H	Font0_ID	1	Index in the FLASH memory of the Font to use, for encoding modes0x01 - 0x04.
0x19	0x09:L	Font1_ID	1	Index in the FLASH memory of the Font to use, for encoding modes0x00 and 0x05, and other non-ASCII fonts for encoding modes0x01 -0x04.
0x1A	0x0A:H	Font_x_Dots	1	Font width, in pixels. Must be even. In encoding modes 0x01 -0x04, must be twice the Width.
0x1B	0x0A:L	Font_Y_Dots	1	Font height, in pixels.
0x1C	0x0B:H	Encode_Mode	1	Character spacing mode is defined by bit7: 0: Automatic spacing. 1: Fixed spacing. Font encoding is defined by bits 6 to 0: 0x00: 8 bit. 0x01: GB2312. 0x02: GBK. 0x03: BIG5 0x04: SJIS 0x05: UNICODE.
0x1D	0x0B:L	HOR_Dis	1	Spacing between characters, in pixels.
0x1E	0x0C:H	VER_Dis	1	Spacing between lines, in pixels.
0x1F	0x0C:L	Reserved	1	0x00

7.3.11 RTC Display

(1)Used to display current date and/or time, in digital format. Uses the internal RTC.

Address	SP Address	Definition	Length (bytes)	Description
0x00		0x5A12	2	0x5A12
0x02		*SP	2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x04		0x000D	2	0x000D
0x06	0x00	0x0000	2	0x0000
0x08	0x01	(X,Y)	4	Upper-left coordinates of the text to display.
0x0C	0x03	Color	2	Text color.
0x0E	0x04:H	Lib_ID	1	Index in the FLASH memory of the ASCII Font to use.
0x0F	0x04:L	Font_Width	1	Font width, in pixels.
0x10	0x05	String_Code	MAx16	Display format string. Use ASCII characters and the Field Codes on the following table. E.g.: Current time =2012-05-02 12:00:00 Wednesday, Y-M-D H: Q: S 0x00, will display“2012-05-02 12:00:00”.M-D W H: Q 0x00, will display “05-02 WED 12:00”.

(2)Used to display current time, in analog format (a radial clock). Uses the internal RTC.

Address	SP Address	Definition	Length (bytes)	Description
0x00		0x5A12	2	0x5A12
0x02		*SP	2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x04		0x000D	2	0x000D
0x06	0x00	0x0001	2	0x0001
0x08	0x01	（X,Y）	4	Center of rotation on the Screen. The center of rotation of the Icon is placed in this point, and pivots around it.
0x0C	0x03	Icon_Hour	2	Index of the Icon showing the Hour hand. 0xFFFF: none.
0x0E	0x04	Icon_Hour_Central	4	Center of rotation on the Hour Icon.
0x12	0x06	Icon_Minute	2	Index of the Icon showing the Minute hand. 0xFFFF: none.
0x14	0x07	Icon_Minute_Central	4	Center of rotation on the Minute Icon.
0x18	0x09	Icon_Second	2	Index of the Icon showing the Second hand. 0xFFFF: none.
0x1A	0x0A	Icon_Second_Central	4	Center of rotation on the Second Icon.
0x1E	0x0C:H	Icon_Lib	1	Index in the FLASH memory of the ASCII Font to use.
0x1F	0x0C:L	Reserved	1	0x00

7.3.12 HEX Data

Used to display numeric information in hexadecimal format, with optional digit separators.

Address	SP Address	Definition	Length (bytes)	Description
0x00		0x5A13	2	0x5A13
0x02		*SP	2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x04		0x000D	2	0x000D
0x06	0x00	*VP	2	Variable Pointer. The data will be displayed in HEX format when half-byte data is greater than 0x9, e.g.: 0x32: display 32, 0xBF: display BF.
0x08	0x01	（X，Y）	4	Upper-left coordinates of the text to display.
0x0C	0x03	Color	2	Text color.
0x0E	0x04:H	Mode	1	.7: BCD encoded switch. 0= off. 1= on. .6-.4: reserved. .3-.0: Byte numbers to be displayed, 0x01 - 0x0F.
0x0F	0x04:L	Lib_ID	1	Index in the FLASH memory of the Font to use.
0x10	0x05:H	Font_x	1	Font width, in pixels.
0x11	0x05:L	String_Code	MAX15	Sequence of characters (ASCII) representing the separators for this Hex Display. The current value (contained on VP) will be shown in hexadecimal, and after each byte, a separator character is inserted. Special characters: 0x00 (blank), 0x0D (new line).

7.3.13 Roll Text

Text scroll function is the variable data scroll display in the specified area with the specified direction .

Address	SP Address	Definition	Length (bytes)	Description
0x00		0x5A14	2	0x5A14
0x02		*SP	2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x04		0x000B	2	0x000B
0x06	0x00	*VP	2	Text Variable Pointer. First three VP must be reserved. Text is saved after the 3rd VP and ended with 0x00 or 0x0F.
0x08	0x01:H	Rolling_Mode	1	0x00: from left to right.
0x09	0x01:L	Rolling_Dis	1	Rolling space in a DGUS cycle, in pixel.
0x0A	0x02:H	Adjust_Mode	1	0x00: Left-alignment. 0x01: Right-alignment. 0x02: Center- alignment. Rolling will stop when the text length smaller than textbox.
0x0B	0x02:L	Reserved	1	0x00
0x0C	0x03	Color	2	Text color.
0x0E	0x04	Xs Ys Xe Ye	8	Upper- left coordinates and lower-right of the area where the text will be displayed.
0x16	0x08:H	Font0_ID	1	Index in the FLASH memory of the Font to use, for encoding modes0x01 - 0x04.
0x17	0x08:L	Font1_ID	1	Index in the FLASH memory of the Font to use, for encoding modes0x00 and 0x05, and other non-ASCII fonts for encoding modes0x01 -0x04.
0x18	0x09:H	Font_X_Dots	1	Font width, in pixels. Must be even. In encoding modes 0x01 -0x04, must be twice the Width.
0x19	0x09:L	Font_Y_Dots	1	Font height, in pixels.
0x1A	0x0A:H	Encode_Mode	1	Character spacing mode is defined by bit7: 0: Automatic spacing. 1: Fixed spacing. Font encoding is defined by bits 6 to 0: 0x00: 8 bit. 0x01: GB2312. 0x02: GBK. 0x03: BIG5 0x04: SJIS 0x05: UNICODE.
0x1B	0x0A:L	Text_Dis	1	Spacing between characters, in pixels.
0x1C	0x0B:H	Reserved	4	0x00

7.3.14 Data Window

The data window indicator displays the data variables in a specified display window, highlighting the selected values.

Address	SP Address	Definition	Length (bytes)	Description
0x00		0x5A15	2	0x5A15
0x02		*SP	2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x04		0x 000C	2	0x000C
0x06	0x00	*VP	2	Variable Pointer, two words. The second one is reserved.
0x08	0x01	V_MIN	2	Minimum value.
0x0A	0x02	V_MAX	2	Maximum value.
0x0C	0x03:H	Integer_Digits	1	Number of digits to the left of the decimal separator.
0x0D	0x03L	Decimal_Digits	1	Number of digits to the right of the decimal separator.
0x0E	0x04:H	Data_Num	1	Display data amount
0x0F	0x04L	Mode	1	0x01: display invalid zero. 0x02: display positive(+). 0x04: cyclic data.
0x10	0x05	（x,y）	4	Center coordinates.
0x14	0x07	Adjust_step	2	Step length.
0x16	0x08:H	Font_x_Dots	1	Font width of unselected data, in pixels
0x17	0x08:L	Font_Y_Dots	1	Font height of unselected data, in pixels.
0x18	0x09	Color	2	Text color unselected.
0x1A	0x0A:H	Font_x_Dots	1	Font width of unselected data, in pixels
0x1B	0x0A:L	Font_Y_Dots	1	Font height of unselected data, in pixels.
0x1C	0x0B	Color	2	Text color selected.
0x1E	0x0C	Reserved		0000

7.3.15 Real Time Curve

Used to plot line graphs.

Address	SP Address	Definition	Length (bytes)	Description
0x00		0x5A20	2	0x5A20
0x02		*SP	2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x04		0x000B	2	0x000B
0x06	0x00:H	Mode	1	0x00: display from right side to left side. Others: display from left side to right side.
0x07	0x00:L	0x00	1	Reserved
0x8	0x01	Xs,Ys Xe,Ye	8	Upper-left coordinates and lower-right coordinates of the curve. Do not display when data is out of range.
0x10	0x05	Y_Central	2	Y center of the curve.
0x12	0x06	VD_Central	2	The corresponding value of 'Y_Central', which is the mean value of the maximum data and minimum data.
0x14	0x07	Color	2	Curve color
0x16	0x08	MUL_Y	2	Vertical Zoom. Range: 0x0000-0x7FFF.
0x18	0x09:H	CHANEL	1	Channel of the curve. Range: 0x00-0x07.
0x19	0x09:L	Dis_HOR	1	Horizontal Increment. Range: 0x01-0xFF.
0x1A	0x0A:H	Pixel_Scale	1	Width of curve, range: 0x00-0x07, in pixel.
0x1B	0x0A:L	Reserved	5	0x00

7.3.16 Basic Graphic

Used to access many graphic manipulation functions, like copy/pasting and shape drawing.

Address	SP Address	Definition	Length (bytes)	Description
0x00		0x5A21	2	0x5A21
0x02		*SP	2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x04		0x0008	2	0x0008
0x06	0x00	*VP	2	Variable Pointer
0x08	0x01	Area	8	Upper-left coordinates and lower-right coordinates of the Graphic. Only valid for Commands 0x0001 -0x0005, 0x0009 -0x000B
0x10	0x05:H	Dashed_Line_En	1	Write 0x00
0x11	0x05:L	Dash_Set	4	Write 0x00
0x15	0x07:L	Pixel_Scale	13	The actual size of pixel. 0x00-0x0F to 1*1-16*16
0x16			12	Write 0x00

The description of the variable data format pointed to by the variable data pointer (variable storage space) is shown in the table.

Address	Definition	Description
VP	CMD	Drawing instructions
VP+1	Data_Pack_Num_Max	Maximum data packet: connection (0x0002), defined as the number of connection lines (the number of vertices-1);
VP+2	DATA_Pack	Data

The description of drawing instruction data packet is shown in the table.

CMD	Function	The description of drawing instruction data packet			
		Address	Length	Define	Description
0x0001	Dot	0x00	2	(x,y)	The coordinate position of dot. The high byte of coordinate x is the judgment condition.
		0x02	1	Color	Dot color
0x0002	Line	0x00	1	Color	Line color
		0x01	2	(x,y)0	The coordinate of the line vertex 0, and the high byte of the x coordinate are the judgment conditions.
		0x03	2	(x,y)1	The coordinate of the line vertex 1, and the high byte of the x coordinate are the judgment conditions.
		0x01+2*n	2	(x,y)n	The coordinate of the line vertex n, and the high byte of the x coordinate are the judgment conditions.
0x0003	Rectangular	0x00	2	(x,y)s	The coordinates of the upper left corner of the rectangle and the high byte of the x coordinate are the judgment conditions.
		0x02	2	(x,y)e	The coordinates of the bottom right corner of the rectangle.

		0x04	1	Color	Rectangular color
0x0004	Rectangle filling	0x00	2	(x,y)s	The coordinates of the upper left corner of the rectangle and the high byte of the x coordinate are the judgment conditions.
		0x02	2	(x,y)e	The coordinates of the bottom right corner of the rectangle.
		0x04	1	Color	Fill color of the rectangle field.
0x0005	Circle	0x00	2	(x,y)s	Center coordinates.
		0x02	1	R	The radius of the circle.
		0x03	1	Color	The arc color.
0x0006	Picture Copy&Paste	0x00	1	Pic_ID	The ID of the page where the cut image area is located; the high byte is the judgment condition
		0x01	2	(x,y)s	The coordinates of the upper left corner of the image cutting area.
		0x03	2	(x,y)e	The coordinates of the lower right corner of the image cutting area.
		0x05	2	(x,y)	The coordinates of the upper left corner of the area in the current page that it is cut and pasted to
0x07**	Icon Display	0x00	2	(x,y)	The coordinate position of display area, and the high byte of the x coordinate is the judgment condition.
		0x02	1	ICON_ID	Icon ID, the location of the icon library is specified by the high byte of instruction. The icon is fixed to not display the background color.
0x0008	Closed Region Filling	0x00	2	(x,y)	The coordinates of point.
		0x02	1	Color	The color of filling the rectangle field.
0x0009	Frequency Spectrum (Vertical line)	0x00	1	Color0	Connect (x0, Y0s) (x0, Y0e) with Color0 color, and the high byte of x0 is the judgment condition.
		0x01	3	X0,Y0s,Y0e	
0x000D	xOR	0x00	2	(x,y)s	The coordinates of the upper left corner of the rectangular area, and the high byte of the x coordinate are the judgment conditions.
		0x02	2	(x,y)e	The coordinates of the lower right corner of the rectangular field.
		0x04	1	Color	The color of xOR of the rectangular field, and 0xFFFF will be inverted.

In the DGUS development software, click the display control_basic graphic display, and then use the mouse to select the display area on the page where you want to display the drawing board, and define the variable address in the right setting menu, and finally use the instruction to realize drawing.

7.3.17 Zone Rolling

Area scrolling is to move the content of the specified area around, the direction of movement can be set.

Address	SP Address	Definition	Length (bytes)	Description
0x00		0x5A24	2	0x5A24
0x02		*SP	2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x04		0x0007	2	0x0007
0x06	0x00	（Xs，Ys）	4	Upper-left coordinates of the area.
0x08	0x02	（Xe，Ye）	4	Lower-right coordinates of the area.
0x0C	0x04	Dis_Move	2	Move space every DGUS cycle. In pixel.
0x10	0x05_H	Mode_Move	2	0x00: move left. 0x01: move right. 0x02: move up. 0x03: move down.
0x12		Reserved	13	0x00

7.3.18 QR Code

Used to display QR Codes generated from the value of the VP.

Address	SP Address	Definition	Length (bytes)	Description
0x00		0x5A25	2	0x5A25
0x02		*SP	2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x04		0x0004	2	0x0004
0x06	0x00	*VP	2	Variable Pointer.
0x08	0x01	(x,y)	4	Upper-left coordinates of the Icons to display.
0x0C	0x03	Unit_Pixels	2	QR Pixel size, in pixels. It's the size of the smallest square on a QR Code. Range: 0x01-0x07.
0x0D	0x05:H	Fix_Mode	1	0x01：display fix at 73*73 pixel. Others: display suit for Unit_Pixels.
0x0E	0X05:L	Reserved	1	00

7.3.19 Brightness

Used to control the brightness of an area of the page.

Address	SP Address	Definition	Length (bytes)	Description
0x00		0x5A26	2	
0x02		*SP	2	Parameter Pointer. 0xFFFF: Disables SP (no run-time modification).
0x04		0x0005	2	
0x06	0x00	*VP	2	Variable Pointer.
0x08	0x01	（Xs，Ys）	4	Upper-left coordinates of the area.
0x0C	0x03	(Xe，Ye)	4	Lower-right coordinates of the area.
0x10-0x1F		Reserved	18	0x00