Introduction

The stepping motor is a kind of mechanical and electrical device that converts electrical pulses into angular displacement. More generally, when the stepping driver receives a pulse signal, it will drive the stepping motor rotate a certain degree according to the setting direction (namely, the stepping angle). You can control the amount of angular displacement by controlling the amount of pulse, so as to achieve the purpose of accurate positioning, and at the same time you can also control motor rotation speed and acceleration by controlling the pulse frequency, so as to achieve the purpose of speed controlling.

Classification of Stepping Motor

Permanent Magnet (PM) : It is generally with two phase, torque and volume are low, the stepping angle is 7.5 degrees or 15 degrees.

Reaction (VR) : It is generally with three phase, which can realize large torque output, the stepping angle is 1.5 degrees commonly, but the noise and vibration are quite enormous.

Hybrid (HB) : It combines the advantages of permanent magnet and reaction and is divided into two and five phase: two phase stepping angle is 1.8 degrees and five phase stepping angle is 0.72 degrees in general. This kind of stepping motor is more widely used.

Technical Parameters

The static indicators of stepping motor

The step angle:

It means that every time control system sends a step pulse signal, the fixed angle of a certain permanent magnet stepping motor is 3.75°/7.5° (the value of half step driving is 3.75 °, the whole step driving is 7.5 °). The step angle can be called " fixed angle of a stepping motor ", it is not necessarily the actual angle of a working motor, the actual angle relates to the driver.

The phase number:

It refers to the number of coil group inside a motor. At present, the commonly used stepping motors are two-phase, three-phase, four-phase and five-phase motors. The step angle of a motor varies along with phase number. Generally, the step angle of two-phase motor is 0.9 ° /1.8 °, three-phase is 0.75°/1.5° and five-phase is 0.36°/0.72°. When there is no subdividing driver, users mainly meet their requirements of the step angle by selecting different phase number motors. If using subdividing driver, the "number" of phase number will become meaningless, users only need to change subdividing number of drives, then the step angle can be shifted.

The number of beats:

It refers to the pulse number or conducting state needed to complete the periodic changes in a magnetic field, and it can also be defined as the pulse number that a motor turns a certain step angle. Let's take four-phase motor for an example, four phase and four beats operation mode, namely, AB - BC - CD - DA - AB, four phase and eight beats operation mode is A - AB - B - BC - C - CD - D - DA - A.

The dynamic indicators of stepping motor

The precision of stepping angle

It refers to the error between the actual value and the theoretical value when a motor turns a certain step angle. Expressing as a percentage: the angle error / step angle* 100%. This value varies with the number of beats, when the motor runs by 4 beats, it should be within 5%, 8 beats within 15%

Out of step

When the drive pulse number for the motor is not equal to the steps during operation, we will call this out of step.

The misalignment angle

It refers to the deflective angle between rotor axis and stator axis. There definitely exists misalignment angle when a motor is running. The error caused by misalignment angle can't be solved even we use subdividing drives.

Driving Modes of Stepping Motor

The driving mode of stepping motor is also known as exciting mode, divided into full step excitation and half step excitation. The former one also can be divided, one-phase(one-beat drive) and two-phase excitation(full step drive); the latter also refers to one-two phase excitation(one step drive). Suppose every rotation needs 200 pulse signal to excite, then each excitation signal can make the stepping motor rotate 1.8 °.

The comparison of the three drive mode:

Drive mode	Step Angle	Power Consumption	Advantages and disadvantages
Single step	5.625	P	Simple control, low consumption, but the minimum output torque, vibration is larger, when step easy alienation
Full step	5.625	2P	Maximum power consumption, large output torque, small vibration, step is stable

Half step	2.8125	1.5P	Performance between step taken in single drive and the drive, only half the stepping Angle, smooth operation, the most widely used

The timing tables of the three drive mode:

Wire/step	Step1	Step2	Step3	Step4	Step5	Step6	Step7	Step8
Blue/A	1	1	0	0	0	0	0	1
Pink/B	0	1	1	1	0	1	0	0
Yellow/C	0	0	0	1	1	1	0	0
Orange/D	0	0	0	0	0	1	1	1

(c). Half step

Wire/step	Step1	Step2	Step3	Step4
Blue/A	1	0	0	0
Pink/B	0	1	0	0
yellow/C	0	0	1	0
orange/D	0	0	0	1

Single step

Wire/step	Step1	Step2	Step3	Step4
Blue/A	1	0	0	1
Pink/B	1	1	0	0
Yellow/C	0	1	1	0
orange/D	0	0	1	1

Full step

The reduction stepping Motor

Diameter: 28mm

Voltage: 5 v

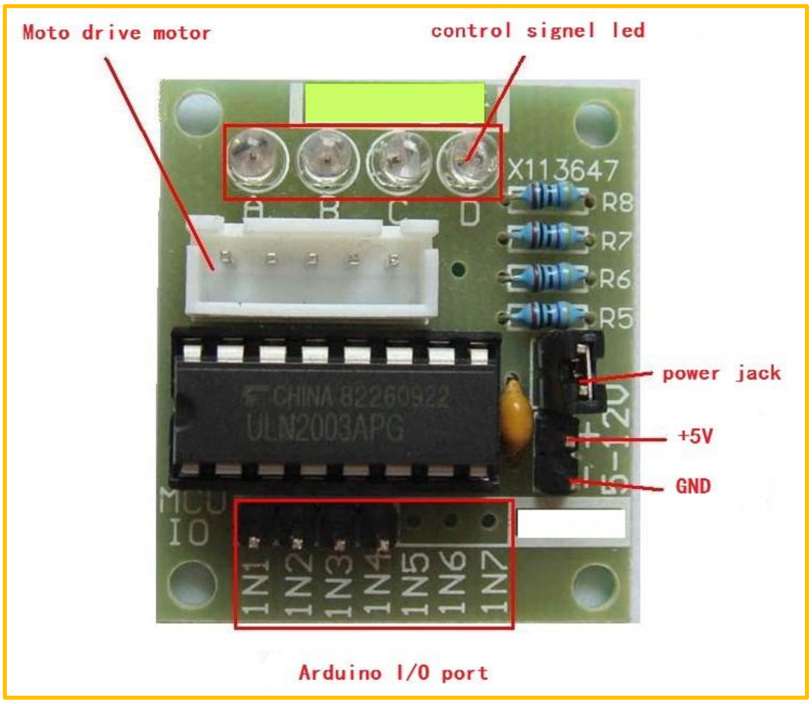
The step angle: 5.625 x 1/64

The reduction ratio: 1/64

The no-load power consumption of the stepping motor is under 50mA, it is equipped with a 64 times speed reducer, namely, 64 drive pulse. If the external belt rotates a circle, the stepping motor spindle needs to rotate 64 circles due to the 1:64 reduction gear in the motor.

The output torque is massive, so heavy load could be driven, it is suitably used on development board. Notice: This stepping motor is equipped with a 64 times speed reducer, the rotational speed is slower compared with those without a speed reducer. For the convenience of observation,we can stick a piece of cardboard on the output shaft.

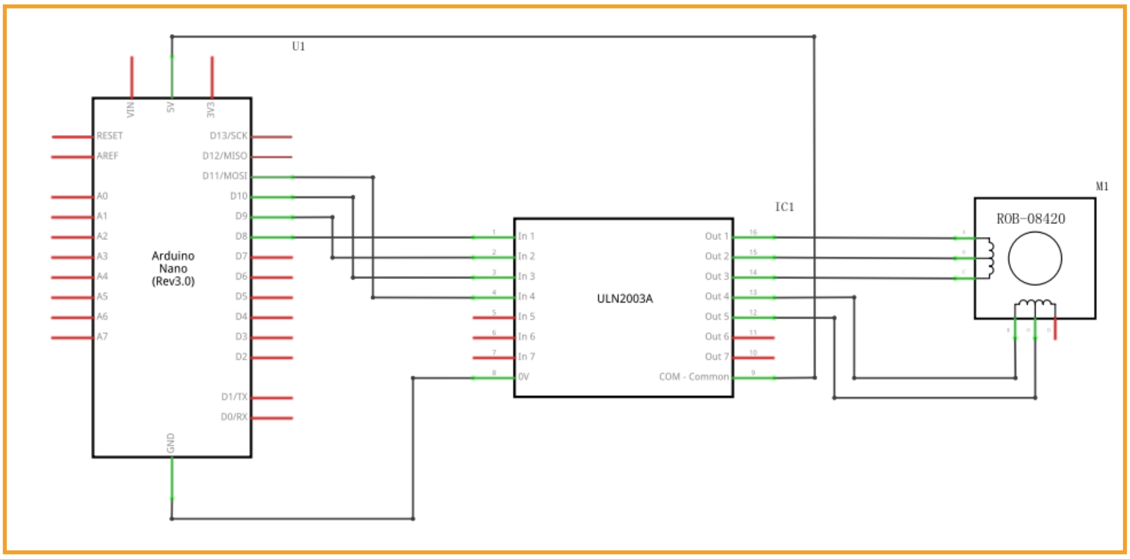
The stepping motor driver board:



Component List

- Emakefun Arduino Nano mainboard
- Breadboard
- USB cable
- Stepper Motor *1
- X113647 *1
- Some wires

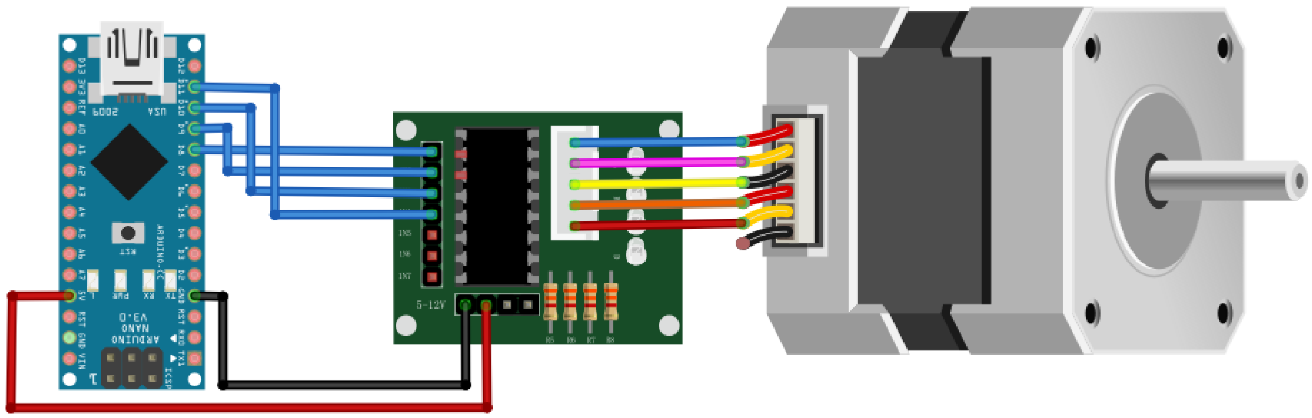
Schematic Diagram



Wiring of Circuit

Arduino Uno R3	ULN2003
8	IN1
9	IN2
10	IN3
11	IN4
GND	-
VCC	+

ULN2003	Stepper Motor
A	Blue
B	Pink
C	Yellow
D	Orange
VCC	Red



Principle of Program

We choose half step drive, so the stepping motor is driven according to the above diagram (c). Half step means the step angle shorten to half of the original: 5.625/2 = 2.8125. To make the stepping motor equipped with speed reducer rotate a whole circle, it takes (360/5.625/2) * 64=8256 pulses.

Code

```
#define  STEPPER_MOTOR_A   8
#define  STEPPER_MOTOR_B   9
#define  STEPPER_MOTOR_C   10
#define  STEPPER_MOTOR_D   11

#define  STEPPER_MODE_4    4
#define  STEPPER_MODE_8    8
boolean StepperDir = 0;

int stepperSpeed = 5 , CurrentStep = 0 ; //set Stepper motor speed  delayms
int stepsum = 0 ;
void setup()
{
  pinMode(STEPPER_MOTOR_A, OUTPUT);
  pinMode(STEPPER_MOTOR_B, OUTPUT);
  pinMode(STEPPER_MOTOR_C, OUTPUT);
  pinMode(STEPPER_MOTOR_D, OUTPUT);
  Serial.begin(115200);
  Serial.println("Stepper Motor And Start :");
}

void loop()
{
  switch ( CurrentStep )
  {
    case 0:
      digitalWrite(STEPPER_MOTOR_A, LOW);
      digitalWrite(STEPPER_MOTOR_B, HIGH);
      digitalWrite(STEPPER_MOTOR_C, HIGH);
      digitalWrite(STEPPER_MOTOR_D, HIGH);
      break;
    case 1:
      digitalWrite(STEPPER_MOTOR_A, LOW);
      digitalWrite(STEPPER_MOTOR_B, LOW);
      digitalWrite(STEPPER_MOTOR_C, HIGH);
      digitalWrite(STEPPER_MOTOR_D, HIGH);
      break;
    case 2:
      digitalWrite(STEPPER_MOTOR_A, HIGH);
      digitalWrite(STEPPER_MOTOR_B, LOW);
```

```
    digitalWrite(STEPPER_MOTOR_C, HIGH);
    digitalWrite(STEPPER_MOTOR_D, HIGH);
    break;
  case 3:
    digitalWrite(STEPPER_MOTOR_A, HIGH);
    digitalWrite(STEPPER_MOTOR_B, LOW);
    digitalWrite(STEPPER_MOTOR_C, LOW);
    digitalWrite(STEPPER_MOTOR_D, HIGH);
    break;
  case 4:
    digitalWrite(STEPPER_MOTOR_A, HIGH);
    digitalWrite(STEPPER_MOTOR_B, HIGH);
    digitalWrite(STEPPER_MOTOR_C, LOW);
    digitalWrite(STEPPER_MOTOR_D, HIGH);
    break;
  case 5:
    digitalWrite(STEPPER_MOTOR_A, HIGH);
    digitalWrite(STEPPER_MOTOR_B, HIGH);
    digitalWrite(STEPPER_MOTOR_C, LOW);
    digitalWrite(STEPPER_MOTOR_D, LOW);
    break;
  case 6:
    digitalWrite(STEPPER_MOTOR_A, HIGH);
    digitalWrite(STEPPER_MOTOR_B, HIGH);
    digitalWrite(STEPPER_MOTOR_C, HIGH);
    digitalWrite(STEPPER_MOTOR_D, LOW);
    break;
  case 7:
    digitalWrite(STEPPER_MOTOR_A, LOW);
    digitalWrite(STEPPER_MOTOR_B, HIGH);
    digitalWrite(STEPPER_MOTOR_C, HIGH);
    digitalWrite(STEPPER_MOTOR_D, LOW);
  default:
    digitalWrite(STEPPER_MOTOR_A, LOW);
    digitalWrite(STEPPER_MOTOR_B, LOW);
    digitalWrite(STEPPER_MOTOR_C, LOW);
    digitalWrite(STEPPER_MOTOR_D, LOW);
    break;
}
if ( StepperDir ) {
  CurrentStep++;
}
```

```
  else {
    CurrentStep--;
  }
  if ( CurrentStep >= STEPPER_MODE_8 ) {
    CurrentStep = 0;
  }
  if ( CurrentStep < 0 ) {
    CurrentStep = STEPPER_MODE_8 - 1 ;
  }
  delay(stepperSpeed);
  stepsum++ ;
  if ( stepsum == 4096 )
  {
    Serial.println(stepsum);
    stepsum = 0 ;
    delay(5000);
  }
}
```

Experiment Result