

## Hardware Reconfiguration and Simulation of Satellite Attitude Control System

Menglin Sheng<sup>1</sup>, Sanghai Guan<sup>1</sup>, Shihua Yu<sup>2</sup>, Chunyu Cheng<sup>1</sup>, Peng Sun<sup>1</sup> and Kaiyu Wang<sup>1+</sup>

<sup>1</sup> Electrical and Information Engineering, Dalian University of Technology, Dalian, 116024, China

<sup>2</sup> Computer Science Technology College, Hulunbeier University, Hunlunbeier, 021008, China

**Abstract.** A mathematical model is established for implementing hardware reconfiguration and simulation of an attitude control system for three-axis stabilized satellite. The satellite attitude quaternion, dynamics equation and kinematics equation are defined and a closed-loop system used for attitude control is designed. Computer simulation with MATLAB and FPGA instead of space borne equipment is realized, using FPGA for the host controller and some peripheral modules. The hardware simulation results are good, proving the feasibility of the scheme. Adding hardware reconfiguration into the design provides a new way for FPGA simulation of satellite attitude control system.

**Keywords:** satellite attitude control, PID, FPGA, serial communication, dynamic reconfiguration.

### 1. Introduction

A satellite, irrespective of its size and function, orbits with certain attitude and speed in space, and must be able to quickly adjust its posture as needed to complete different tasks. Therefore, the satellite attitude control system plays a very important role in the whole satellite system. Most of the existing satellite attitude control systems are software-based with an on-board computer. Once hardware malfunction happens, backup computers are needed to take control. Using FPGA instead of computers for satellite attitude controlling can help to reduce the load of the host processor [1], improve system performance, and reduce system size, weight, and cost as well. With the increasing maturity of technology, FPGA's performance is substantially improved and its resources are more and more rich and efficient, they provide a reliable hardware environment for system design. In addition, the attitude control system upgrade is very convenient with FPGA dynamic reconfiguration. By generating different reconstruction backups, FPGA can configure appropriate backups and finish system upgrade when tracking system operation states at the same time.

### 2. Modeling of Three-Axis Stabilized Satellite

The main structure of a three-axis stabilized satellite includes dynamics module, kinematics module, attitude determination module, quaternion differential solving module, host controller module, power module, and so on. These modules form a closed-loop system, as shown in Figure 1.

In order to accurately describe the indicators and conduct satellite attitude control, an attitude control system mathematical modelling is needed. This modelling includes establishing a coordinate system, determining a description method of the satellite attitude and describing the movement of the satellite. Common coordinate systems include geocentric inertial coordinate system, orbit coordinate system and local coordinate system [2]. Quaternions are used to represent the satellite attitude. A quaternion is a four-dimensional parameter including both scalar part and vector part [3], which is defined as formula (1).

---

<sup>+</sup> Corresponding author. Tel.: + 041184708788; fax: + 041184708788.  
E-mail address: wkaiyu@dlut.edu.cn.

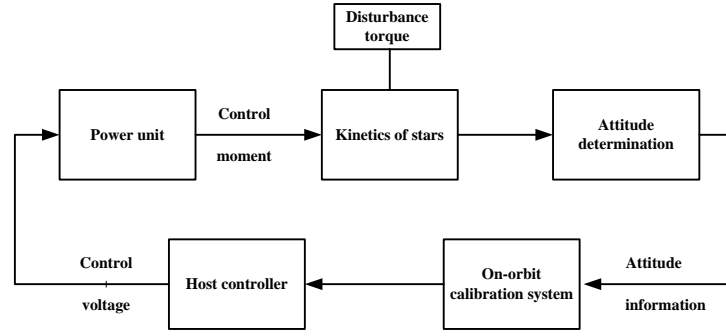


Fig. 1: Overall system block diagram.

$$q = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} \cos(\phi/2) \\ e_x \sin(\phi/2) \\ e_y \sin(\phi/2) \\ e_z \sin(\phi/2) \end{bmatrix} = \begin{bmatrix} \cos(\phi/2) \\ e \sin(\phi/2) \end{bmatrix} \quad (1)$$

where  $\phi$  is the Euler angle of rotation  $e_x, e_y, e_z$ , are the Euler orientation cosines on x, y, z-axis (orbit coordinates), respectively. These four parameters satisfy the orthogonality constraint equation [4], as defined in formula (2).

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \quad (2)$$

A quaternion can be written as formula (3) as well.

$$q = q_0 + q_1i + q_2j + q_3k = q_0 + q_v \quad (3)$$

where  $q_0$  is the scalar part and  $q_v$  is the vector part, they constitute an attitude quaternion.

$\otimes$  represents the quaternion multiplication, which is defined as formula (4).

$$q \otimes p = [q]p \quad (4)$$

where

$$[q] = \begin{bmatrix} q_0 & -q_v^T \\ q_v & q_0 I_3 + [q_v \times] \end{bmatrix} \quad (5)$$

$$[q_v \times] = \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix} \quad (6)$$

Converting orbit coordinates to satellite local coordinates:

$$\begin{aligned} R_b &= \left[ q_v q_v^T + (q_0 I - [q_v \times])^2 \right] R_o \\ &= \begin{bmatrix} q_1^2 + q_0^2 - q_3^2 - q_2^2 & 2(q_1 q_2 + q_0 q_3) & 2(q_1 q_3 - q_0 q_2) \\ 2(q_1 q_2 - q_0 q_3) & q_1^2 + q_0^2 - q_3^2 - q_1^2 & 2(q_2 q_3 + q_0 q_1) \\ 2(q_1 q_3 + q_0 q_2) & 2(q_2 q_3 - q_0 q_1) & q_3^2 + q_0^2 - q_1^2 - q_2^2 \end{bmatrix} R_o \\ &= C_o^b R_o \end{aligned} \quad (7)$$

The astral motion model is made up of kinematics equations and dynamics equations.

Let  $q_o^b$  represent the attitude quaternion in satellite local coordinate system relative to the orbit coordinate system, the kinematics equation of the satellite attitude can be defined as formula (8).

$$q_o^b = \frac{1}{2} q_o^b \otimes \omega_{ob}^b = \frac{1}{2} q_o^b \otimes (\omega_{ib}^b - C_o^b \omega_{io}^o) \quad (8)$$

where  $\omega_{ob}^b$  is the angular velocity in local coordinate system relative to orbit coordinate system;  $\omega_{ib}^b$  is the angular velocity in local coordinate system relative to geocentric inertial coordinate system;  $\omega_{io}^o$  is the angular velocity in orbit coordinate system relative to geocentric inertial coordinate system;  $C_o^b$  is the transformation matrix from orbit coordinate system to local coordinate system.

The kinematics equation can be deduced from the theorem of moment of momentum [5]. The theorem states that for a point in inertial space, the change rate of its angular momentum equals to the sum of all torques act on it, which is:

$$\frac{dH}{dt} = T \quad (9)$$

where  $H$  is the angular momentum of the satellite relative to geocentric inertial coordinate system,  $T$  is the total torque. Generating formula (9) into satellite local coordinate system, the dynamics Euler equation can be get:

$$\frac{d_I H}{dt} = \frac{d_B H}{dt} + [\omega \times] H = T \quad (10)$$

where  $T$  is the total torque in local coordinate system,  $H$  is the angular momentum in local coordinate system.

$H = I\omega$ , where  $I$  is a parameter matrix of moments of inertia:

$$I = \begin{bmatrix} I_x & -I_{xy} & -I_{xz} \\ -I_{xy} & I_y & -I_{yz} \\ -I_{xz} & -I_{yz} & I_z \end{bmatrix} \quad (11)$$

In this paper, the attitude determination uses both a star sensor and a gyroscope. The direction of starlight in star sensor coordinate system is shown below:

$$S^s = \frac{1}{\sqrt{p_x^2 + p_y^2 + f^2}} \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} \quad (12)$$

where  $p_x$  and  $p_y$  are starlight images on CCD surface,  $f$  is the focal length. When satellite local coordinate system is coincident with star sensor coordinate system, the observational equation (on the earth) of satellite local coordinate system is:

$$S^b = S^s = NS^I + \Delta S \quad (13)$$

where  $S^I$  is the unit direction vector of starlight which can be inquired in the ephemeris;  $N$  is the transformation matrix from inertial coordinate system to satellite local coordinate system;  $\Delta S$  can be approximated as Gaussian noise, which represents the measurement error of the star sensor and meets the following criteria:

$$\begin{aligned} E\{\Delta S\} &= 0 \\ E\{\Delta S \Delta S^T\} &= \sigma^2 \left[ I_{3 \times 3} - (AS^I)(AS^I)^T \right] \end{aligned} \quad (14)$$

Gyroscope is another attitude determination device used. Its speed model is described as formula (15).

$$\omega_g(t) = \omega(t) + \Delta\omega(t) \quad (15)$$

where  $\Delta\omega(t)$  is the measurement error, including constant drift, white noise, etc.

The power unit is made up of reaction flywheels. Three flywheels are installed along three orthogonal axes respectively. The satellite attitude control is realized by the angular momentum exchange between the flywheels and the satellite [6-8].

The host controller is the most important part of satellite attitude control system, whose performance directly affects the performance of the whole system. PID controller, which is one of the classic controllers, is chosen in this paper [9]. The deviation amount is calculated based on the actual input and output, which then applied to determine the control variables [10]. The relationship is shown as below:

$$error(t) = rin(t) - yout(t) \quad (16)$$

The transfer function equation of PID control rate is defined as formula (17):

$$G(s) = \frac{U(s)}{E(s)} = k_p error(t) + k_d \frac{derror(t)}{dt} + k_i \int_0^t error(t) dt \quad (17)$$

where  $k_p$  is the scale factor,  $k_i$  is the integration time constant and  $k_d$  is the derivative time constant. By adjusting these three parameters, the attitude control can be achieved. If the satellite quaternion changes from an initial value  $q$  to a set value  $q_c$  and the change value is  $q_e$ , the relationship between them is as following:

$$q = q_c \otimes q_e \quad (18)$$

If the angular velocity changes from an initial value  $\omega$  to a set value  $\omega_c$  and the change value is  $\omega_e$ , the relationship between them is:

$$\omega = \omega_c + \omega_e \quad (19)$$

The power unit is made up of three flywheels installed along three orthogonal axes, so:

$$H = I\omega + Ch \quad (20)$$

where  $C = I_{3 \times 3}$  is the flywheel matrix;  $h$  is the flywheel control torque. Generating formula (18), (19) and (20) into formula (8) and (10), the satellite attitude control model based on the change values can be obtained:

$$q_e = \frac{1}{2} q_e \otimes \omega_e \quad (21)$$

$$I\dot{\omega}_e + I\dot{\omega}_c + [\omega_e \times]H + [\omega_c \times]H = -u + M \quad (22)$$

where  $I$  is the parameter matrix of moments of inertia,  $u = Ch$ ,  $M$  is the torque caused by outside disturbance.

Therefore, the PID control rate of the attitude control is:

$$u = -I\dot{\omega} - [\omega_e \times]H - [\omega_c \times]H + k_d\omega_e + k_q q_e \quad (23)$$

According to Lyapunov stability analysis, nonlinear term  $(-I\dot{\omega}_c - [\omega_c \times]H)$  can be regarded as interference which can be removed and  $[\omega_e \times]H$  has no effect on the stability, so the PID control rate can be rewritten as:

$$u = k_d\omega_e + k_q q_e + k_g \int q_e dt \quad (24)$$

The integral term increases the complexity of calculation while there is not much effect if remove it. The resulting PID control rate is described as formula (25):

$$u = k_d \omega_e + k_q q_e \quad (25)$$

### 3. Hardware Implementation of the Attitude Control System

Verilog HDL programming is used for the above PID control algorithm with the host controller module realized on SPARTAN-3E FPGA development board and other modules realized by building simulation modules using MATLAB/Simulink. RS232 serial interface is used for communication which makes MATLAB-FPGA co-simulation possible. The simulation block diagram is shown in Figure 2.

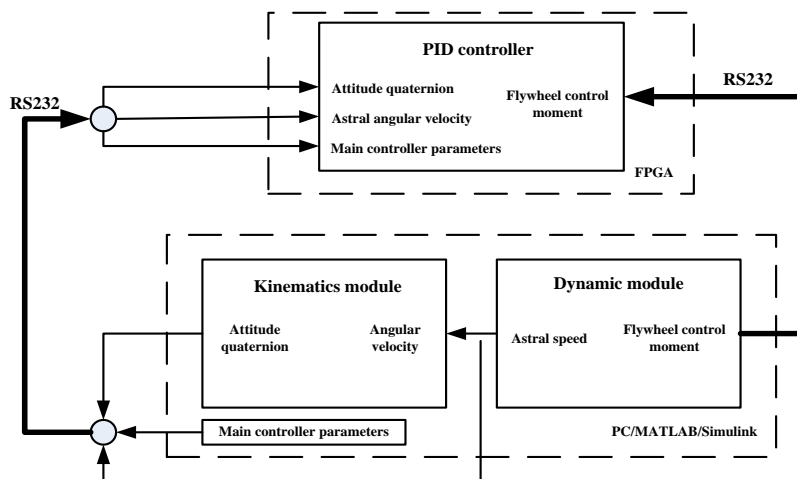


Fig. 2: Co-simulation system design diagram.

Which can be seen in the above diagram, three kinds of input parameters are needed in each operation cycle of the FPGA: the vector part of the quaternion  $(q_1, q_2, q_3)$ , the angular velocity  $(\omega_1, \omega_2, \omega_3)$ , and the host controller parameters  $k_p$  and  $k_d$ , while the output is the calculated torque  $T(t_1, t_2, t_3)$ . To facilitate the processing, all data are converted to 32-bit signed integers and inputted to the FPGA through RS232 serial interface. Data conversion and distribution modules, which are called the input/output buffers, are needed between the serial interface and the controller because RS232 serial interface can only receive 8-bit-wide data. The host controller which performs the attitude operation is connected in the middle of the input and output buffers. In addition, synchronizing signals are added and set to the rising edge. Actions of the latter module are triggered by synchronizing signals from the former, which help to retain the sequence of streaming data. The overall hardware design is as shown in Figure 3.

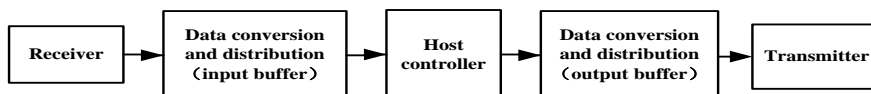


Fig. 3: The overall design diagram of the hardware part.

Internal programs of the receiver and the transmitter are designed according to the UART communication protocol. 115200 baud rate is used in this paper, and the 50M system clock is divided by 27 to obtain the detection signal whose frequency is 16 times the baud rate.

Whenever the input buffer detects a synchronizing signal from the receiver, it starts to store and combine the input 8-bit data until the data are enough to be calculated. Then the input buffer will output a synchronizing signal and send the combined data to the host controller. On the other side, the output buffer splits the calculated results received from the host controller into sequential 8-bit data and send them to the transmitter.

The host controller calculates according to formula (25). Six 32-bit signed multiplications and three 32-bit signed additions are needed to be instantiated, because MATLAB/Simulink is able to identify at most 32-

bit signed numbers, including one sign bit and lower 31 bits representing the number. The input buffer, the host controller and the output buffer are connected and packaged together as a processing module. The receiver, the transmitter, the frequency module and the processing module are connected and instantiated in the top-level module.

#### 4. MATLAB/Simulink-FPGA Co-simulation

An attitude control system for three-axis stabilized satellite is built which using FPGA to realize host controller module and MATLAB/Simulink to build dynamics module, kinematics module, attitude determination module, quaternion differential solving module, power module, and so on. Receiver and transmitter modules enable Simulink and FPGA to communicate properly; Data type conversion module converts input data to integers; Rate conversion module ensures that the baud speed of Simulink matches that of FPGA. All the above modules form a closed-loop system where FPGA and MATLAB/Simulink are connected in series.

In the MATLAB/Simulink-FPGA co-simulation system, the simulation conditions are set as follows: the initial attitude angle of the satellite is  $\text{fei0}=[0, 42.7, 5.6]^*\text{dtr}$  (where dtr is the conversion coefficient between angle and radian, the same as below); the target attitude angle is  $\text{fei1}=[10, 30, 60]^*\text{dtr}$ ; the initial angular velocity is  $\text{Ww0}=[0.1 \ 0.1 \ 0.1]^*\text{dtr}$ ; the gain of the power unit is 1; the host controller parameters are determined as  $k_d=0.08$  and  $k_p=500$  based on the research results; the simulation time is 25s; the simulation algorithm is the default algorithm; the baud rate is 115200 Baud, the sampling time of the receiver is 0.00001s and other sampling times are 0.0001s; the data type is int32; the data format is [1,3] and previous data is continuing to use when no data receiving. The simulation results are shown in Figure 4, 5, 6, 7 and 8.

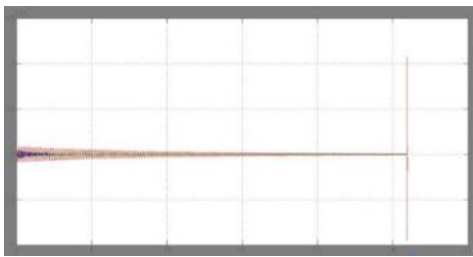


Fig. 4: The simulation results of the moment.

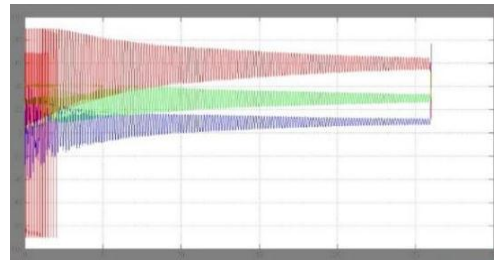


Fig. 5: The simulation results of the attitude Angle.

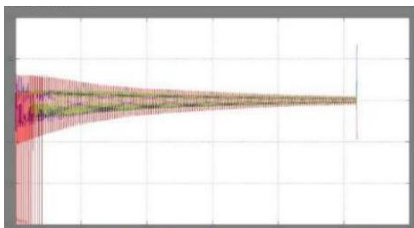


Fig. 6: The simulation results of the error of attitude Angle.

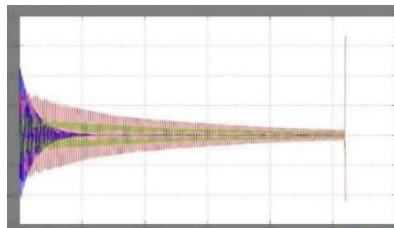


Fig. 7: The simulation results of the angular velocity.



Fig. 8: The Enlarge figure of the error of attitude Angle.

With the simulation time increasing, the torque applied by the power unit, the attitude angle error and the angular velocity decrease gradually and tends to zero, while the attitude angle gradually approaches the target, which can be seen from Figure 4, 5, 6 and 7. Figure 8 is an enlarged view of Figure 6, which shows that the range of the attitude angle error is decreasing and less than 6 as time increasing to 25s which achieves the desired result. The simulation of the attitude control system is proved to be successful.

#### 5. Reconfigurable Analysis

Reconstruction is to generate IP cores for designed files, where the IP cores are called reconstruction backups. The hardware reconstruction is realized by calling different reconstruction backups. In this paper, the key parameters include host controller parameters and power unit parameters. Different reconstruction backups are generated with different system parameters and these backups are able to be retrieved based on

different system demands. Furthermore, the entire algorithms can be redesigned and reconstruction backups are generated when the system performance improved by a new algorithm. The system can upgrade easily by calling new backups.

## 6. Conclusion

In this paper, a mathematical modelling of satellite attitude is completed and an attitude control system for three-axis stabilized satellite is built using PID controller and MATLAB/Simulink. Verilog HDL programming is used for the PID control algorithm and RS232 serial interface is used for FPGA communication which achieve MATLAB-FPGA co-simulation. By observing the simulation waveforms, it can be found that with the simulation time increasing, the torque applied by the power unit, the attitude angle error and the angular velocity decrease gradually and tends to zero, while the attitude angle gradually approaches the target. The range of the attitude angle error is also decreasing and less than 6 as time increasing to 25s which is a very good result. FPGA is proved to be effective when using for satellite attitude control. Moreover, the reconstruction idea is added to the system design and the feasibility of reconstruction is discussed.

## 7. Acknowledgements

This work was supported by the National Natural Science Foundation of China (No. 61003175), “the Fundamental Research Funds for the Central Universities” of China and the scientific research project of the Inner Mongolia Autonomous Region colleges and Universities (No. NJZY13325).

## 8. References

- [1] Dai Yan. Satellite attitude control system design and simulation based on FPGA [D]. Harbin: Harbin Institute of Technology, 2007.
- [2] Yang Feng. A study of attitude determination and control system (ADCS) of Three-axis stabilized satellite [D]. Xi'an: Northwestern Polytechnical University, 2006.
- [3] Xu Fan. Research on attitude control system and simulation of a certain satellite [D]. Harbin: Harbin Institute of Technology, 2006.
- [4] Lv Jianting. Attitude control algorithms research for three-axis stabilized satellite [D]. Harbin: Harbin Institute of Technology, 2007.
- [5] Cao Yang. Research on attitude determination and control system of three-axis stabilized satellite [D]. Harbin: Harbin Engineering University, 2012.
- [6] Feng Yushu. Simulation of staring imaging attitude tracking finite time control of TV satellite [D]. Changsha: National University of Defense and Technology, 2015.
- [7] Jiang Rui, Wei Jiaolong, and Cen Chaohui. Simulation modeling of satellite attitude control system based on quaternion feedback [J]. *Journal of System Simulation*, 2009, **21**(19): 6260-6265.
- [8] Wang Ningqiang, and Liu Xiangdong. Research on hardware in the loop simulation for satellite attitude control system [J]. *Computer Simulation*, 2005, **22**(10):78-81.
- [9] Qian Yong. Studies on the attitude determination and control system of high precision three-axis stabilized satellite [D]. Changsha: Northwestern Polytechnical University, 2002.
- [10] Liu Jinkun. *Advanced PID control and matlab simulation* [M]. Beijing: Electronic Industry Press, 2004.