

# Design of Brushed Motor Position Loop Control System Based on Incremental PID

Tianyi Zhang \*, Peng Lu

*Tianjin University of Science and Technology, China;*

*E-mail: \* 2679483816@qq.com*

*www.tust.edu.cn*

## Abstract

The motors are an important part of the transmission and control system. With the development of technology, the focus of motors in practical applications has shifted to precise control of speed, position and torque. This paper focuses on the control system design of the brush motor position loops, using STM32 microcontroller and incremental PID to achieve accurate control of the motor stopping moment position. The software is designed in C programming. It is divided into four parts: System Clock Configuration Functions, Principal Functions, System Tick Timer Interrupt Callback Function, Position closed loop PID Control Design Function. The hardware part is based on the STM32F103VET6 core board.

*Keywords:* DC brush motor, STM32, closed-loop control, PID

## 1. Introduction

For many reliable technical reasons, DC-powered brush motors have been largely replaced by electronically controlled brushless motors. In spite of this, in less demanding or cost-sensitive applications, Brushed motors are still an effective solution.

By combining the Brushed Motor with the Basic Motor Driver IC, brushed motors and end products allow for many additional operations and better protection. This paper describes the design of position loop hardware and software control system, for brush motor incremental PID. The function of this system is to rotate the motor, and change the stopping position after pressing the key.

For YS-F1Pro demo board hardware, we have obtained a program on the rotation of hard stone motors. We have optimized the changes based on this program. The obtained source program is optimized to be migrated to the core board containing the STM32F103ZET6 chip. This is used to drive the motor to rotate according to programmed instructions. control intelligent household device and monitor the information of intelligent household device.

## 2. The Hardware Selection

The entire hardware system consists of three parts: the motor, the hardstone driver board, and the black core board containing the STM32F103VET6.

### 2.1. Driver board selection

For the driver board, choose the motor driver ULN2003 from the YS-F1Pro development board. ULN2003 is a high-current drive array. It is mostly used in control circuits of microcontrollers, smart meters, PLCs, digital output cards, etc. It can directly drive relays and other loads. Its input 5V TTL level and output up to 500mA/50V. ULN2003 can be directly connected to TTL and CMOS circuits at 5V operation, it can directly process data that would otherwise require a standard Logical Buffer to process.

The ULN2003 is a family of high-voltage, high-current Darlington transistor arrays, it features high current gain, high operating voltage, wide temperature range, and high load capacity. It is suitable for all types of systems

requiring high speed and high power drives. It can drive brushed DC motors of up to 600W (60V10A).

The schematic diagram of the driver chip ULN2003 is shown in Fig.1.

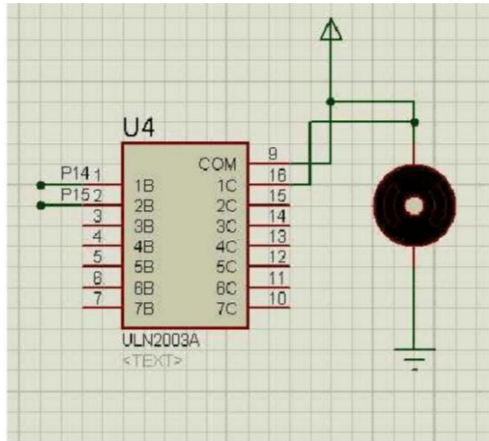


Fig.1. The schematic diagram of the driver chip ULN2003

## 2.2. Motor selection

For the motor, We chose the GM37-545 Hall-coded geared motor, which has a precision reducer for better coil amplification, and features high power and high torque. the GM37-545 Hall-coded geared motor is shown in Fig.2.



Fig.2. the GM37-545 Hall-coded geared motor

## 2.3. Core chip selection

For the core board, the chip selected is the STM32F103VET6. which is the Cortex-M3 ARM core and has a 100-pin. The chip has 128KB flash memory and 20KB RAM. It can through SWD or JTAG with KEIL 4 to download the program and online debugging<sup>1</sup>. And it has more than 80 general-purpose IO ports, multiple timers and serial ports, which fully meet the

design of the required pin and memory requirements. It has the following characteristics:

- Supported peripherals: Timer, ADC, SPI, I2C and USART.
- Operating Voltage 2.0V ~ 3.6V.
- Operating temperature range: -40 °C ~ 85 °C.

The design of the main control chip is shown in Fig.3.

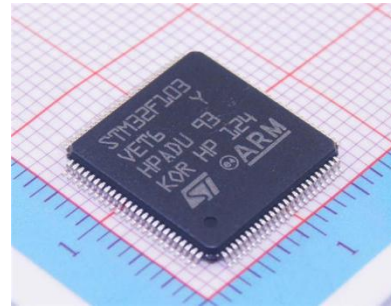


Fig.3. STM32F103VET6 chip

## 3. Software Design

Software design mainly includes the design and writing of programs, the application of functions in programs, etc.

### 3.1. Program Block Diagram

The program block diagram of system is shown in Fig.4.

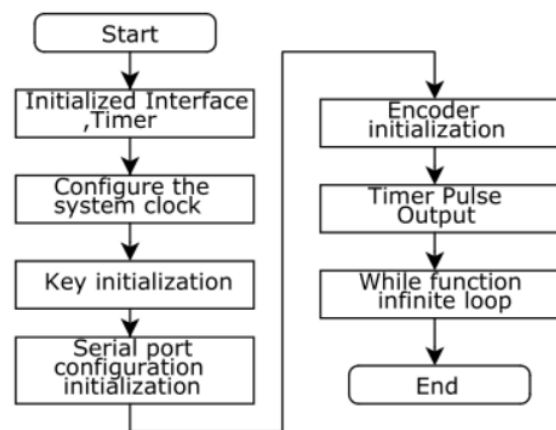


Fig.4. program block diagram

The first step is to configure the Flash interface and system ticking timers, keys, serial communication, encoder initialization, and so on. Then configure the

timer pulse output. Finally, set the LCD program to an infinite loop with the while statement.

### 3.2. System Timer Interrupt Callback Function

This function calculates PID results based on a velocity loop period of 60ms, and no change in limiting speed. It uses PWM duty cycle to determine the current motion direction. Note that every time a timer occurs, the interrupt enters the callback function once.

### 3.3. Position closed loop PID control function

The function is enter the current control amount and output the target control amount. First, the deviation is calculated, and the deviation dead-band is used to determine if the deviation exists. The “iIncpid” is calculated according to the formula for the incremental PID, and the final error is stored for the next calculation<sup>2</sup>. Main procedures for incremental PID control is shown in Fig.5.

```

/* Dead Zone Limits */
if ((iError<50) && (iError>-50))
    iError = 0;
iIncpid=(int32_t) ((sPID.Proportion * (float)iError) // Term E[k]
    - (sPID.Integral * (float)sPID.LastError) // Term E[k-1]
    + (sPID.Derivative * (float)sPID.PrevError)); // Term E[k-2]

sPID.PrevError = sPID.LastError; // Store errors for the next calculation
sPID.LastError = iError;
return(iIncpid); // Return the incremental value

```

Fig.5. Main procedures for incremental pid control

This program uses the PID formula to calculate the deviation iError:

Deviation = set value - measured value

If  $-50 < \text{deviation} < 50$ , it can be considered 0.

The PID formula is as follows:

$$u_k = K_p * e_k + K_i \sum_{j=0}^k e_j + K_d (e_k - e_{k-1}) \quad (1)$$

PID incremental value = current error E[K] - last error E[K-1] + last error E[K-2]

$$\Delta u_k = u_k - u_{k-1} = A e_k - B e_{k-1} + C e_{k-2} \quad (2)$$

The last error is stored for the next calculation, returning an infinite loop of incremental values.

## 4. Project Experiment

### 4.1. Motor control theory

The drive board adopts H-bridge driving circuit, the directional control of brushed DC motor is shown in the figure, when Q1, Q4 conduction, Q2, Q3 closed, the current direction is Q1 → motor → Q4, then the motor is positive, when Q2, Q3 conduction, Q1, Q4 closed, the current direction is Q3 → motor → Q2, then the motor is reversed. Motor Forward Rotation Diagram is shown in Fig.6. Motor Reverse Diagram is shown in Fig.7.

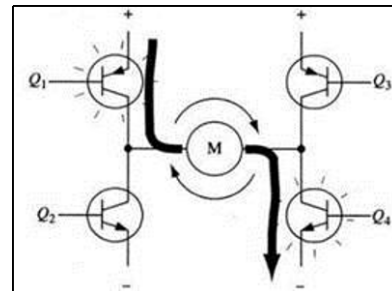


Fig.6. Motor Forward Rotation

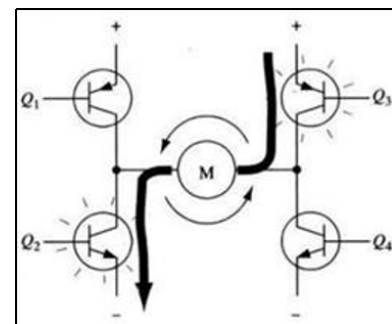


Fig.7. Motor Reverse

### 4.2. Experimental procedures

After hardware selection and software design, hardware and software connections are needed to complete the experiment.

There are the following steps:

- Establishment Project
- Create new Usart workgroups and files and link settings.
- Importing software design programs.

Change the optimization program according to experimental requirements and schematic diagrams, design main functions and sub-functions, program the system clock configuration function, main function, system drop timer interrupt callback

function, and position closed-loop PID control design function.

- Connecting Hardware Devices.  
Connect the core board, motor and YS-F1Pro drive board according to the given pin diagram and datasheet.

The experimental results that can be obtained:

1. Press KEY1: motor start up
2. Press KEY2: motor stop
3. Press key 3 and add 1 from the motor's stop position.
4. Press key4 and subtract 1 from the motor stop position.

## 5. Conclusion

This experiment is to write software design for position ring control of brushed motor (incremental PID) using C program. The program includes the main function, timing interrupt callback function, clock configuration function, position ring design function, etc. Connect hardware devices to complete the experiment. Use PID function to calculate the motor stop position.

## References

1. Yuxing Ouyang , Fengzhi Dai , Yiqiao Qin , et al. Integral Design of Intelligent Home Equipment. *Proceedings of International Conference on Artificial Life and Robotics*, 2017, 22: pp.148-151.
2. Zhiwang Chen, Baohua Liu, et al, *Getting Started with STM32 Embedded Microcontrollers*, YanShan University, Hebei, China, 2014.5.