Research on Vehicle ECU Application Program Update System Based on SAE J1939 Protocol

Cheng'e LUO
School of Information Engineering,Wuhan Huaxia Institute of Technology,Wuhan, China
e-mail:luochenge2012@163.com

***Abstract*—Aiming at the problems of cumbersome update operation, high hardware cost, complicated protocol design, and inability to roll back if the update fails, a set of automotive electronic control unit ECU application update system was developed. The software of the upper computer update control platform was developed, and the target ECU was realized based on the STM32 controller. The system structure mode was designed as the client-server mode, and the two ends of the system were connected through the USB-CAN adapter to realize the real-time communication of the system. Based on the startup control program and the In Application Programmable (IAP) method, the target ECU Flash programming was carried out to realize the convenient update operation.Handshake communication, packet serial number verification, retransmission, application storage partition and other designs realize stable and reliable communication process, and updates can be rolled back.**

Keywords- Automotive Electronic Control Units; Recommended Operating Procedures for Road Vehicles; Controllers; Client-Server; Startup Control Programs; In-Application Programming.

I. INTRODUCTION

The most common application of embedded systems based on the Controller Area Network (CAN) bus protocol is in the field of automotive electronics. Automotive electronics is also called ECU (Electronic Control Unit, ECU) [1]. It is composed of large-scale integrated circuits and is used in power control systems, body electronic control systems, etc.[2][3]. In order to improve the performance, stability, or fix vulnerabilities of in-vehicle ECUs, automakers usually update the in-vehicle ECU applications

II. Key technology research

SAE J1939 is established on the basis of CAN2.0B. Its frame structure design follows the CAN extended data frame structure design. There are two types of frame types: connection management frame and data frame. The connection management frame is used to start and close the connection, or use In control flow[4]. The connection management frame TP.CM has a request frame (Connection Mode Request to Send, TP.CM_RTS), a response frame (Connection Mode Clear to Send, TP.CM_CTS), an end frame (End of Message Acknow ledgment, TP.CM_EndOfACK), and connection Release frame (Connection Abort, TP.CM_CA) and broadcast frame (Broadcast Announce Message, TP.CM_BAM), a total of 5 types of frames. Among them, TP.CM_RTS frame is used to send connection request, TP.CM_CTS frame is used for response, TP.CM_EndOfACK frame is used for end confirmation, TP.CM_CA frame is used for connection release, and TP.CM_BAM frame is used for broadcast information[4].

The definition of P.CM_RTS frame data field is shown in Figure 1:

Start byte	field	Lengt h byte	Description and requirements
0	Ctl	1	Control byte,Value16
1	Total byte	2	Total bytes of pre sent valid data
3	Total pack	2	Pre sent TP Number of DT frames
5	PGN	3	Value 60416,PF=236,PS=0

Figure 1 TP. CM_ RTS frame data domain

The definition of the TP.CM_CTS frame data field is shown in Figure 2:

Field	Length	Description And Requirements
1	1 byte	control byte，value is 17
2	1 byte	Maximum number of packets or frames
3	1 byte	Sequence number of the next packet/ or frame
4-5	2 bytes	Reserve
6-8	3 bytes	PGN,value is 60416

Figure 2 TP.CM_CTS frame data field definition

The data field definition of TP.CM_EndOfACK frame is shown in Figure 3:

Field	Length	Description And Requirements
1	1 byte	control byte,value is 19
2,3	2 bytes	Total number of bytes
4	1 byte	Total package/frame number
5	1 byte	Reserve
6-8	3 bytes	PGN,value is 60416

Figure 3 TP.CM_EndOfACK frame data field definition

The data transmission information frame (Transport Protocol-Data Transfer Message, TP.DT) is transmitted by the initiator. The definition of the TP.DT frame is shown in Figure 4:

2473-3636/22/$31.00 ©2022 IEEE
DOI 10.1109/DCABES57229.2022.00068

Parameter	Description And Requirements
Data length	8 bytes
Data Page,DP	value is 0
PDU Format,PF	value is 235
PDU Specified field,PS	value is 0
default Priority,P	value is 7
Parameter Group Number,PGN	60160 (00EB0016)
Packet or Sequence Number	value is 1-255(1byte)
Data	7 bytes

Figure 4 TP.DT frame definition

III. SYSTEM SCHEME DESIGN AND REALIZATION

A.System Protocol Design

The definition of P.CM_RTS frame data field,control byte(value is16,1 byte),Total number of bytes(2 bytes), byte total packet or frame number(1 byte), Maximum number of packets or frames(1 byte),PGN(value is 60416,3 bytes)[4]. The definition of the TP.CM_CTS frame data field,control byte(value is 17,1 byte),Maximum number of packets or frames(1 byte),Sequence number of the next packet or frame(1 byte),Reserve(2 bytes),PGN(value is 60416,3 bytes)[4]. The data field definition of TP. CM_End OfACK frame, control byte(value is 19,1 byte),Total number of bytes(2 bytes),Total package/frame number(1 byte),Reserve(1 byte),PGN(value is 60416,3 bytes)[4].Transport Protocol-Data Transfer Message, TP.DT，Packet or Sequence Number(1 byte),valid data(7 bytes)[4].

B.Overall Design of the System Scheme

The system adopts the upper computer/lower computer mode, the upper computer is a PC, as the update server, the lower computer is the target ECU to be updated, and the ECU is the embedded system device. The PC is connected to the USB-CAN interface card through the USB interface [5].

The overall structure of the system is shown in Figure 5.

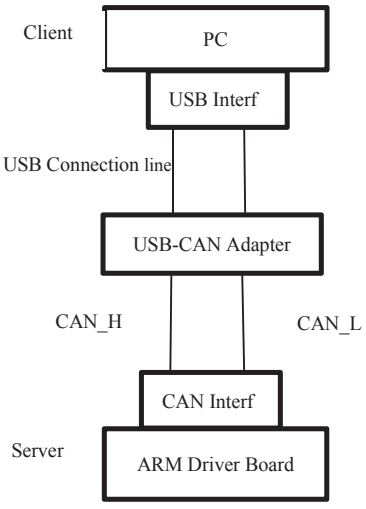

Figure 5 The overall structure design of the system

C.Detailed Design of the System Scheme

The System communication step design: 1. The upper computer needs to send an update request frame TP.CM_RTS to the lower computer; 2. The lower computer sends the response frame TP.CM_CTS after receiving the TP.CM_RTS frame; 3. The upper computer receives the TP.CM_CTS frame Then send the data frame TP.DT; 4. The lower computer sends the end frame TP.CM_EndOfACK after receiving the TP.DT frame; 5. The upper computer ends the update operation after receiving the TP.CM_EndOfACK frame. The system peer-to-peer communication sequence is shown in Figure 6.

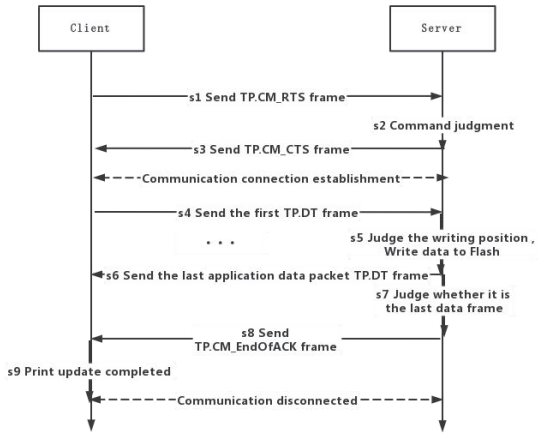

Figure 6 System communication sequence diagram

The The upgrade control program design of the upper computer software design.1. initialize the CAN control module, and open the CAN receiving thread; 2. Send the update data file, and initiate an update request to the lower computer. 3. Determine whether the TP.CM_CTS frame has been received. If it has not been received, enter the retransmission mechanism and retransmit the TP.CM_RTS frame.4.Enter the data transmission cycle, each transmission of a TP.DT frame needs to determine whether to receive the TP.CM_CTS frame and TP.CM_EndOfACK frame, if the TP.CM_EndOfACK frame is received, the update is completed and the lower computer restarts.

The overall hardware structure of the lower computer.This system uses the STM32F103VET6 microcontroller chip [6] as the microcontroller of the system's lower computer.

BootLoader subprogram design of the lower computer.1. System initialization.2. Determine which application program to start in Flash according to the value of the judgment flag. 3. Determine whether there is an application program in the application program storage block SRAM.

The design of the application program subprogram of the lower computer.1. First judge whether there is a CAN frame arrives. 2. If you receive the data receiving interrupt from the CAN port, enter the CAN receiving interrupt processing function to read the data in the CAN receiving buffer area. 3. Judge whether this data is the TP.CM_RTS frame sent by the host computer. If yes, enter the TP.CM_RTS frame processing program, and return to the

TP.DT frame detection and judgment place after processing to determine the next received frame. 5. If it is a TP.DT frame, enter the TP.DT frame processing program, otherwise end.

The subprogram design of the lower computer TP.DT frame processing.1. Determine whether the frame number of the received data is consistent with the frame number of the TP.DT requested by the last sent TP.CM_CTS. 2. Judge whether it is the first TP.DT frame. If it is the first frame, first select the storage address of the data according to the value of the judgment flag. Here, if the value of the flag can be divided by 2, it can be stored in The storage area of APP2, otherwise it is stored in APP1. 3. After writing 1 frame of data.If the last TP.DT frame is written, a TP.CM_EndOfACK frame is sent to the upper computer to request the end of the transmission.

D.System Solution Realization

According to the determined overall structure of the system, the system is further analyzed from the physical structure to obtain the functional structure block diagram of the system.

The software of the upper computer of the system adopts the Windows operating system and the Microsoft Visual Studio2010 platform for software development, and adopts the C/C++ programming language for programming code. The main CAN structure OBJ and CAN interface function API used by the upper computer upgrade control program. The VCI_CAN_OBJ structure is used to transmit CAN frames in the VCI_Transmit and VCI_Receive interface functions. The VCI_INIT_CONFIG structure defines the configuration of initial CAN. This system sets the CAN frame as a data frame, the frame length is 8, the frame format is an extended frame, and the value of the TP.CM_RTS frame ID is set to 0x1cec0053. The first byte of the TP.CM_RTS frame data field is the control byte, and the value is set to 0x10, which is 16. The second and third bytes are the total number of pre-sent bytes. The total number of bytes is calculated by reading the total number of bytes of the sent file. The sent file data is the new application data pre-written into the program memory of the lower computer. The 4th and 5th bytes are the total number of pre-sent CAN frames. The sixth, seventh, and eighth bytes are the parameter group number PGN of the CAN frame, the value is 0x00ec00, and the value of the parameter group number PGN is set according to the SAE J1939-71 protocol for the regulation and description of the PGN value, where the value represents communication The target is a certain ECU of the automobile power system. The ID value of the TP.DT frame is assigned as 0x1ceb0053, and the first and second bytes of the data field are obtained by the frame sequence number of the request for the next frame carried in the third byte of the TP.CM_CTS frame sent by the lower computer. The 3-8th bytes of the data field are the effective data transmitted, which are obtained by sequentially reading the data in the CAN sending buffer.

The lower computer needs to develop two functional programs, named BootLoader program and application program, respectively. The BootLoader program completes the function of selecting and starting the application program, and the application program completes the function of updating the Flash application program of the RAM driver board through the CAN network. The development of the ARM driver board program of this system adopts the firmware function library and hardware related materials provided by the STM32 chip manufacturer, the software development platform is Keil µVision 4.10, and the process-oriented thinking of C language is adopted to develop. The main structures CanRxMsg and CanTxMsg, CanRxMsg mainly defines the CAN frame in the CAN receive buffer.All members of the structure CanTxMsg have the same definition as that of the structure CanRxMsg. The interface function u8 CAN_Transmit (CanTxMsg* TxMessage) is used for CAN frame transmission. void CAN_Receive(u8 FIFONumber, CanRxMsg* RxMessage) is used to receive CAN frames. Flash_Status Flash_ErasePage (u32 Page_Address) is used to erase the data stored in Flash. Flash_Status Flash_ProgramHalfWord(u32 Address, u16 Data) is used to write data into Flash.

The BootLoader program completes the application startup selection, and judges by dividing the value of data_read by 2 remaining. If the remaining 2 cannot be divided, the data received by the lower computer is stored in the Flash_APP2_ADDRESS storage address, otherwise, it is stored in the Flash_APP1_ADDRESS storage address.

IV. TEST EXPERIMENT AND RESULT ANALYSIS

System software and hardware joint debugging test operation steps: 1. Burn the BootLoader program to the ARM driver board; 2. Burn the application program to the ARM driver board; 3. Compile the update program; 4. The lower computer burns the above 2 programs After that, the system restarts; 5. The upper computer transmits the prepared update program data file through the control program through the control program through the interface of the upgrade control platform, and transmits a frame of data packets to the lower computer, and at the same time prints the upgrade related information in the serial port print form. Observe the information prompt of the data transmission process.According to the experimental test observations, the upper computer printed the "Upgrade Complete" message after the system was updated, and all the 3 LED lights of the lower computer were on. At the same time, the judgment flag value was 0 from the serial port window information, because every time the update is completed The value of the lower computer restart judgment flag is increased by 1. At this time, the BootLoader program selects to start APP1, which is the update program written last time, so it is verified that the update program has been running normally after the lower computer restarts.

Analyze the performance of system design optimization points through test data:

1. This system uses the first 2 bytes of the 8 bytes of the TP.DT frame data field for frame sequence number definition, and the last 6 bytes as valid data. The system can transmit a large data packet of about 384kb. This system has been experimentally verified with about 11kb data transmission, and the entire data transmission process is more than 1 minute, which verifies the good real-time performance of the data transmission of this system. 2.

Communication mechanism design. The upper computer adopts a single-frame data transmission mechanism. Each frame of data from the upper computer received by the lower computer needs to be checked. If the check is correct, the data will be written into the Flash. If it is not correct, an error will be prompted. The stable communication process confirmed the correctness of this design. 3. The processing of the retransmission mechanism. The correct writing of data from the lower computer confirms the correctness of this design. 4. Transfer file type processing. After testing and verifying that the binary .bin file is used for transmission, the correctness of the design is confirmed by the absence of reading data errors during the transmission process. 5. Flash partition design. In order to ensure that the new application does not cover the old application, Flash is divided into two application storage areas APP1 and APP2, and the new application writing location is determined by the design judgment flag. The correctness of the design is verified by the old application check after the update. 6. Verification of the successful update. The application data before and after the update is compared for consistency in bytes. When the comparison is consistent, the update is proved to be successful, otherwise the update fails. If successful, the lower-level computer will give a light prompt, and all the LEDs 1-3 of the lower-level computer will be lit, otherwise it will fail. Secondly, after the update is completed, the lower computer judges the value of the flag to change, and immediately runs the latest application after restarting, which can immediately verify the running effect of the latest application.

Through the experimental test, the functional test cases, stability and reliability test cases set in the experiment are all executed, the update can be successfully executed, and the original application is not covered after the update. The functional correctness of the system, the stability and reliability of the system. In the performance test, the communication response time between the upper computer and the lower computer is very fast, the ms level is very high, and the real-time performance is very high. In terms of transmission rate, according to the data transmission baud rate set by the host computer platform, with the increase of the transmission file, the average transmission rate can be close to the set baud rate value. The maximum baud rate supported by the lower function of this experiment is 115.2Kbps, which meets the test requirements. If the read and write rate of the hardware device is faster, a higher baud rate can be set, and the transmission rate is faster. This rate constraint is limited by the hardware performance. It is not on the system software. In subsequent experiments, chips with better read and write performance can be selected for experiments.

Ⅴ. CONCLUSION

Aiming at the problem of updating the ECU application program of the vehicle network.This research designs an improved ECU application program update system. The proposed system protocol system is based on the SAE J1939 protocol method, which makes the system business design simple and can be directly applied to the vehicle CAN network without additional protocol compatibility and conversion; Hardware design cost is low. The system has few transmission nodes, and the wired transmission network ensures real-time and stable communication. The research has practical application value for the software upgrade of the ECU in the vehicle network. It is not bound by the underlying hardware protocol and can be widely used. Due to the short period of time, it is necessary to continuously find and solve problems in practical applications to improve the system.

REFERENCES

[1] LIU Z,ZHANG T.Research on automatic lane change method based on vehicle network information[J].Journal of Chongqing University of Technology(Natural Science),2020,34(4):11-17.

[2] Zhao Haijian, Gan Meng. Research on Flash programming technology in embedded systems [J]. Computer Engineering and Design, 2006, 26(11): 3006-3009.

[3] Yu Tianqi, Hu Jianling, Jin Jiong, Yang Jianfeng. Intrusion detection method of in-vehicle CAN network based on mobile edge computing [J]. Computer Science, 2021, 48(1): 34-40.

[4] SAE, J1939-21-2001, data link layer of SAE International Advanced Mobility Engineering Society Land-Sea-Air and Space Ground Vehicle Recommendations [S]. United States: Copyright SAE International, 2001:1-47.

[5] YUE Bin-bin,Li Xiang-yang.CortexM3 USB-CAN Development of the USB-CAN Converter Based on CortexM3[J].Computer Engineering & Science,2012,34(5):68-72.

[6] ZHANG Xiang,YANG Dong-sheng.Research pf Porting and Low-power for mbed OS Based on STM32[J]Journal of Chinese Computer Systems,2020,41(3):564-568.