



# Design and Implementation of a Pipeline-Based Data Scheduling Method for Spacecraft with Multiple Data Buses

Sheng Yu<sup>✉</sup>, Duo Wang, Zejing Lv, Dan Wang, Zhenhui Dong, and Xiongwen He

Beijing Institute of Spacecraft System Engineering, CAST, Beijing, China  
Yusheng86@outlook.com

**Abstract.** The MIL-STD-1553B data bus is widely used in modern spacecraft. Since a 1553B bus can support 31 remote terminals (RT) at most, a complex spacecraft possibly has multiple 1553B buses, and these buses need to work simultaneously. How to efficiently manage multiple 1553B buses by a single bus controller (BC) is a challenging problem for such spacecraft. To solve this problem, we design and implement a pipeline-based data scheduling method. First, we introduce the procedure of data transmission and the concept of preemptive scheduling. Second, we break the procedure into modules and propose the pipeline-based data scheduling method. Third, we implement scheduling software based on the method and evaluate the performance of the method through experiments. We successfully applied the software on the Tianhe core module and achieved good performance.

**Keywords:** 1553B data bus · Pipeline · Data management · On-board network

## 1 Introduction

Data bus is the key part of the On-Board Data Handling (OBDH) system of a spacecraft, and the MIL-STD-1553B data bus is widely used in modern spacecraft [1]. The 1553B bus is fault-tolerant and highly reliable, and it can respond to data transmission command in real time. These features make the 1553B bus very suitable for building spacecraft on-board network [2, 3]. The maximum data transmission rate for a 1553B bus is 1M bit/s. To achieve the maximum capacity of a 1553B bus, it is important to design an effective data scheduling scheme.

The 1553B bus has two different work modes: the time synchronization mode and the time asynchronous mode. The time synchronization mode requires strict time synchronization between BC and RTs, and divides a transmission period into time slices [4]. In each time slice, specified types of data transmission are arranged, so that data transmission delays are strictly bounded. The time asynchronous mode does not require strict time synchronization, and does not use time slices. BC can start data transmission whenever necessary, thus the time asynchronous mode is more flexible [5]. However,

it cannot guarantee data transmission delay. The technique of priority queue and preemptive scheduling can be used to improve the efficiency and delay of the asynchronous mode.

According to the MIL-STD-1553B standard, one 1553 bus can support 31 RTs at most. The number of devices on a spacecraft is increasing rapidly, so one bus cannot satisfy the need of the information system of a complex spacecraft, and multiple 1553 buses need to work simultaneously. For example, the Tianhe core module uses 6 first-layer 1553 buses and one BC (the central unit) is in charge of all 6 buses. To achieve time synchronization of BC between RTs on 6 buses is very difficult, so the time synchronization mode is not suitable and the asynchronous mode should be used. The biggest challenge is to effectively schedule data transmission task on 6 buses by a single BC, and to achieve minimum transmission delay for time critical missions.

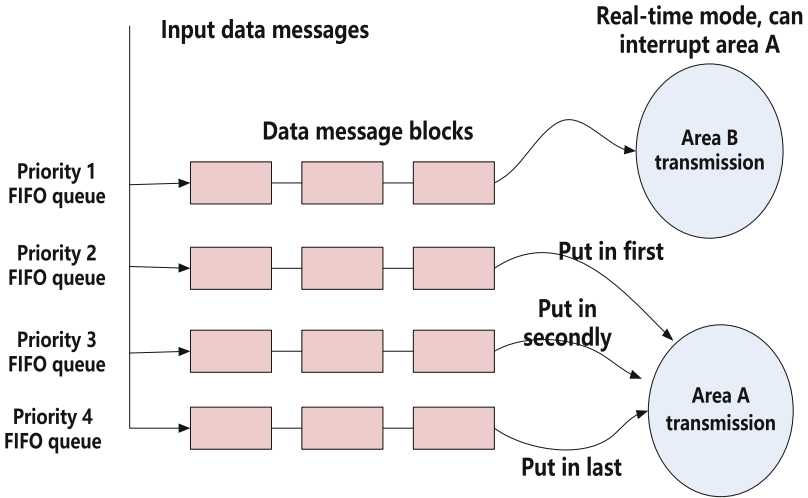
In this paper, we propose and design a pipeline-based scheduling method for multiple 1553 buses. First, we introduce the procedure of data transmission and the concept of preemptive scheduling. Second, we break the procedure into modules and propose the pipeline-based scheduling method. Third, we implement scheduling software based on the method and evaluate the performance of the method through experiments. We successfully applied the software on the central unit of the Tianhe core module and achieved good performance.

## 2 Data Transmission Procedure

According to the MIL-STD-1553B standard, there are 10 types of data transmission and 5 out of the 10 types are frequently used. The 5 types of data transmission are listed below:

- BC to RT data transmission: BC sends a receiving command word and several data words to RT, and RT replies with a status word.
- RT to BC data transmission: BC sends a transmission command word to RT, and RT replies with a status word, followed by several data words.
- $RT_a$  to  $RT_b$  data transmission: BC sends a receiving command word to  $RT_b$ , and sends a transmission command word to  $RT_a$ , then  $RT_a$  sends data words to  $RT_b$ .
- Mode command with data word: BC sends a transmission command word to RT, then RT replies with a status word and a data word.
- BC to RT broadcast transmission: BC sends a receiving command word (RT address is 0x1F) to all RTs, followed by several data words. RTs with broadcast function reply with a status word.

To increase the data transmission efficiency, we adopt the preemptive scheduling method. All data messages are assigned with priority 1 to 4, and they are put into 4 separate FIFO (First In First Out) queues according to priority type. Priority 1 is the highest priority, and BC always uses area B of the 1553 bus to send priority 1 messages. Priority 4 is the lowest priority, and BC uses area A to send messages with priority 2 to 4. The transmission of messages in area B can interrupt messages in area A, so data messages with priority 1 are transmitted in a real-time mode. Telecommand data, time stamp, and attitude control data are typical data with the highest priority. The preemptive scheduling method is illustrated in Fig. 1.



**Fig. 1.** The preemptive data scheduling method of a 1553 bus

Each bus has 4 FIFO data message queues. Thus, for a BC controlling 6 buses, there are 24 data message queues to manage. Within each data transmission period, the data transmission procedure of each bus can be divided into 4 modules. The first module is called data organizing module. In this module, BC organizes all data messages needed to transmit within this time period, assigns data message type and priority, and puts these messages into FIFO queues. The second module is called data write-in module. In this module messages in FIFO queues are written into the area A or area B of the 1553 bus RAM. Since the RAM size of 1553 bus is limited, only a certain number of messages can be written at a time. In our case, the number is 64, if the number of messages to send exceeds 64, then these messages need to be transmitted multiple times. The third module is data transmitting module. The 1553 bus chip transmit all data messages in its RAM according to their types, and each time at most 64 data messages can be transmitted. The module does not require the CPU of BC to process. The last module is data processing module. After data messages are transmitted, BC checks the status of transmission, and if transmission result is successful, BC saves the received data and processes or further transmits the data.

### 3 The Pipeline-Based Data Scheduling Method

To efficiently manage multiple data buses, we propose a pipeline-based bus data messages scheduling method based on the 4 modules of the data transmission procedure. A very important resource of a BC is the CPU time of the BC, and our target is to minimize the occupation of the CPU time for all bus data transmissions. The idea is based on the observation that the data transmitting module of each bus does not require the CPU time of BC, so BC can parallel process the data transmitting module of all buses.

The pipeline-based scheduling method uses a process and multiple threads. First, BC creates a process to manage bus data transmission. In each transmission period, the process deals with the data organizing module of all buses. Second, each thread performs the data written-in module. When enters the data transmitting module, a thread uses a semaphore to suspend itself and waits for the end of the transmission. At the same time, another thread can take the CPU and run the data written-in module for another data bus, and then suspend itself too. Third, when the data transmission is finished, the thread is activated again and continues to perform the data processing module. The pipeline-based scheduling method is illustrated in Fig. 2.

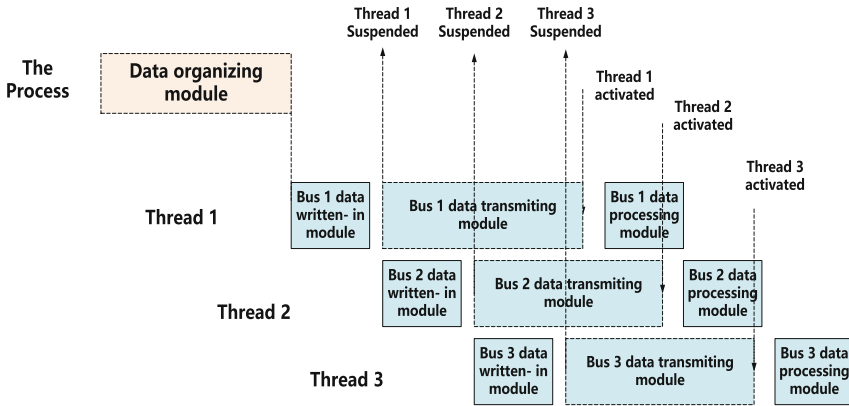
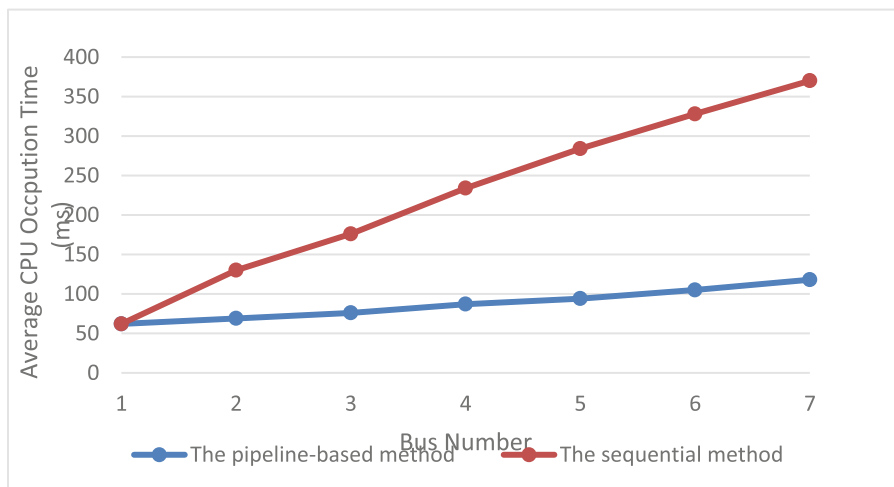


Fig. 2. The illustration of the pipeline-based scheduling method

### 4 Performance Evaluation

We implemented the BC scheduling software based on the proposed method and evaluate the performance of the method through experiments. The performance of the pipeline-based scheduling method is compared to the sequential schedule method that all 4 modules of each bus are process by CPU of the BC sequentially and the performance is evaluated by CPU occupied time of each method within a typical transmission period.

In our experiments, we use BM3803 as the CPU of the BC, and the main frequency of the CPU is 100 MHz. CPU cache is disabled. On each 1553B bus, the BC transmits 4K bytes data to a RT in every transmission period. The data length of each message over 1553B is 64 bytes, so each bus transmits 64 data messages. The same version of BC software is used in each experiment. We increase the number of 1553B bus controlled buy the BC, and we track the average time used by the CPU to send all data messages We compare the time used in two methods and the result is showed in Fig. 3. The results show that as the number of 1553B bus increases, the pipeline-based method can achieve better performance.



**Fig. 3.** The performance of the pipeline-based scheduling method vs the sequential method

## 5 Summary

In this paper, we design and implement a pipeline-based data scheduling method for spacecraft with multiple buses. The method is used on the Tianhe core module of the Chinese space station. We introduce the procedure of data transmission and the concept of preemptive scheduling. We break the procedure into modules and propose the pipeline-based scheduling method. Finally, we implement scheduling software based on the method and evaluate the performance of the method through experiments, and the results show that the proposed method can achieve good performance.

## References

1. Yu, S., Zhou, B., Niu, J., et al.: Design and implementation of a high efficiency space packet routing algorithm on a spacecraft. In: 12th International Conference on Wireless and Satellite System (WISATS) (2021)
2. Lan, L., Liu, X., Sheng, Y., et al.: Research on routing mechanism for Inter-spacecraft 1553B bus network of space station system. *Spacecraft Eng.* **29**(2), 95–101 (2020)
3. Lan, L., Sun, Y., He, X., et al.: Optimization method of command management between equipment of large spacecraft. *Spacecraft Eng.* **30**(4), 85–90 (2021)
4. Zhang, Y., Yang, P., Yang, Z.: Design of a kind of 1553B bus communication protocol based on time synchronization. *Spacecraft Eng.* **30**(2), 88–95 (2021)
5. Dong, Z., Qiang, M., Mao, L., et al.: RAM fault tolerant design of 1553B bus chip for HXMT satellite. *Spacecraft Eng.* **27**(5), 107–113 (2018)