Research of Application Simulation in Satellite Data Management System Based on SpaceWire

Session: Test and Verification
Short Paper

Wang Rui, Li Guoliang
China Aerospace Engineering Consultation Center
Cao Song
Center for Space Science and Applied Research, China Academy of Science
E-Mail: hustwr@yahoo.com.cn

Abstract

We use SMAT to design a model for the prototype of satellite data management system based on SpaceWire. This model is capable of dealing with situation where data rates vary in a large scope and space data systems in accordance with CCSDS international standard have multiple transaction structures. We use this model to make simulations of SpaceWire with the features of high speed transmission, network routing and redundance in linkage. In order to examine the feasibility of our experiments and the system performance, we make tests to provide system designers with more quantitative evidences.

1. Introduction

The tasks of Satellite data management system include program control of satellite, command delay, Payload Operation and Management, systems on satellite, and etc[1]. The AOS standard of Consultative Committee for Space Data Systems （CCSDS）integrates multiple data into a satellite data network, and it provides three services including data management, data routing and data transmission channel.

SpaceWire is being used successfully on many space exploration missions[2][3]. However, there are several problems SpaceWire must face in the design of a practical satellite data system, such as what kind of topological structure can improve efficiency and reliability of data exchange [4].

However, research of the problems above cannot rely on the actual network construction [5]. It is difficult to complete the research of performance and parameter test above in a relatively fixed physical network. While system simulation is a very good research means to resolve these problems.

2. Technique and Tools for Discrete Events Simulation

Based on the objective of system analysis, the system simulation establishes a simulation model which can describe the system structure or process of action and can be expressed by some logic or mathematical equation on the basis of analyzing the nature and relationship of the elements in the system. Accordingly, Experiments or quantitative analysis are made to obtain all the information required for proper decision-making. By the simulation model, system simulation can successfully solve the system problems such as forecasting, analysis and evaluation for some object systems which are difficult to establish the physical and mathematical model.

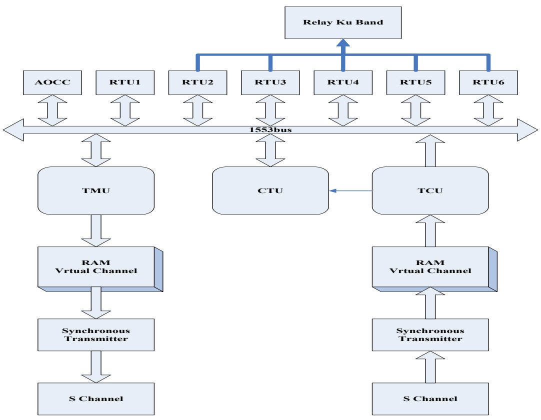
Based on the OMNET++, we developed the System Modeling & Analysis Tool. The tool can do the analysis, quantitative evaluation, validation and optimization for the system-level design. The whole process of system simulation is also supported by the tool.

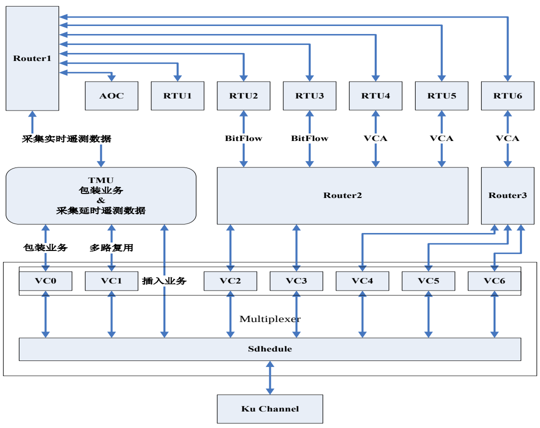
3. System Modeling

Satellite Conventional Orbiting System (COS) is composed with five units including Central Control Unit (CTU), Telecommand Unit (TCU), Telemetry Unit (TMU), Attitude & Orbit Control Circuit (AOCC) and Remote Terminal Unit (RTU), all of which are connected with serial bus (Bus 1553 in practice).

CTU is in charge of transaction management, bus dispatch, etc of satellites. TMU is designed to collect real-time telemetering data from sub-systems and send them into downlink virtual channels and to the surface through channel S, while generating delayed telemetering packages according to specified sampling rates. TCU sends indirect remote instructions to devices respectively through uplink virtual channels, RTU is used to carry platforms and actual payloads from which large amount of data generated is sent to the surface through wave band “Ku” by specified devices. The diagram below demonstrates the structure of COS model.

Since CTU, TMU and TCU are different units taken from one single CPU, and the amount of uplink data is small, the uplink part with mono functions can be facilitated into remote instruction generators. The CTU and TCU are integrated into TMU.





With SpaceWire and FPGA, the bottleneck of application in AOS can be solved. The diagram above demonstrates how to use SMAT to build a SpaceWire and FPGA based AOS downlink model, in which all routers are SpaceWire based, Router1 is in full-duplex mode, Router2 and Router3 are in simplex mode, TMU is the real-time or delayed telemetering data collect unit, and Multiplexer is combiner.

3.1 Data Source Model

In satellite electronic devices, all payloads units, telemetering units and telemetering units can be seen as data generators (data sources) obeying specified rules. AOS system is made up with eight transaction modules including routing module, internet module, package module, multiplex module, bit-stream module, virtual-channel-access module, virtual-channel-data-unit module and injection module. SMAT provides data source modules for all AOS transactions. As the structure of routing transaction is extremely complex and this transaction involves communication between multi-system and multi-satellites, it is rarely used in practice.

3.2 Router Model

Each node of the router contains four ports: Data input port “In”, Data output port “Out”, Node-state port “Status” for the situation when sending data from one node to another node and the port “Busy” to set the node refuses to accept data.

The module “Nod_In” receive data-input-request, if the output node being requested is servicing for the other input nodes, then return “Busy” flag, do not discard the data, enter the waiting service sequence， until the data is transferred then return “Finished” flag; If the corresponding FIFOBuffer of the output node has overflowed, then discard the data and return “Overflow” flag; Or else send the data to the corresponding module “Nod_Out”, set the flag to “Busy”, and then return “Finished” flag until receive the "No Busy" signal.

The module “Nod_Out” transmitted out the data it has received, and then send “No Busy” signal to the module “Nod_In”.

The Module "Error" produce interference signal according to the rules have been given, if detected signal “Noise” during data transfer process, then re-send the data and record the length of the re-send data simultaneously, to test fault tolerance ability of the router link-layer.

3.3 TMU Model

The Module “TMU ” collect telemetry data from each load, and then pack the data unit（E－SDU）, is arranged by byte and is not delimit by CCSDS structure ,into statute data unit E－PDU (CCSDS package) by fixed position . And then multiplexing with the other CCSDS packages to produce M-PDU package. And simultaneously extract data in the required Sampling rate, generate delayed telemetry data. For the data required real-time transmission, the “TMU” combined them into Insert data to generate IN-SDU.

In addition to the functions mentioned above, the module “TMU” also distributed telemetry command (Telecommand) to the corresponding load units. In this experiment, "TC Generator" replaced the actual uplink telecommand based on the ground control rules and produce regular telecommand.

3.4 Multiplexer Model

The Multiplexer schedule the VCDU data units of all virtual channels，add current IN-SDU header and VCDU header，in this way generate the VCDU. If there is none of the VCDU data units in the virtual channels, we should fill “0” in the VCDU data units and generate a leisure data frame. Coding the VCDU or leisure data frame, we can finally generate the PCA_PDU to be sent.

4. Experiment Scheme

Models of the system are all achieved by SMAT simulation tool. And the models are placed in the model library. Different simulation systems can be constructed according to the purposes of experiments. In this paper, the simulation system is composed by the data transmitter and data receiver. Before running the simulation model there are the following work needed to carry out:

4.1 Verification of model accuracy

The verification of the correctness of the simulation model is to test the model constructed which can really represent the basic performance of a real system (or the system designed) or not. The correctness validation process for the simulation model is a process of repeated comparisons between the model and practical system. And the difference generated by comparison is used to improve and modify the model in order to make the model gradually approach to the actual system. The process will not stop until the simulation model is recognized as the true representative of the real system.

4.2 Confirmation of Data Payload

In our simulation experiment, the primary objective is to add data payload to the model. Since the amount of data sources is huge, we pick up several transaction data to be the object in channel combiner according to the features of AOS and the need of our research.

The table below shows the data sources employing AOS structure.

N.o	Data source	Service type	Affiliated	Data length (Bytes)	Speed (Mbit/s)	Distribution (s)	Virtual channel
1	Camera	B_PDU	RTU2	—	150		2
2	Env_Science		RTU3	—	50		2
3	Mat_Science	VCA_SDU	RTU4	—	30		3
4	Life_Science		RTU5	—	20		3
5	TV Video*3	B_PDU	RTU6	—	2.048		4，5，6
6	EngineeringTM	Package、multi duplicate	AOC	131934	-	4	0
7	Audio		RTU6	15360	-	2	0
8	Delay TM		ALL	Sampling	-		1
9	TM*7		ALL	40	-	0.512	0
10	Power		AOC	80	-	0.512	0
11	Self_manage		RTU1	10	-	0.512	0
12	Insert	IN_SDU		64	-	-	0

4.3 Simulation Model Performance Statistical Index Architecture

The final statistical indicators of performance extracting from our simulation experiment are composed with:

- Data transmitting availability of downlink channel
- Data transmitting immediacy
- Data capacity
- Data transmitting fault tolerance

4.4 Confirmation of Experiment Scheme

In this experiment, the model designed on the system's every function is followed AOS system architecture's function and target. The algorithm of the virtual channel schedule is a key points of the AOS channel combiner. A reasonable schedule algorithm is a assurance that AOS channel combiner will finish each complex task in order. The efficiency of the algorithm will make great affect on the system. In the modeling course, we compared several schedule and finally take the schedule of priority and overtime, which take the longest delay time as the standard of the transfer in real time.

We analyze downlink channel data transmission availability, data delay, data load capacity and data effective transmission rate with interference, which are captured in our experiment. Then we compare our results with the simulation outputs taken from conversional satellite orbiting system, while we can see the performance is improved in many aspects. Furthermore, utilizing the SpaceWire routers and without high speed CPU (DSP), it is feasible to break through the restriction of data stream, realize the design plan of AOS system, and improve the fault tolerance in the link layer.

5. Summary

We in this paper suggest a method to implement AOS system with SpaceWire routers, and make simulation experiment using real data source with our designed system model. The experiment shows the bottleneck in AOS system (low throughput capacity and low CPU speed) can be eliminated with the deployment of SpaceWire routers. By adjusting the topology structure, we can get system performances with different topology and same data payload, and thus optimize the system design.

6. Reference

[1] Parkes S, Rosello J, SpaceWire-Links, nodes, routers and networks, DASIA 2001 -Data Systems in Aerospace, Proceedings of the Conference, 2001.

[2] R Amini, E Gill, G Gaydadjiev, The Challenges of Intra-Spacecraft Wireless Data Interfacing, 57th International Astronautical Congress.

[3] W.J. Dally and C. Seitz, "The Torus Routing Chip", Distributed Computing, vol. 1, no. 3, 1986.

[4] SM Parkes, P Armbruster, SpaceWire: a spacecraft onboard network for real-time communications, Real Time Conf., 2005.

[5] C. McClements, S.M. Parkes, and A. Leon, “The SpaceWire CODEC,” International SpaceWire Seminar, ESTEC Noordwijk, The Netherlands, November 2003.