S-type speed control curve based on the number of pulses

Zhizheng Zhang¹, Yuanhua Yu*¹
¹School of Life Science and Technology, Changchun University of Science and Technology, Changchun, 130022
¹18844182207@163.com, *¹yuyuanhua8888@126.com

Abstract—The stepper motor is a kind of motor that converts electrical pulse signals into corresponding angular displacements or linear displacements, with high accuracy and easy control. According to the characteristics of stepper motor controlled by pulses, this paper designs an S-type speed control method based on the number of pulses, then tests and analyzes it in a motion system which controlled by STM32. The results show that: the change of the speed for the motion system is compared. It is stable and basically conforms to the designed quadratic function curve, and there is no out-of-step phenomenon in the test. While retaining the great stability and acceleration performance of S-type acceleration and deceleration, this control method greatly reduces the calculation difficulty. It is more suitable for microcontrollers and embedded control systems.

1. Introduction
The stepper motor is a kind of motor that converts an electrical pulse signal into a corresponding angular displacement or linear displacement [¹]. Each time a pulse signal is input, the rotor rotates an angle or one step forward, and its output angular displacement or linear displacement is the same as the input. The number of pulses is proportional, and the speed is proportional to the pulse frequency [²]. Due to the wide application of stepper motors, the control and use of stepper motors have been extensively studied. When the stepper motor stops suddenly, starts suddenly, or changes its speed suddenly, the rotation of the rotor cannot be matched due to the sudden change of the input pulse or the improper pulse frequency, resulting in out-of-step, overshoot and other phenomena, resulting in the movement position errors will also cause a certain impact on the components of the motion mechanism and affect the service life. In order to control the error in the movement and make the running process more stable, the researchers used the step method to decelerate the DC motor [³], the adaptive critical neuro-fuzzy controller with the magnetic field The signal provided by the encoder for feedback control [⁴][⁵] and other methods, which have achieved good results. However, in actual use, the control system of a stepper motor often only has limited computing capabilities and is not suitable for complex operations for error compensation. Practice has shown that appropriately reducing the speed of the stepper motor can significantly improve the stability of the stepper motor, but a too low speed will make the movement efficiency low. Adding the acceleration and deceleration process when designing the driver can greatly improve the movement efficiency while ensuring the smooth operation of the stepper motor.

S-Type acceleration/deceleration is a speed change method in which the speed change curve with time is close to the shape of "S" in the process of acceleration or deceleration. Take the acceleration process as an example. The acceleration gradually increases to a certain value and then remains unchanged. After a period of time, it gradually decreases until the speed drops to zero. At the same
time, the speed is also divided into three types: acceleration, uniform acceleration and deceleration. Change phase. The traditional S-type acceleration/deceleration is adjusted based on time, which does not completely conform to the characteristics of pulse control and the purpose of positioning for the stepper motor, and the calculation amount is large. Therefore, the S-type based on the number of pulses is designed. The speed control method solves the above-mentioned shortcomings well.

2. Design of S-type speed control curve based on the number of pulses

The number of pulse cycles received by the driver is directly proportional to the number of steps of the stepper motor. That is, under the condition that the subdivision is unchanged, the stepper motor rotates one step every time the driver receives a pulse, so the speed of the stepper motor is also proportional to the single-chip microcomputer. The output pulse frequency is proportional. The S-shaped acceleration and deceleration curve based on the number of pulses changes the traditional quadratic curve S-shaped acceleration and deceleration algorithm from adjusting the motion state based on time to adjusting the motion state based on the number of pulses, and assigns it at different travel positions. Different pulse frequencies can achieve the purpose of controlling the speed of the stepper motor.

2.1. Algorithm design

Assuming that in a movement process, the total number of steps of the stepper motor is equal to \(N\) and the total number of pulses that the MCU needs to output to the stepper motor driver is also equal to \(N\). So the whole movement process could be divided into three parts: acceleration, constant speed and deceleration. Since the change trend of the absolute value of the speed is the same during the acceleration process and the deceleration process, the acceleration process could be taken as an example to design the speed change curve with respect to the number of pulses. During the acceleration process, it is divided into three stages according to the change of acceleration:

The first stage \((0 \leq n \leq n_1)\) is an acceleration movement with a gradual increase in acceleration. The acceleration is positive. At this stage, the acceleration \((a(0))\) changes linearly with the increase in the number of advancements (\(k\) is the acceleration rate and \(a_0\) is the initial acceleration in the following formulas):

\[
a(n) = kn + a_0
\]

\[
v(n) = \int_0^n a(n)dn + v_0 = \frac{1}{2} k n^2 + a_0 n + v_0
\]

The second stage \((n_1 < n \leq n_2)\) is the uniform acceleration movement stage. After the first stage runs \(n = n_1\), it reaches the maximum acceleration preset by the program or completes the numbers of step which could be used to acceleration. After this time, the absolute value of the acceleration reaches the maximum and will be no change. It starts accelerating uniformly.

\[
a(n) = a_{\text{max}} = kn_1 + a_0
\]

\[
v(n) = \int_{n_1}^n a_{\text{max}}dn + v(n_1) = (kn_1 + a_0)n - \frac{1}{2} k n_1^2 + v_0
\]

The third stage \((n_2 < n \leq n_3)\) is an acceleration movement with a gradual decrease in acceleration. The second stage runs until \(n = n_2\) reaches the deceleration time calculated or preset by the system. The acceleration starts to decrease with the pulse number, then the acceleration drops to 0 when \(n\) is equal to \(n_3\). After the acceleration is completed, it enters the state of uniform motion:

\[
a(n) = -kn + kn_2 + a_{\text{max}}
\]

\[
v(n) = \int_{n_2}^n a(n)dn + v(n_2) = -\frac{1}{2} k n^2 + (kn_1 + kn_2 + a_0)n - \frac{1}{2} k (n_1^2 + n_2^2) - (kn_1 + a_0)(n_1 - n_2)
\]
2.2. Design of control program

The design of the control program is based on STM32F103VET6 which was produced by STMicroelectronics[6], and it cooperates with the DM422C stepper motor driver and 42CM08 stepper motor produced to realize and verify related control functions. STM32F103VET6 is a microcontroller designed by ST company based on the Cortex-3 core. It has 4 general-purpose timers, two advanced timers and two basic timers. It can achieve multiple PWM or timer pulse outputs and has the USART function can realize the communication between the upper and lower computers and feedback the current output status to the upper computer in the test.

The realization process of the S-type acceleration and deceleration curve based on the number of pulses mainly includes the following three processes:

(1) Obtain the parameters required by the motion mode: set the initial acceleration and initial velocity as the initial state parameters of the motion; set the maximum acceleration and maximum acceleration as the judgment conditions for the acceleration and deceleration state switching; obtain the required running distance as the termination condition of the entire movement.

(2) Calculate the number of steps required in the process of acceleration and deceleration as the judgment condition for switching from acceleration to constant speed, and from constant speed to deceleration.

(3) Calculate the current acceleration and speed, and control the timer to generate corresponding pulses of the corresponding frequency to drive the motor to run.

According to the above three process, the entire control program is divided into five parts: the unit for obtaining and calculating eigenvalue, the unit for generating pulses, the unit for recording the number of pulses, the unit for calculating acceleration and the unit for calculating speed.

(1) The unit for obtaining and calculating eigenvalue: This part acquires the eigenvalues for the entire movement process, including the given initial acceleration \((a_0)\), initial velocity \((v_0)\), maximum acceleration \((a_{\text{max}})\), maximum velocity \((v_{\text{max}})\) and running distance \((x)\); calculates the total number of steps (total number of pulses) in the running process \((N)\), The total number of pulses during acceleration \((N_a)\) and the number of steps required during deceleration \((N_d)\). This design used a stepper motor to drive the screw rod which the diameter is 12mm and the pitch is 4 mm to complete a linear motion, and the stepper motor driver adopts the non-subdivision mode \((1.8°/\text{step})\), and the total number of pulses can be calculated:

\[
N = \frac{x}{4} \times \frac{360}{1.8} = 50x \tag{7}
\]

Since the acceleration process is divided into three stages of acceleration increases, constant acceleration, and acceleration reduction, the velocity-pulse number relation of the above three stages is calculated by definite integral, and the speed change amount of each stage \((\Delta v_1, \Delta v_2 \text{ and } \Delta v_3)\) can be obtained:

\[
\Delta v_1 = \int_0^{n_1} a(n)dn = \frac{1}{2}k n_1^2 + a_0 n_1 \tag{8}
\]

\[
\Delta v_2 = \int_{n_1}^{n_2} a_{\text{max}}dn = (kn_1 + a_0)(n_2 - n_1) \tag{9}
\]

\[
\Delta v_3 = \int_{n_2}^{n_3} a(n)dn = \frac{1}{2}k(n_3 - n_2) - kn_2 - a_0(n_2 - n_3) \tag{10}
\]

Since the acceleration change amount \((\Delta a)\) and the acceleration change \((k)\) rate phase are the same in the acceleration phase and the deceleration , the number of pulses required for the two phases
is also the same, that is \( n_1 = n_3 - n_2 \), \( n_1 = \frac{1}{k}(a_{\text{max}} - a_0) \). According to the speed change amount \((\Delta v)\), the total number of pulses in the acceleration process \( (N_a) \) can be calculated:

\[
\Delta v = v_{\text{max}} - v_0 = \Delta v_1 + \Delta v_2 + \Delta v_3
\]

(1)

\[
N_a = n_3 = \frac{1}{a_{\text{max}}} \left[ (v_{\text{max}} - v_0) + \frac{1}{k}(a_{\text{max}} - a_0) \right]
\]

(2)

The change trend of the absolute value of acceleration during increases and acceleration reduction is same, so \( N_a = N_d \).

(2) The unit for generating pulses: This part is responsible for recording the current number of pulses that have been executed \( (n) \), and switching the motion state according to the execution progress and execution status. If the number of pulses currently executed is greater than or equal to the number of pulses required in the acceleration phase, the motion state will be switched from acceleration to constant speed; if the number of remaining pulses \( (N - n) \) is less than or equal to the number of pulses required in the deceleration phase, the motion state will be switched from constant speed to deceleration. If the number of pulses currently executed is greater than or equal to the total number of pulses, the current motion process is ended, and the characteristic values of this motion are returned to zero. When the total number of pulses is not enough to complete the entire acceleration and deceleration process, in the acceleration phase, if the currently executed pulse number is greater than or equal to half of the total number of pulses, the motion state will be directly switched from acceleration to deceleration.

![Fig. 1 Flow chart of pulse counting unit](image)

(3) The unit for recording the number of pulses: This part is responsible for controlling the timer to generate pulses of specific frequency. The design used the inversion of the level of a specific IO port in the timer interrupt to generate a pulse signal. Each time an interrupt is entered, the corresponding IO port is flipped. Level, that is, a complete pulse signal can be obtained by entering the interrupt twice. Compared with the pulse output mode of configuring the timer to output the PWM signal, the method of timer control flipping IO is more convenient to record the number of output pulses, and is not limited by the timer and IO port functions, simple, reliable, and easy to implement. Changing the timer
configuration parameters (reload value and clock prescaler coefficient) can change the time interval of a single flip of IO, output pulse signals of different frequencies, and achieve the purpose of speed change. The clock prescaler coefficient is fixed in the design, and the purpose of frequency conversion is achieved by calculating the reload value required by the current speed.

(4) The unit for calculating acceleration: This part is responsible for calculating the acceleration value in each movement state. Since the acceleration and deceleration have the same change trend in the absolute value of the acceleration, this unit can be used to calculate the acceleration. Taking the acceleration phase as an example, the acceleration phase is divided into three parts, acceleration increases, constant acceleration, and acceleration reduction. After the start, it enters the acceleration increases phase, the acceleration \( (a) \) increases linearly with the number of pulses \( (n) \), that is the acceleration increases by \( k \) science the number of pulses increases by 1. If the current acceleration is greater than or equal to the maximum acceleration. The acceleration state is switched from acceleration increases to constant acceleration; the acceleration remains unchanged after entering the constant acceleration stage. If the number of remaining acceleration pulses \( (N_a - n) \) is less than or equal to the number of pulses required for the acceleration to drop to 0 \( \left( \frac{a_{\text{max}} - 0}{k} \right) \), then the motion state will be switched to the acceleration reduction stage; when the acceleration drops to 0, the acceleration state will be ended and the state will be directly switched to the uniform speed state. This unit operation will not be used.

In particular, if the number of pulses executed is greater than or equal to the total number of pulses \( \frac{1}{4} \) but the acceleration is still less than \( a_{\text{max}} \) or the number of pulses currently executed is equal to or exceeded \( \frac{1}{2} N_a \) but the acceleration has not yet increased \( a_{\text{max}} \), the acceleration will not increase and it will directly switch to deceleration motion.

Fig. 2 Flow chart of acceleration calculation unit
The unit for calculating speed: This part is responsible for calculating the speed of the current stage, and provides the corresponding reload value with a timer (TIMx_arr). This unit is also divided into three stages: acceleration, uniform and deceleration. In the acceleration phase, \( v \) increases by \( a \) for every 1 increase in the pulse count; if \( v \) is greater than or equal to \( v_{\text{max}} \), \( v \) will be set to \( v_{\text{max}} \).

In the deceleration phase, \( v \) reduces by \( a \) for every 1 increase in the pulse count until the preset number of pulses is completed, and the speed is returned to zero.

The operating speed of a mechanism powered by a stepper motor depends on the pulse frequency output by the controller, and the pulse frequency is positively correlated with the movement speed. If the speed is \( v/(\text{mm/s}) \), according to the used screw pitch (4mm) and the controller subdivision setting (not subdivision), the corresponding pulse frequency can be calculated:

\[
f = \frac{v \times 360}{4 \times 1.8} = 50v
\]

(3)

The timer frequency is determined by the reload value (arr) and the clock prescaler coefficient (psc):

\[
f_{\text{TIM}} = \frac{72000000}{(TIM_{\_arr} + 1)(TIM_{\_psc} + 1)}
\]

(4)

According to the pulse generating unit, the timer generates a complete pulse for every two cycles:

\[
f = \frac{1}{2} f_{\text{TIM}}
\]

(5)

Therefore, the relationship between \( arr \) and \( v \) is:

\[
TIM_{\_arr} = \frac{36000000}{50v \times (TIM_{\_psc} + 1) - 1}
\]

(6)

Then assign the reload value to the timer to start the next timing cycle.

### 3. Realization of speed control curve

Set parameter values as shown in Tab. 1. According to the algorithm and control program designed above, the required parameters are: initial velocity \( (v_0) \), maximum velocity \( (v_{\text{max}}) \), initial acceleration \( (a_0) \), maximum \( (a_{\text{max}}) \), acceleration (the acceleration given here is defined as the rate of change of velocity with the number of pulses), and total displacement \( (x) \). Set the acceleration rate of change \( (k) \) is equal to one, the timer clock prescaler coefficient \( (TIM_{\_psc}) \) is equal to 799.

According to the design principle, the change of speed is based on the increase of the number of pulses, so the unit of acceleration here should be the derivative of speed to pulse number, namely: \( \text{mm}/(s \times \text{step}) \).

| Parameter | Value |
|-----------|-------|
| \( v_0/(\text{mm/s}) \) | 2 |
| \( v_{\text{max}}/(\text{mm/s}) \) | 50 |
| \( a_0/(\text{mm}/(s \times \text{step})) \) | 0 |
| \( a_{\text{max}}/(\text{mm}/(s \times \text{step})) \) | 0.4 |
| \( x/\text{mm} \) | 100 |

According to the designed algorithm, the length unit mm needs to be converted into the number of steps. According to the screw used in the actual test it can be calculated that 1mm=50 steps. Then the total number of pulses calculated according to the total displacement is: \( N = 5000 \), \( v_0 = 100 \times \text{step}/s \), \( v_{\text{max}} = 2500 \times \text{step}/s \). When debugging, with each time the speed calculation is completed, the current speed and acceleration values are respectively output through the serial port, and the curve of the speed changing with the number of pulses can be drawn. The change trend of speed with the number of pulses during the whole movement process is shown in Fig. 3. It can be seen that the speed changes...
rapidly during acceleration and deceleration, but the interval between the beginning of the speed change and the end of the change is relatively gentle, and there is no obvious turning point. The overall trend is consistent with design requirements.

Fig. 3 Speed change curve with pulse number

Fig. 4 shows the curve of acceleration changing with the number of pulses. It can be seen that in a velocity change cycle, the absolute value of acceleration starts increases linearly in the first stage from 0, remains unchanged for a period of time after the acceleration reaches the maximum, and then decreases linearly until finally returns to zero. The overall trend is in line with the design goal.

Fig. 4 Acceleration curve changing with pulse number

As the acceleration and deceleration process have the same change trend, the acceleration process is taken as an example for analysis. The overall change trend of the speed during the acceleration stage is shown in Fig. 5. It can be seen that the acceleration process includes three processes of acceleration increases, constant acceleration, and acceleration reduction. The overall acceleration process is relatively smooth, and there is no turning point of sudden changes in speed and acceleration, which is compared with the S-type acceleration curve.

For the acceleration phase curve, the three processes are drawn the curve and fitted for every one, and the relationship between the speed of each phase and the number of pulses is obtained as shown in Fig. 6.

Fig. 5 Speed change curve with pulse number during acceleration stage
The relationship between the speed and the number of pulses in the actual operation of the above three stages can be obtained by fitting:

\[
v(n) = \frac{1}{2} n^2 + \frac{1}{2} n + 100 \quad (0 \leq n \leq 20)
\]

\[
v(n) = 20n - 90 \quad (20 \leq n < 121)
\]

\[
v(n) = -\frac{1}{2} n^2 + 193.5n - 7230 \quad (121 \leq n \leq 141)
\]

It can be seen from the speed and pulse number output by the controller that the acceleration and deceleration process conforms to the general characteristics of the quadratic S-shaped acceleration and deceleration curve, and the goal is basically realized.

In order to test the stability of the designed motion control method, the repeat positioning accuracy of the motion system was measured with a dial indicator. The measurement method is as follows: firstly set a position in the control program and control the movement mechanism to move to this position; secondly adjust the position of the dial indicator after zeroing and contact the measuring head of the dial indicator with the side wall of the tray; next move the dial indicator slowly and rotating the dial indicator pointer to "100" in the first circle, fix the dial indicator and adjust to zero again; finally turn on the driver, control the movement mechanism to reciprocate with this point as the focus and record the value of the table when the movement mechanism gets to this position.

| Frequency | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    | 10   |
|-----------|------|------|------|------|------|------|------|------|------|------|
| Error     | 0.010|-0.003| 0.009| 0.004| -0.003| -0.010| -0.008| -0.005| 0.005| -0.008|
| Frequency | 11   | 12   | 13   | 14   | 15   | 16   | 17   | 18   | 19   | 20   |
| Error     | -0.002| 0.004| 0.005| -0.001| -0.003| -0.004| 0.001| 0.004| -0.007| 0.005|

It is designed to use a stepping motor with a screw rod for testing, and straight line distance for one step is 0.02mm. After measurement, the motion accuracy of the motion mechanism used in the test that matched with the designed algorithm can reach ±0.01mm which can be seen that no out-of-step phenomenon occurred in the test, so the S-type speed control curve based on the number of pulses has a good control effect on the unstable factors of the stepper motor in high-speed motion.

4. Conclusion

A stepping motor speed control curve based on the number of pulses is designed according to the characteristics of the stepping motor. The curve is based on the number of pulses output by the processor. The acceleration process is divided into three processes: acceleration increases, constant
acceleration, and acceleration reduction. The acceleration increases process enables the stepper motor to achieve the maximum acceleration through a smooth process; the constant acceleration process enables the stepper motor to accelerate at the maximum acceleration to achieve a rapid increase in speed; the acceleration reduction process can be smoothly at the end of the acceleration process reduce the acceleration to a minimum, and achieve a smooth transition from acceleration to a uniform speed. In the test, a dial meter is used to measure the repeated positioning accuracy of the motion mechanism. The results show that the error of multiple motion is less than 0.01mm and the single displacement is less than 0.02mm, that is, there is no out-of-step phenomenon.

However, it can be seen from the fitting curve obtained in the implementation process that the actual speed-pulse number curve is different from the curve of the theoretical calculation. This is because in the calculation process, in order to reduce the calculation amount, the calculation method of direct integration of the acceleration is not used. But it is caused by calculating the current acceleration first, and then superimposing the current acceleration with the speed of the previous step to get the current speed. For example: according to the acceleration curve to calculate the integral, when the pulse type is 1, the speed should be 100.5, that is, the timer frequency is 100.5hz at this time, but according to the actual output data, the speed at this time is 101, that is, the timer frequency at this time Is 101hz.

The advantage of using the speed superposition method is that the logic is simpler in writing the control program, the calculation is easier and the operation process is more stable and reliable. The disadvantage is that there is a certain gap between the actual speed curve and the theoretical calculation, and there is still a slight sudden change from acceleration to uniform speed. Since in general practical application scenarios, such tiny mutations will not affect the speed and position accuracy of the entire motion, the algorithm can be applied to general medical equipment and automatic auxiliary equipment, smart homes and other aspects [7]. It has a certain practical significance. The increasingly sophisticated stepper motor control technology also provides more possibilities for the design of application scenarios such as robots, autonomous driving, and smart homes.

Acknowledgement
This study was funded by the Jilin Scientific and Technological Development Program (CN) (Grant Number 20200708119YY) and Scientific and Technological Research Project of Jilin Province Education Department (Grant Number JJKH20210849KJ)

References
[1] Silaban Freddy Artadima and Budiyanto Setiyo and Raharja Wahyu Kusuma. Stepper motor movement design based on FPGA[J]. International Journal of Electrical and Computer Engineering (IJECE), 2020, 10(1) : 151-159.
[2] Johnson Electric S.A.; Patent Issued for Linear Stepper Motor, Device And Method For Manufacturing The Linear Stepper Motor (USPTO 10,601,294)[J]. Journal of Engineering, 2020.
[3] Dadi Venkataraao and Peravali Swapna and Busi Rambabu. Conversion of geared DC motor into stepper motor using switching delay time signals[J]. SN Applied Sciences, 2021, 3(4).
[4] A MUKHERJEE et al. Hybrid controller for precision positioning application[J]. Sādhanā: Published by the Indian Academy of Sciences, 2020, 45(11) : 726-735.P.
[5] Ghanoomi et al. Robust precise trajectory tracking of hybrid stepper motor using adaptive critic-based neuro-fuzzy controller[J]. Computers and Electrical Engineering, 2020, 81(C) : 106535-106535.
[6] Jiaju Zhang,Zhen Zuo. Design and Implement of Multi-channel Stepping Motor Control System Based on STM32 Serial Communication[C]//Proceedings of the 4th International Symposium on Application of Materials Science and Energy Materials.The 4th International Symposium on Materials Science Applications and Energy Materials, 2020:467-474.DOI:10.26914/c.cnkihy. 2020.064250.
[7] Srinivasan V.P. et al. Design and fabrication of dual axis writing machine[J]. Materials Today: Proceedings, 2021, 45(P7) : 6743-6749.