MAE/ECE 5320 Mechatronics

2025 Spring semester

Lecture 05

Tianyi He
Utah State University

Content

- Arduino microcontroller background
- Arduino Uno
- Example: Mini Segway control system hardware
- DC motor actuator
- Step and servo actuator

Content

- ❑ Arduino microcontroller background
- ❑ Arduino Uno
- ❑ Example: Mini Segway control system hardware
- ❑ DC motor actuator
- ❑ Step and servo motor actuator

Arduino microcontroller background (1)

Open-source electronics platform https://www.arduino.cc/

- Easy-to-use hardware and software, programming, and implementation
  1) Arduino can work with Matlab and Simulink
  2) Arduino software IDE

- Many extensible hardware with wide application scope
- Internet of things, smart home, DIYs

Arduino microcontroller background (2)

Arduino Programming method

IDE software

- ❑ *Programming based on hardware specifications*
- ❑ *Faster compiling, computing codes*
- ❑ *User-defined control sequence*
- ❑ *Difficult for complex system*

https://www.arduino.cc/en/Tutorial/BuiltInExamples
A collections of simple and basic examples

https://create.arduino.cc/projecthub A project hub of various applications

Programming with MATLAB and SIMULINK

- ❑ Graphic programming in Simulink
  - More interactive and understandable
  - Real-time visible signal response
  - Programming on diagram
  - Physics based programming
  - MATLAB convert blocks to C/C++ code, thus easier for complex system
- ❑ Compatible with control design, signal *processing, off-line simulations, etc.*

Note that this class programs Arduino in Matlab and Simulink

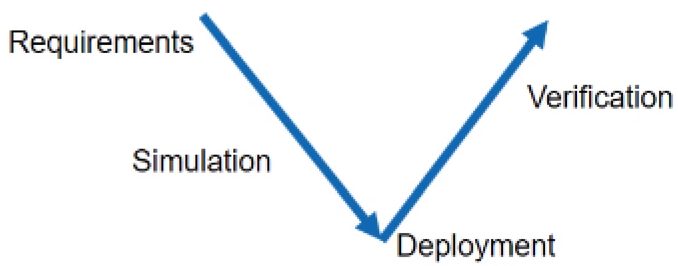
Why programming via MATLAB

- Save time and resources on hardware programming
- Deploy vehicle dynamics in Simulink
- Visual programming by blocks and results
- Validation, test by hardware-in-loop simulation accelerates development
- a coding standard is required for ISO 26262 compliance (automotive)

- MATLAB has developed interface with Arduino, dSPACE, etc controllers of many platforms.
- Accelerated coding process to deal with booming coding lines.

Apollo ≈145,000 lines

Autonomous car ≈ 100-500 million lines



Codes taking Apollo to the Moon

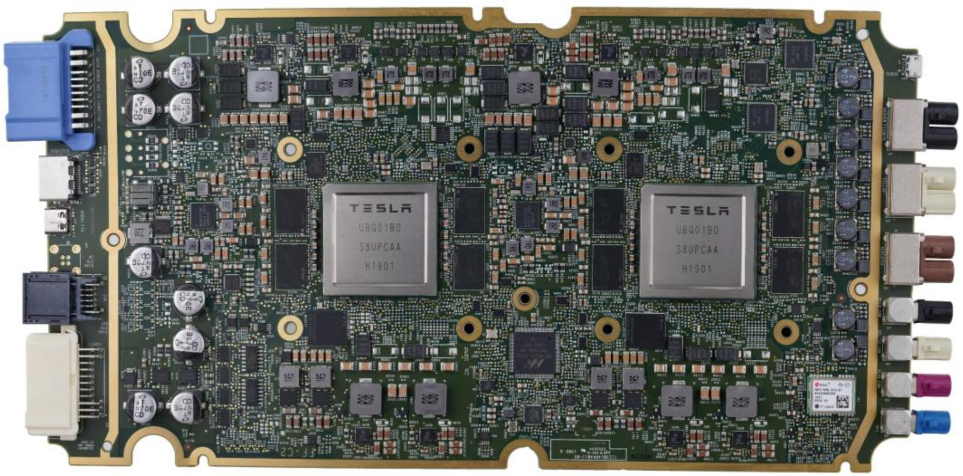
Some Control Hardware

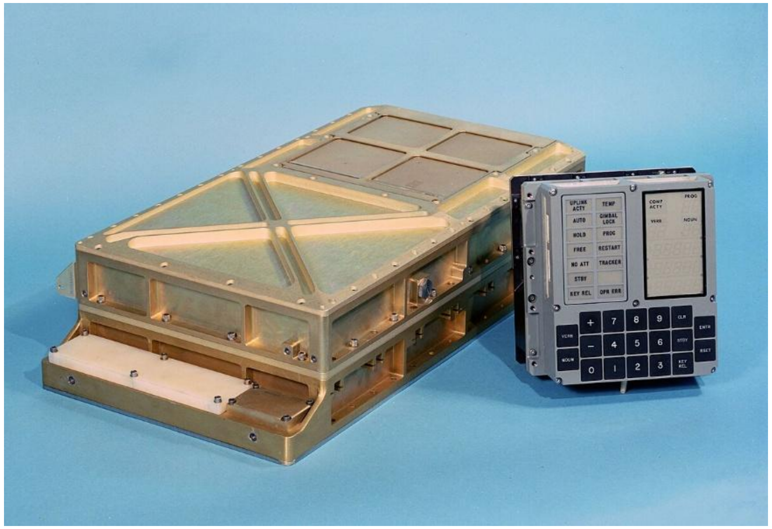
Tesla's vehicle control unit (VCU)

- CPU, 20 cores, 2.35 GHz and 3 NN processor (2.2GHz)
- 16 GB RAM

Apollo's guidance computer

- CPU 2.048 MHz
- 2048 words RAM
- 36,864 words ROM





Example: How automative vehicle control is developed

https://www.youtube.com/watch?v=_UzCuU8CjqY&ab_channel=MATLAB (link for additional materials)

https://www.mathworks.com/videos/adas-and-automated-driving-development-in-matlab-and-simulink-1599585866446.html (second video to watch)

Arduino microcontroller background (3)

Arduino support from Simulink

With MATLAB® Support Package for Arduino® Hardware, you can use MATLAB to interactively communicate with an Arduino board. The package enables you to perform tasks such as:

- ❑ Acquire analog and digital sensor data from an Arduino board
- ❑ Control other devices with digital and PWM outputs
- ❑ Drive DC, servo, and stepper motors (also supports Adafruit Motor Shield)
- ❑ Access peripheral devices and sensors connected over I2C or SPI
- ❑ Communicate with an Arduino board over a USB cable or wirelessly over Wi-Fi, Bluetooth
- ❑ Build custom add-ons to interface with additional hardware and software libraries

Content

- ❑ Arduino microcontroller background
- ❑ Arduino Uno
- ❑ Example: Mini Segway control system hardware
- ❑ DC motor actuator
- ❑ Step and servo motor actuator

Arduino Uno

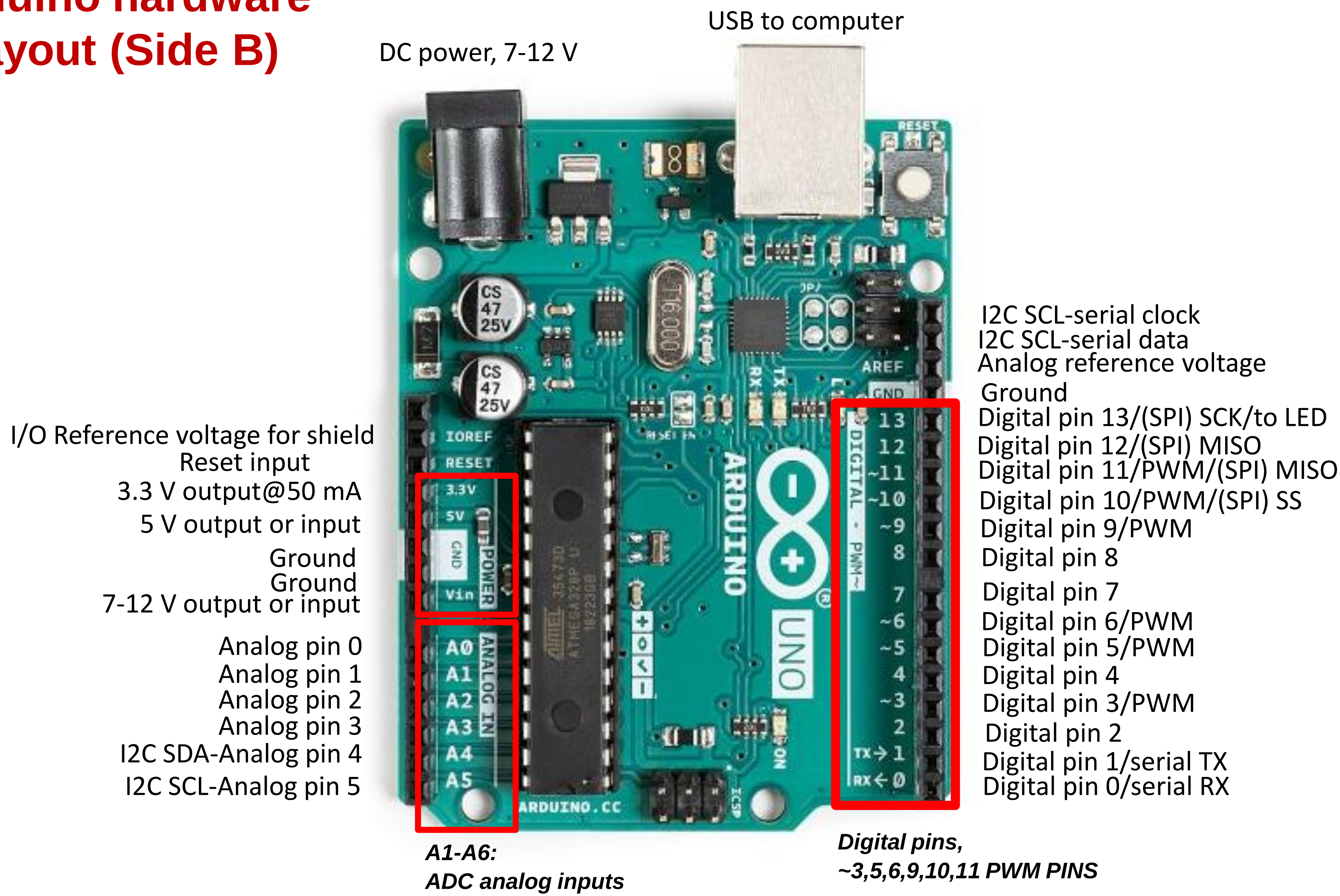
Internet of Things:

- ATMEL 8-bit microcontroller : Atmega328P
- 14 digital I/O pins, 6/14 can be used as PWM digital pins;
- 6 analog inputs
- Simulink or IDE for software development, we will mainly use the Simulink to program in this class
- Wifi version **Arduino Uno WiFi Rev2**

External link: https://store.arduino.cc/usa/arduino-uno-rev3

Arduino Uno

Arduino hardware layout (Side B)



Summary-pinout

I/O(Input and output):

Serial: pin 0 (RX) is used to receive and pin 1 (TX) is used to transmit serial data.

External interrupts: pin2 and pin 3 can be configured to trigger an interrupt.

PWM: pin 3,5,6,9,10,11. (has '~', *[tilde]* before pin number), 8-bits

SPI: 10(SS), 11(MOSI), 12(MISO), 13(SCK) to support SPI communication

LED: pin 13, high-value ON, low-value , OFF

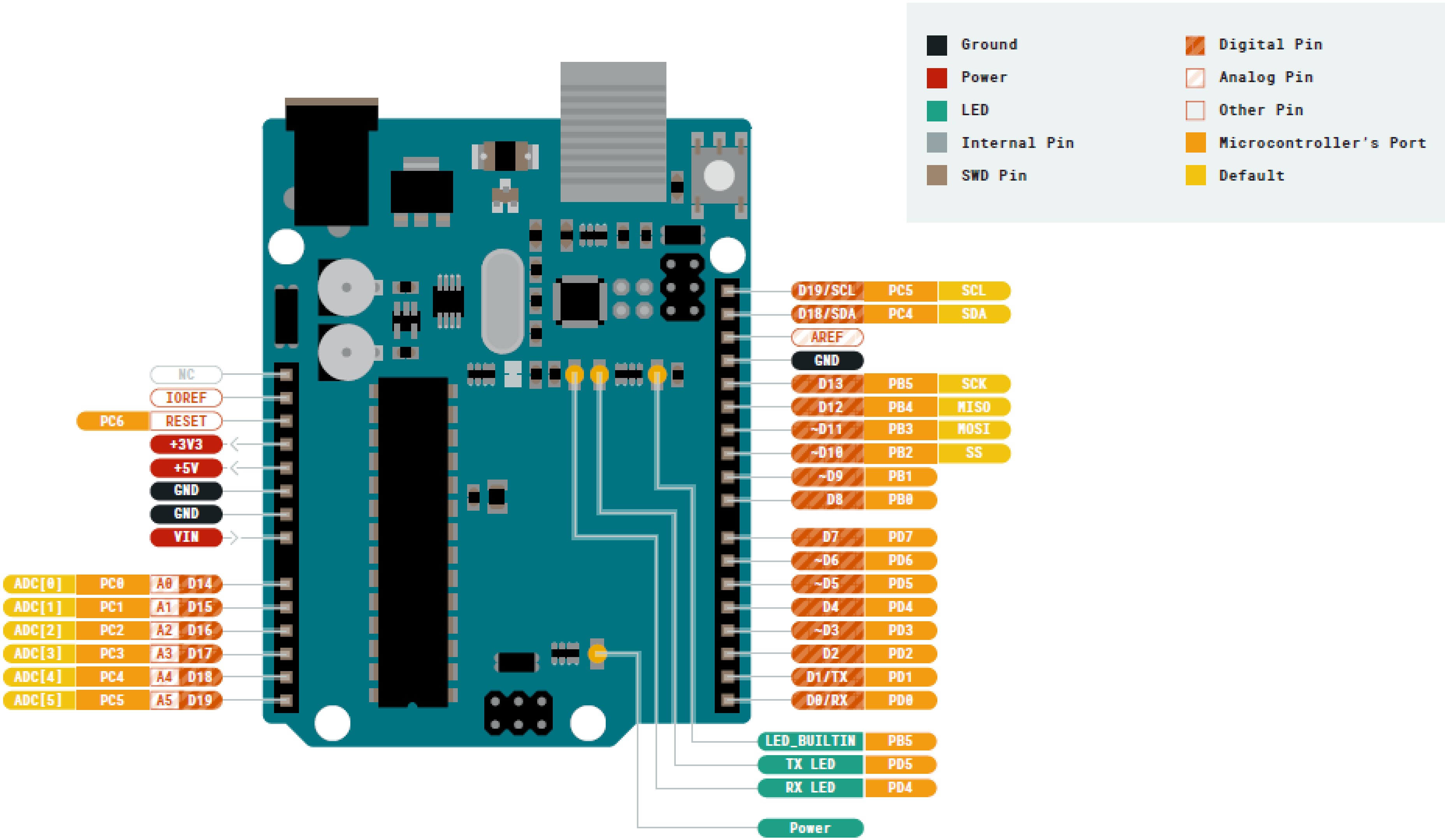
Analog inputs: A0-A5

10-bits resolution, meaning that 1024 divided for default voltage range (0-5V)

AREF: external reference voltage. (0-ref voltage)

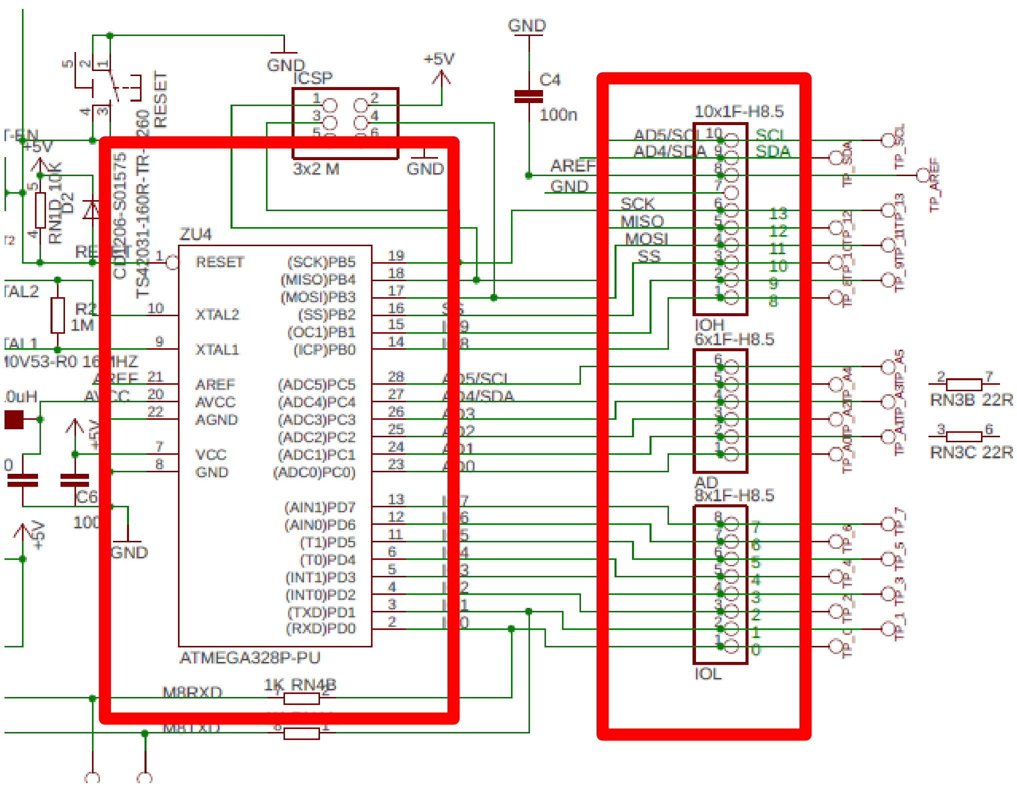
RESET: trigger to reset the microcontroller

Arduino Uno pinout



Arduino Uno

Commercial chips and other mechatronic parts are usually developed and packaged. It is essential to learn how to read technical document. Here, we use Arduino Uno as example, you can find document by https://content.arduino.cc/assets/UNO-TH_Rev3e_sch.pdf .



microcontroller

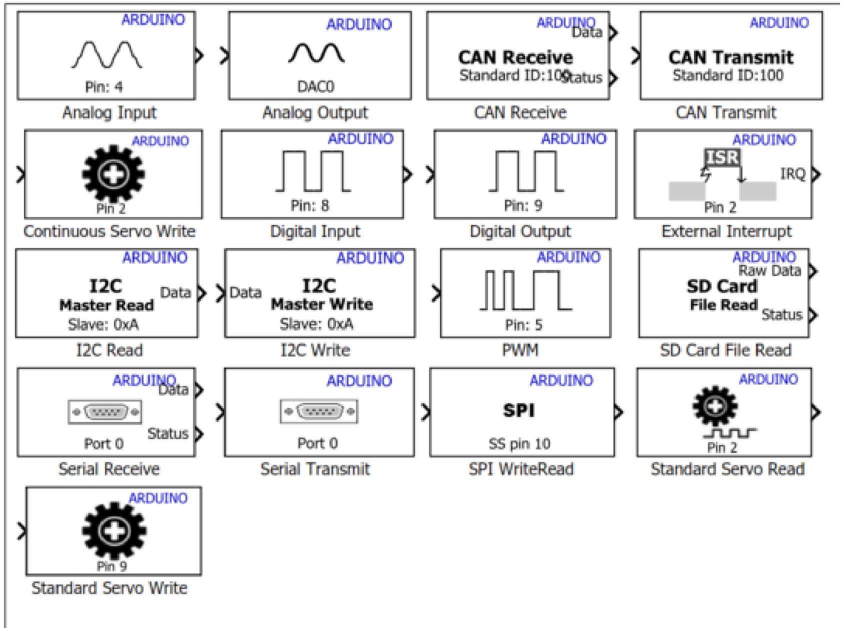
extension pin

Arduino Uno

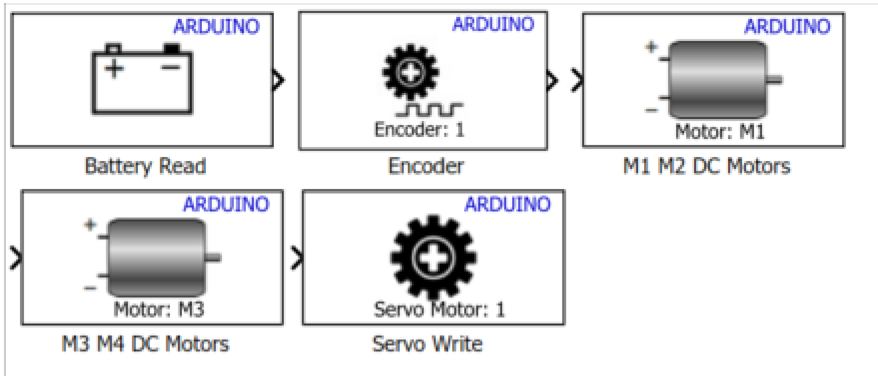
Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

MATLAB Arduino library

Common



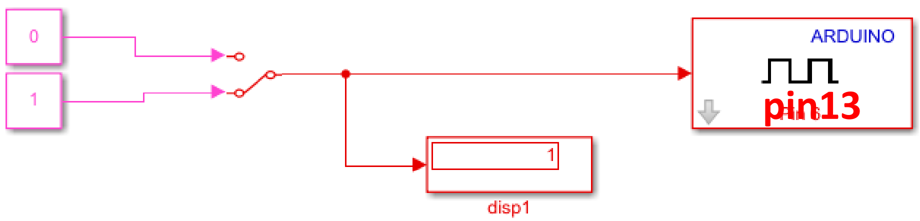
motors

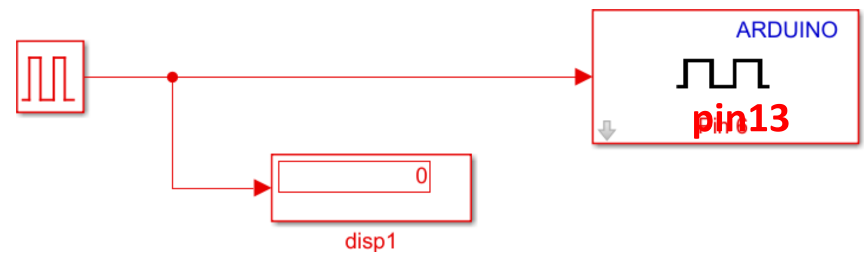


sensors

Arduino Uno

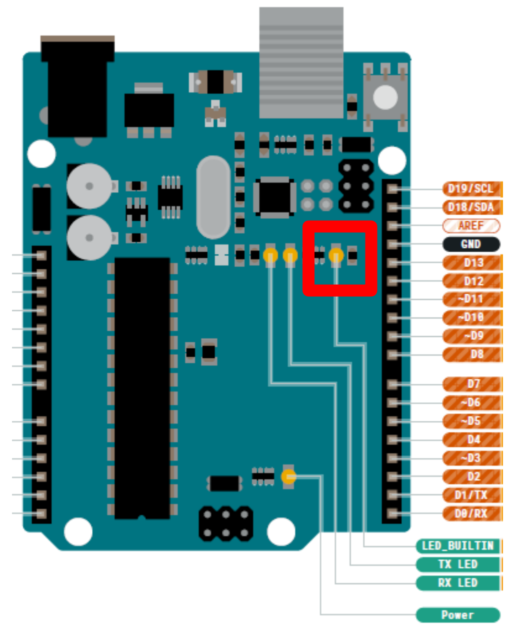
LED control by Arduino Uno using Simulink



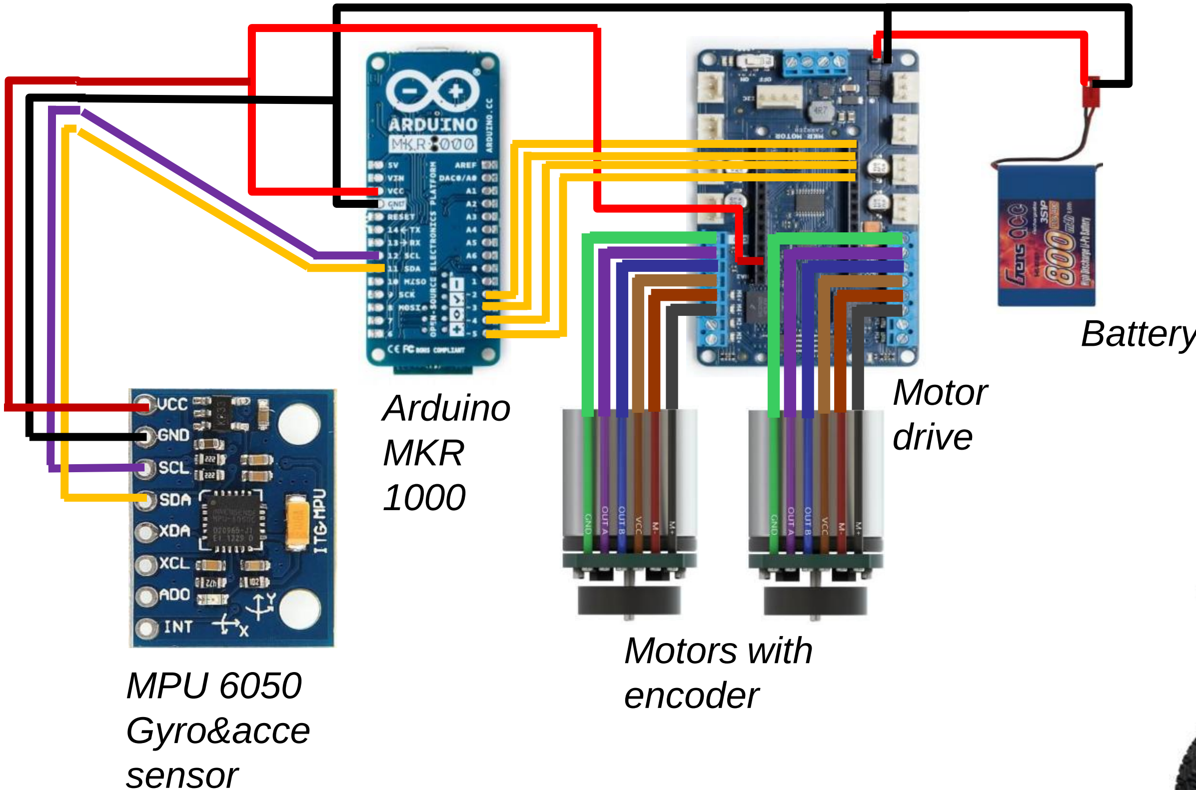


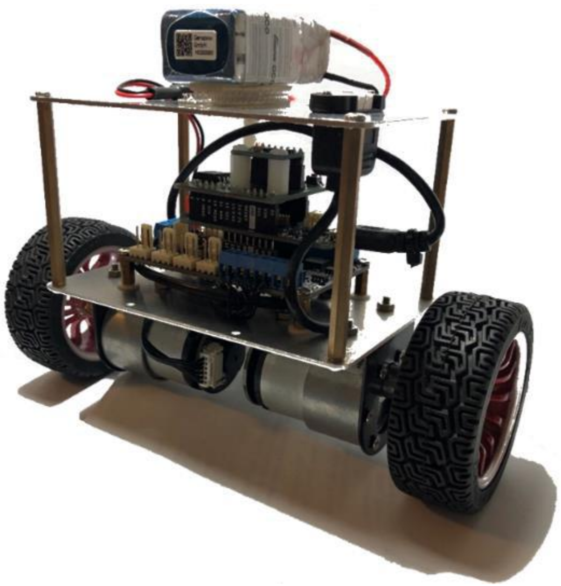
Use switch to control LED ON/OFF;
When input is 1 (high), LED is ON;
When input is 0 (low), LED is off.

Use pulse signal to control
LED period illumination;
Input is 1 or 0 for one second in loop.

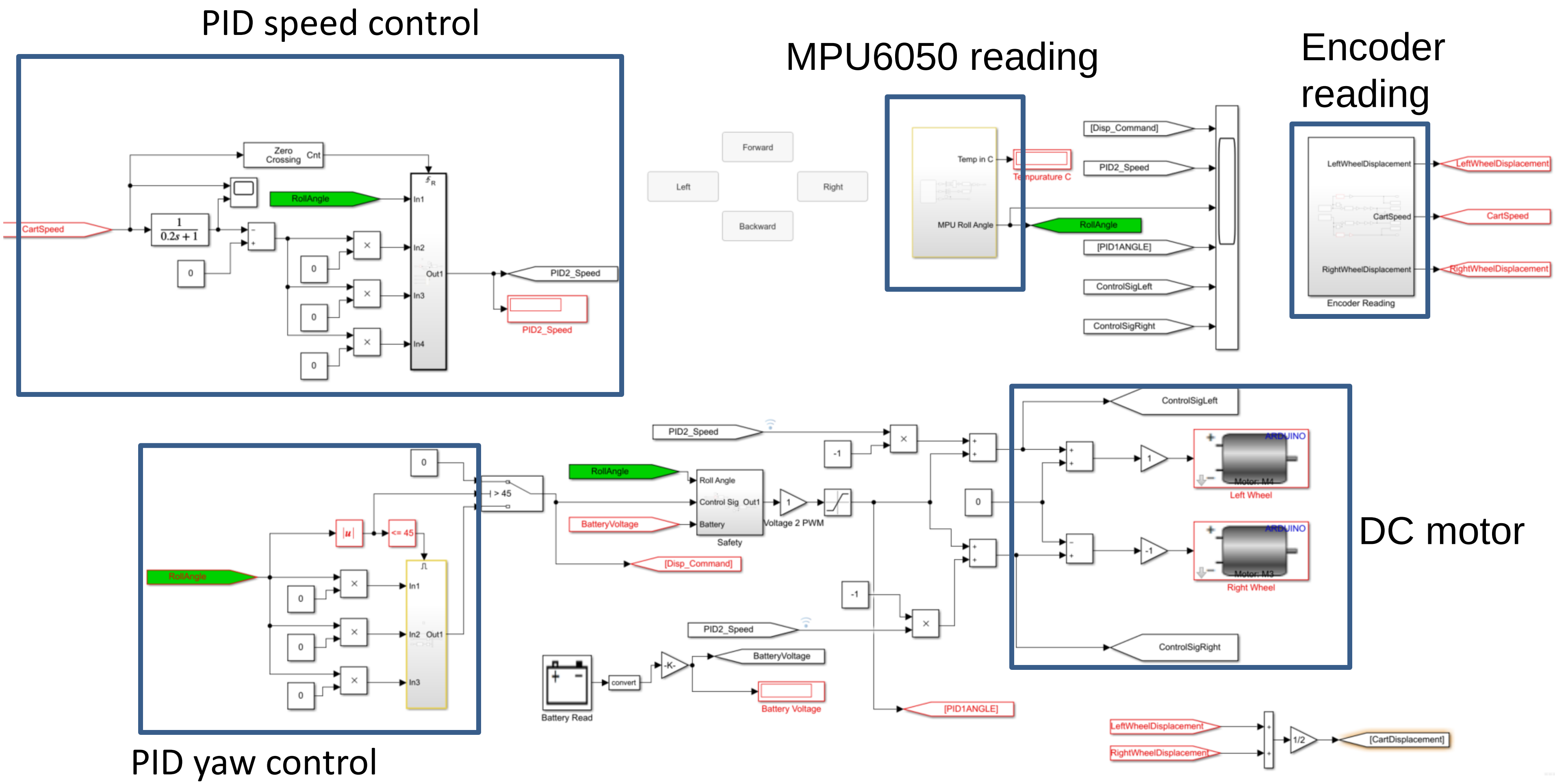


Example: Control system of Segway





Example: Control system of Segway



Content

- Arduino microcontroller background
- Arduino Uno
- Example: Mini Segway control system hardware
- DC motor actuator
- Step and servo motor actuator

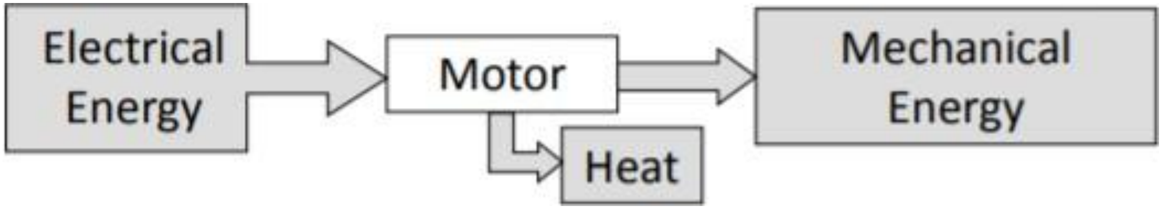
DC motor (1)

A few actuator examples:

1. AC/DC motor
2. Servo motor
3. Step motor

$\}$ *Electric motor*

4. Hydraulic and pneumatic actuator
5. Combustive actuator



Due to friction and resistance

An electric motor converts energy from electrical to mechanical one based on electromagnetism.

DC motor (2)

AC and DC motors

Alternating current → AC motor
high-power level, high power density, variable speed applications

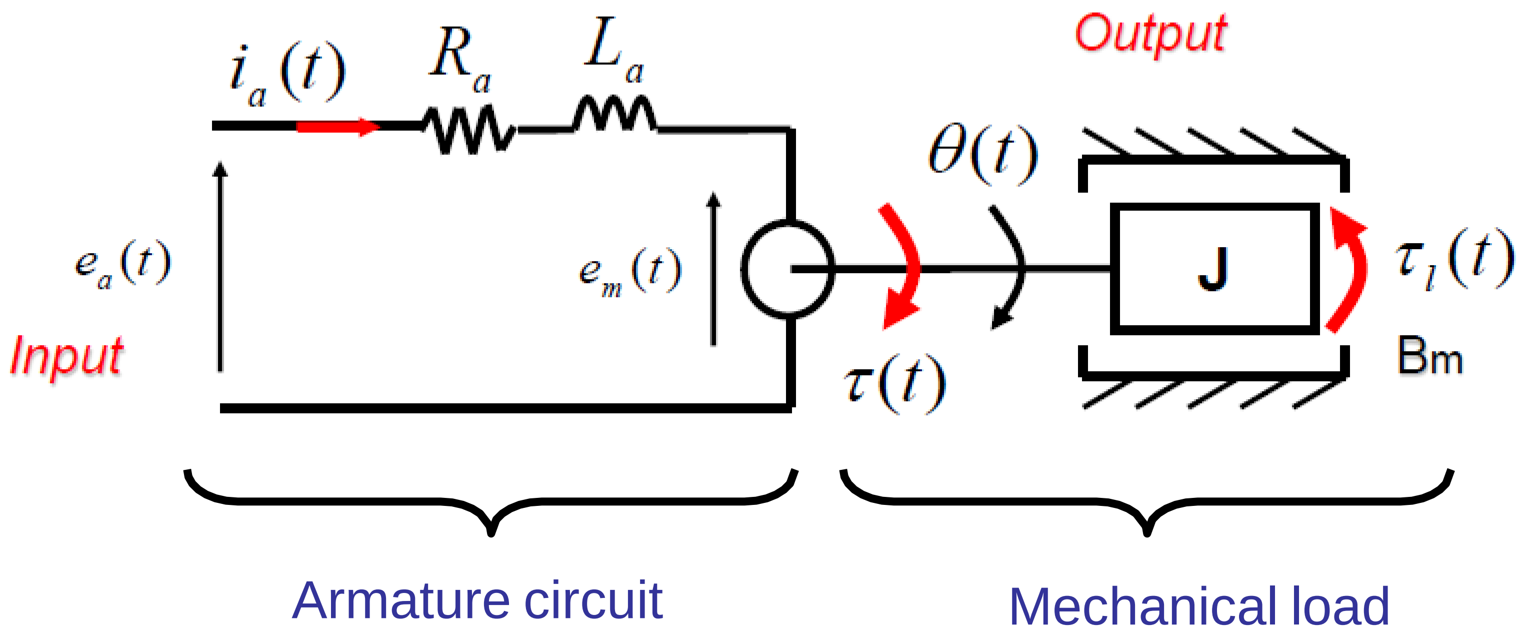
Direct current → DC motor
low-power, high torque, variable speed, mechatronic applications

Basic characteristics of DC motors

- ❑ With fixed load, motor shaft speed proportional to control voltage
- ❑ With fixed control voltage, shaft speed proportional to torque load
- ❑ Shaft torque proportional to motor current, independent of motor voltage
- ❑ Motor efficiency mainly impacted by its electric resistance and mechanical friction

DC motor control will be covered in next lecture (week).

Modeling – DC Motor (lecture 1)



e_a :	applied voltage	θ :	angular position
i_a :	armature current	ω :	angular velocity
e_m	back EMF voltage	J :	rotor inertia
(counter-electromotive force/voltage)		B :	viscous fricton

Modeling – DC Motor Transfer Function (lecture 1)

Note: In many cases $L_a << R_a$. Then, an approximated TF is obtained by setting $L_a = 0$.

$$\frac{\Omega(s)}{E_a(s)} = \frac{K_\tau}{(L_a s + R_a)(Js + B) + K_\tau K_m} \approx \frac{K_\tau}{R_a(Js + B) + K_\tau K_m}$$

$$=: \frac{K}{Ts + 1} \left(K := \frac{K_\tau}{R_a B + K_m K_\tau},\ T = \frac{R_a J}{R_a B + K_m K_\tau} \right)$$

$$\frac{\Omega(s)}{T_l(s)} = \frac{-(L_a s + R_a)}{(L_a s + R_a)(Js + B) + K_m K_\tau} \approx \frac{-R_a}{R_a(Js + B) + K_m K_\tau}$$

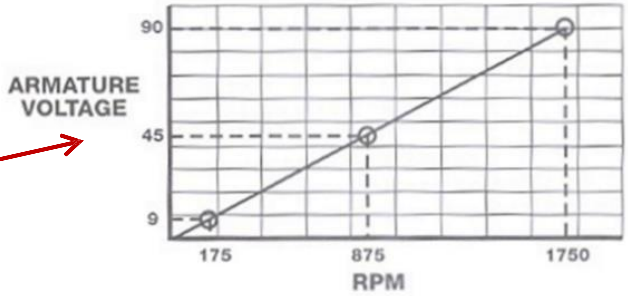
2nd order system ➡ *1st order system*

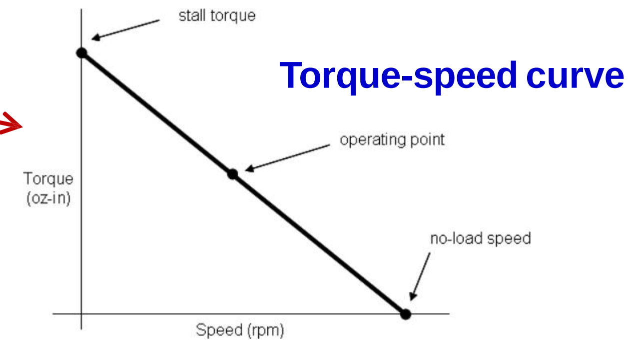
$$\Omega(s) = G_1(s) E_a(s) + G_2(s) T_l(s)$$

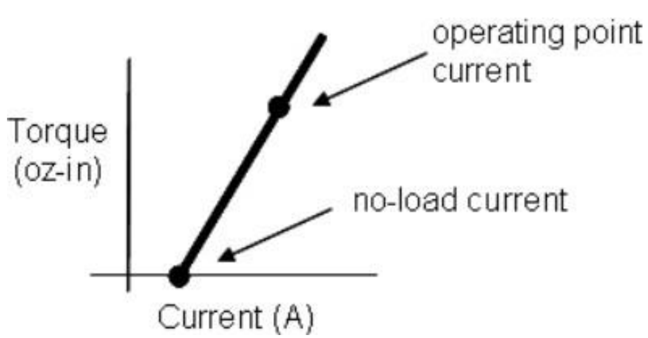
DC motor (3)

Basic characteristics of DC motor

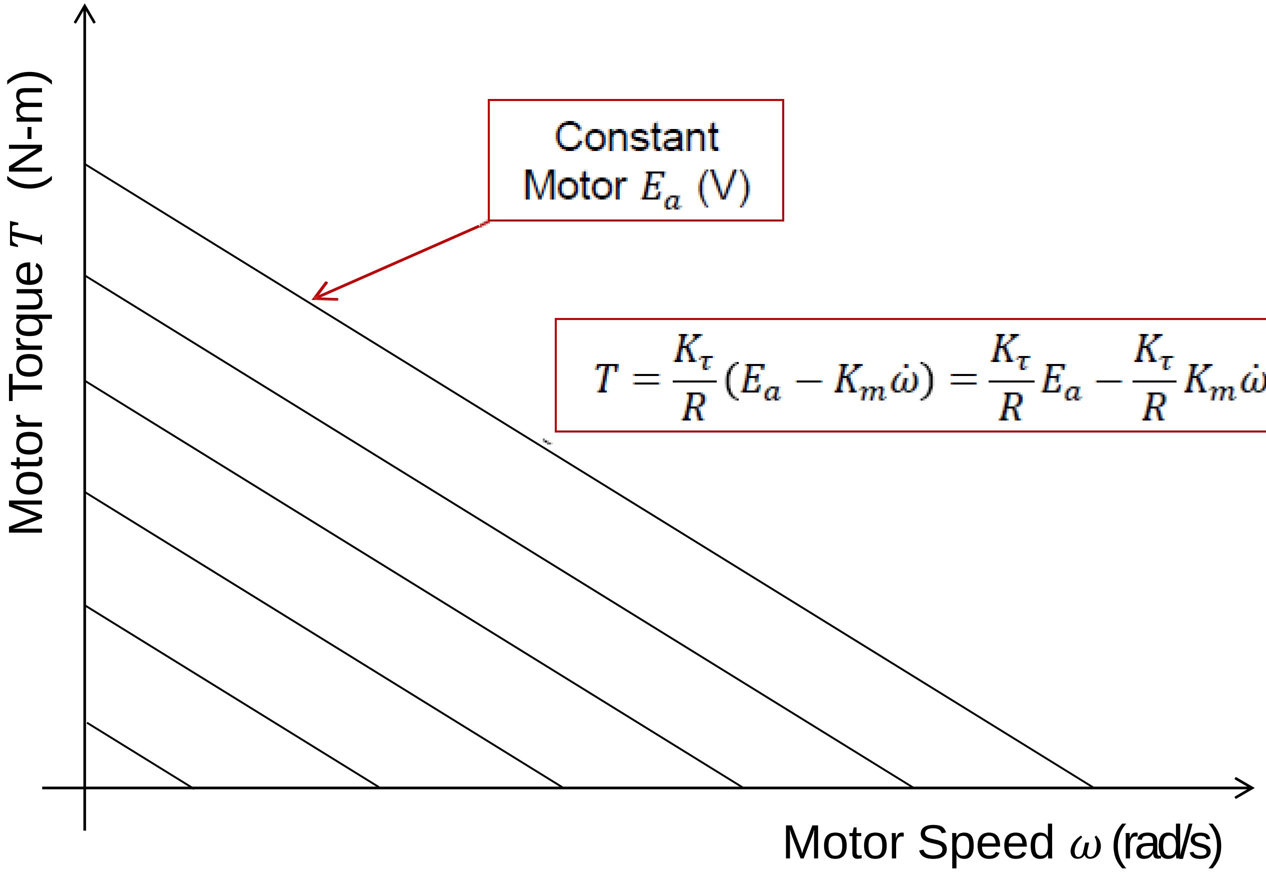
- ❑ With fixed load, motor shaft speed proportional to control voltage
- ❑ With fixed control voltage, shaft speed proportional to torque load
- ❑ Shaft torque proportional to motor current, independent of motor voltage
- ❑ Motor efficiency mainly impacted by its electric resistance and mechanical friction







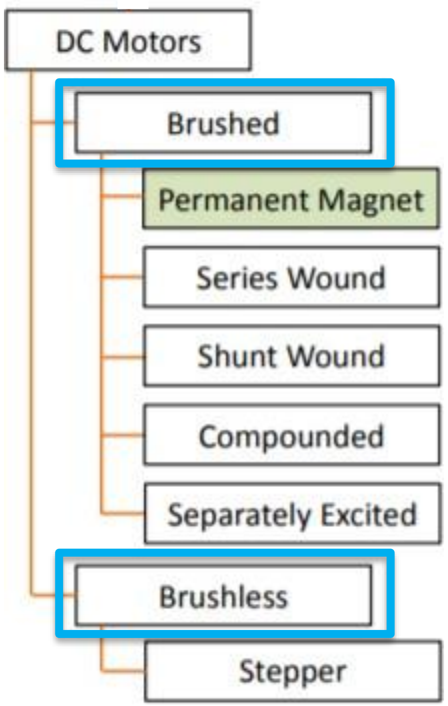
Modeling – DC Motor SS Characteristics

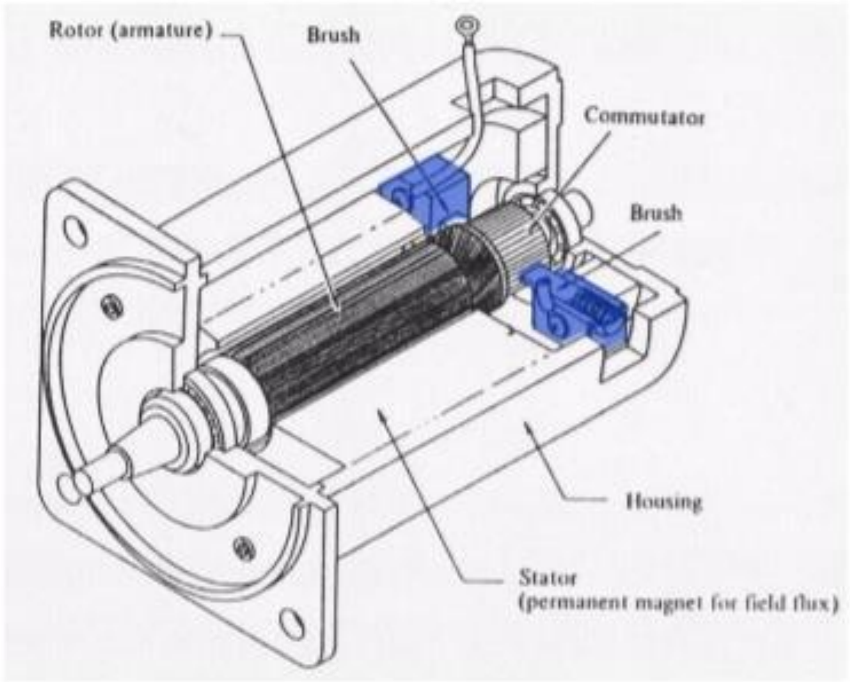


$$T = \frac{K_\tau}{R}(E_a - K_m\dot{\omega}) = \frac{K_\tau}{R}E_a - \frac{K_\tau}{R}K_m\dot{\omega}$$

DC motor (4)

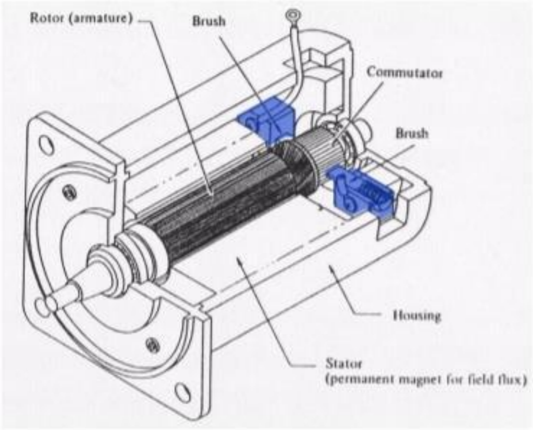
- *Brushed*
- *Brushless*

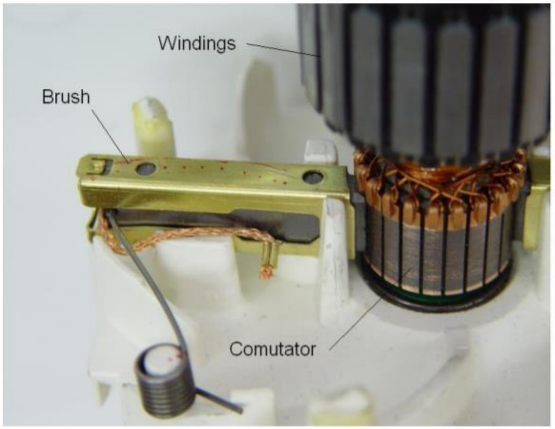




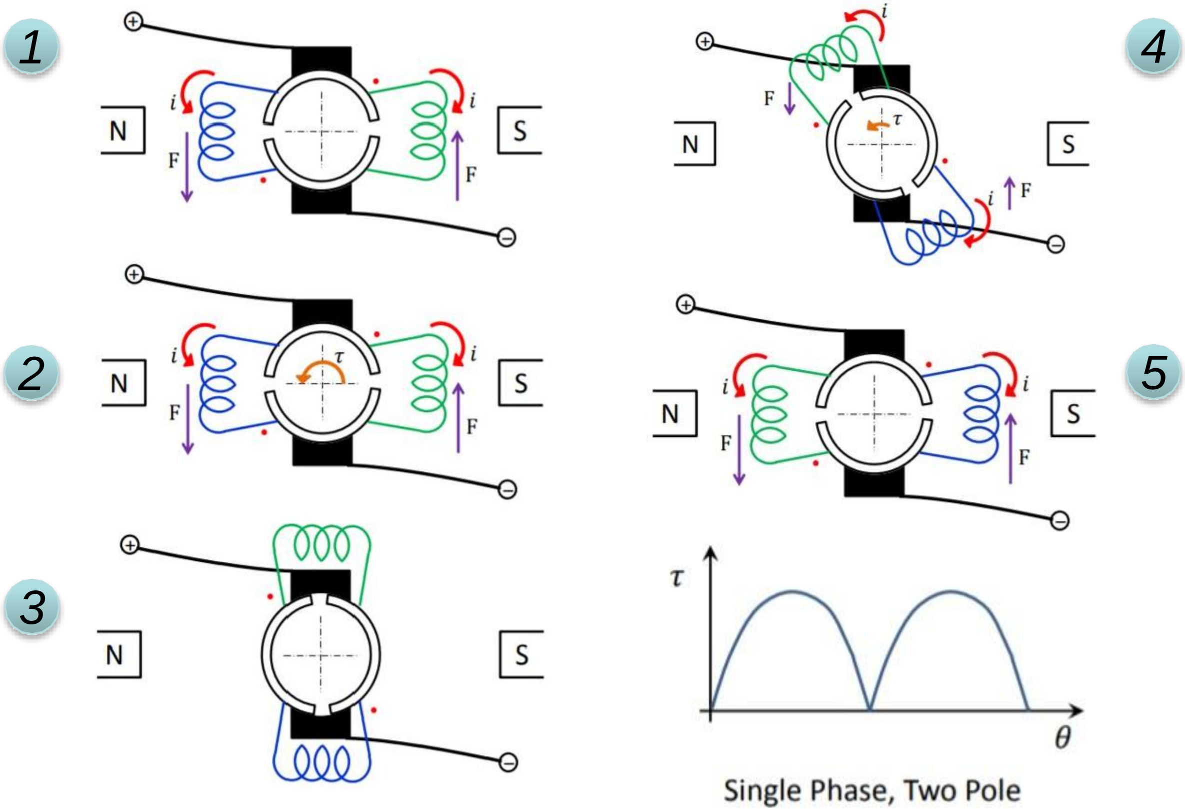
DC motor (5) - Brushed

The current direction is changed by commutator when torque is zero, in order to keep output torque remaining at the same direction.



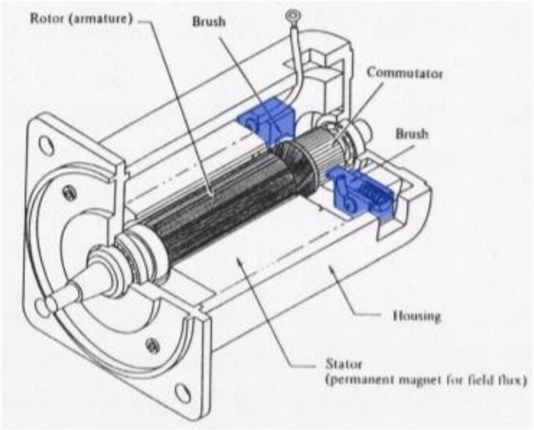


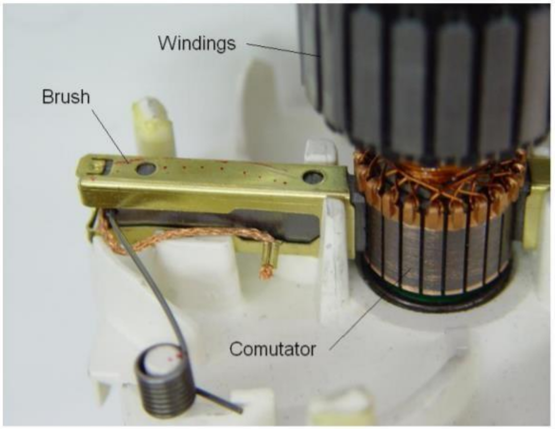
Consider a two-pole commutator case

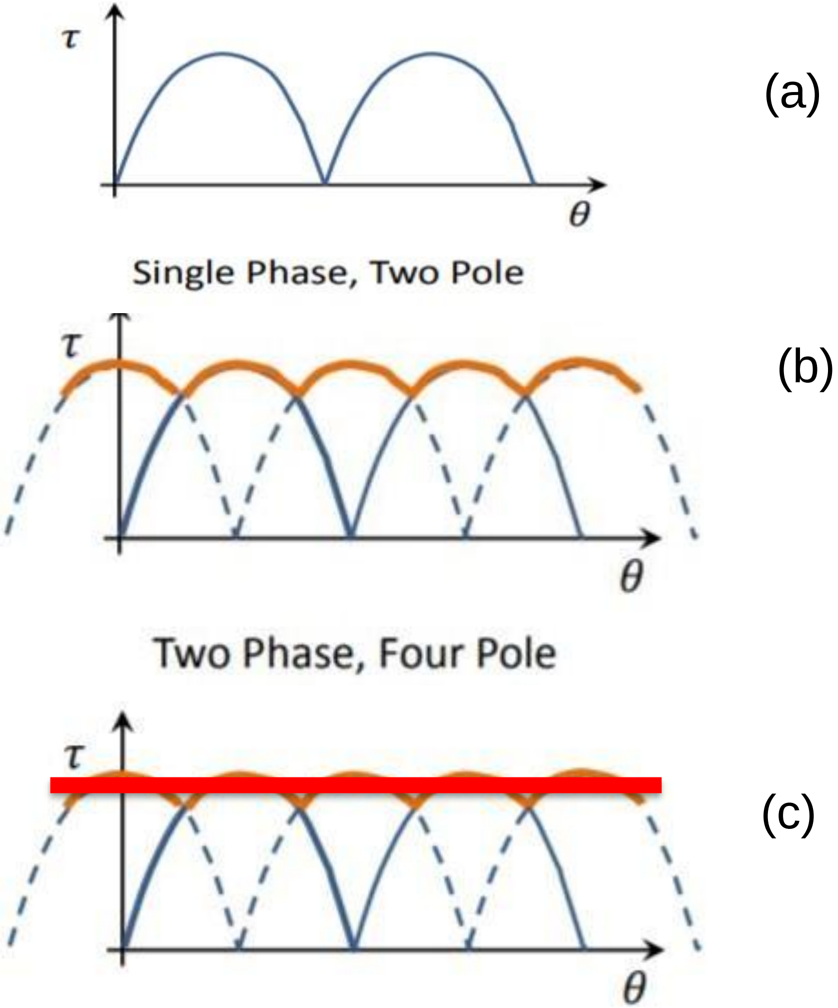


DC motor (6) - Brushed

Note that a DC motor with a two-pole commutator (see (a)) has large torque ripple, as the number of poles increases, the torque ripple reduces and can be considered as constant (see (b) and (c)).



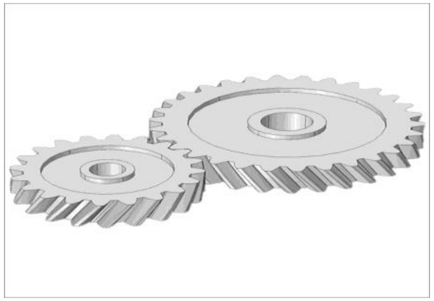




Geared DC motor

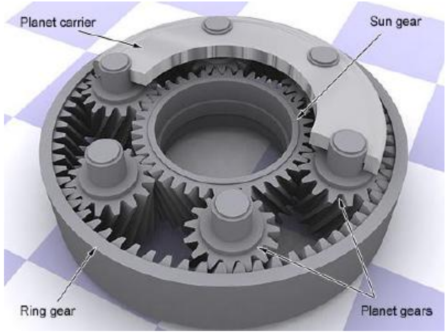
Gear types:

- ❑ Parallel shaft gear



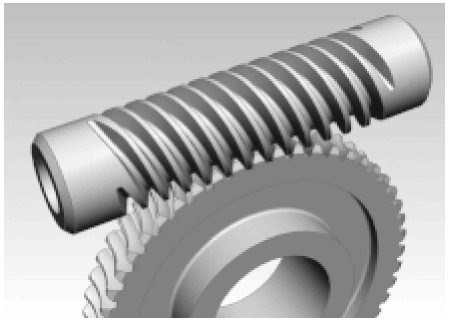
Spur gear
Helical gear

- ❑ Planetary gear



Reduce rotation speed to obtain higher torque

- ❑ Worm gear



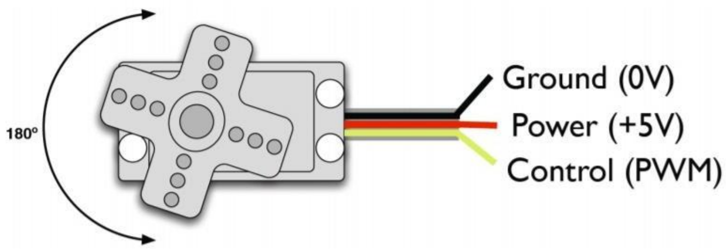
Content

- Arduino microcontroller background
- Arduino Uno
- Example: Mini Segway control system hardware
- DC motor actuator
- Step and servo motor actuator

Step and servo motor (1)

Servo motors

1. Gear system to reduce speed;
2. Potentiometer (voltage divider) for position feedback
3. AC or DC motor
4. Motor control electronics



Step and servo motor (2)

Servo motors

1. Gear system to reduce speed;
2. Potentiometer (voltage divider) for position feedback
3. AC or DC motor
4. Motor control electronics

Advantages of servo motors

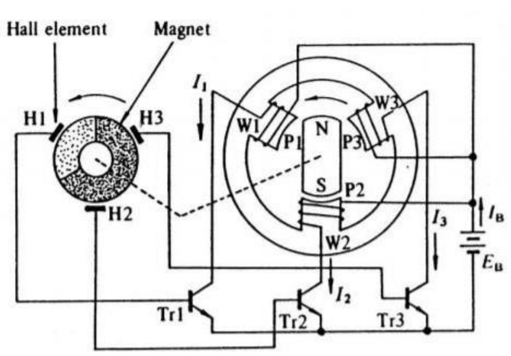
- *High intermittent torque output, high torque to inertia ratio, and high speed*
- *Work well for velocity and position control*
- *Available in all sizes*
- *Smooth rotation at lower speed*

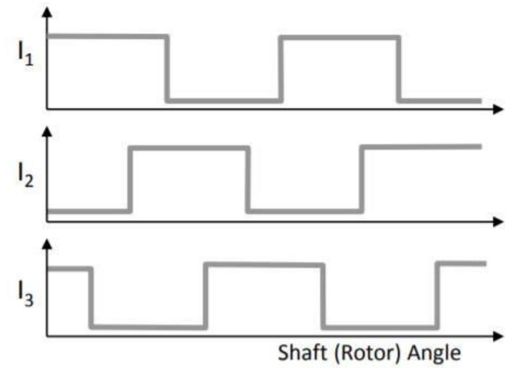
Disadvantages of servo motors

- More expensive than stepper motors
- Requiring tuning control loop for optimal performance
- Not suitable for hazardous environments (e.g., in vacuum)
- Excessive current could result in partially damaging DC servo motor

Step and servo motor (3)

Brushless DC motors-step motor





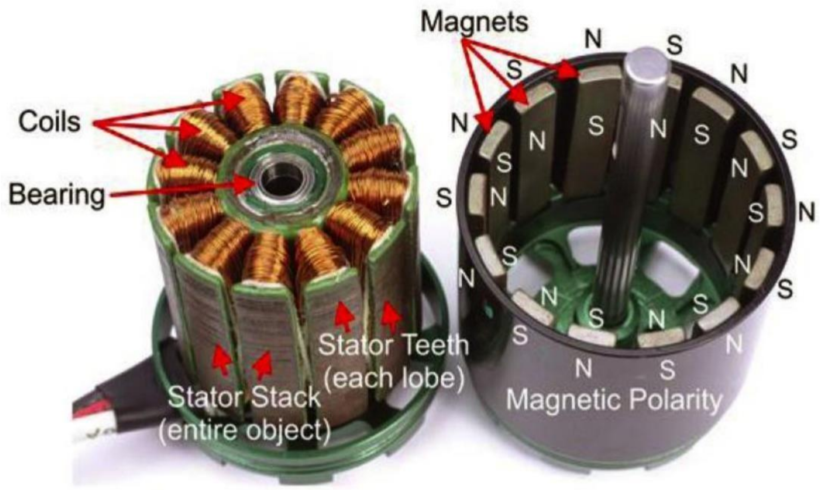
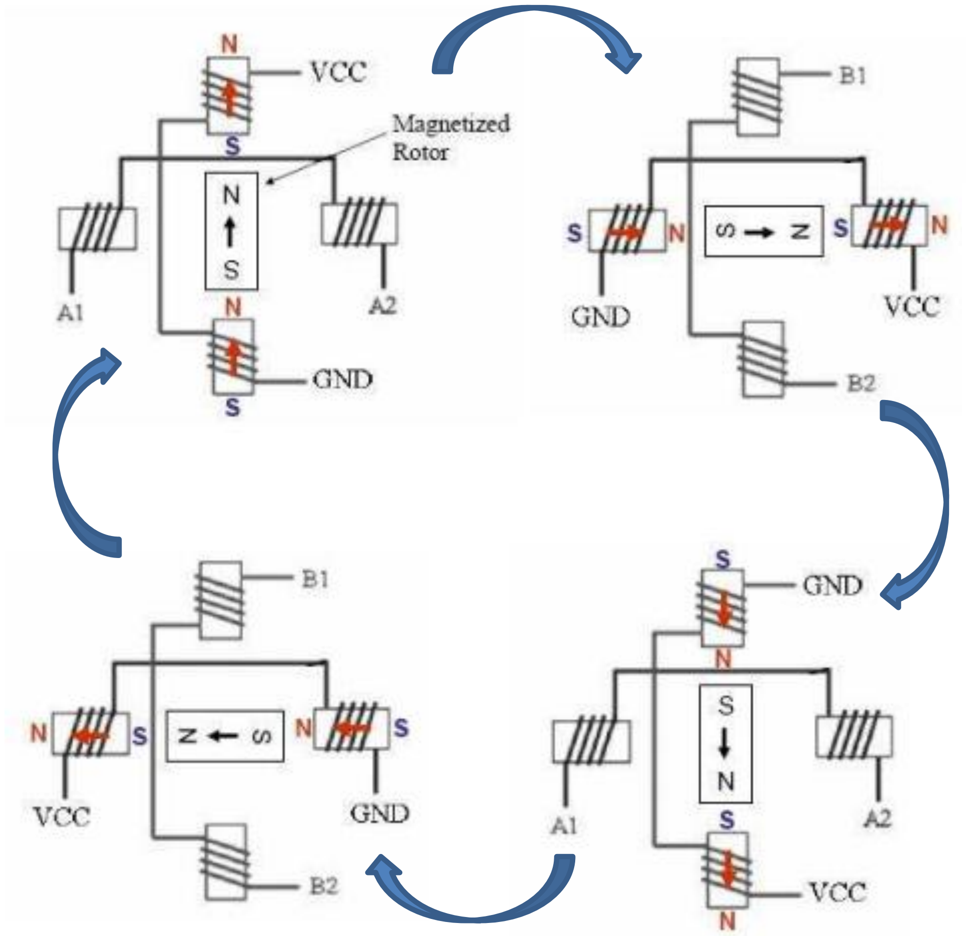


Image: http://www.rcuniverse.com/magazine/reviews/1344/BrushlessMotors7.jpg

Digital step pulses are converted to discrete mechanical movement.

Principles of step motor

Control step motor by phase sequence

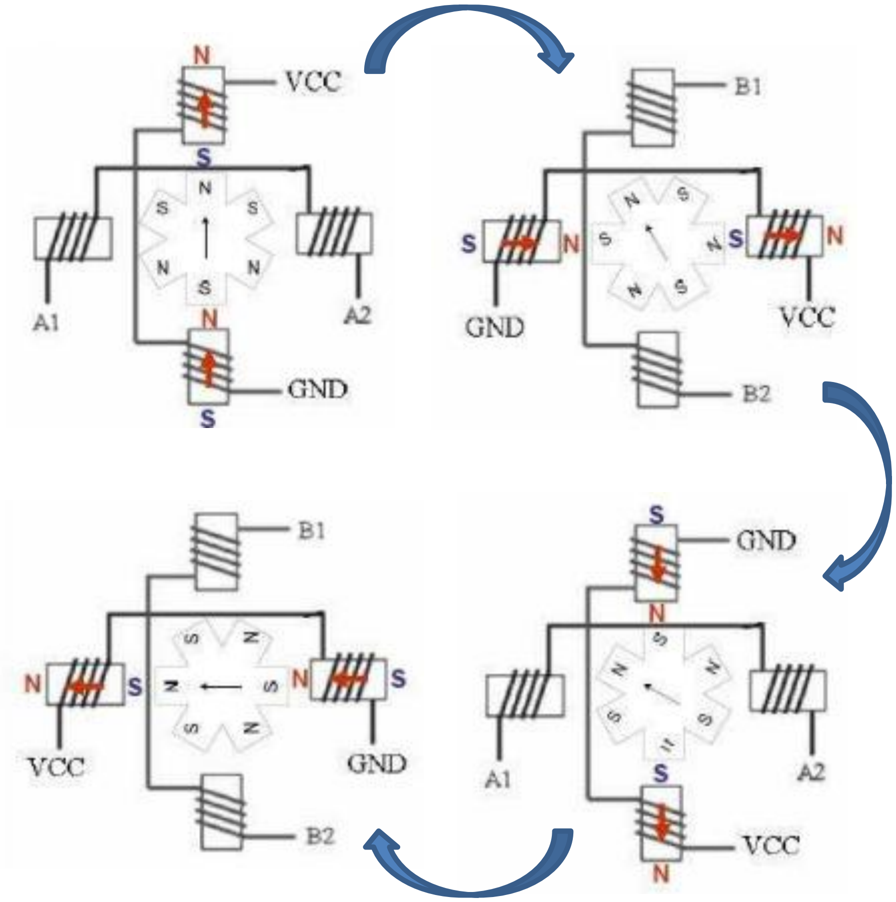


Step	A1	A2	B1	B2
1	NC	NC	+	–
2	–	+	NC	NC
3	NC	NC	–	+
4	+	–	NC	NC

NC = Not Connected

Resolution of step motor

The resolution of step motor is determined by the number of rotor poles and phases of stator;
Each step 30°



Angular resolution of n phase stator and rotor poles

$$\text{Angular Resolution} = \frac{360^\circ}{n \times (\#\text{ of poles})}$$

Pros and cons

Pros:

- Don’t need a feedback, controlled by phase sequence
- Easy to maintain
- Cost less
- Digital control

Cons:

- Cannot correct position error if over-loaded;
- Slower (1500~3000 rpm) than servomotor (3000~6000 rpm)
- Overheat issue

Next lecture will talk about the control of DC motor by feedback, and digital control of step motor

Lab 3 & 4

More exercises in lab sessions

- ❑ Install support for Arduino hardware in MATLAB and Simulink
- ❑ Get familiar with Arduino Uno pins, MATLAB & Simulink library, and Arduino Simulink library
- ❑ PID control of DC motors by Arduino and MATLAB