A Design of Communication Interface of ADS-B for UAV

Chao Wei, Yanchen Dai
Research Institute of Electronic Science and Technology
University of Electronic Science and Technology of China
Cheng Du, China
uestc_weichao@foxmail.com, daiy777@163.com

Abstract—Through analyzing the small UAV (Unmanned Aerial Vehicle) flight control system based on CAN (Controller Area Network) bus, the paper focuses on the study of flight control system bus architecture, and puts up an ADS-B (Automatic Dependent Surveillance-Broadcast) architecture design which is suitable for this type of UAV. Transplants the mature ADS-B system with a focus on the design of CAN bus interface, and details the initialization of the CAN bus controller SJA1000 module and read-write module processing flow, designs and verifies the correctness of register read-write module on FPGA development platform according to the register read-write timing of CAN bus controller. Solved the key technologies of ADS-B system transplantation and verifies its correctness.

Keywords—CAN bus; UAV; ADS-B; FPGA

I. INTRODUCTION

ADS-B is one of the main means of aviation surveillance, which has become an important part of ICAO Future Air Navigation Systems[1]. It mainly implements air-to-air surveillance, ADS-B system is formed by many ground stations and airborne stations, accomplishes data bi-directional communication by network, multipoint to multipoint. Secondary Surveillance Radar ground station construction meets the problems such as high cost, site restrictions, high running guarantee, especially in the mountain or desert, ADS-B technology is no doubt the complement and the best alternative of Secondary Surveillance Radar. Unmanned aerial vehicle system is widely used in the field of civil and military, and some of the payload, range and service ceiling meet or exceed conventional manned aerial vehicle, such as the United States MQ-X and RQ-X series of unmanned aerial vehicle systems, these unmanned aerial vehicles should operate under the authority of the appropriate air traffic control departments, and one airborne ADS-B transponder also becomes a standard equip. In the flight control system of UAV research field, some research and practice in domestic uses CAN bus as the flight control system bus.

In this paper, with the flight control system with CAN bus interface author proposed a UAV ADS-B system structure design with CAN bus interface, described communication interface design between ADS-B system and the UAV flight control system, and verified the correctness of the interface design of critical steps.

II. UAV FLIGHT CONTROL SYSTEM

Distributed system using modular building with a single bus has a feature of easy to maintain and upgrade. MIL-STD-1553B bus which is commonly used in avionics has only one bus controller to manage the entire bus run, so there is a potential danger of bus failure, and its price is high. Another ARINC 429 bus while is not managed by a single bus controller, but only for point-to-point connection, this will no doubt increase the system wiring length and the weight of the system, and its fastest transfer rate is 100Kbps. Unmanned aerial vehicle system is very sensitive to the cost and weight, so both buses above cannot be used. CAN bus is a multi-master bus, and the fastest bus transfer rate is up to 1Mbps, which has the complete physical layer and data link layer protocol, and is easy to form a modular distributed system; it has the complete error management and high reliability. Moreover, the CAN bus is applied extensively and maturely in automobile electronic and industrial control fields, its superiority is a result of long practice. Based on the analysis above, the CAN bus applied to the UAV flight control system is a reasonable choice. The Institute of Flight Control of Nanjing University of Aeronautics and Astronautics has the successful experience of developing a UAV flight control system based on CAN bus. Their design is based on ARM7 processor as the main controller. The system is formed by inertial measurement unit, GPS module, wireless measurement and control unit, air data sensor and others. This modular distributed bus architecture has the good scalability, and CAN bus system can close the node automatically without affecting the normal operation of other nodes in the current operation in case of node failures. The UAV ADS-B that acts as a CAN node mount on the bus, to realize data transmission with the flight control system. The system hardware structure diagram as shown in Figure 1[2].

III. DESIGN OF CAN BUS COMMUNICATION INTERFACE

Research Institute of Electronic Science and Technology of UESTC with independent intellectual property rights of a certain type of airborne ADS-B system through the ARINC429 bus interface for avionics system. In this paper, combined with the characteristics of UAV flight control system based on the CAN bus redesigned the bus interface, and the bus interface design using dynamic priority algorithm to make up for the
deficiency of CAN bus at real-time in the condition of high load rate, and the dynamic priority strategy is reflected in the interface circuit of data transceiver module[3][4].

A. Initialization module

The initialization operation of SJA1000 occurs at system power on, hardware reset or main controller trigger software reset. Initialization process, including the mode register set, clock divider register set, acceptance code register set, acceptance mask register set, bus timer set and output register set. The ADS-B system has a high requirement for real-time, initialization register set to interrupt mode. To ensure data transmission with high reliability, the mode register set to PeliCAN mode, in this mode the SJA1000 support for error analysis and system diagnosis function, can make the system easy to maintain and optimize. The initialization flow chart as shown in figure 2.

B. Sending module

Sending module includes sending main module and sending interrupt module. Storage medium in message transmission process contains two parts, sending cache in SJA1000 and the temporary storage in FPGA, the temporary storage empty fullstate by setting the "further_message" flag state mark. Enter the sending process to enable sending interrupt register of CAN Controller then judge the "transmition buffer state" bit value of the state register of SJA1000. If the sending buffer released continue to write message to the cache, and set the command register "sending request" sign; If the sending buffer is not released turn to handle data to be sent in temporary storage and set "further_message" flag, after the current message successfully completed generate an interrupt, then the interrupt program continue sending message in temporary storage. The CAN controller SJA1000 generates a interrupt after successfully sent a message, sending module read status register of the CAN controller after the SJA1000 received the interrupt signal, if the result of judgment is sending interrupt then enter the transmit interrupt service program, continue to judge whether the "further_message" flag is set or not to determine whether there are messages in the temporary register waiting to be sent, if the "further_message" flag was set then clear the flag and send the message to the sending buffer of CAN controller, then set the "sending request" bit of command register to complete the sending process. Sent over the SJA1000 generates interrupt again, loops until the temporary without more sending message is the end. Sending module processes and interrupt handling process is shown in figure 3.

C. Receiving module

The CAN bus controller SJA1000 according to the setting of acceptance code register and acceptance mask register
automatically receive data to store into the receiving FIFO, at the same time to generate interrupt informs the main controller to read data. After receives the interrupt, FPGA read interrupt register of CAN bus controller read and judge whether the interrupt was triggered by “receiving FIFO full” state, if it is, then receive messages from the FIFO and save the current messages, and then release the receiving FIFO, release the arbitration lost capture registers and the error code capture register. The receiving module flow chart is shown in figure 4.

Figure 4. Receive message flow chart

IV. SJA1000 REGISTER READ-WRITE MODULE

The key points of CAN bus interface design is the design of CAN bus controller SJA1000 internal register read-write module, accurately reading and writing control register and data register is a prerequisite to achieve communication function. This design in strict accordance with the SJA1000 register reading and writing timing, use a state machine to implement reading and writing function.

A. SJA1000 reading and writing timing

The signal lines of SJA1000 are address/data multiplexing bus AD0–AD7, address latch signal ALE, chip select signal CS, the read enable RD, write enable WR, reset input RST, mode selection MOD and interrupt output INT. For general consideration of the design using Intel Mode to read and write register. In read and write process, firstly sends address of register to SJA1000, and sends the address latch signal, SJA1000 correctly latch corresponding register address. After sent reading address in the reading process, pull down the read enable and chip select signal, then SJA1000 correctly latch corresponding register address. After sent reading address in the reading process, pull down the read enable and chip select signal, SJA1000 correctly latch corresponding register address. After sent reading address in the reading process, pull down the read enable and chip select signal, then SJA1000 automatically sends out data of the corresponding registers for the main controller to read from the bus. After sent writing address in the writing process, pull down the read enable and chip select signal and sends data to the bus to complete the register write operation. Reading timing as shown in Figure 5, writing timing as shown in Figure 6 and the timing parameters as shown in Table 1.

Table 1. Timing parameters

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>PARAMETER</th>
<th>MIN.</th>
<th>MAX.</th>
<th>UNIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>fosc</td>
<td>oscillator frequency</td>
<td>~24 MHz</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tADn(AL)</td>
<td>address set-up to ALE/AS LOW</td>
<td>8 ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tH(L-A)</td>
<td>address hold after ALE LOW</td>
<td>2 ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tW(AL)</td>
<td>ALE/AS pulse width</td>
<td>8 ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tRDQV</td>
<td>RD LOW to valid data output</td>
<td>50 ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tRDIQ</td>
<td>data float after RD HIGH</td>
<td>30 ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tIWH</td>
<td>input data valid to WR HIGH</td>
<td>8 ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tWHRI</td>
<td>input data hold after WR HIGH</td>
<td>8 ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tWH</td>
<td>WR HIGH to next ALE HIGH</td>
<td>15 ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tAHL</td>
<td>ALE LOW to WR LOW</td>
<td>10 ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tALR</td>
<td>ALE LOW to RD LOW</td>
<td>10 ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tW(P)</td>
<td>WR pulse width</td>
<td>20 ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tW(R)</td>
<td>RD pulse width</td>
<td>40 ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tCLW</td>
<td>CS LOW to WR LOW</td>
<td>0 ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tCLR</td>
<td>CS LOW to RD LOW</td>
<td>0 ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tWHC1</td>
<td>WR HIGH to CS HIGH</td>
<td>0 ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tRCH</td>
<td>RD HIGH to CS HIGH</td>
<td>0 ns</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

a. Do not care about the specific value
B. The function verification of register read-write submodule

In this design, the read and write enable signal ‘read_trigger’ and ‘write_trigger’ of the upper level module to the reading and writing submodule are single cycle pulse, who respectively trigger the reading operation and writing operation for specific registers. The upper module sends out a "read_trigger" signal at the same time send out corresponding register address, then data can be read from the bus. The upper module sends out a "write_trigger" signal and sends out the corresponding register address and data to be written to realize the register writing function. The FPGA used in this paper is Cyclone IV E EP4CE15F17C8, and this design combined with the the optimum timing performance of SJA1000’s register to set the register read-write module frequency to at most 50 MHz. Using Quartus II integrated logic analysis tool SignalTap II to grab SJA1000 register read or write real-time oscillograph, as shown in Figure 7. In the module "can_pin" is defined as the bi-directional IO port, when the bidirectional port is set to output data inputs to the "can_pin" or SJA1000 data/address multiplexed bus from the inner port "can_in" . In the figure, the sixteen hexadecimal number “09h” in the figure is the address of the test register of SJA1000, and the data sixteen hexadecimal number “AAh” is the test data written into or read out from the test register. Pulls up "write_trigger" for a cycle and sends the register address and data to write to complete the register write function. Pulls up "read_trigger" for a cycle and sends the register address, then the corresponding register data is sent to the SJA1000 data/address multiplexed bus, and, at the same time, pulls up “can_dir” to set the bus direction to input, and data can be read into the FPGA. The waveform in Figure 7 proves correctness of register read-write module.

V. CONCLUSION

Through the analysis of UAV flight control system bus architecture, the article raises an ADS-B system which is node initialization module and the read-write module process, suitable for CAN bus interface of UAV platform. Details CAN designs the register read-write module according to the register read-write timing of the CAN bus controller SJA1000 and verify its correctness.

![Figure 7. Register read-write submodule timing simulation](image)

REFERENCES

学霸图书馆
www.xuebalib.com

本文献由“学霸图书馆-文献云下载”收集自网络，仅供学习交流使用。

学霸图书馆（www.xuebalib.com）是一个“整合众多图书馆数据库资源，
提供一站式文献检索和下载服务”的24小时在线不限IP图书馆。

图书馆致力于便利、促进学习与科研，提供最强文献下载服务。

图书馆导航：
图书馆首页     文献云下载     图书馆入口     外文数据库大全     疑难文献辅助工具