Bootl

This paper introduces a general Bootloader implementation method of the TLE989X series. Bootloader can remotely upgrade the product firmware (program) through any communication port, which solves the problem that the MCU needs to dismantle the device or professional personnel, special tools, and on-site operation. The Bootloader provided this time integrates part of UDS (14229, 15765 specifications) services with the TSMaster host computer to download APP programs through the CANFD interface.

The content of this article includes: how to download the APP program through Bootload with the host computer. TSMaster was used as the host computer of Bootloader, and UDS protocol was used to transfer APP.HEX file to MCU (the underlying communication protocol was CANFD). The Bootloader program parses the data packets transmitted from the host computer, combines the APP code packets, and writes them into the target Flash space in order. The Bootloader program will automatically quit running after the APP program in the target Flash area is started successfully. The APP starts working. The flow chart of the process is as follows Figure 1:

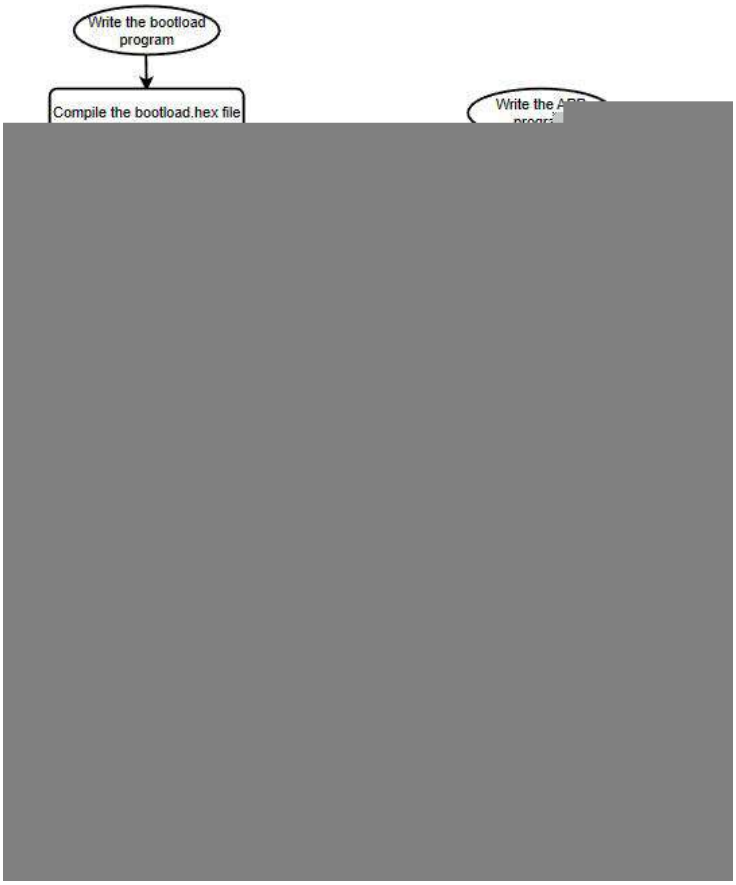
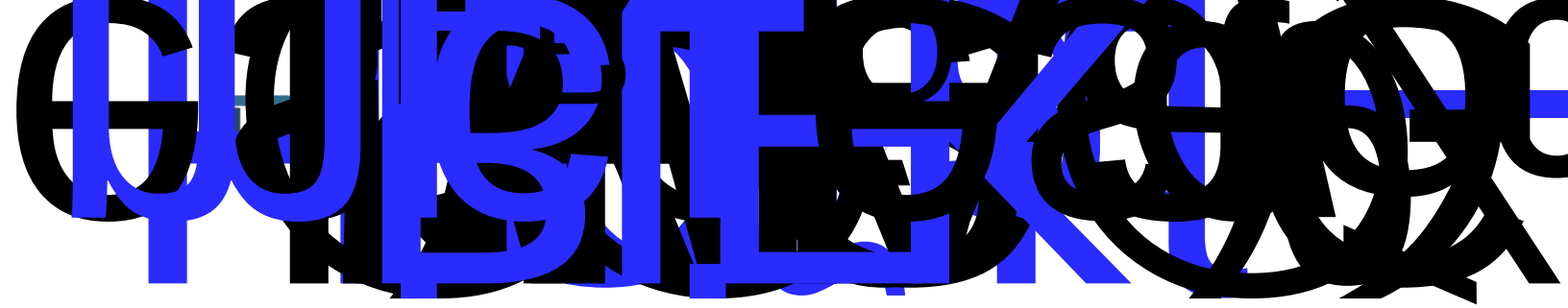


Figure 1



UDS (Unified Diagnostic Services) diagnostic protocol is a communication protocol in the environment of an automotive electronic ECU, which is specified in ISO14229. At present,

37	Request Transfer Exit
38	equest

The storage space of Infineon TLE989X series microcontroller is designed to be arranged according to linear address. The benefit of this is that RAM, ROM, Flash and register addressing is more convenient and intuitive. The physical address range for IROM1 is from 0x11000000 to 0x11006000, and for IROM2 is from 0x12002000 to 0x12040000,IRAM1 has the physical address range 0x18000000 to 0x18002000, and IRAM2 has the physical address range 0x18002000 to 0x18007C00. The specific situation is shown in Figure 3 below.

Figure 3

The Bootloader of Infineon TLE989X series microcontroller is placed in the low address space of Flash starting from 0x11000000 address. After MCU is powered on, it will automatically start from Bootloader to check whether APP program code exists, and then jump to APP for execution. If it does not exist, it enters bootload and waits for TSMster to initiate UDS service request. The APP program is placed in c

Figure 4

Figure 5

The interrupt vector table is essential at program startup and during the execution of interrupt service functions, which is equivalent to the "directory" of the program. In particular, 0x00000100-0x00000103 holds the stack space MSP - the value of the stack top pointer. Also, 0x0000 0104-0x0000 0107 holds the pointer to the Reset_Handler function. The following screenshot shows the contents of a partial Vector.c file

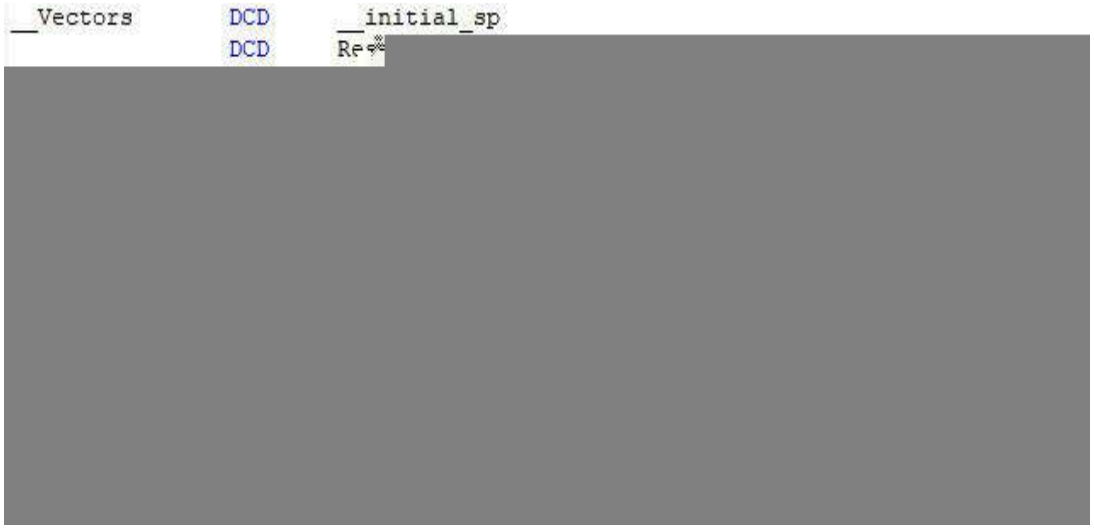


Figure 6

When the Bootloader boots the APP, it needs to use the interrupt vector table of the APP. After the APP starts, it needs to switch the Bootloader's vector table to the APP's own vector table. This process is called vector table remapping. In the conventional program, after the interrupt is generated, the hardware automatically addresses the corresponding interrupt service function entry and jumps to the interrupt function execution after the stack is pushed in the interrupt field. The jump code is as follows Figure 7:

Figure 7

The microcontroller is powered on and started, the CAN is initialized, and the delay is 50 milliseconds to wait for whether to enter the Bootload mode. If the Bootload mode is not entered, check whether there is an APP program at the address 0x12002000. If the APP exists and there is no UDS service request, the MCU restarts and jumps to the address 0x12002000 to execute the APP program. If the APP program is not detected and there is no UDS service request, the MCU

[redacted]

[redacted] the host [redacted] microcontroller is reset to detect whether the APP code is downloaded. If the APP download is completed, the bootload mode will exit and the APP code program will [redacted] 0x12002000. If the download is not completed, the current state will still be bootload mode. If the APP program has been downloaded before, the APP program can be directly re-downloaded in APP mode without power down and restart. If there is an error in the APP, it can power off and restart. Before the 50ms delay ends, it can send the instruction to enter bootload mode to download the APP again. The flow chart of Bootload downloading APP is as follows Figure 8:

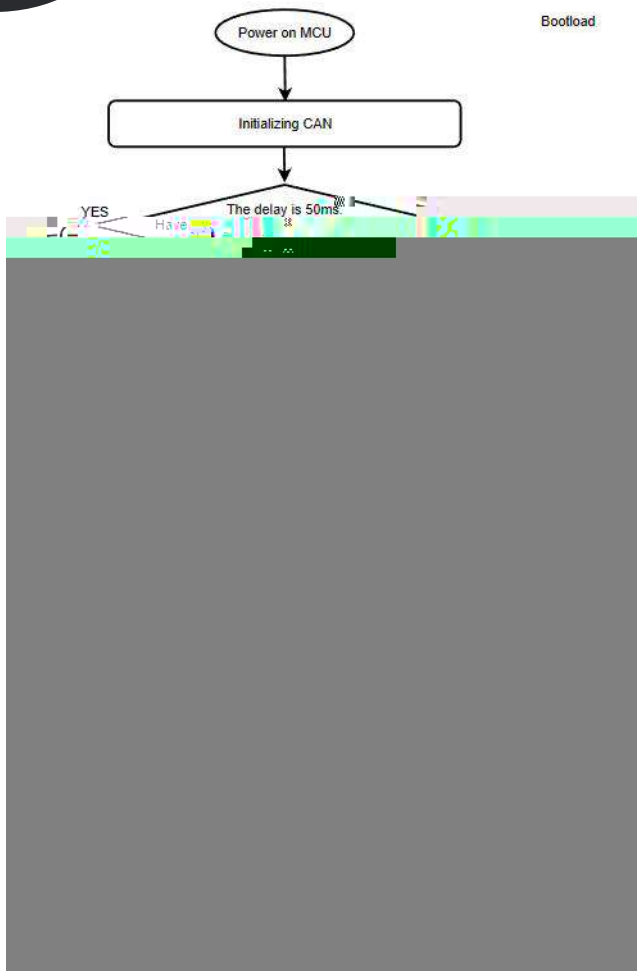


Figure 8

Get a new [redacted] state 0, then state to download bootload code, when the bootload code download success

APP program. If APP is downloaded successfully, it will enter state 2, the MUC will jump from bootload mode to APP mode, and the APP will start to work. If the APP code needs to be re-downloaded, if the process of downloading APP is interrupted or an error occurs, the MCU will re-enter state 3 and wait for the APP code to be re-downloaded until the re-download of APP is successful, then enter state 2 and run the APP program.

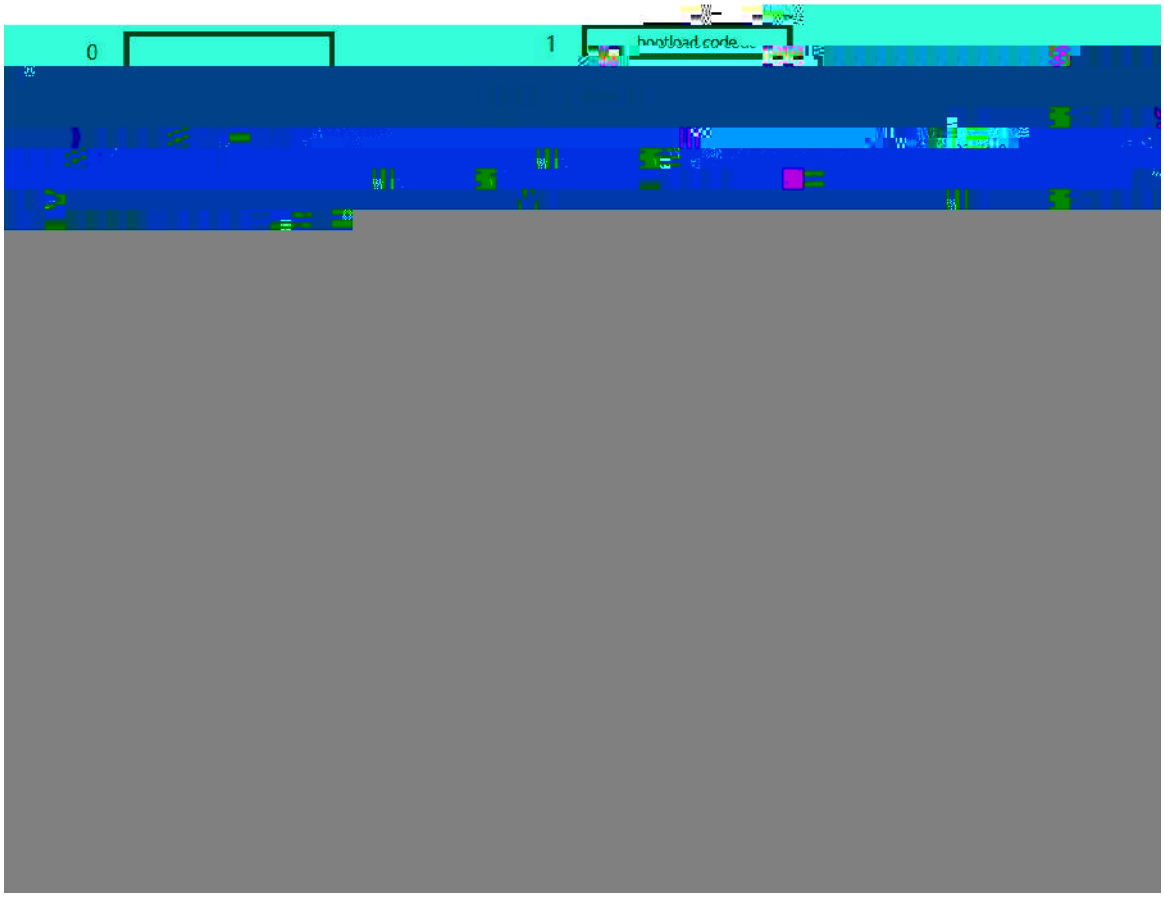


Figure 9

As shown in Figure 10 below, APP1, APP2, bootload, TSmaster_bootload, and bootload usage documents are provided in the file. APP1 file and APP2 file are APP routines for LED flashing at different frequencies. TSmaster_bootload file is the configured TSmaster host computer software routine, combined with bootload can realize the function of downloading APP. The bootload file contains the bootload source code.

Figure 10

Step 1: The chip is connected to the downloader.

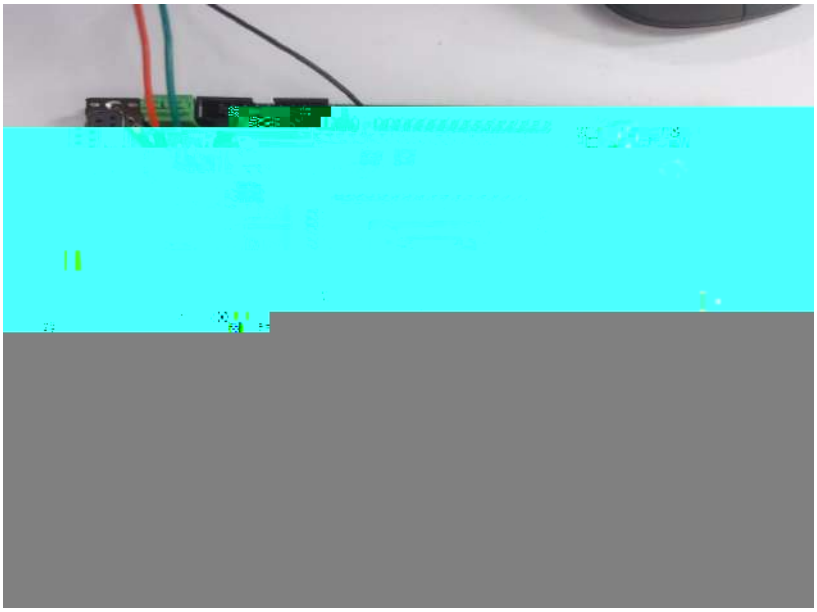
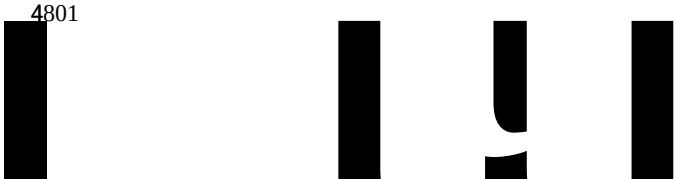


Figure 11

4801



Step 1: Connect the CANFD channel of the same star TC1012P to the CANFD channel of the MCU.

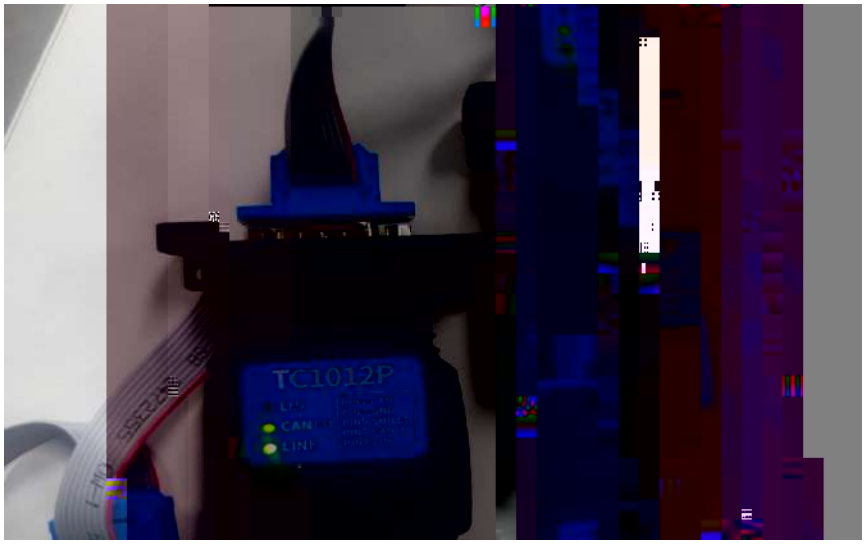


Figure 14

Step 2: Open boot_TSmaster file, click Hardware - > Channel selection, select CAN, configure the number of channels in the application, and finally configure the hardware channel selection to TOSUN TC1014 CANFD channel 1 .

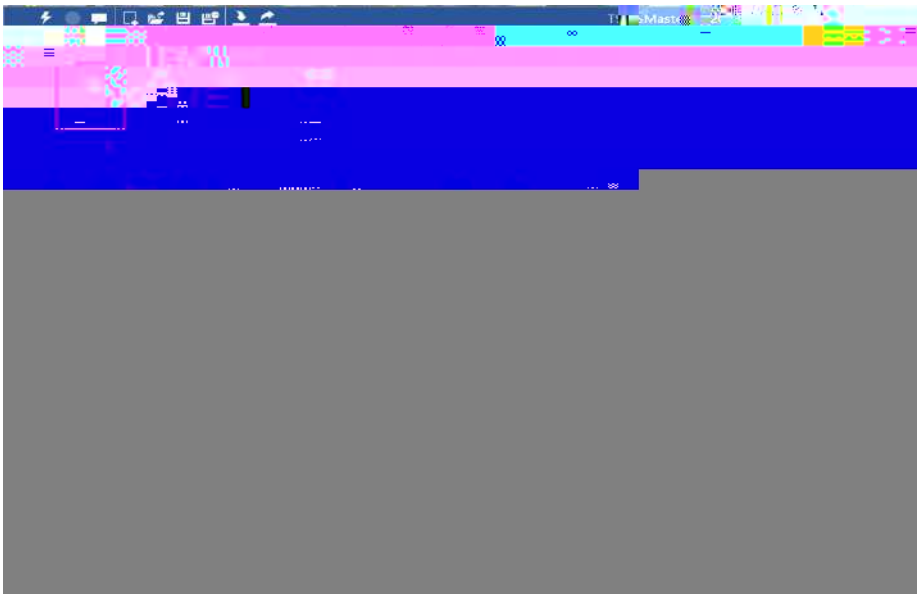


Figure 15

Step 3: Go to App - > Diagnostic Module - > Basic Diagnostic Configuration - >343637 Download File - > File Path to lok

nai TOSUN Technology Ltd.

Step 1: Open TSMaster's diagnostic module.

Figure 19

Step 1: Click the Transport Layer (ISO TP) button. Step 2: Go to the Diagnostic Transport layer page. Step 3: Configure the bus type as CAN/CANFD; Channels are configured on demand (Channel1) The request ID is configured to 0x755. The request ID type is set to Standard. The reply ID is configured to 0x7FF. The reply ID type is configured to be Standard. Function ID is configured to 0x7DD. The function ID type is set to Standard. Padding bytes can be configured arbitrarily. The maximum DLC of FC is set to [15]64Bytes. FD variable baud rate can be checked (checked to speed up the transmission rate).

Figure 20

Step 1: Click the Transport Layer (ISO TP) button. Step 2: Go to the diagnostic service layer page. Step 3: Configure P2Time(on demand) Step 4 Configure the online parameters of the diagnostic instrument (configure according to requirements). Step 5 config the seed key for the 27 service.

Figure 21

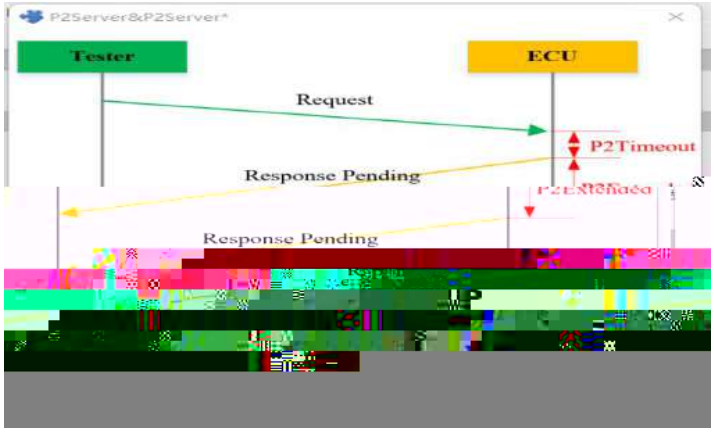


Figure 22

The Key generation rule can be modified by modifyip خ

The first step is to enter the automatic diagnosis process interface. The second step is to create a UDS FLOW. The third step is to load the UDS service request process (the request download process should conform to the UDS specification). Step 4 Click the Download button to start downloading the *.hex file. The blue box gives feedback on service requests and service responses.

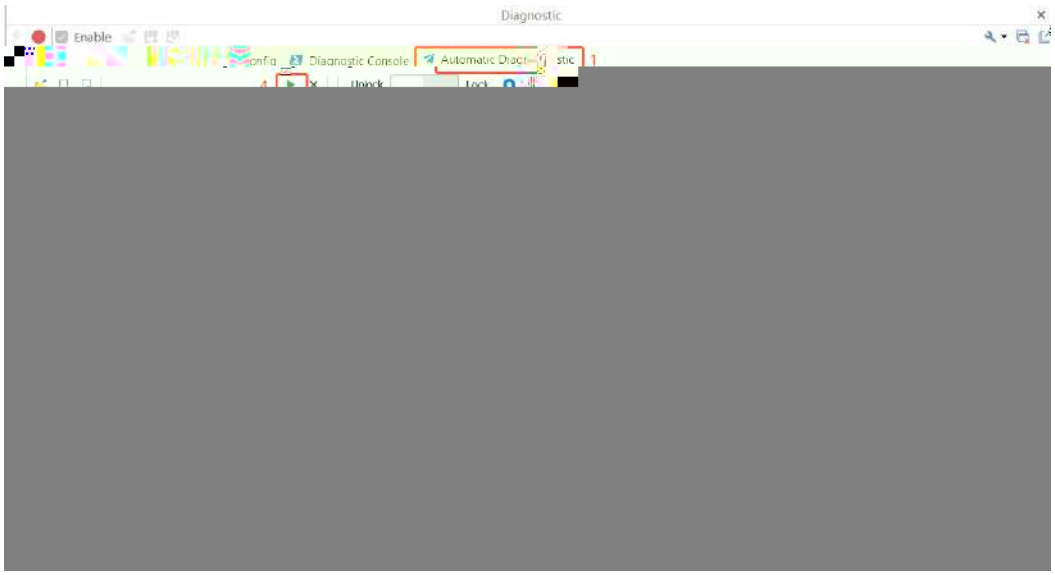


Figure 28

The bootload program is downloaded to the MCU, and then bootload with the TSMster host computer can realize the purpose of downloading APP through CANFD

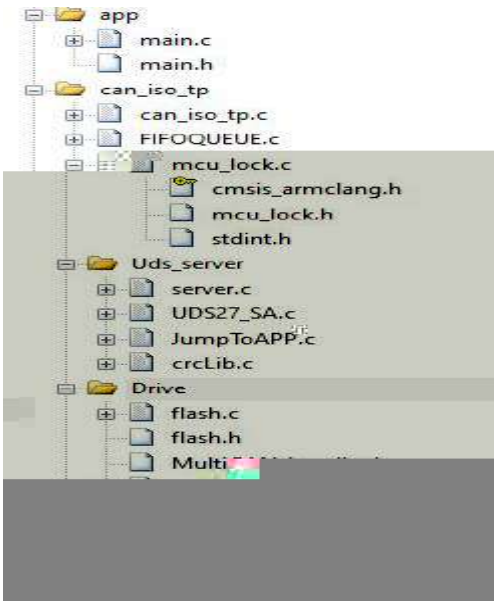
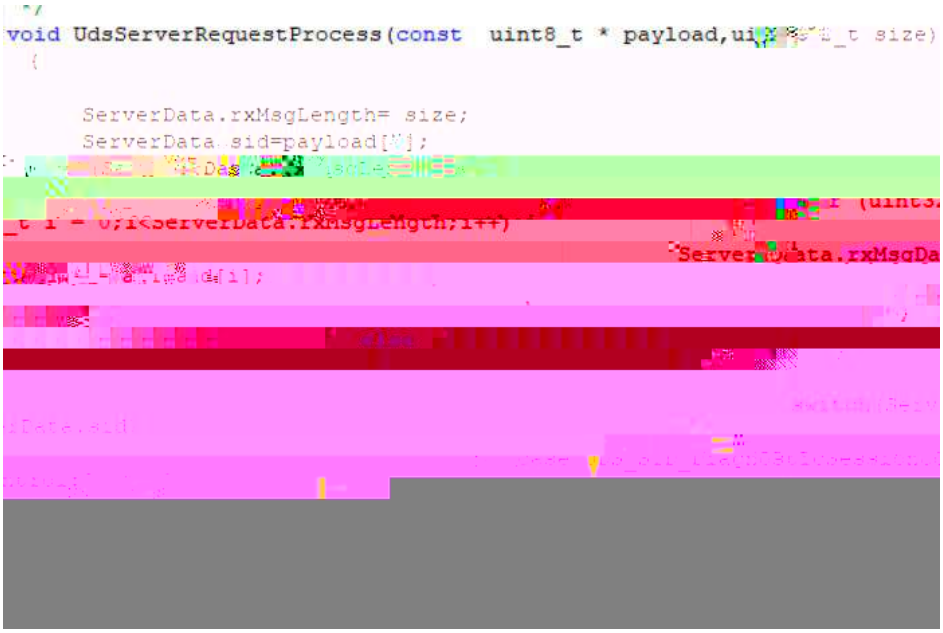


Figure 29


void UdsServerRequestProcess(const uint8_t * payload,
ServerData.rxMsgLength= size;
ServerData.sid=payload[0];

Fi

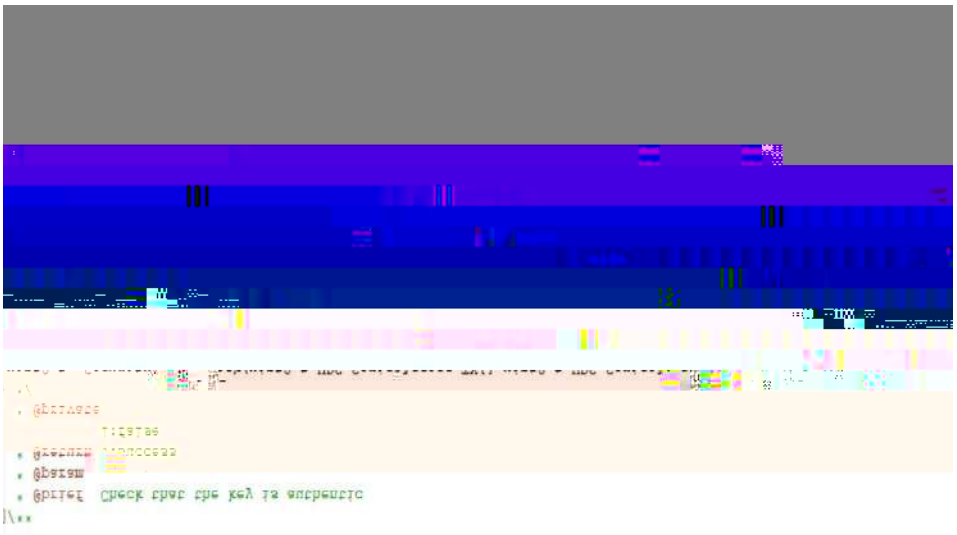


Figure 32

Test entries	expected result	actual result	Remark/explanation
(using TSMaster) Use the host computer software to burn the APP application	The APP functions normally	OK	
After the APP has been flushed, it can be flushed multiple times in a row (e.g., 5 or more times)	The APP functions normally	Flush more than 5 times, OK	
Bootloader software modifies the start address of App (within the normal address range)	It can be brushed normally, the APP jumps correctly, and the function runs normally	OK	Addresses start from 0x12002000 to 12040000
When the App is flushed for the first time, power down operation is performed on the ECU during the connection phase. After power on again, you can re-brush again.	The APP functions normally	OK	
When the App is flushed for the first time, power down operation is performed on the ECU in the erase Flash stage of APP program flushing.	The APP can be flushed normally	OK	

®o

4801 209
Room 209, No 4801 Caoan RD. Shanghai
www

(ditto above) and the communication (ditto above), and the normal flush can be performed again.			
During the flush process, CANH is disconnected, and after recovery, it can be flushed normally again.	The APP can be flushed normally	OK	
During the flush process, the CANH short-circuits the power supply, and after recovery, it can flush normally again.	The APP can be flushed normally	OK	

1.A mock

OSUN

4801 209

Room