Research Article

Design of Motion Controller for Satellite-Borne Data Transmission Antenna

Lijiao Li [iD],[1] Ran Shi,[2] Xiaohui Zhang,[2] Jiyu Ma,[2] and Jun Zhang [iD][1]

[1]*College of Information Science and Engineering, Central South University, Changcha, Hunan 410083, China*
[2]*Shanghai Aerospace Control Technology Institute and Shanghai Engineering Research Center of Inertia, Shanghai 201109, China*

Correspondence should be addressed to Jun Zhang; junzhang@csu.edu.cn

In order to realize stable, reliable, and high-precision motion control of the satellite-borne data transmission antenna, a FPGA-based motion controller is designed and achieved. The controller receives commands through the RS-422 asynchronous transmission serial port and performs speed planning autonomously. The controller also performs closed-loop position control with the collected resolver angle and controls two antennas synchronously or independently in a subdivision driving method. The controller is firstly designed using the hardware description language VHDL, simulated in ModelSim software. Then, it is connected to the stepping motors through the LMD18200H bridge chip by using aerospace-grade FPGA, controlling the data transmission antennas to regulate the angle, go to zero-position, or stop rotation. The simulation and experiment results show that the design can control the antennas accurately and stably. The accuracy of angle regulation reaches 0.0085° under the condition of 64 subdivisions and 100 : 1 reduction ratio.

1. Introduction

The satellite-borne data transmission antenna is a passive antenna, which is mainly used for communication between remote sensing satellites and ground stations. When the satellite is normally orbited, the data transmission antenna expands according to the overall requirements of the satellite. The antenna receives control commands, makes motion to realize beam pointing, and completes the task of transferring data to the ground stations [1]. As the key technology of the whole system, the motion control plays a decisive role in the stable and high-precision operation of the mechanism.

PID control is the classic algorithm in the antenna control system; Hui-Feng and Jian-Jiang proposed a proportional-fussy-proportional integral controller, which reduced system overshoot and improved the dynamic performance of the system [2]; Ya-ting et al. designed an adaptive fuzzy PI controller, which had the advantages of fast system response, strong antiload interference capability, and good robustness [3]. But in general, the PID control algorithm relies on the set of three parameters of proportional, integral, and differ-

ential. The system gradually stabilizes in the process of fluctuation; the overshoot is inevitable. Another control method that is commonly used now is to use a stepping motor and a reducer to control the mechanism by means of subdivision driving. Yu-tian and Wei-dong proposed a microstepping mode antenna rotation control technology, focusing on the principle of subdivision control and its implementation method [4]; Wen-hui proposed a structural design scheme for a biaxial antenna pointing mechanism, mainly for the feeder structure and antenna mechanism design [5]; Yong and Jian considered the nonlinear factors of the drive circuit, modeled and simulated for the mechanism, and obtained the pointing precision, torque fluctuation, speed fluctuation, and single-step response of the antenna mechanism [6].

Based on the above research, this paper designs and implements a motion controller which is suitable for the satellite-borne data transmission antenna. The controller takes the field programmable logic device FPGA as the core; its logic function is written with the hardware description language VHDL. The controller receives commands from
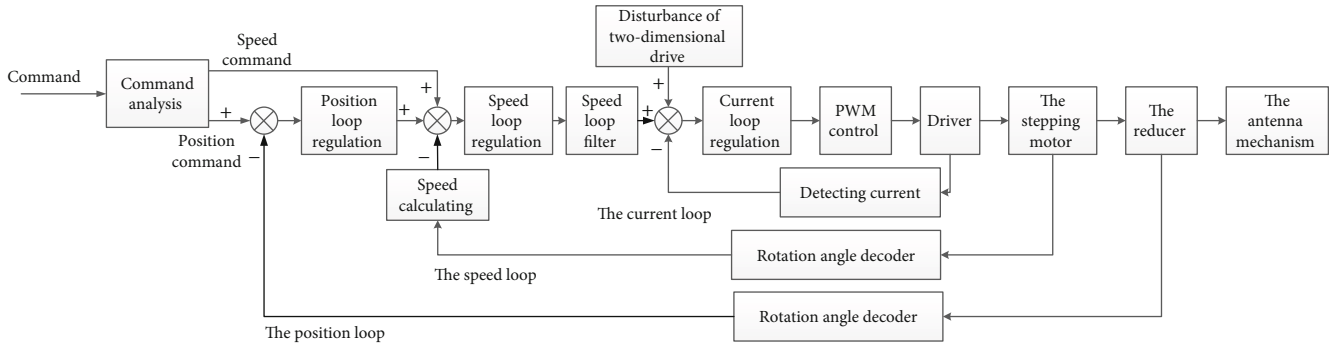
FIGURE 1: Block diagram of the motion controller.

the host computer through the RS-422 serial port and performs closed-loop motion control with the collected resolver angle. The design is simulated in ModelSim and verified on the FPGA development board, stepping motor, and related peripheral circuits.

2. System Requirements and Analysis

The satellite-borne data transmission antennas generally work in space, but the electromagnetic and radiation environments in space are relatively harsh. When high-energy particles enter the sensitive area of the semiconductor device, the logic of the device is reversed, resulting in incorrect calculation results, abnormal operation, and even catastrophic accident. Therefore, the aerospace-grade antifuse FPGA device is selected as the core board, which can effectively avoid the single-particle inversion effect.

There are three types of motors for aerospace vehicles: brush DC motor, brushless DC motor, and stepping motor. The brush DC motor has good starting performance, but there is a danger of sparking due to the brush and the commutator. The brushless DC motor has good dynamic performance, but the electronic circuit of the driver is complicated, and its weight, size, and power consumption are large. The stepping motor is an open-loop control element, driving a step angle every time an electric pulse signal is received. The stepping motor is adopted because of its advantages of open-loop control, high precision, holding torque when stopping rotation, and strong anti-interference [7]. This design chooses a two-phase hybrid stepping motor with the step angle of 1.8° and a take-off function.

When the stepping motor operates in a single step, there may be problems such as out-of-step, overshoot, large torque fluctuation, and severe low-frequency oscillation [8]. For this reason, the controller adopts the scheme of autonomous trapezoidal speed planning; acceleration and take-off/brake speed can be set, which can be used to obtain the optimal speed curve during the experiment and reduce the risk of out-of-step and overshoot. At the same time, the controller drives the stepping motor in the 64 subdivision method, which not only improves the accuracy of angle regulation when in go to zero-position but also slows down the vibration and reduces the running noise [9]. Outside the controller, a 100 : 1 reduction ratio reducer is used to increase the output

torque. In this condition, the minimum step angle of the mechanism is 1.8°/64/100 = 0.00028125°.

In order to ensure the reliability of the design, a closed-loop position control based on the resolver angle is designed. After the angle regulation command is completed, the current position is compared with the collected resolver angle. If the values are different, the current position is updated to the resolver angle. And the controller makes speed planning again until the two numbers are equal. At the same time, in order to meet the requirements of aerospace equipment, multiple redundant design techniques are adopted. First, the two-receiving technique is adopted when receiving the commands from the host computer. If the two commands received are the same, one is taken. Otherwise, retransmission is required. Second, the command fault-tolerance design is adopted; the parity bit, the command, and the command parameter error bit are set to detect the command error. Third, the backup for the stepping motor is adopted; the primary one works normally. When the primary fails to work, the standby starts to work. Finally, asynchronous reset and synchronous release are adopted to avoid a metastable state, and the software reset is also adopted.

3. Design of Motion Controller

As shown in Figure 1, the motion controller of the data transmission antenna is mainly composed of the control of the position loop, the speed loop, and the current loop [10]. The position loop completes the angle regulation of the antenna and achieves precise location by detecting the resolver angle. The speed loop mainly plans the speed curve for the antenna and controls the speed according to the speed curve. The current loop is mainly used for the subdivision control of the stepping motor, achieves smooth rotation, and meets the requirements of the motor work current.

The structural diagram is shown in Figure 2. The motion controller consists of a resolver angle acquisition module, RS-422 single-byte receiving module, multibyte receiving module, command decoding module, telemetry information acquisition module, multibyte sending module, RS-422 single-byte sending module, current acquisition module, ADC control module, zero-position signal acquisition module, position acquisition module, position control module, speed control module, step control module, PWM subdivision control module, and PWM pulse control module.
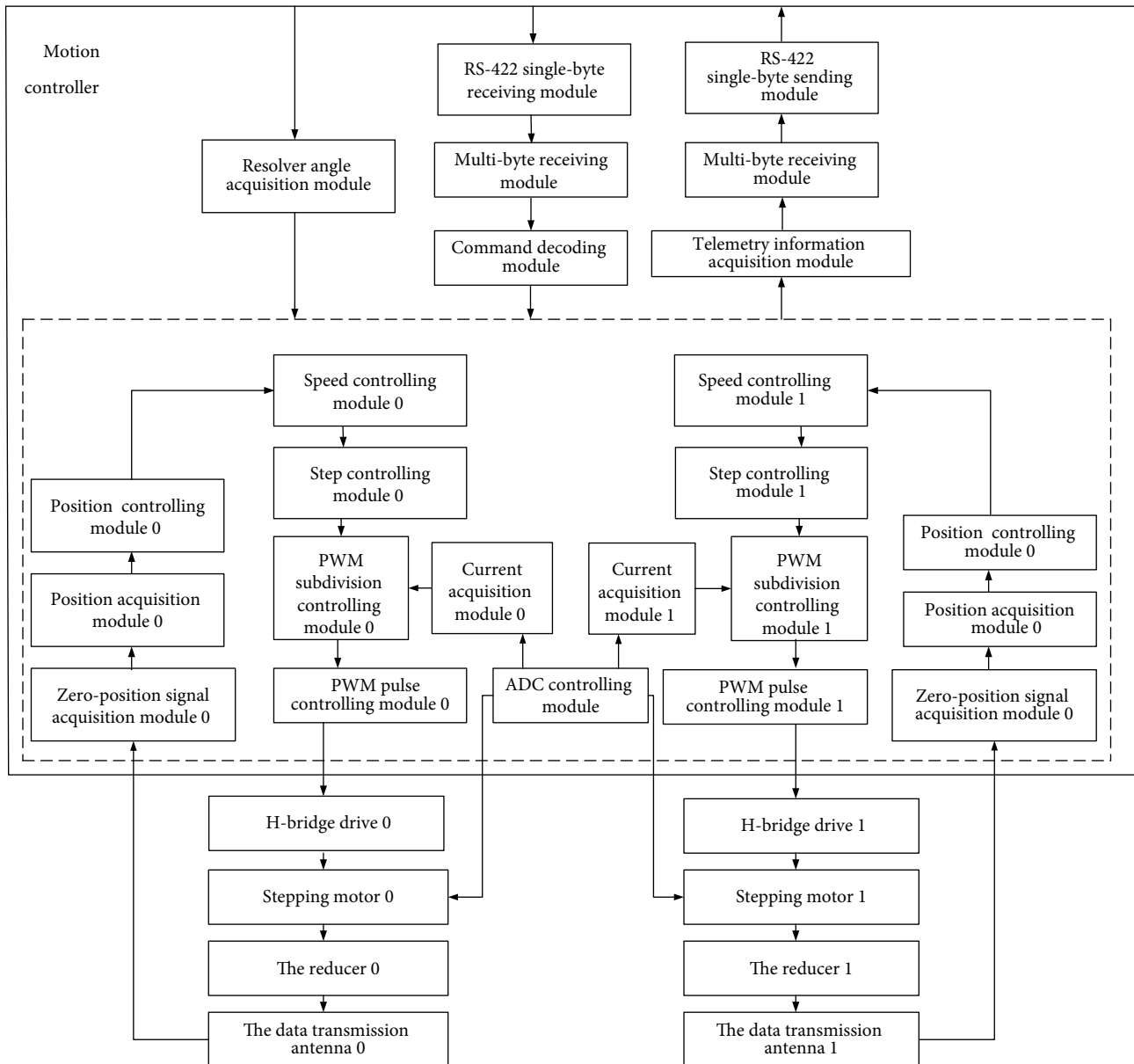
FIGURE 2: Structural diagram of the motion controller.

The processing flow of the motion controller is as follows. The RS-422 single-byte receiving module receives a single byte from the host computer through the RS-422 asynchronous transmission interface. The multibyte receiving module assembles the bytes into command. The command decoding module decodes the command, gets the command parameters and control signals, and sends them to corresponding modules. The resolver angle acquisition module collects the resolver angle of the mechanism and sends it to the position controlling module. The position controlling module receives the position signal of the PWM pulse controlling module and the zero-position signal of the zero-signal acquisition module and updates the internal position register. The speed controlling module plans the speed according to the zero-position signal, command parameters,

and the controlling signals and calculates the time interval of each step. Then, the step controlling module generates the pulse signal according to the time interval. The PWM subdivision controlling module uses the lookup table to obtain the duty value for two phases and calculates the period numbers of the high and low levels. The PWM pulse controlling module controls the output of the PWM pulse according to the LMD18200 timing requirement. The motion controller drives the stepping motor using the driving chip LMD18200 and connects it to the data transmission antennas through the reducer, ultimately realizing the controlling of the data transmission antennas.

3.1. Command Sending, Receiving, and Decoding. The RS-422 single-byte receiving module and the RS-422 single-byte

0	D0	D1	D2	D3	D4	D5	D6	D7	X	1

The start bit   The 8-bit data   The parity bit   The stop bit

FIGURE 3: Format of single-byte data.

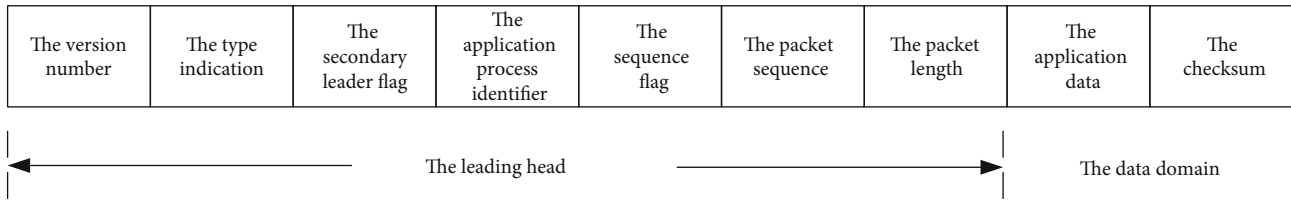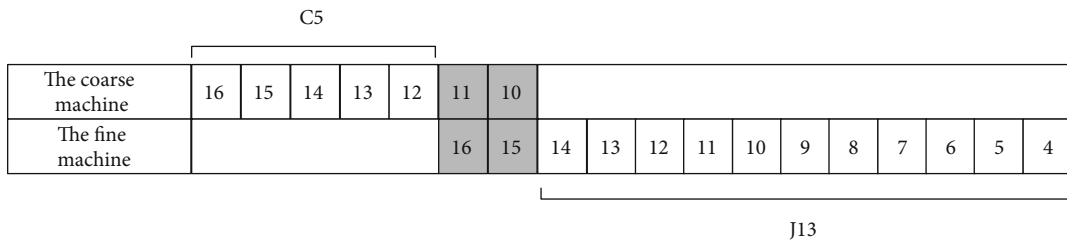sending module transfer data according to the RS422 serial communication protocol; the transmission rate is 115200 bps [11]. The transmission format of single-byte is shown in Figure 3, which includes a 1-bit start bit, an 8-bit data bit, a 1-bit parity bit, and a 1-bit stop bit [12]. The start bit defaults to 0; the check bit is obtained by the odd check method; the stop bit defaults to 1.

When receiving the command, the RS-422 single-byte receiving module receives single-byte data from the RS-422 asynchronous transmission interface. Then, the multibyte receiving module assembles the bytes into the multibyte data packet. When the check code of the data packet is incorrect or the byte is not sent for a long time, the command multibyte receiving module requests a retransmission. After the assembly is successful, the data packet is sent to the command decoding module. The format of the multibyte data packet is shown in Figure 4.

The data packet mainly includes the leading header and the data domain. In the leading header, the version number is set to 000, indicating for source packet. The type indication is 1 for the controlling packet or 0 for the telemetry packet. The secondary leader flag is 1 when there is a secondary header or 0 when there is not. The application process identifier is compiled according to the location of the module and is used for module identification. The sequence flag is set to 11. The packet sequence indicates the serial number of the controlling packet in the relevant packet sequence; the packet length is the length of the controlling packet, which is up to 250 bytes. In the data domain, the application data is 32-bit data, including a 16-bit integer part and a 16-bit fractional part. The checksum is used to perform an XOR check on the application data.

After receiving the multibyte data packet, the command decoding module decodes it according to the format of multibyte data. The commands are mainly controlling commands and debugging commands, wherein the controlling commands include angle regulation command, going to zero-position command, stopping rotation command, and telemetry command. When decoding the command, if it is an angle regulation command, the angle information is extracted and converted into the step numbers. The position register is updated, and the target position information is output to the speed controlling module. If it is a going to zero-position command or stopping rotation command, the go-zero signal or the stop signal is output to the speed controlling module. If it is a telemetry command, the telemetry information collection module initiates an information collection request.

In addition to the above controlling commands, the design also provides the internal debugging commands, including debugging information acquisition command, software reset command, acceleration parameter setting command, acceleration reciprocal setting command, uniform speed setting command during the going to zero-position, the take-off/brake speed setting command, step period calculation cumulative amount setting command, safety current parameter setting command, duty ratio proportional parameter setting command, and current open-loop setting command, which are used to debug parameters such as acceleration, initial accumulated amount, accumulated amount, and safe current. The internal debugging commands are used during the debug phase to obtain optimal parameters for best performance while the system is running.

3.2. Acquisition of Resolver Angle. The resolver angle is stimulated and decoded by the resolver. The stimulate module generates a sine signal by using the Venturi bridge oscillation circuit, and the sine signal is amplified by the MSK2541B. The decoding module decodes the output sine and cosine signal into a digital quantity by the decoding chip RDC19220. The digital quantity is formed to an 18-bit resolver angle by the combination of coarse and fine machines and is sent to the FPGA through the RS422 serial port [13, 14]. The motion control performs the antenna angle servo control according to the resolver angle and realizes the precise positioning of the position and the limit requirement of the mechanism. The resolver angle decoding adopts the combination of coarse and fine machines, 16 bits per machine, and combines the bits into 18-bit code. The synthesis method is shown in Figure 5 [15].

When combining, the 11th and 10th bits of the coarse machine and the 16th and 15th positions of the fine machine are judged, and the C5 (up to 5 digits) of the coarse machine and the J13 (up to 13 digits) of the fine machine are combined into 18 bits. The combined 18-bit number 0x00000~0x3FFFF represents the angle 0~360°. The accuracy of combined angle resolution is $360/2^{18} = 0.00137°$.

3.3. Position Control. In order to prevent inaccurate positioning caused by out-of-step and overshoot, the closed-loop position control based on the resolver angle is designed [16, 17]. During the movement, the position controlling module receives the step signal and updates the internal position register. When the motion is completed, the internal position register is compared with the collected resolver angle. If the values are different, it indicates that the motion does not reach the target position. Then, the internal position register is updated to the resolver angle, and the controller makes speed planning again until two numbers are equal.

At the same time, the position control module receives the zero-position signal of the position acquisition module, which is detected by the zero-position sensor. The zero-position signal is a standard TTL level, generally low level. It turns to the high level when the mechanism reaches the rising edge of the zero position and turns to the low level after the mechanism continues to rotate the scan pulse width by 4.5°. The zero-position signal is deburred by three high levels and three low levels to ensure reliability. That is, when the zero-position signal changes to three consecutive high levels after three consecutive low levels, it is judged as a pre-zero-position signal; when

The version number	The type indication	The secondary leader flag	The application process identifier	The sequence flag	The packet sequence	The packet length	The application data	The checksum

The leading head ← The data domain

FIGURE 4: Format of multibyte data.

	C5																	
The coarse machine	16	15	14	13	12	11	10											
The fine machine						16	15	14	13	12	11	10	9	8	7	6	5	4

J13

FIGURE 5: Combination of coarse and fine machines.

FIGURE 6: Possible speed curve.

the zero-position signal becomes three consecutive low levels after three consecutive high levels, it is judged as a true-zero-position signal. When the true-zero-position signal is received, the internal position register is cleared.

3.4. Speed Control

3.4.1. Speed Planning. The main function of the speed controlling module is to plan the speed according to the received target angle, zero-position signal, or stop-rotating signal and calculate the time interval of each step according to the speed curve. Among them, the speed planning is performed according to the trapezoidal speed curve of the stepping motor. The ideal trapezoidal speed curve is the isosceles trapezoid; that is, the acceleration section is equal to the deceleration section [18]. The possible speed curve is shown in Figure 6.

Figure 6 shows the speed curves for three cases. The description of each case is as follows.

(1) Segment 0A. Assuming that the initial state is the idle state, the go to zero-position command is executed during the forward angle regulation. When executing the first angle regulation command, the motion controller first jumps directly to the take-off speed, then performs the forward acceleration to reach the uniform speed and maintains it. When the uniform speed section passes the number of steps which equal the total steps minus 2 times the acceleration section, the controller enters the deceleration state. The go to zero-position command is received before reaching the target position, and the controller maintains the deceleration state until reaching the brake speed, then maintains the brake speed. When the true-zero-position signal is received, the controller jumps the speed to 0 to realize going to zero-position.
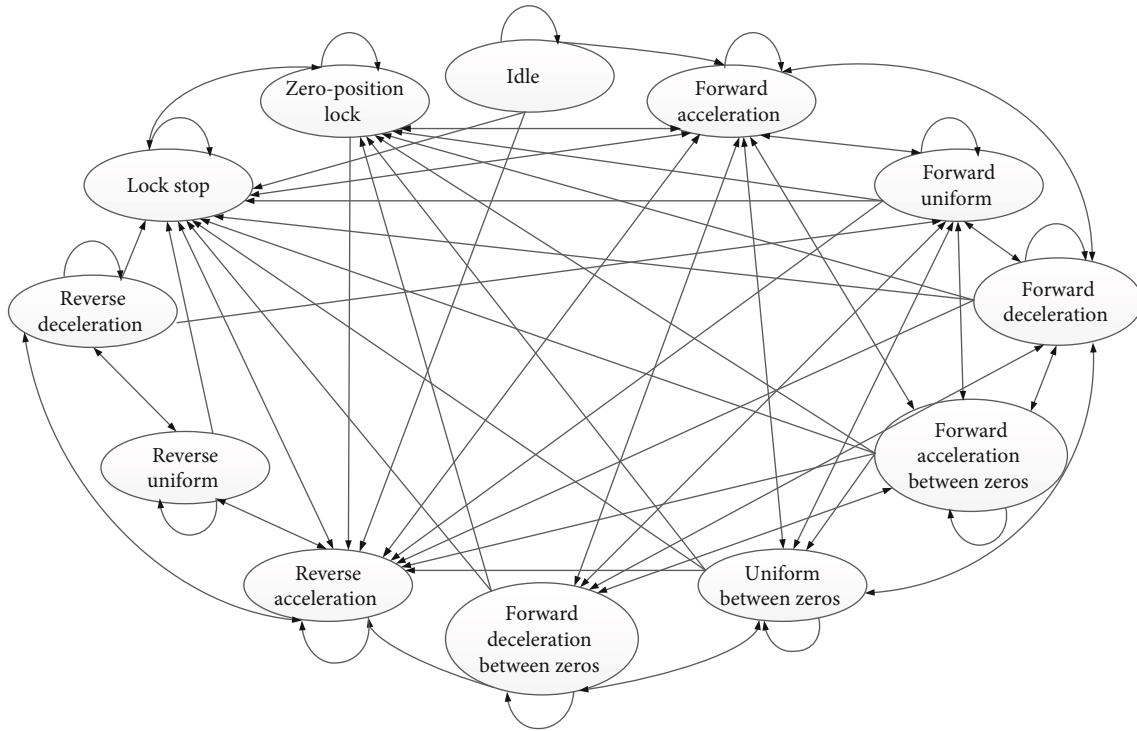
FIGURE 7: State machine of speed planning.

(2) Segment AB. Assuming that the initial state is the idle state, a small angle regulation command is executed. The controller first jumps directly to the take-off speed, then performs the forward acceleration. When the acceleration section passes the number of steps which are half of the total steps and the uniform speed is not reached, the controller enters the deceleration state. The speed is reduced to the brake speed, then jumps to 0, realizing the angle regulation and entering the lock-stop state.

(3) Section BC. Assuming that the initial state is the idle state, a large angle regulation command is executed. The controller first jumps to the take-off speed, then performs the forward acceleration to reach the uniform speed and maintains it. When the uniform speed section passes the number of steps which equal the total steps minus 2 times the acceleration section, the controller enters the deceleration state. The speed is reduced to the brake speed, then jumps to 0. Due to the position closed-loop control, if the collected resolver angle which is the actual position is not inconsistent with the target position at this time, the controller enters the angle regulation again until the actual position is the same as the target position.

Since the stepping motor can be rotated forward and reverse, the received commands have three types, and the new command could overwrite the old command and many other reasons; there are many other possibilities for the speed curve, which are not detailed here.

3.4.2. State Machine of Speed Planning. The speed planning is realized by the state machine. There are 12 states: forward acceleration, forward uniform, forward deceleration, forward acceleration between zero-positions, forward uniform between zero-positions, forward deceleration between zero-positions, reverse acceleration, reverse uniform, reverse deceleration, idle, lock-stop, and zero-position lock. The state machine and its transition are shown in Figure 7.

Since the state machine is complicated, Figure 7 only shows the 12 states and the transition between the states. The transition conditions are omitted, and the double-headed arrows indicate that the states can be converted to each other. The control of the data transmission antenna is mainly angle regulation, going to zero-position, and stopping rotation. The control logic is as follows. The angle regulation command calculates the number of steps according to the target angle and plans the speed according to the trapezoidal curve. When the speed reaches the uniform speed, the controller records the number of steps taken by the acceleration section and then maintains the uniform speed until the number of steps taken by the uniform section is the total number of steps minus 2 times the number of steps taken by the acceleration section and finally performs the deceleration to reach the target position. If the acceleration does not reach the uniform speed but has passed the half of the total number of steps, the controller enters the deceleration state immediately. When the go to zero-position command is received, if the current speed is greater than the brake speed, the controller enters the deceleration state, and if not, enters the acceleration state. When the pre-zero-position signal is received, if the current speed is greater than the brake speed, the controller enters the deceleration state and maintains the uniform speed when the speed is reduced to brake speed. When the true-zero-position signal is received, the speed jumps to 0.
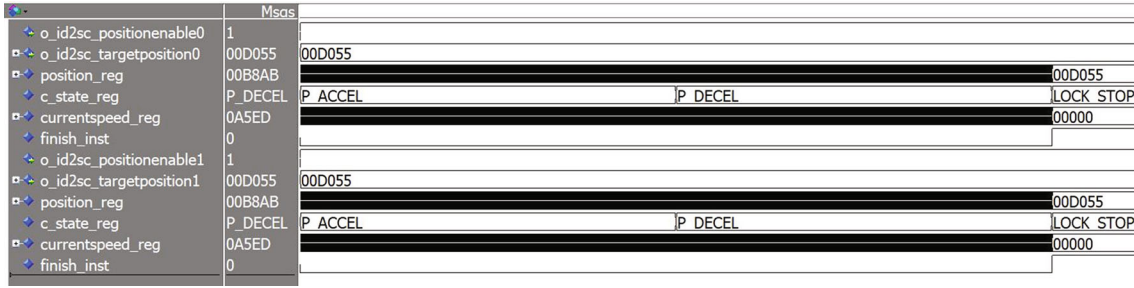
FIGURE 8: Simulation waveform of 15° angle regulation.

The stop rotation command is generally used when the angle regulation command is completed.

3.4.3. Step Interval Calculation. After completing the speed curve planning, the module calculates the time interval of each step based on the speed curve. The speed curve may contain three conditions: uniform acceleration, uniform speed, and uniform deceleration. The time interval can be obtained by calculating the reciprocal of the speed during the uniform. During the uniform acceleration, time interval could be calculated according to kinematic Equation (1) [19]. Equation (1) is transformed into Equation (2) and Equation (3). In the equation, x is the distance, v_0 is the initial speed, a is the acceleration, t is the time, and Δt is the step interval:

$$x = v_0 t + \frac{1}{2} a t^2, \tag{1}$$

$$t = \frac{\sqrt{v_0^2 + 2ax} - v_0}{a}, \tag{2}$$

$$\Delta t = \sqrt{\frac{v_0^2}{a^2} + \frac{2}{a} x} - \sqrt{\frac{v_0^2}{a^2} + \frac{2}{a}(x-1)}. \tag{3}$$

Time t is calculated according to Equation (2), so step interval Δt is calculated as Equation (3). The hardware design is based on Equation (3), including registers such as speed, initial speed accumulated value, speed accumulated value, speed accumulated sum, and speed accumulated sum square root value. Each time the time interval is calculated, only the speed accumulated value and its square root value are calculated, and the time interval is obtained by subtracting the previous calculation result.

3.5. PWM Subdivision and Timing Control. The subdivision control is essential to change the square wave current in the phase coil of the stepping motor to stepped sine wave so that the combined magnetic field rotates at the microstep. In the design, the equivalent area method is used, replacing the sine wave with the rectangular pulse sequences, then calculating the width and interval of each pulse, and storing the data in the ROM table for subdivision driving [20]. Due to the symmetry of the sine wave, only 1/4 sine wave is calculated; the duty cycle change in the four cases of positive increment, positive decrement, reverse increment, and reverse decrement could be obtained with phase change and direction control.

The PWM pulse is driven by the LMD18200 chip. According to the datasheet, the pulse width of the BRK, PWM, and DIR signals must be greater than $1 \mu s$, and the switching of the three signals cannot occur at the same time; it needs to be separated by at least $1 \mu s$. Set the global state machine to control signals and meet timing requirements.

4. Analysis of Simulation and Experiment

In order to debug and test the designed motion controller, a system test platform is written, which simulates the RS-422 interface for command transmission and real-time monitoring of current information. The controller can control two data transmission antennas independently or synchronously, but in order to verify the integrity of the function, the simulation is set to control two antennas synchronously. The simulation is performed in ModelSim software, and the simulation diagrams and results are analyzed as follows.

In the simulation diagram, the positionenable is the angle enable signal, which changes from low level to high level when the angle regulation command is received. The targetposition is the target position register and stores the target position. The position_reg is the current position register and stores the current position. The go_zero is the return-to-zero-position signal; when the go to zero-position command is received, it changes from low level to high level. The rotatedir signal is the rotation direction register. When it is 0, it indicates forward rotation; when it is 1, it indicates reverse rotation. The zero-position signal changes from low level to high level when the pre-zero-position signal is received and changes from high level to low level when the true-zero-position signal is received. The c_state_reg is the current speed status register and indicates the current speed state. The currentspeed_reg is the current speed register, which stores the current speed value. The finish_inst is the command completion flag register; when it is high level, it indicates that the command is completed. The xb_valid_reg is the effective signal of the resolver angle; when it is 1, it indicates that the effective resolver angle is received.

4.1. Simulation

4.1.1. 15° Angle Regulation. The command is set to control two antennas for 15° angle regulation synchronously. The target angle 15 is set as F0000 (hexadecimal number) according to the 16-bit integer 16-bit decimal data format, multiplies by the number of step of 1 degree to get the total
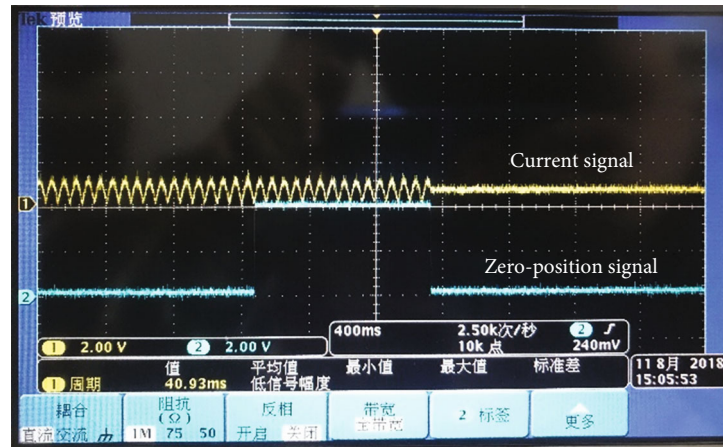
FIGURE 9: Simulation waveform of 235° angle regulation.

FIGURE 10: Simulation waveform of go to zero-position during 60° angle regulation.

FIGURE 11: Simulation waveform of go to zero-position during 60° angle regulation.

FIGURE 12: Stepping motor drive board.

FIGURE 13: Resolver angle acquisition board.

FIGURE 14: Wave of current and zero-position signal.

number of steps D055. The stepping motor 1.8 degrees corresponds to one step, but since the $100:1$ reducer is set and 64 subdivisions are performed for each step, the number of steps corresponding to $1°$ is (decimal number in the formula) $64 * 100/1.8 = 3555.55555$. In order to ensure the accuracy, the above result is saved as DE38E in the format of 12-bit integer plus 8-bit decimal. The total number of steps is calculated as (hexadecimal number in the formula) F0000 $* DE38E = D055520000$; round off 24 decimal places, only the integer part D055 is reserved, which is the total number of steps. The speed value is stored in the form of 1-bit sign bit, 15-digit integer, and 16-bit decimal. The uniform speed is set to 10000 (hexadecimal number), converted to the actual speed of $1°/s$, and the take-off speed is 8000 (hexadecimal number), converted to the actual speed of $0.5°/s$. The following simulation calculation methods are similar and will not be described in detail. The simulation waveform is shown in Figure 8.

It can be seen from the simulation waveform that the initial state is the idle state. When the angle regulation command is received, the positionenable changes from low level to high level, and the controller starts the speed planning. First, it jumps directly to the take-off speed of 8000 and enters the forward acceleration state. When the speed has not reached the uniform speed, but half of the total steps has been passed, the controller enters the deceleration state. After the target position is reached, the finish_inst changes from low level to high level, the angle regulation command is completed, and the lock-stop state is entered.

4.1.2. 235° Angle Regulation. The command is set to control two antennas for $235°$ angle regulation synchronously. The target angle is set to EB0000, and the number of steps is CBFE3. The simulation waveform is shown in Figure 9.

It can be seen from the simulation waveform that the initial state is the idle state. When the angle regulation command is received, the positionenable signal changes from low level to high level, and the controller starts the speed planning. First, it jumps directly to the take-off speed of 8000, enters the forward acceleration state, and accelerates to

reach the uniform speed of 10000 and maintains it. After the final deceleration reaches the target position, the controller stops, and the finish_inst changes from low level to high level, indicating that the angle regulation command is completed.

4.1.3. Go to Zero-Position during 60° Angle Regulation. First, the command is set to control two antennas simultaneously for $60°$ angle regulation, the target angle is set to 3C0000, and the number of steps is 34155. The go to zero-position command and the zero-position signal are given during the angle regulation process, and the simulation waveform is shown in Figure 10.

It can be seen from the simulation waveform that the initial state is the idle state. After the angle regulation command is received, the positionenable changes from low level to high level, and the controller starts the speed planning. First, it jumps directly to the take-off speed of 8000, enters the forward acceleration state, and accelerates to reach the uniform speed of 10000 and maintains it. Then, it maintains the uniform when the go to zero-position command is received and enters the deceleration state between zero-positions when the pre-zero-position signal is received. The speed is reduced to the brake speed of 8000 to maintain the uniform and jumps directly to 0 and remains in the zero-position lock state when the true-zero-position signal is received.

4.1.4. 30° Angle Regulation with Twice Planning. First, the command is set to control two antennas simultaneously for $30°$ angle regulation, the target angle is set to 1E0000, and the number of steps is 1A0AA. After the angle regulation is completed, the resolver angle which is inconsistent with the current position is given, the controller planning speed again. The simulation waveform is shown in Figure 11.

It can be seen from the simulation waveform that the initial state is the idle state. After the angle regulation command is received, the positionenable changes from low level to high level, and the controller starts the speed planning. First, it jumps directly to the take-off speed of 8000, enters the forward acceleration state, and accelerates to reach the uniform speed of 10000 and maintains it. After the final deceleration reaches the target position, the resolver angle is received at
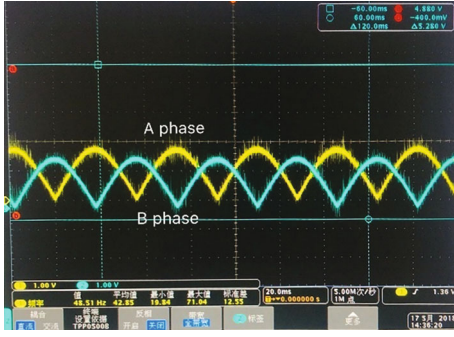
FIGURE 15: Current wave of stepping motor two phases.

the same time, which is inconsistent with the current position, the position_reg value is updated, and the speed planning is performed again. After the completion, the position counter is equal to the target counter, the finish_inst changes from low level to high level, and the controller enters the locked state.

4.2. Experiment. In order to verify the designed motion controller, the hardware circuit system is built, including the FPGA main control board, a current acquisition circuit, resolver angle acquisition board, stepping motor drive board, and stepping motors. And the code is downloaded to FPGA for the experiment. Figure 12 is the stepping motor drive board, Figure 13 is the resolver angle acquisition board, and Figures 14 and 15 are the waveforms measured by an oscilloscope.

Experiment results show that the motion controller can accurately perform angle regulation, go to zero-position, and stop rotation according to the commands. Figure 14 is wave of current and zero-position signal when going to zero-position. When the pre-zero-position signal arrives, the stepping motor decelerates, and the current signal changes. When the true-zero-position signal arrives, the stepping motor locks and the current signal becomes a constant value. Figure 15 shows the current wave of stepping motor two phases during the angle regulation process. Due to the characteristics of the drive chip, the waveform has only the forward portion. It can be seen from the figure that two phases have a phase difference of 90 degrees, and the waveform is close to the standard forward sine wave, indicating that the duty cycle of the subdivided PWM changes according to the sine wave law.

In order to verify the design accuracy, physical tests are performed under the condition of work frequency of 22.1184 MHz, take-off/brake speed of 0.5°/s, and acceleration of 0.05°/s². In the rotation range of 0-270°, the angle regulation test was performed at intervals of 60° in the case of forward rotation and reverse rotation. The test flow is as follows: the angle regulation command is sent through the serial port, and the resolver angle is collected after the mechanism reaches the target position, and the error value of the set speed is calculated. The test data and analysis are shown in Table 1.

It can be seen from Table 1 that the position error of the mechanism during the forward and reverse motions is

TABLE 1: Transitions selected for thermometry.

Situation	Target position (°)	Actual position (°)	Position error (°)
Forward rotation	0	0.0045	0.0045
	60	59.99175	-0.00825
	120	119.9925	-0.0075
	180	180	0
	240	239.9895	-0.005
	270	270	0
Reverse rotation	270	270	0
	210	210.0083	0.0083
	150	149.9963	-0.0037
	90	90.00225	0.00225
	30	29.99475	-0.00525
	0	0.0045	0.0045

between ±0.0085°. There are deviations because the accuracy of the resolver angle acquisition is only 0.0013725°, and the resolver and the mechanism have deviations during the installation process, and there is friction during the movement.

5. Conclusion

From the perspective of aerospace applications, this paper designs and implements a motion controller for satellite-borne data transmission antennas. Firstly, the overall framework of the motion controller is given. At the same time, several specific measures of redundant design are given according to the reliability design of the aerospace device. Then, the design of modules such as command transmission and reception, command decode, speed control, position control, and PWM subdivision control and PWM timing control is introduced in detail. The simulation and experiment results show that the motion controller can control the antennas for angle regulation accurately and stably and control two antennas synchronously or independently. The accuracy of angle regulation reaches 0.0085° under the condition of 64 subdivisions and 100 : 1 reduction ratio. At present, the design has been applied to a satellite-borne data transmission antenna control system.

Data Availability

The experiment data are included in the manuscript. But the resource code used to support the findings of this study has not been made available because there is confidentiality agreement between Jun Zhang and the Shanghai Aerospace Control Technology Institute, China.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] G. Yue, *Design and study on servo controller for satellite data transmission antenna pointing mechanism, [M.S. thesis],*, Shanghai Jiao Tong University, China, 2013.

[2] L. Hui-Feng and Q. Jian-Jiang, "Design on P-fuzzy-PI controller for antenna control system," *Radio Engineering*, vol. 41, no. 7, pp. 45–47, 2011.

[3] H. Ya-ting, Z. Yong, and Y. Jian-hong, "Application of fuzzy-PI control in antenna servo system," *Micromotors*, vol. 48, no. 4, pp. 84–87, 2015.

[4] W. Yu-tian and Y. Wei-dong, "Rotation control technique applied to satellite-borne SAR antenna," *Chinese Journal of Electron Devices*, vol. 41, no. 3, pp. 576–581, 2018.

[5] Z. H. A. N. G. Wen-hui, *Structural design analysis and test of a spaceborne data transmission antenna, [M.S. thesis]*, Xi Dian University, China, 2013.

[6] Z. H. O. U. Yong and Z. H. A. N. G. Jian, "Modeling and simulation of driving transmission antenna," *Chinese Space Science and Technology*, vol. 34, no. 6, pp. 31–37, 2014.

[7] M. Xiu-Juan, Z. Zhi-hao, W. Xue-Yi, and L. Yan-Li, "Current controlling subdivision driving system for stepper motor based on FPGA," *Control Engineering of China*, vol. 24, no. 1, pp. 16–21, 2017.

[8] T. Li, "Design of subdivision driving system for stepper motor," *Mechanical and Electrical Information*, vol. 27, no. 6, pp. 90-91, 2017.

[9] J. Sheng-guang, "Design of PWM generator for stepper motor based," *Electric Drive*, vol. 44, no. 2, pp. 47–50, 2014.

[10] C. Liu, G. Zheng, and X. Meng, "Design of servo-control system on a moving base: pointing control to antenna based on aeroboat," in *2010 International Conference on Measuring Technology and Mechatronics Automation*, pp. 1035–1037, Changsha City, China, 2010.

[11] S. Yan and Z. Shi-long, "The design points of RS232-RS422/RS485/TLL signal converter," *Journal of Shandong Institute of Commerce and Technology*, vol. 12, no. 2, pp. 92–94, 2012.

[12] N. Jia, H. Chen, and J. Tian, "A design of configurable multitype flight data acquisition system," in *2018 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC)*, pp. 1–5, Qingdao, China, 2018.

[13] C. Yebing, X. Jing, L. Yu, Z. Fanquan, and F. Junyong, "Research on exciting amplification of rotary transformer for electric servo system," *Modern Electronics*, vol. 39, no. 24, pp. 167–171, 2016.

[14] W. Kester, *Resolver-to-digital Converters: Tutorial [R] US: Analog Device*, Anolog Devices, Inc., 2013.

[15] Z. Wenxiang, L. Yujing, Z. Wei, and M. A. Liming, "The principle and application of precise-crude combination shaft angle conversion of the double channel revolving transformer," *New Technology and New Process*, vol. 6, pp. 72–76, 2018.

[16] R. Hino, T. Le Caer, F. Le Mentec, P. Fajardo, and N. Janvier, "A position encoder processing unit," in *16th International Conference on Accelerator and Large Experimental Control Systems*, Barcelona, Spain, October 2017.

[17] T. H. Phuong, M. P. Belov, and T. Dang Khoa, "Model predictive controller based on Laguerre functions for large radio telescope servo control system," in *2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, pp. 1003–1007, Moscow, Russia, 2018.

[18] D. Zhang, J. Wang, L. Qian, and J. Yi, "Stepper motor open-loop control system modeling and control strategy optimization," *Archives of Electrical Engineering*, vol. 68, no. 1, pp. 63–75, 2019.

[19] O. K. Hancioglu, M. Celik, and U. Tumerdem, "Kinematics and tracking control of a four axis antenna for Satcom on the Move," in *2018 International Power Electronics Conference (IPEC-Niigata 2018 -ECCE Asia)*, pp. 1680–1686, Niigata, Japan, 2018.

[20] L. Lijiao, S. Ran, M. Jiyu, S. Zhaoxia, T. Huimin, and Z. Jun, "Design of PWM subdivision driver for multi-axis stepper controlling," in *2018 13th World Congress on Intelligent Control and Automation (WCICA)*, Changsha, China, 2018.