Design of stepper motor driver based on STM32

Haowen Liu\textsuperscript{1a*}, Ripeng Li\textsuperscript{1b}

\textsuperscript{1}School of Electrical Information Engineering, Northeast Petroleum University, Daqing 163318, China

\textsuperscript{1a*}haowenliu2021@126.com, \textsuperscript{1b}liripeng678@163.com

Abstract: In the field of industrial control, stepping has been widely used as an actuator. For this article, based on STM32F103 as the control core, the functions of start and stop, forward rotation, reverse rotation and speed regulation of the stepping motor are realized through four buttons. The system uses L293D to drive a stepper motor, LED0 and LED1 two status lights display the motor's forward and reverse rotation status, and the digital tube displays the current stepper motor speed. The whole system includes L293D module, button module, status light module and speed display module. The whole system is modeled, theoretically analyzed and simulated based on the drive control principle of the stepper motor, and the precise and stable control performance of the system under different occasions is further verified through experiments.

1. Introduction
Stepping motor is a kind of micro-motor that is controlled by pulse signal and can convert the pulse signal into corresponding angular displacement or linear displacement. Since the power supply inputs an electric pulse to the motor each time, the motor will advance a small step, and the way of rotation is stepping, so it is called a stepping motor. Because the power input is a pulse-like voltage, it is sometimes called a pulse motor. The author's control circuit uses ST's STM32F103 microcontroller as the core of the system. \cite{1} The STM32F103 series is a 32-bit RISC (Reduced Instruction Set Computer) low-power core based on the high-performance ARM\textsuperscript{\textregistered}Cortex\textsuperscript{\texttrade}M4F. The operating frequency is up to 168MHz. It supports all ARM single-precision data processing instructions and data types of single-precision floating-point units. It also implements a complete set of DSP instructions and memory protection units, thereby improving the security of the application.

2. System design plan and main components
The hardware part is mainly based on STM32. The stepper motor selected in this design is a four-phase stepper motor. Through the combination of software and hardware, the start-up pause, forward and reverse rotation, acceleration and deceleration functions of the stepper motor are realized, and Place the status line of the motor's steering and speed level on the digital tube. The design is composed of STM32, keyboard control module, digital tube display module, motor drive module and power supply module, the overall framework of the design is shown in Fig.1. The keyboard control module, digital tube display module, and motor drive module are the core content of this design.
The control circuit uses ST’s STM32F103 microcontroller as the core of the system; the drive chip is L239D, which contains a four-channel logic drive circuit inside, which is a special driver for two-phase stepper motors and four-phase stepper motors, which can be used at the same time. To drive two two-phase stepper motors or one four-phase stepper motor, the analog timing signal can be directly provided by the IO interface of the single-chip microcomputer;

The display circuit adopts LED digital tube. The display module of LED digital tube adopts LED seven-segment common anode or common cathode digital tube, which can carry out dynamic display and static display. The pulse sequence output by STM32 drives the digital tube display through the pull-up resistor. The digital tube dynamic display mode has the advantages of high brightness, fast response speed, long service life, simple hardware circuit, simple programming, etc., but it occupies more I/O ports of the single-chip microcomputer.

Stepper motor selection 28BYJ-48 stepper motor is a four-phase eight-beat motor, the input voltage range is DC 5V-12V. When the stepping motor inputs a series of continuous pulse signals, the stepping motor rotates the corresponding angle according to the corresponding operating state. For example, when the energized state of the input stepper motor changes once, the corresponding rotor will also rotate through a corresponding step angle and be connected to it.

3. Hardware design

3.1. STM32 minimum system circuit
STM32 minimum system is the most basic circuit for system operation and debugging, which mainly includes power supply circuit, clock crystal oscillator circuit, reset circuit and JTAG simulation interface debugging circuit. The chip uses a 25M external high-speed crystal oscillator as the system clock source, which is input to the internal clock tree for frequency division and frequency multiplication for use by various peripherals. The reset circuit adopts RC plus buttons to realize power-on reset and manual reset. Since the STM32 chip has integrated power-on reset and voltage detection reset circuits, the external reset is mainly to realize the manual reset function.

3.2. Keyboard control circuit
The system is equipped with 4 buttons to set and control the related parameters of the asynchronous motor’s frequency conversion speed regulation. According to the requirements of the control system function, four control buttons need to be set, namely: K1–K4, to control the start and stop, forward rotation, reverse rotation and speed regulation functions of the stepping motor respectively. When the control button K2 or K3 has an input signal, STM32 detects that the state of the P1.0 port or P1.1 port changes from high to low, and the internal program calls the corresponding start-stop program or turn to the program, and then controls the step into the start-stop and steering working state of the motor.
3.3. Digital tube display circuit
The digital tube display, also known as the LED display, is a device that can display fields composed of multiple light-emitting diodes, and each segment of the LED display needs to be driven by a driving circuit. In this control system, we need to use the STM32 FSMC interface to drive the LED display. This interface can well expand various types and speeds of external memory. When connecting different external memory types, it will send out according to the speed of the signal. Matching data, address, control signal. The LED used in this article uses FSMC_NE4 as the chip select signal, in fact, the LED is controlled as SRAM.[3]

3.4. Serial communication circuit
The serial port is a widely used device communication protocol on computers and instrumentation equipment. It can view, print and output related information by communicating with the upper computer. STM32's serial communication is relatively simple to implement. As long as the baud rate, data bit length, parity bit and other information are set, data transmission can be completed by sending and receiving two wires. However, because the processor chip outputs a TTL level signal, and the PC serial port is an RS-232 level, the hardware needs to add a level conversion chip MAX323 to achieve two-way voltage conversion.

4. Software design

4.1. Design of the main program
The software program of the system completes the control task through the control of the hardware equipment. Before writing the software program, the system needs analysis must be carried out to determine the functions to be realized by the system and the final task indicators of the system. In this system, the software mainly realizes functions such as driving the motor, data processing, and communication.

4.2. System initialization
The STM32 peripherals required by this design include PA port, PB port, PC port and timer port, so the clock of the peripheral must also be set. Because the PA port, PB port, and PC port are on the APB1 system bus peripherals, and the timers TIM2 and TIM3 are on the APB2 system bus peripherals, the clock frequencies of the APB1 and APB2 buses must be set. After frequency division of APB1 and APB2, SYSCLK is converted into a system clock that can be received by peripherals and TIM. In the clock initialization subroutine, first select the mode of the system clock, that is, set the system clock to HSE mode, then set the AHB clock equal to the system clock, and set the low-speed or high-speed AHB, and finally the clock initialization flowchart is shown in Fig.2.
4.3. AD conversion program design
Including the result of the conversion, the ADC needs a stabilization time TSTAB before starting the accurate conversion. The timing diagram is shown in Fig.3. After starting ADC conversion and 14 clock cycles, the EOC flag is set. When the 16-bit ADC data register is initialized in the ADC setting, first turn on the AHB peripheral clock enable register DMA1 clock, define the conversion data volume, and set Initialization of DMA channel 1, independently select the sampling time of the channel, set the watchdog on channel 1, turn on ADC1 and start conversion, apply continuous conversion mode, use DMA mode, and then start to convert regular channels to complete the conversion from analog to Changes in the amount of data. [4]

4.4. Motor control interrupt program design
Read the value through A/D, the conversion value is -180-180. If the conversion value is greater than 0, the stepper motor rotates forward, otherwise the stepper motor reverses. If the conversion value is
greater than 170, the maximum speed of the stepper motor will rotate forward. If the conversion value is less than -170, the maximum speed of the stepper motor reverses. If the read A/D conversion value is the same as the displayed value, continue to read the A/D value. If the read value is different from the displayed value, the display will be recalculated.

4.5. LED display program design
If you want to light up a certain point on the LCD screen, you actually set a certain bit in the display RAM area corresponding to that point, so you must determine the row address and column address of the point. The row address of the MzL02-12864 liquid crystal display module is actually the page information, each page should have 8 rows; and the column address represents the abscissa of the point, which is arranged from left to right on the screen, one of the pages Byte corresponds to one column (8 rows, that is, 8 points), up to 128 columns, according to this relationship in the program to control the display of the LED display. [5] In the displayed main program, the font color and background color are displayed on the TFT display screen by quoting the following program.

5. Analysis and debugging
Proteus software is made by Lab Center Electronics, a British company. The use of this software realizes a complete design from design ideas to actual product production, greatly improving the efficiency of electronic design work. Software design we adopt C language design, and the programming software adopts Keil MDK software of Keil company. The main program involving the pulse signal driven by the stepper motor is shown in Fig.4 and the Proteus hardware simulation is shown in Fig.5.
6. Conclusion
This design mainly studies the circuit design of the stepping motor control system based on STM32. This design uses STM32 as the control core. The pulse signal generated by it is amplified by the L239D chip to drive the stepping motor. At the same time, the rotation and speed of the stepping motor are displayed synchronously by the digital tube, and the stepping motor is realized by the corresponding button. Start and stop, forward and reverse, acceleration and deceleration functions. In the process of this design, the hardware circuit is designed to be simple, and the advantages of the software part are fully utilized to meet the requirements of the system. The actual test shows that the performance of the designed system is better than that of the traditional stepper motor controller. It has the characteristics of simple circuit structure, high reliability, strong practicability, convenient control, and high cost performance. It has a broad development prospect.

References
[1] Li Ning. Development and application of STM32 processor based on MDK [M]. Beijing: Beijing University of Aeronautics and Astronautics Press, 2008:35-47
[2] Feng Tao, Li Qing, pan yuedou, et al. Design of trapezoidal acceleration and deceleration curve planning and control experimental system of stepping motor [J]. Coal mining machinery, 2020, 41 (07): 23-25

[3] He Changjun, Yang Zidong, Yan Zhenqi, et al. Stepping motor dynamic acceleration and deceleration control based on STM32 [J]. Light industry machinery, 2020, 38 (02): 45-50

[4] Wang zongpei. Stepping motor and its control system [M]. Harbin: Harbin Institute of Technology Press, 2009

[5] Zhou Jie, Chen Weihai, Yu Shouqian. Application of arm based embedded system in robot control system [J]. Microcomputer information, 2007, (2): 45-47