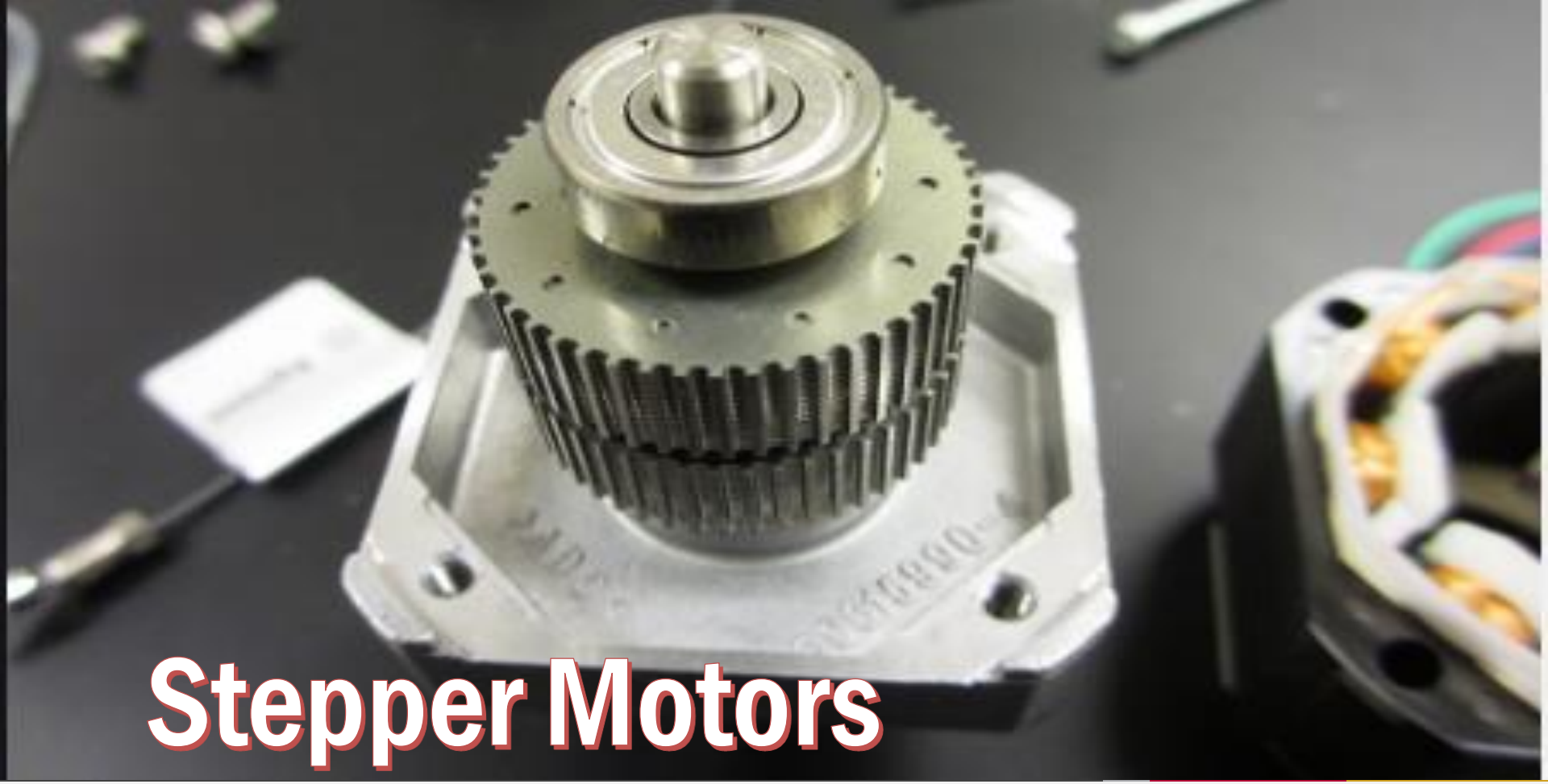
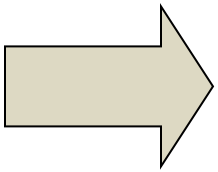


Stepper Motors

ECET 20900

IUPUI

Outline



Stepper Motors

- Advantages/Disadvantages
- Operation
- Driving the Stepper Motor through Code
- Flowchart Operation
- A very Simple Program
- Interface Problems

Stepper Motors

- There are many kinds of electric motors
  - See Wikipedia article
- In many motors, the rotor spins continuously, with no way of precisely controlling the motor's rotational position
- Stepper motors and servos are two widely used kinds of motors whose position can be precisely controlled
- Today we will talk about stepper motors

Stepper Motors: Advantages

Advantages:

- **Positioning**
  - Since steppers move in precise repeatable steps, they excel in applications requiring precise positioning such as:
    - 3D printers
    - CNC: computer numerical control
      - automated control of factory tools and machining
    - Camera platforms
    - X,Y Plotters
    - Some disk drives also use stepper motors to position the read/write head.
- **Speed Control**
  - Precise increments of movement also allow for excellent control of rotational speed for applications such as:
    - process automation and robotics
- **Low Speed Torque**
  - Normal DC motors don't have very much torque at low speeds. A Stepper motor has maximum torque at low speeds:
    - So, they are a good choice for applications requiring:
    - low speed with high precision

Stepper Motors: Disadvantages

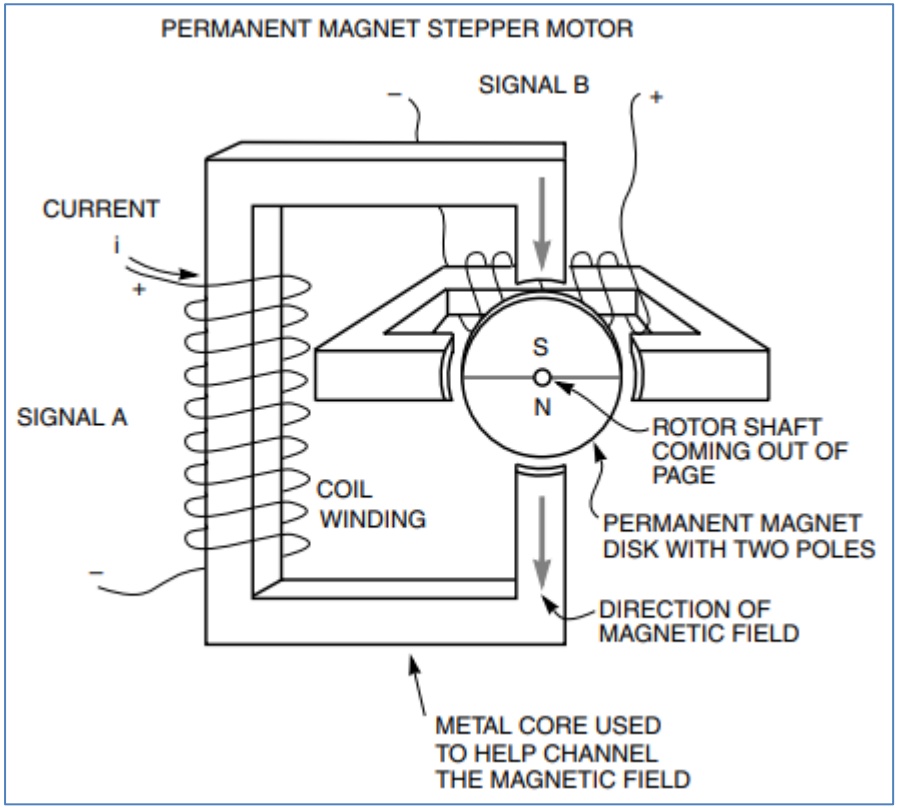
Dis-Advantages:

- **Low Efficiency**
  - Unlike DC motors
    - Stepper motor current consumption is independent of load
    - They draw the most current when they are doing no work at all
    - Because of this, they tend to run hot
- **Limited High-Speed Torque**
  - In general
    - Stepper motors have less torque at high speeds than at low speeds
    - Some steppers are optimized for better high-speed performance….
      …..but they need to be paired with an appropriate driver to achieve that performance
- **No Feedback**
  - Unlike servo motors
    - Most steppers do not have integral feedback for position
    - Although great precision can be achieved running ‘open loop’….
      ….. Limit switches or ‘home’ detectors are typically required for safety and/or to establish a reference position

https://learn.adafruit.com/all-about-stepper-motors/what-is-a-stepper-motor

Stepper Motors Operation

- Generally, a stepper motor consists of:
  - a stator,
  - a rotor with a shaft,
  - and coil windings.
- The stator is a surrounding casing that remains stationary and is part of the motor housing,
- The rotor is a central shaft within the motor that actually spins during use.
- Rotor shaft will turn when current is applied to appropriate input contact signal lines.
- Four contacts signal lines (a, $\bar{a}$, b, $\bar{b}$)



Stepper Motors Operation

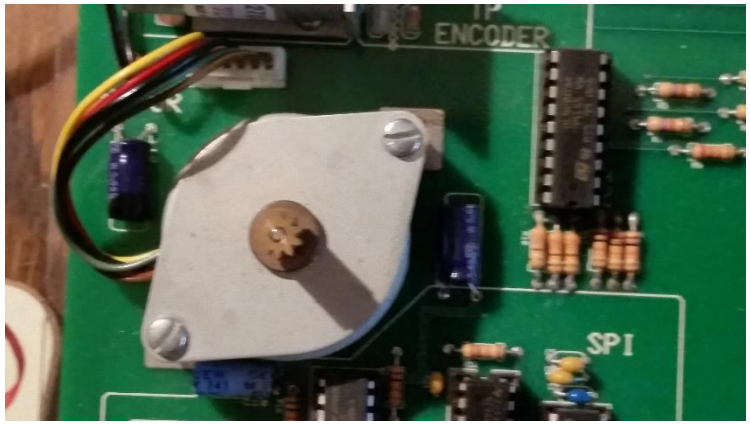
- A stepper motor when digitally controlled, allows precise control over the position of the motor's rotor.
- Changes in the digital input, rotates the motor's rotor by a precise amount, which is called a step or step angle.
- Depending on the motor, this step angle may be as small as 1° (or less) or as large as 45°.
  - On our motors, the step angle is 7.5°.
  - Meaning it takes 48 steps to turn our stepper motor one full rotation.

$$360° / 7.5° = 48$$

- Some more common stepper motors have step angles of 1.8°, 3.6°

Stepper Motor:

- Our training board's stepper motor
  - Stepper motor: PF35T
  - http://nipponpulse.com/catalog/document/5852ff62232b5_pf35t.pdf
  - Motor Enabled switch must be in the left position.
    - (jumper block J1, found just above the DC motor on our boards)

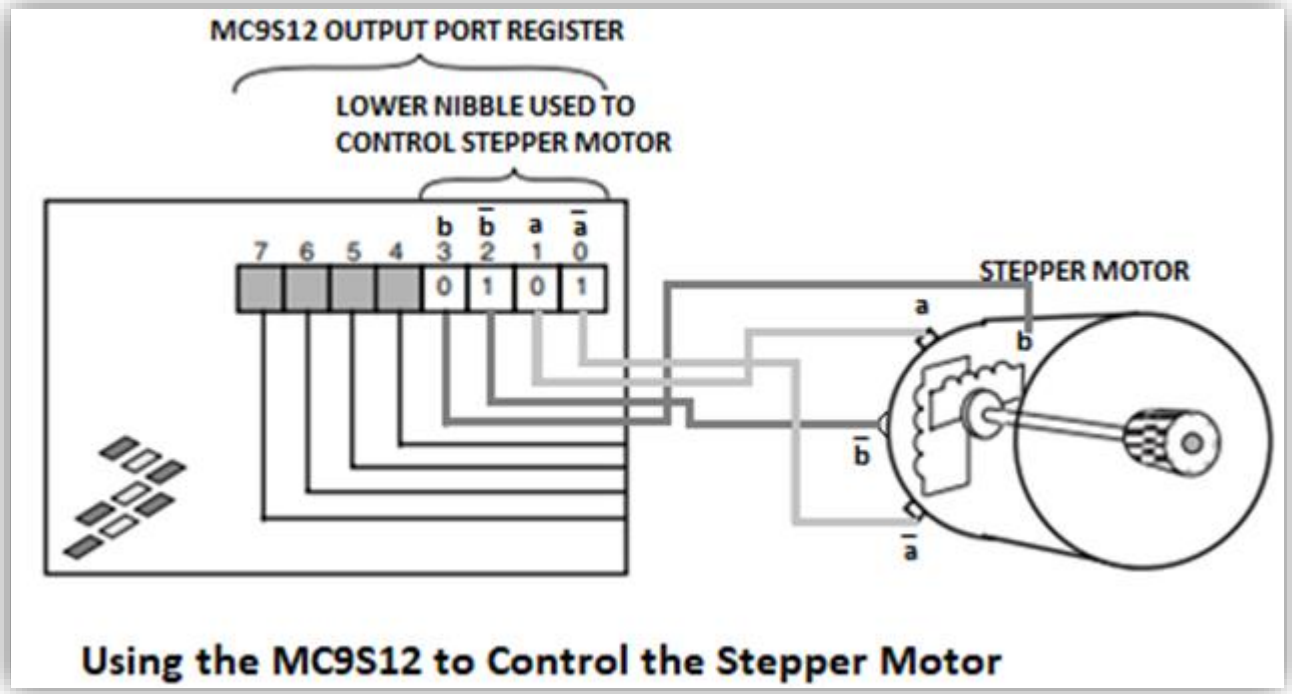


Specifications

Specifications	Unit	PF35T-48			
Type of Winding		Unipolar		Bipolar	
Excitation Mode*		Full step (2-2)			
Step Angle	°	7.5 ±5%			
Steps Per Revolution*		48			
Rated Voltage	V	12	5	12	5
Resistance	Ω	70 ±7%	12 ±7%	72 ±7%	16 ±7%
Inductance	mH	30	6.5	60	6.2
Holding Torque	mN·m	18	18	27	27
Rotor Inertia	kg·m²	2.7 x 10^{-7}			
Starting Pulse Rate*	pps	600			
Slewing Pulse Rate*	pps	610			
Operating Temp. Range	°C	-10 to +50			
Temperature Rise*	°C	70			
Weight	g	77			

Stepper Motors: Driving the Stepper Motor through Code

- Our training board's stepper motor
  - 4-signal pin, two-phase Permanent Magnet (PM) stepper motor, two poles on the PM disk
  - Values to drive our stepper motor are: 0x05, 0x09, 0x0A, 0x06 (0101, 1001, 1010, 0110)
- With an appropriate algorithm, we can use pins Port D[3:0] of the MC9S12 to produce the waveforms needed to drive a stepper motor.
- Where pins Port D3:B, D2:$\bar{B}$, D1:A, D0:$\bar{A}$



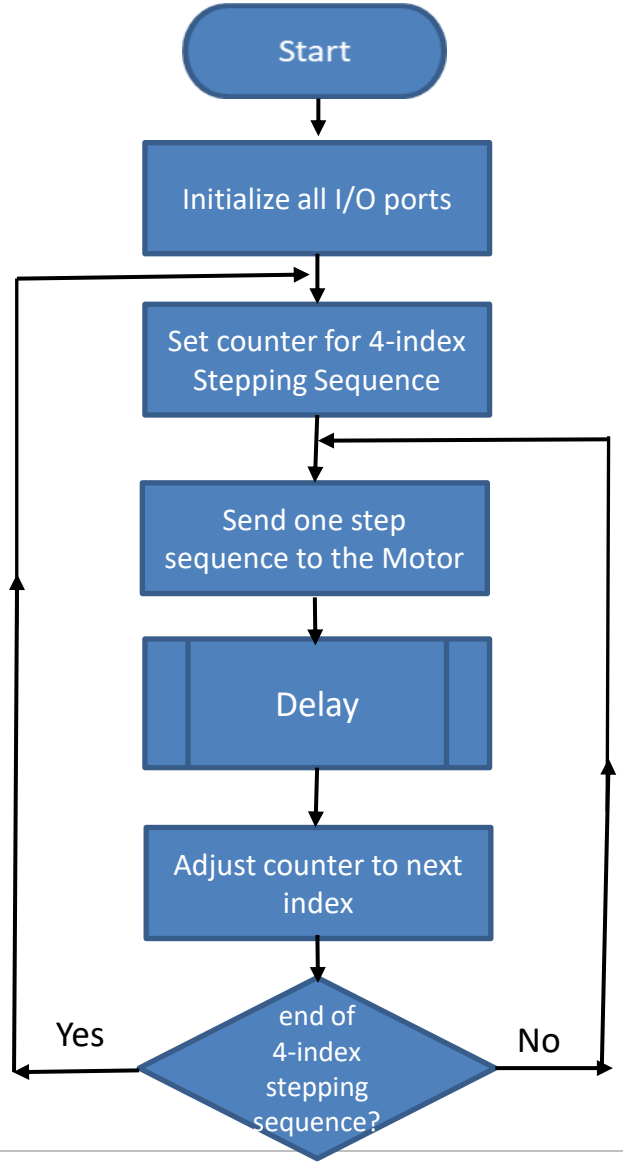
https://www.nxp.com/docs/en/application-note/AN2974.pdf

Stepper Motors: Driving the Stepper Motor through Code

- With an appropriate algorithm, we can use Port D pins D[3:0] of the HCS12 to produce the waveforms needed to drive a stepper motor.
- The general flow of the algorithm can be similar to the flow of a state machine:
  - Put list of values in an array (assembly list)
  - Initialize the I/Os
  - Set the bits in register D to a state: a value in the list array
  - Wait a discrete amount of time
  - and set the bits in register D to the next state: next value in the list array
  - For each change in the microcontroller register state
    - a change is produced in the waveform that causes the motor to rotate a fixed amount.
    - The period of time required between register states will vary depending upon the motor and the performance desired,
    - but is usually on the order of milliseconds
- Remember the following:
  - If the delay between changes to the microcontroller register states is too short,
    - the motor will not physically be able to move fast enough to keep up with the register state changes.
  - A delay that is too long could create a motor response with noticeably rigid movements and choppy noises with each step.

https://www.nxp.com/docs/en/application-note/AN2974.pdf

Stepper Motor: Flowchart Operation



- Stepper Motor “High-Level” Operation

Stepper Motors: With a very Simple Program

```
void main(void)
{
    unsigned char stepValues[ ] = {0x05,0x09,0x0A,0x06};
    char index = 0;

    DDRD |= 0x0F;

    for(;;)
    {
        PORTD = stepValues[index];
        Delay();
        index++;
        if(index >= 4)
        {
            index = 0;
        }

        _FEED_COP(); /* feeds the dog */
    } /* loop forever */
  /* please make sure that you never leave main */
}
```

Stepper Motors: Interfacing Problems

- Two interfacing problems arise when driving a motor from a digital system:
  1. Motors introduce a great deal of electrical “noise” into a system. This noise can disrupt the operation of sensitive digital circuits such as the MC9S12
  2. Motors consume more current than most digital outputs can supply. Therefore, we can’t connect the motor directly to the MC9S12’s output pins
  3. For our Training board we have a Darlington Transistor motor driver array to protect the microcontroller’s pins from the excessive current
  4. A well specified motor driver is the success to a motor in a digital driven system.

Summary

- Stepper Motors
  - Advantages/Disadvantages
  - Operation
  - Driving the Stepper Motor through Code
  - Flowchart Operation
  - A very Simple Program
  - Interface Problems