



DWIN Technologies
Professional, Creditable, Successful

DWIN WIFI module protocol instruction

Version 1.2.6

DWIN Technologies

2021/5/28



1. Overview

In order to make the intelligent LCD screen easily and conveniently connected to the Internet, DWIN set up its own cloud server, and on top of the DGUS system, will connect the Internet interface integration, so that the future development of intelligent DWIN screen with network function will be very simple and convenient.

Currently supported LCD platforms are T5UID1, T5UID2 and T5L_ASIC. Supported products include 86 boxes of products, network screen, etc.

The advantages of DWIN WIFI scheme are mainly reflected in the following four aspects:

(1) Development efficiency

Based on the DGUS architecture, after the single computer function is realized, the single chip microcomputer zero code can realize the network distribution and data uploading. Engineers just need to plan the location of the data. The complex JSON protocol parsing is processed in the cloud, which saves the resources of SCM and improves the development efficiency. Customers can focus on the core functionality of the product.

(2) Stability

The protocol between the main chip (GUI), WIFI module and MQTT server has been connected in the scheme, forming a UI+IOT integrated solution, which reduces the number of links in customer development, reduces the cost of communication, reduces the uncertainty in development, and improves the stability of the whole scheme.

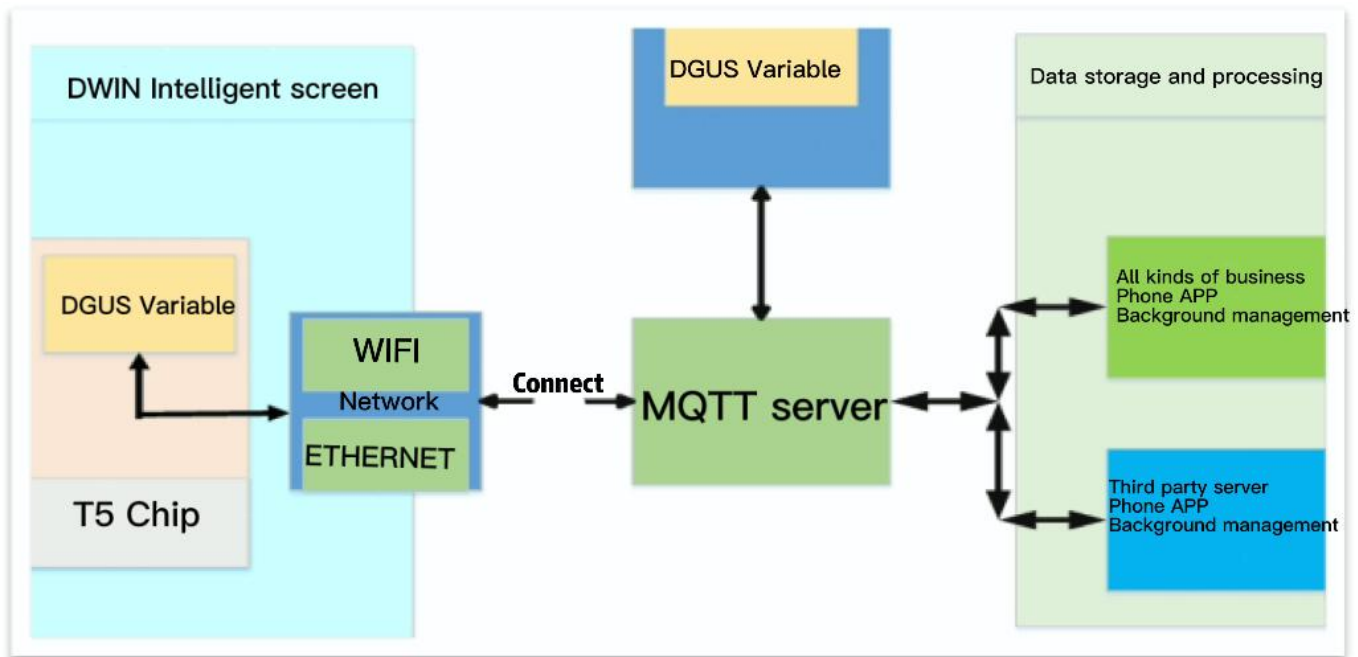
(3) Cost

As a part of Devon's Internet of Things solution, WIFI module faces tens of thousands of customers in different industries, forming a scale advantage. Direct cooperation with WIFIÐERNET chip factory to ensure cost advantage.

(4) A variety of options

The scheme provides two networking methods, WIFI and Ethernet, which can meet different networking needs. Ethernet is the main module. When the Ethernet is connected, the module will automatically close the WIFI connection, and when the Ethernet is off, it will start WIFI again. To provide customers with the best quality network experience.

1.1 System block diagram



- (1) Through simple configuration, the smart screen equipped with DWIN WIFI module will connect the MQTT server through the "connection" channel.
- (2) WIFI module will transfer the DGUS register variable of DWIN smart screen up and down with the corresponding DGUS variable on the server.
- (3) Mobile terminal and background management conduct data communication and interaction between servers through MQTT server. The mobile terminal mainly realizes remote UI function and variable modification function. Backstage management can achieve variable modification control and file upgrade functions.
- (4) The third party server can be the client's server, by providing interface can operate the DGUS variable, file upgrade.

1.2 Access

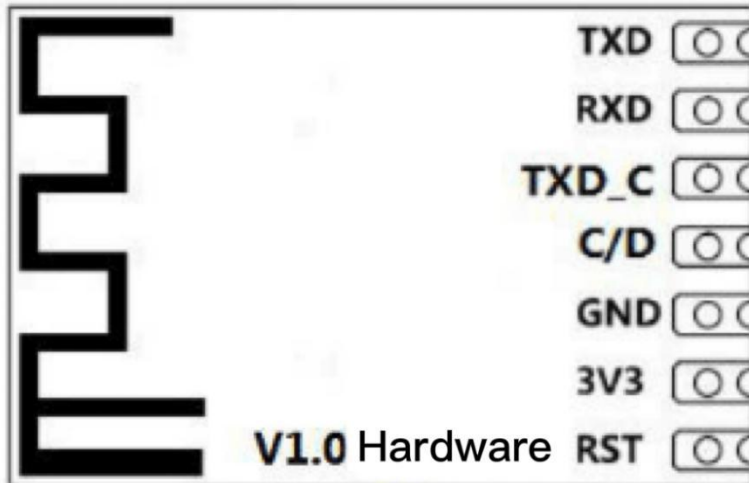
DWIN WIFI module and MQTT server and data relay server are required to use DWIN. The server that the business handles data storage and processing can use either Devon's or the client's server.

1.3 Characteristic Parameter

Features of WIFI module include:

- (1) Main chip ESP8266+4MB Flash
- (2) Support 802.11 (2.4GHz), frequency range 2.4g ~ 2.5g (2400M ~ 2483.5m)
- (3) The type of antenna includes PCB on-board antenna
- (4) The maximum transmission power is 14dBm
- (5) Serial port transmission rate 921600BPS (device side transmission data rate)
- (6) Ethernet card interface (please refer to the documentation of ETH - ESP32 for the hardware interface diagram of the Ethernet card)

1.4. Hardware interface



The definition of the V1.0 hardware package is shown in the figure above, with the foot pitch 2.0mm.

WIFI module should be equipped with VCC (3.3V), GND, TXD, RXD, C/D, TXD_C, /RST 7 pins.

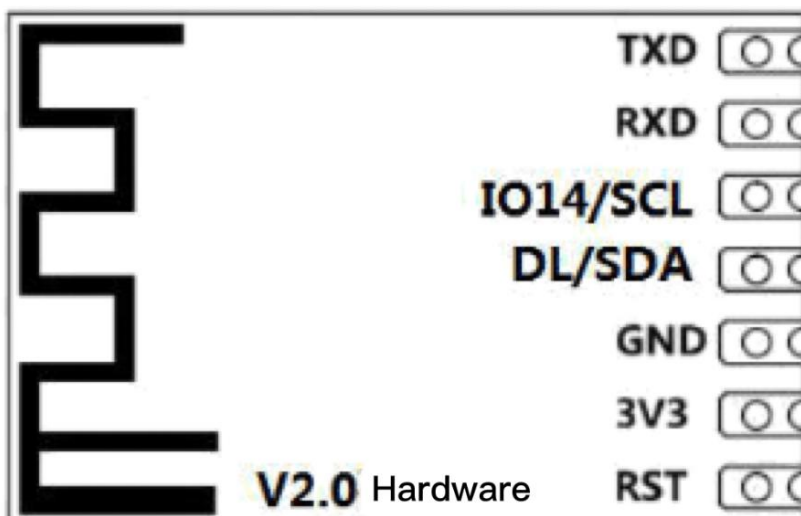
TXD, RXD: Data channel, baud rate 921600.

TXD_C: Nothing to do with the business, internal information printing output pin, suspended processing.

C/D: WIFI module download enable, internal pull up resistance 10K. When the pin is low when the module is powered on, it means that the boot mode is entered. Normal use of hanging can be.

3V3, GND: Need to ensure peak current 300mA.

/RST: reset pin, need to connect the reset chip for reliable reset. Low level reset WIFI module.



The definition of the V2.0 hardware package is shown in the figure above. Pin spacing is 2.0mm.

Differences:

IO14/SCL: SCL of I2C interface, output of WIFI module.



DL/SDA: WIFI module download enabled, internal pull-up resistance 10K. When the pin is low when the module is powered on, it means that the boot mode is entered. Normal use of hanging can be.
I2C interface SDA, if enable I2C interface, need to turn off UART1 TX function.

1.4.1. T5UID1_86 platform and WIFI module Interface

The T5UID1_86A platform is an application of the standard T5UID1 platform on 86 boxes, with additional definition of IO and serial port to connect to WIFI. I ended up using only UART4 and nothing else.

T5_IO	WIFI_IO	Instructions
IO2	C/D	Connect the resistor, hardware configuration download enable, actually do not need, can be suspended
IO3		Power on button detection
UART4_TX	--	A serial port
UART4_RX		A serial port
IO1	RXD	Debugging information output, do not need to pay attention to, the actual use of less than, can be suspended
UART6_RX	TXD	

1.4.2. T5L hardware platform interface

T5L_IO	WIFI_IO	instructions
IO18	C/D	Connect resistor, hardware configuration download enable
IO19	TXD_C	Debug information output, do not need to ignore
UART1_TX	RX	Serial data
UART1_RX	TX	Serial data

2. Development steps

2.1. Determine platform

Platform	Instruction	Product example
----------	-------------	-----------------



T5UID1 T5UID2	1. It is necessary to install 22.bin file and load it 22. The bin file. 2. Need to implement instruction 8283 in OS. Can be To use the given routine code.	TR028C11W04、TC035C21W04 TR028C12W00、TC035C22W00
T5L-ASIC	Just configure the 22.bin file and load the 22.bin file	

2.2. Stand-alone equipment program development

As the original DGUS stand-alone development, no special treatment.

2.3. Access cloud operate

Compile 22.bin files to meet network access requirements, including product ID, display network connection status icon, display version number and MAC address, etc.

2.4. Mobile terminal and cloud background development

Log on to <http://merchant.dwinhmi.com.cn/> for background management and development of the App.



3.DGUS platform interface define

3.1. DGUS system read-write interface

DGUS systems generally interact with external systems through in-memory data. The memory variable space is 0x0000-0xFFFF, where 0x0000-0x0FFF is DGUS

System area, 0x400-0x4FF in the system area is the network communication interface. The 0x1000-0xFFFF region is the user memory region.

The bus interface of external devices reading and writing DGUS system is realized by UART, mainly 82H,83H instructions.

It should be noted that the length of 82H/83H instruction is only 1 byte, that is, the maximum 256 bytes, excluding the frame header and the frame tail and the CRC, the maximum net charge is about 240 bytes. Therefore, if there is a large amount of data to interact with the DGUS system, it needs to be broken down into small packets for interaction. For example, the cloud 4K data is sent to the WIFI module, which needs to be broken into small packages and then written into the DGUS system variable in turn.

3.2. DGUS system network interface definite

In the DGSU system, the address of the network part is defined in the region 0x400-0x4FF, as follows:

Definition	Address	Length (word)	Recommended value (HEX)	Instructions
Network interface switch	0x400	1	5AA5	0x5AA5 indicates that the network communication interface is enabled.
RAM ALARM	0x401	3	0000 0000 0000	D5-D4:0x5AA5 indicates that the specified RMA spatial data is immediately started once and uploaded to the server. D3-D2: RMA variable memory word address to be uploaded. D1-D0: The word length of the RMA variable memory to be uploaded. (Currently up to 4KByte)
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.
	0x411	2	00000001	4Bytes.Device manufacturer ID, DWIN factory assigned, 0xFFFF: 0000 - 0xFFFF: FFFF segment reserved.
	0x413	1	0001	2Bytes.The classification of each manufacturer's equipment is according to the DWIN classification standard equipment classification.
	0x414	2	User-defined	4Bytes.The individual device number of each type of device.
	0x416	26	DGUS2_ T5UID1	Device description text, up to 52Bytes.



Device description (RMA mapping)	0x430	1	012C	RMA automatically refreshes the server interval 0x0000-0xFFFF in 0.1 seconds. 0x0000 indicates that no automatic refresh is required.
	0x431	1	1000	RMA (mapped to the server's variable memory) read space start address, word address, out of bounds can not be read. (Since we are running out of space, now we have no limit on reading space)
	0x432	1	0004	RMA read space size in 128Words with a maximum size of 64KWords and 0x0000 indicates forbidden reading.
	0x433	1	2000	RMA write space start address, can overlap with the read space, can not write out of bounds.
	0x434	1	0004	RMA write space size in 128Words, Max 64KWords, 0x0000 indicates write forbidden.
	0x435	3	All are 00	Reserve
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.
	0x439	1	0064	Remote upgrade packet timeout timer configuration in 0.1 second.
	0x43A	4	5A 0001 0000 0000F0	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	<p>Remote upgrade Buffer interface definition:</p> <p>D3:0x5A indicates that the remote upgrade Buffer is valid.</p> <p>D2:Upgrade mode 0x00= the communication side is responsible for verifying the data CRC, and the error frame informs the host to resend.</p> <p>D1: The number of buffers available for remote upgrades, 0x01-0x10, up to 16.</p> <p>D0: The starting address of Buffer0 is high byte (word address) and low address is 8bit 0x00.</p>
				Each Buffer is fixed in 2304 words (0x900) space, which is arranged in the back row.
				<p>A single Buffer definition (the first 512 bytes is the control interface, the next 4KB is the data):</p> <p>D0:0x5A means to start the remote upgrade of this buffer once, and clear it after CPU processing.</p> <p>D1: Remote upgrade space selection, 0x00-0x03, one of 4 remote upgrade spaces.</p> <p>D2-D5: Write the destination address of the remote upgrade space. The lower 12 bits are 0 (aligned to 4KB).</p> <p>D6-D7:Data byte length, 0x0001-0x0FFF.</p> <p>D8-D9: Data CRC checksum.</p> <p>D10-D511: Reserved.</p> <p>D512: Data starts, up to 4096 bytes.</p>
0x44C	4	All are 00	Reserve	
Device description QR code	0x450	48	All are 00	QR_Code device QR code.



Communication device description	0x480	16	User-defined	<p>D31: 0x5A indicates that the communication device description data is valid.</p> <p>D30: The encoding method and length of the device description text. .7-.6 Edit: Code mode 0x00=UNICODE 0x01=GBK, GBK is recommended. .5-.0 Description: The length of the text is 0x00-0x14.</p> <p>D29: Communication device category 0x01=WIFI .</p> <p>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 files, 0x00 means the worst, 0x07 means the best.</p> <p>D27-D20: 8Bytes communication device MAC address, low (D20) alignment.</p> <p>D19-D0: Description of the communication device in text format, up to 20Bytes. Use hexadecimal to describe the information as the corresponding ASCII character.</p>
Communication device	0x490	8	User-defined	Communication device customization, such as WIFI user name, password.

4. WIFI Memory Interface Define

The address 0x400-0x47F is the DGUS system communication interface, and the address 0x480-0x4FF is the function and status interface of the specific communication module or device.

Different network modules and devices have different definitions. The following is the definition of WiFi module.

Definition	Address	Length (word)	Recommended value (HEX)	Instructions
Communication device description	0x480	16	User-defined	<p>D31: 0x5A indicates that the communication device description data is valid.</p> <p>D30: The encoding method and length of the device description text. .7-.6 Edit: Code mode 0x00=UNICODE 0x01=GBK, GBK is recommended. .5-.0 Description: The length of the text is 0x00-0x14.</p> <p>D29: Communication device category 0x01=WIFI.</p> <p>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 files, 0x00 means the worst, 0x07 means the best.</p> <p>D27-D20: 8Bytes communication device MAC address, low (D20) alignment.</p> <p>D19-D0: Description of the communication device in text format, up to 20Bytes. Use hexadecimal to describe the information as the corresponding ASCII character.</p>
WIFI hard_set	0x490	8	User-defined	Restarting WiFi module is valid, used during restart initialization.



				<p>D15-D14: The baud rate setting is reserved. Currently fixed at 921600bps, it cannot be set.</p> <p>D13-D9: Reserved.</p> <p>D8: WIFI module switching Debug firmware 5A means valid.</p> <p>D7: D0: WIFI module type identification information Default "DWD100".</p>
WIFI cfg_set	0x498	4		<p>D7: Write 5A to start the WIFI distribution network. The distribution network is completed, and the value clears 0.</p> <p>D6: 0x5A indicates the start network name, password way to connect to the router (network name and password are saved in 0x4b0).</p> <p>D6: 0xA5 means that SCANF scan function is enabled and the result is placed at 0x4D0.</p> <p>D7-D6:As long as the start once, after completion, D7-D6 will be cleared at the same time, the two Internet access mode is not start at the same time, if all is 5A, then with one key network priority.</p> <p>D5: Network Time Synchronization Enable 5A means enabling DUSII with RTC(0x9C); 5B stands for Enable RTC Library RTC (0 xf430).</p> <p>D4-D2: D4:0x5A indicates that UDP port Settings are valid.</p> <p>D3-D2: UDP port number.</p> <p>[D1] WIFI module automatically upgrades 5A to enable.</p> <p>[D0] Soft Reset 5A Start Reset Reset Complete Clear 0.</p>
Reserve	0x49C	4		<p>D7:5Ameans start electronic table broadcasting, every 5 seconds.</p> <p>D5:D4means the electronic table card ID.</p>



				D3:5A means open weather forecast.
Network status	0x4A0	8		<p>[D15:D14](0x4A0) is reserved, and the WIFI version number is displayed in ASC code at 0x487.</p> <p>[D13:D12](0x4A1) Distribution network status feedback. 00 :No network distribution . 01 :Start network distribution . 02 :Distribution network in distribution network . 03 :Distribution network success . 0F :Distribution network failure.</p> <p>[D11:D10](0x4A2) Network connection status. 00:Did not get username and password. 01: WIFI routing. 02: WIFI module is in upgrade. 03: Connect to server. 04: Log in server. 05: Connected to cloud.</p> <p>[D9:D8](0x4A3) Network strength (same as 481)</p> <p>[D7:D6] WIFI connection router error. 01: The router does not exist. 02: Password error. 03: Connection timeout. D5: D2 retention.</p> <p>[D1:D0](0x4A7) keep the counter, each cycle (< 1s) will increase by 1. DGUS finds that the value has not changed for a long time, so it can be considered that the WIFI module has crashed. In order to proceed to the next processing.</p>
Program running state	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 parameters	0x4B0 0x4D0	32 48		0x4B0: SSID, the end must end 0xFF. 0x4C0: SN, the end must end at 0xFF. 0x4D0:Save the name of the SSID separated by \n.



Registers such as 400H, 490H, etc. that DGUS issues commands to WIFI will be actively read by WIFI module every 0.1s-0.2s.

The status of WIFI module is such as 480H and 4A0H registers. WIFI module will write to the DGUS register every 3-5 seconds.

Operating Instructions:

- (1) Edit the 22. Bin file template to modify the values you need.
- (2) Network name and password setting of router: register 498H write 005AH enable, 4B0H set SSID and password; You can also unset, in the application to start a key network.
- (3) T5UID1 platform needs to add instruction code of 8283 to the OS, and it needs to use interrupt to detect WIFI serial port. For details, refer to the routine. The T5L platform does not require any processing.

UI operation instructions :(need to make ICONS for status indication)

I. One-click network configuration page of mobile phone:

(1) Touch control: start the one-key network distribution button (write 0x5A00 according to the address of build return 0x498).

(2) Display:

- One-key distribution network status icon (0x4A1 address: 00 no distribution network 01 start distribution network 02 distribution network in 03 distribution network success 0F distribution network failure);
- WIFI networking status icon (0x4A2 address: 00 not obtained user name and password 01: WIFI router connected successfully 02: WIFI module in upgrade 03: connected to server 04: logged in server 05: connected to cloud).
- WIFI module MAC address (0x482-484 address: hexadecimal display, 12-byte ASCII length display).
- WIFI module version number: (0x487 address: ASC code string display, length of 18 bytes).

II. manually enter the network name and password

(1) Touch control:

- Network name text input (address 0x4B0), network password text input (address 0x4C0).
- Start the network connection key (write 0x005A as build return address 0x498).

(2) Display:

- WIFI networking status icon (0x4A2 address: 00 not obtained user name and password 01: WIFI router connected successfully 02: WIFI module in upgrade 03: connected to server 04: logged in server 05: connected to cloud).
- WIFI module MAC address (0x482-484 address: hexadecimal display, 12-byte ASCII length display).
- WIFI module version number: (0x487 address: ASC code string display, length of 18 bytes).

After that, the WIFI module will automatically surf the Internet and start data synchronization.

Work process of WIFI module serial port side:

(1) Read configuration information: After the WIFI module is powered on, read 0x400-4FF configuration information, and enter the next state after reading it.

(2) Status between WIFI module and DGUS:

a. Lowest priority 0: every 0.1-0.2 seconds to read the DGUS variable (83 instructions), 400H, 490H and so on involved in the DGUS instruction. The necessary state of the WIFI module is written into DGUS by 82 instruction every 3-5 seconds.

b. Second low priority 1: whether the synchronization time with the server interface is on time, and then periodically upload the specified big data.

c. Second low priority 2: check whether the RMA alarm interface is enabled, and then carry out the specified big



data trigger upload.

d.Second high priority 3: distribution network, network timing, network status change trigger uploaded to DGUS, such as in the distribution network, network login, once the status change, will be immediately written to DGUS.

e.High priority 4: file upgrade. After receiving the first package of file upgrade package, the general task scheduling will be suspended to ensure that resources are sent to DGUS through MQTT and serial ports, and other 82H/83H instructions are not allowed to be generated during subcontracting to DGUS.

Work process of MQTT side of WIFI module:

(1) Power on initialization, waiting for 512Bytes configuration word.

(2) Wait to connect to the router.

(3) Decide whether to upgrade the WIFI module. If the WIFI module needs to be upgraded, the WIFI module will be upgraded.

(4) Access the REG registration API through the HTTP protocol and get the user name and password to login the MQTT server.

(5) Re-establish the connection and log in to the MQTT server with the user name and password.

(6) Subscribe to the topic channel and send online notifications.

(7) The establishment of data channel is completed, and 82H and 85H read-write instructions can be carried out with the server, register synchronization can be carried out, and file Buffer can be carried out with 92H and 93H instructions.

5. WIFI And Server Communicate

The maximum buffer left for one frame in WIFI module is 5K(5120), that is, the maximum packet length between the WIFI module and the server is 5K.

5.1 Overview

(1)Why is the synchronization of the current data not answered on the command?

- a. The data is centered on the device, and the data of the device is based on it. The equipment itself has an automatic timing reporting mechanism. The data on the mobile phone is only used for UI display, and cannot reflect the real device status.
- b. Data is transmitted in one packet. This means that when the device sends data to the server, each frame is trying to transmit as much data as possible.
- c. If the mobile phone controls a certain operation and the device does not receive it, the mobile phone can first lock the display data into the request state, and then display the real value to cover the display value after the timeout (for example, 5 seconds). The data of the mobile terminal is considered unreliable, and the real data will always be reported and synchronized by the device.
- d. No response mechanism, will simplify the synchronization mechanism of each process, will simplify the complex problems, and finally (automatic report if it is 30s) will be the state of the device to display all the display terminal.

(2)Without a reply, how can the convergence of data be achieved?

The automatic convergence of data is realized mainly through the active and regular reporting of equipment.

- a. The server writes data to the device, there will be no reply.
 - b. However, after the server writes data to the device, it may also receive new data immediately. It is the automatic reporting function triggered by the device after it finds the variable change and sends out 0x85 instruction, which is the variable upload of the whole block.
 - c. If the data sent by the server to the device is not updated, it will not be automatically reported. The instruction was not answered. The device does not notice the variable change and does not immediately trigger the report, so the server does not receive the 0x85 instruction.
- If the instruction differentiates between write and read, it is to pass the register.

(3)Currently, instructions that WIFI module and server side can support are as follows:

- a. The server writes data to the device 0x82. If the WIFI-ETH module receives the server's instruction of 82, it writes data to the device. This instruction has no answer at present (if it does, it should be 0x83 according to the protocol, and the answer is "OK" (4F 4B)).
- b. The server reads data request 0x84 to the device, and the WIFI-ETH module receives instruction type 0x84 from the server. After the WIFI-ETH module receives this instruction, it will reply to the server 0x85 reply instruction and take the data with it. After the server receives the 85 instruction, it needs to write the data to the mapped memory block of the server.
- c. The device writes data to the server
- d. The device reads data to the server

In fact, in the end, data only needs to be transferred from device to server, or from server to device. The function



described by 3 is replaced by instruction 0x85. The function described by 4 is replaced by the 0x82 instruction.

In the end, only 0x82, 0x84 and 0x85 instructions are retained. The 0x82 instruction realizes the data transfer from the server to the device, the 0x85 instruction realizes the data transfer from the device to the server, and the 0x84 instruction realizes the server actively querying the registers of the device.

5.2. Frame structure

5.2.1. RAM write

RAM write operation: mainly uses RAM to write, especially when the server writes to device RAM

Frame header(2B)	Remaining frame length(2B)	Frame type	Address(2B)	Word data flow
AA55	MAX 5117	0x82:Write	XXXX	

RAM write reply (currently not implemented) :

Frame header(2B)	Remaining frame length(2B)	Frame type	Address(2B)	Word data flow
AA55	MAX 5117	0x83:Write	XXXX	4F4B

5.2.2. RAM read operation

The use of active query means that the server queries the device.

The sender

Frame header(2B)	Remaining frame length(2B)	Frame type	Address(2B)	Word data flow
AA55	5	0x82:Read	XXXX	

The responder (the responder can be used as an instruction for active reporting by the device)

Frame header(2B)	Remaining frame length(2B)	Frame type	Address(2B)	Word data flow
AA55	MAX 5117	0x82:Read response	XXXX	



5.2.3. Upgrade BUF frames

Server to device:

Frame header(2B)	Remaining frame length(2B)	Frame type	BID(2B)	BUF structure
AA55	MAX 5117	0x92:Upgrade BUF read	BUF ID number	<p>D0:0x5A means to initiate a remote upgrade of this buffer, and clear it to zero after CPU processing.</p> <p>D1: Remote upgrade space selection, 0x00-0x03, one of four remote upgrade Spaces.</p> <p>D2-d5: Write to the target address of the remote upgrade space, the low 12bit is 0 (aligned to 4KB).</p> <p>D6-D7: Data byte length, 0x0001-0x1000.</p> <p>D8-D9: Data CRC checksum.</p> <p>D10-D511: Reserved.</p> <p>D512: Data start, up to 4096 bytes.</p>

Response:

Frame header(2B)	Remaining frame length(2B)	Frame type	BID(2B)
AA55	MAX 5117	0x92:Upgrade BUF read	BUF ID number

5.3. Initialize Process

Operation process of WIFI and server :(mainly communicate through DGUS business topic)

- (1) After WIFI is powered on, use 0x411-0x413 as PID,MAC address as DID to register with the registration server, get the login information, and then log into the server.
- (2) Subscribe to DGSU business topics.
- (3)Write 0x400-0x4FF data in DGUS to the server via the DGUS business topic.
- (4)The initialization is completed. After that, there will be all data frame structures in the theme.

5.4. Data Communicate

RMA operation, frame type 0x82 or 0x83.

The communication frame header with the server is 0xAA 55 data stream.

After receiving the data stream from the server, the WIFI-ETH module is converted into a DGUS frame structure and reads and writes to DGUS.

5.5. BUF Upgrade

The server converts the corresponding address (4G space) according to the file rules of DGUS. Then compose



the BUF application frame.

The BUF frame as a whole is put into the communication frame as a net charge.

After receiving the BUF, WIFI will reply BUFID.

5.6 UDP485 Bus Use

(1) The upload address 0x402 is set to the upload address such as 0X3800.

(2) Set the length of data to be uploaded by 0x403, for example: 0x0010.

(3) Set 0x401 to 0x7AA7. After sending, it will automatically reset to zero.

Data will be broadcast to all devices in the LAN through UDP, other devices will get the data, and write the data to the 3800 address of the local unit to cover the original data, the specific use of the data, the device decided.

5.6.1. UDP bus frame structure write

UDP write operation: mainly use RAM write, write to the device RAM.

Frame header(2B)	Remaining frame length(2B)	Frame type	Address(2B)	Word data flow
AA55	MAX 1024	0x72:Write	XXXX	

In order to recognize multiple destination addresses on a given host and allow multiple applications to work on the same host and send and receive data packets independently, the user datagram protocol UDP is designed.

UDP uses the underlying Internet protocol to send messages and provides the same unreliable connectionless packet transport service as IP. It does not provide message arrival confirmation, ordering, or flow control.

UDP Helper can relay and forward broadcast messages on the specified UDP port. It converts the broadcast message on the specified UDP port into a single broadcast message and sends it to the specified server, acting as a relay.

5.6.2 Message format

In the UDP protocol hierarchy model, UDP is above the IP layer. The application accesses the UDP layer and then uses the IP layer to transmit datagrams. The data part of the IP packet is the UDP datagram. The IP layer header specifies the source and destination host addresses, while the UDP layer header specifies the source and destination ports on the host. A segment transmitted by UDP consists of an 8-byte header and a payload field.

The UDP header consists of four fields, each of which takes two bytes, including the source port number, the destination port number, the packet length, and the checksum.

Introduction to LAN device protocols

Hardware equipment according to the actual business needs, the development of LAN UDP protocol. UDP protocol format is divided into reading and writing two ways, the upper computer (Android, iOS or PC) sent, the MCU has the corresponding feedback data.

The upper computer sends the write instruction (0x72), and the device returns the instruction (0x73), which is the feedback of the written data content. The content is the same.

The upper computer sends the read instruction (0x74), and the device returns the instruction (0x75), which is the content of the read data, and the content is the value of the read memory address.

5.6.3 UDP read-write protocol format

Own MAC address (12 bytes) + instruction header AA55 (2 bytes) + subsequent content length (2 bytes) + command word (1 byte) + read/write address (2 bytes) + content (no read instruction).

12 bytes MAC address, which is the ASCII code corresponding to the MAC address content.

The command word is the write instruction 0x72-- >; 0 x74; Read instructions 0 x73 - & gt; 0x75.

5.6.4 Equipment broadcast

The device automatically reports the discovery instruction, and the default sending port is 5556, which can be changed by the user.

Configuring the UDP port for the +1 that configures the UDP port is the receive port

Equipment since found (radio) 01 deviceNum, clientIP, clientPort, cardId

Own MAC address (12 bytes) + instruction head AA55 (2 bytes) + follow-up length (2 bytes) + command word 0 x01 (1 byte) + character content deviceNum, clientIP, clientPort, cardId

5.6.5 UDP communication process

The device sends an automatic device discovery command every 5 seconds. The host computer receives at port 5556. If it fails to receive more than 5 times, the device is offline.

The device sends 72,73 read and write commands, and the corresponding data is fed back.

5.7 Weather Forecast Protocol Define

Weather data (network module writes VP variable, weather starting position is 0x0780)

5.7.1 The network module actively obtains weather information

Weather detail data (red part is reported, green part is the weather data sent by the server) :

		0	1	2	3	4	5	6	7	8	9 ~ F
	0x 07 80	Flag bit, time zone	Language , coding	Zip code				Internati onal code			
Basic information	0x 07 90	Flag bit, flag bit	Time zones, year	Month, day	Hour, Minute	Language, coding	City				
Day weather	0x 07 A0	Flag bit, year	Month, day	weather	Temperature	Body feeling temperature	Relative humidity	Wind direction	Air quality	PM 2.5	
+1 day weather	0x 07 B0	Flag bit, year	Month, day	weather	Max temperature	Min temperature	Relative humidity	Wind power	Air quality	PM 2.5	
+2 day weather	0x 07 C0	Flag bit, year									
+3 day weather	0x 07 D0	Flag bit, year									
+4 day weather	0x 07 E0	Flag bit, year									
+5 day weather	0x 07 F0	Flag bit, year									

Note: Replace all retained data with 00.

Time zone: Signed char, range [-12 -- +12], default +8 is China time zone.

Language: unsigned char, range [0-255], default 1 language code 1, Chinese 2, English, 3, Spanish 4, French 5, Arabic 6.

Encode: Unsigned char, range [0-255], default 3 encode code 1, ASCII, 2, UTF8, 3, GBK.

Zip code: ASCII string.

Estimated area code: unsigned double byte int.

Flag bits: all use 5A, other values are invalid.












Year, month, day: The time in the corresponding time zone, unsigned CHAR, 0C0C1E0C04 stands for 12:04 December 30, 2012.

City name: Specifies the language and encoding of the string to a maximum of 22 bytes.

Weather: 2 byte unsigned integer.

Weather phenomenon code comparison table:

Code	Chinese	English	Icon
0	晴（国内城市白天晴）	Sunny(Domestic cities,day)	
1	晴（国内城市夜晚晴）	Sunny(Domestic cities,night)	
2	晴（国外城市白天晴）	Sunny(Foreign city,day)	
3	晴（国外城市夜晚晴）	Sunny(Foreign city,night)	
4	多云	Cloudy	
5	晴间多云	Partly Cloudy	
6	晴间多云	Partly Cloudy	
7	大部多云	Mostly Cloudy	
8	大部多云	Mostly Cloudy	



Code	Chinese	English	Icon
9	阴	Overcast	
10	阵雨	Shower	
11	雷阵雨	Thundershower	
12	雷阵雨伴有冰雹	Thundershower with Hail	
13	小雨	Light Rain	
14	中雨	Moderate Rain	
15	大雨	Heavy Rain	
16	暴雨	Storm	
17	大暴雨	Heavy Storm	
18	特大暴雨	Severe Storm	
19	冻雨	Ice Rain	



Code	Chinese	English	Icon
20	雨夹雪	Sleet	
21	阵雪	Snow Flurry	
22	小雪	Light Snow	
23	中雪	Moderate Snow	
24	大雪	Heavy Snow	
25	暴雪	Snowstorm	
26	浮尘	Dust	
27	扬沙	Sand	
28	沙尘暴	Duststorm	
29	强沙尘暴	Sandstorm	
30	雾	Foggy	



Code	Chinese	English	Icon
31	霾	Haze	
32	风	Windy	
33	大风	Blustery	
34	飓风	Hurricane	
35	热带风暴	Tropical Storm	
36	龙卷风	Tornado	
37	冷	Cold	
38	热	Hot	
99	未知	Unknown	N/A

Temperature: 2 bytes with integer degrees Celsius

Wind direction: 1 E,2 ESE,3 SE,4 SSE,5 S,6 SSW,7 SW,8 WSW,9 W,10 WNW,11 NW,12 NNW,13 N, 14 NNE, 15 NE 16 ENE.

Wind power: Unsigned char.

Air quality index: 2 bytes without conforming integer.

PM25:2 bytes have no conforming integer.



Device actively fetches weather (up to 4 active fetches per day) : 0x0401 write 2AA2.

5.7.2. Obtain payment QR code

QR code location 0x0760.Device access payment QR code: 0x0401 write 2AA3.

5.7.3. Get QR weather configuration code address

QR code location 0x0760.Device weather configuration QR code: 0x0401 write 2AA4.

5.8 TCP data passthrough communication protocol

TCP data passthrough communication (network module writes VP variable, TCP data passthrough communication starts at 0x0730).

5.8.1 0x730

0x730 address is the DGUS system communication interface.

Define	Address	Bytes	Instruction	Recommended value
TCP through communication	0x730	16	D31-D30: 0x5AA5 means open TCP service connection, connect once.	User_defined
			D28: Connection status: 0x00 Not connected in connection at 0x02; 0x03 connection successful 0x04 connection disconnected.	
			D27-D24: Save TCP's IP.	
			D23-D22: The port where the TCP is stored.	
			D21-D20: Specifies the address where TCP writes data. If this address is not filled, the default is 0xD400.	
			D19-D18:0x5AA5 indicates successful TCP write.	
			D17-D16: Save the length of TCP written data.	
			D15 - D0: reserve.	

5.8.2 TCP Passthrough used

(1) need to set the TCP IP address 0x732-0x733 for example: 0xC0A80AD1 decimal: 192.168.10.209.

(2) need to set TCP PORT address 0x734 for example: 0x1A0A decimal: 6666.

(3) It is necessary to set the address 0x735:0xD400 to receive the data sent by the TCP service. This address can be configured in the 22.bin file.

(4) To start TCP connection, set 0x730 to 0x5AA5, and the device will automatically connect to the specified service address.



(5) Receive TCP service data and save the data to the address configured in Step 3 (the maximum length of data received at a single time is 4KB).

(6) After receiving the TCP service data, write the length of the received data into 0x737, and write the data success sign at the same time, 0x736 is 0x5AA5.

(7) Connect the device of T5L to process the above data, and return the data if necessary after processing, perform the following operations:

- 0x402 is set to the address of data to be reported, such as 0xD400;
- Set 0x403 to the length of data to be reported, for example: 0x0010;
- Set 0x401 to 0x7AAC to send; After sending, the value of 0x401 is automatically reset;
- The TCP server receives the data sent from the device connected to T5L, and the data need not be changed in the whole process.

Note: During the process of data sending and receiving, please construct the data in Modbus communication format by yourself.

5.8.3 TCP passthrough process

(1) Write the formatted data to the sending area (address 0x402 stored value).

(2) After setting the sending flag bit (address 0x401, value 0x7AAC), the data in the sending area will be automatically read and sent through TCP.

(3) The receiving area (the value stored in address 0x402) will set the data length (address 0x737) and the data readable flag (address 0x736, value 0x5AA5) to the specified location when receiving complete data.

(4) After receiving the data from TCP, write the data to the receiving area (address 0x735 stored value), and set the data length (address 0x737) and the data readable flag (address 0x736, value 0x5AA5).

(5) The upper application scans the data readable flag (address 0x736) to determine whether new data has been received.

(6) When the upper application reads away the data, the upper application needs to clear the data readable flag.