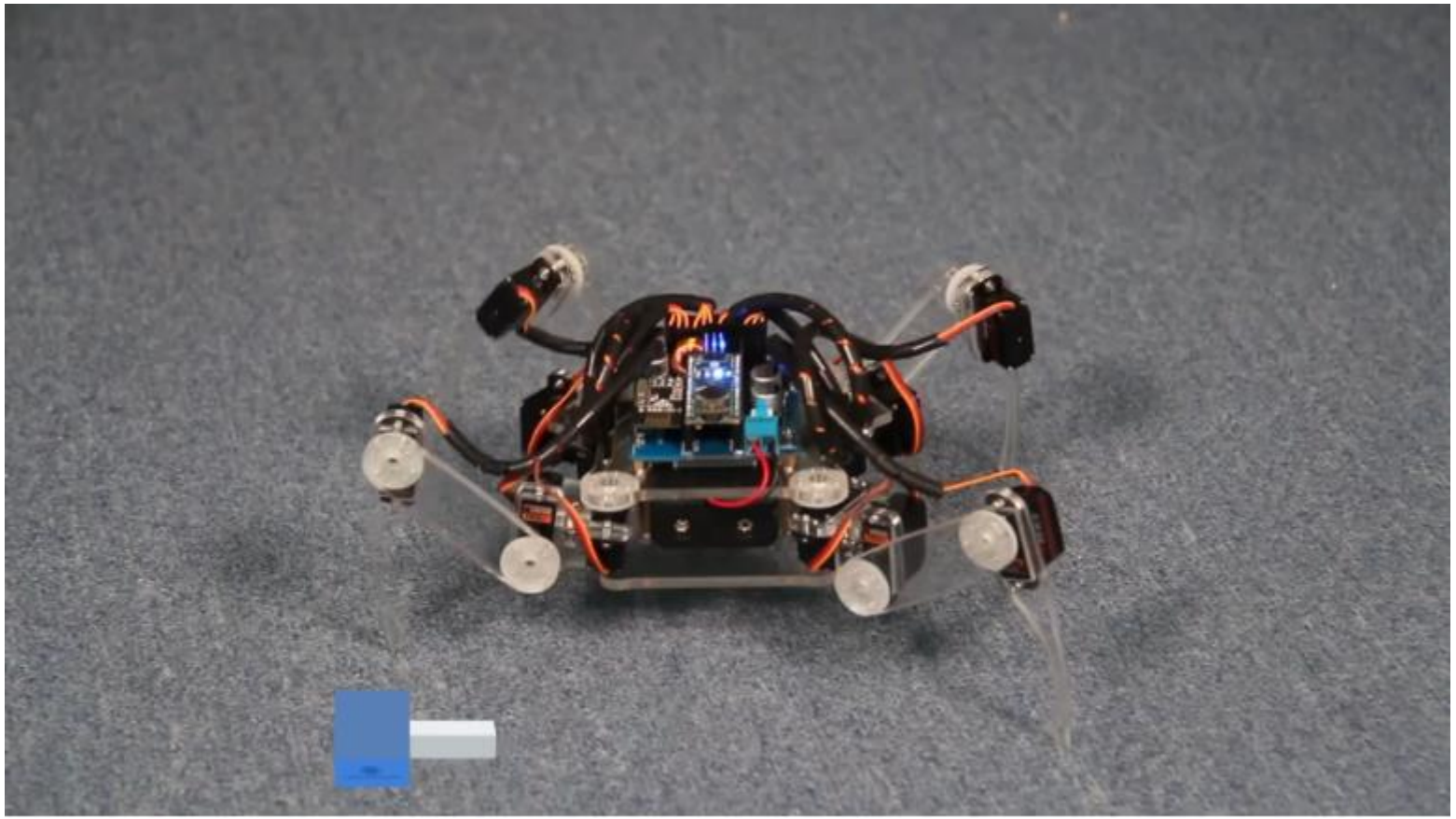
Robotics and Embedded Systems

Technische Universität München

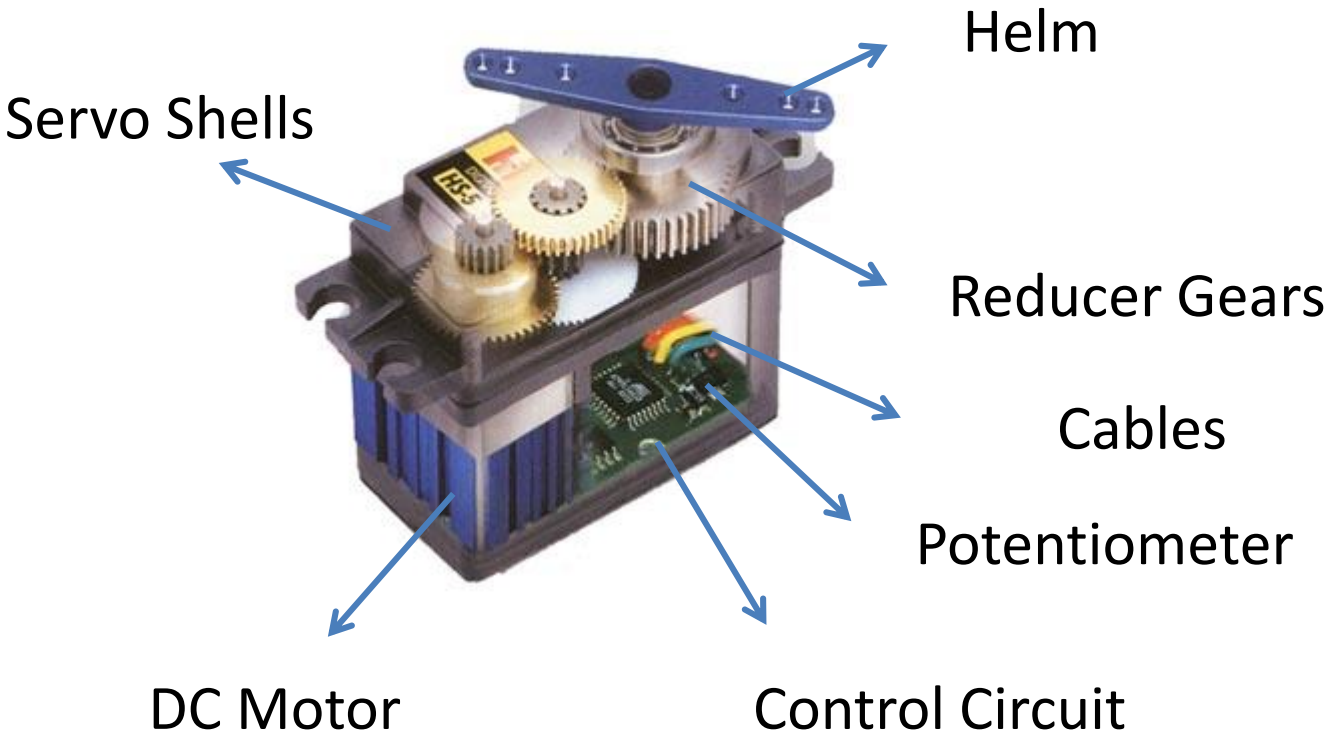
Servo Control —the lab course of Snake Robot

Kai Huang, Long Cheng,
Zhenshan Bing, Mingchuan Zhou

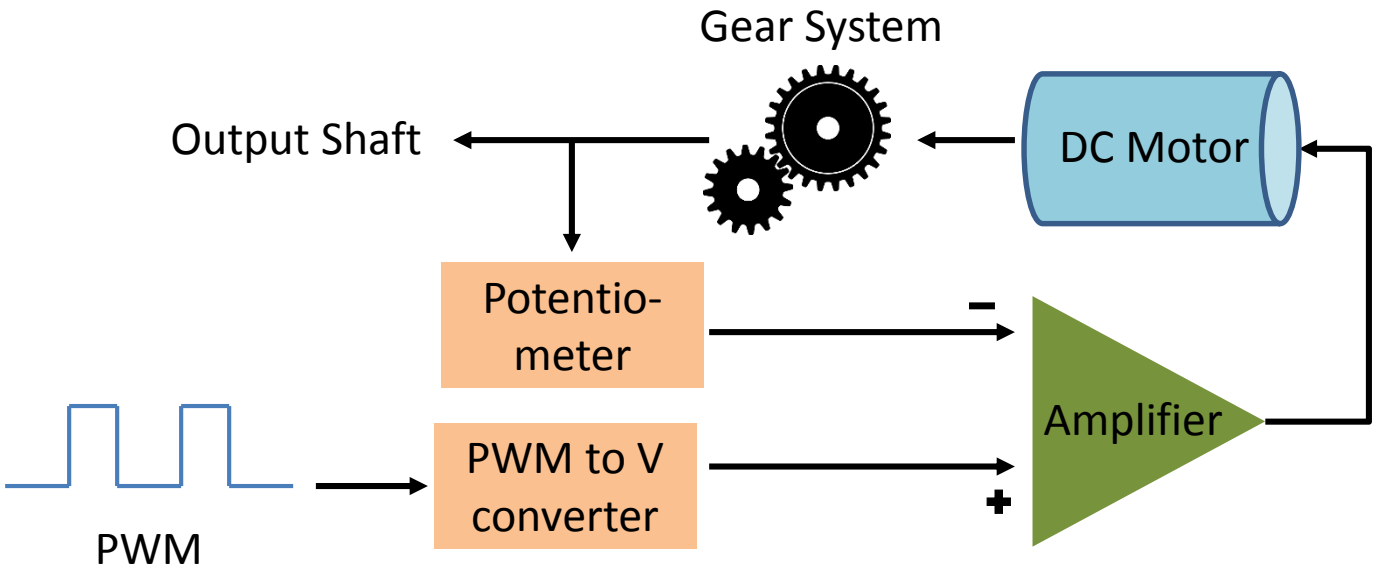
Background



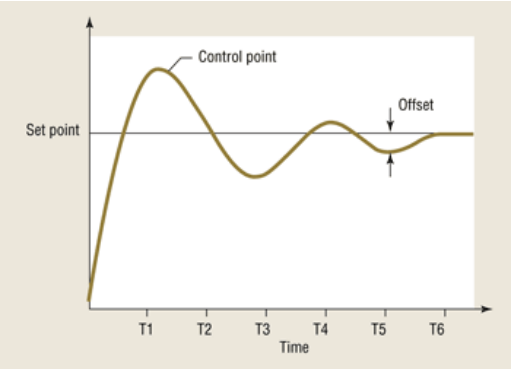
Servo Overview



Servo Principle

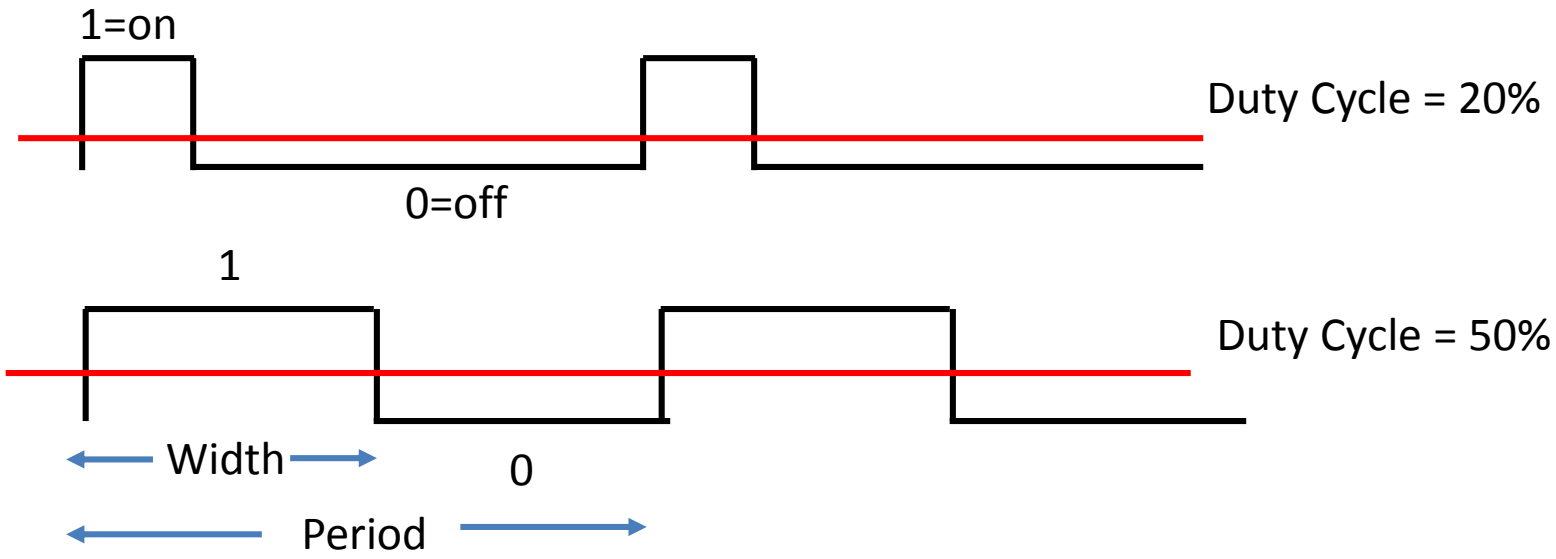


$$V_{error} = (V_{PWM} - V_{potentiometer}) \times K_{Amplifier}$$



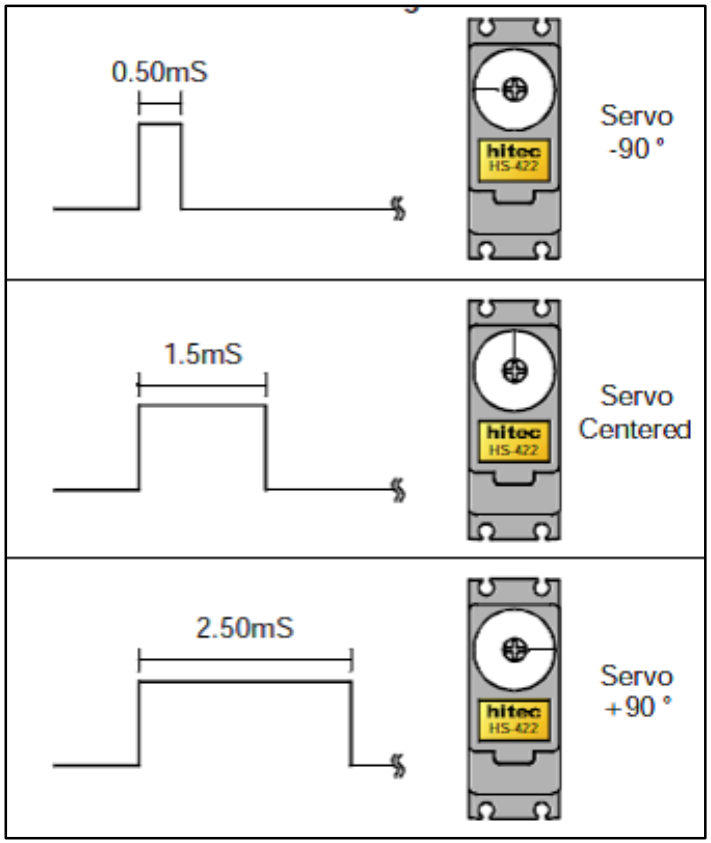
Pulse-Width Modulation (PWM)

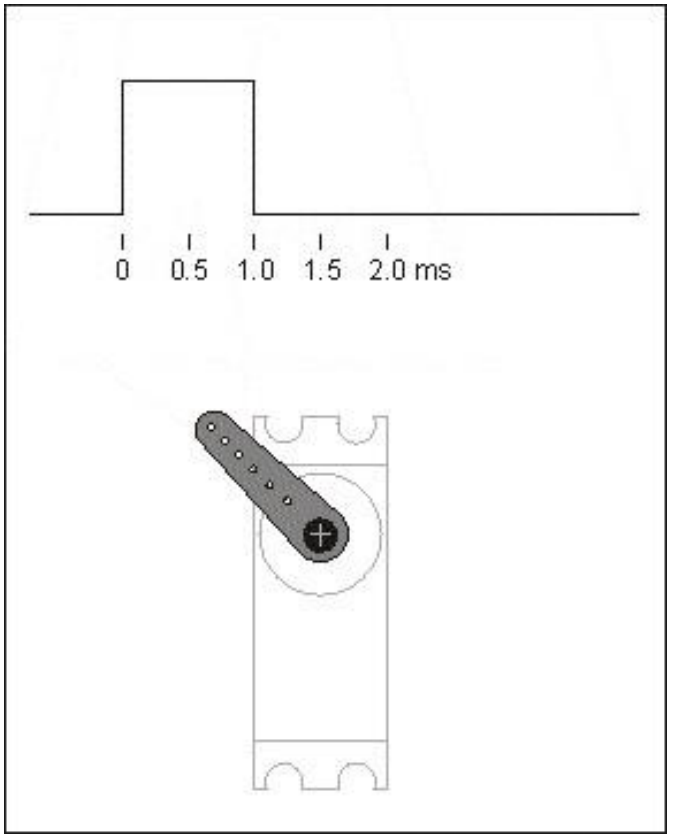
- Output signal alternates between on and off within specific period



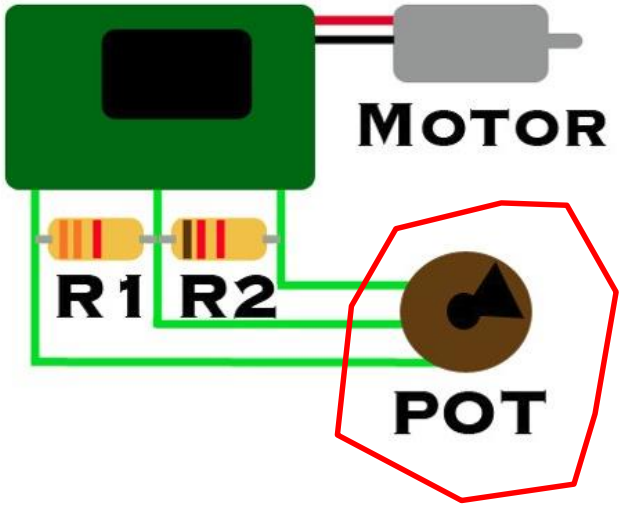
Duty Cycle=$t_{on}/(t_{on}+t_{off})$

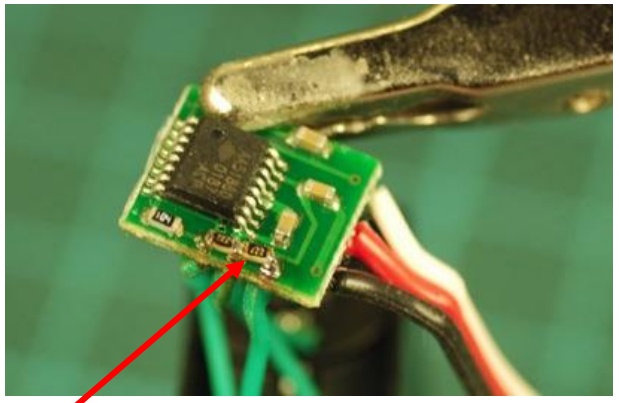
PWM in Servo

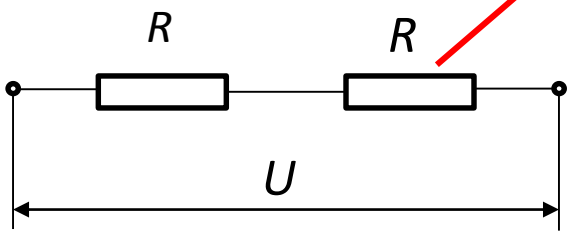

0.50mS
Servo
-90 °
1.5mS
Servo
Centered
2.50mS
Servo
+90 °
hitec
HS-422


0
0.5
1.0
1.5
2.0 ms

Servo Modification

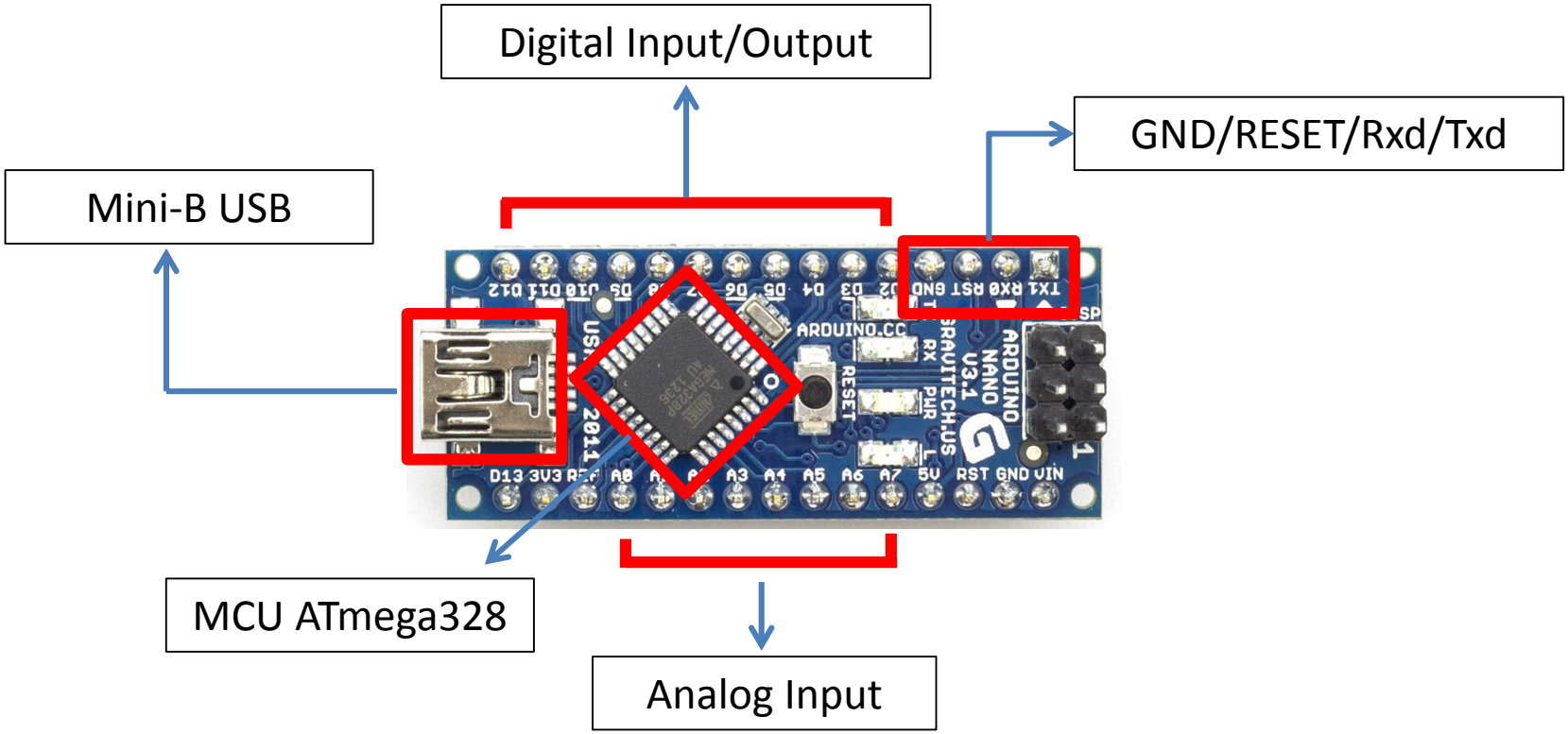






$$R_p = 2R, R \approx 2.5k\Omega$$

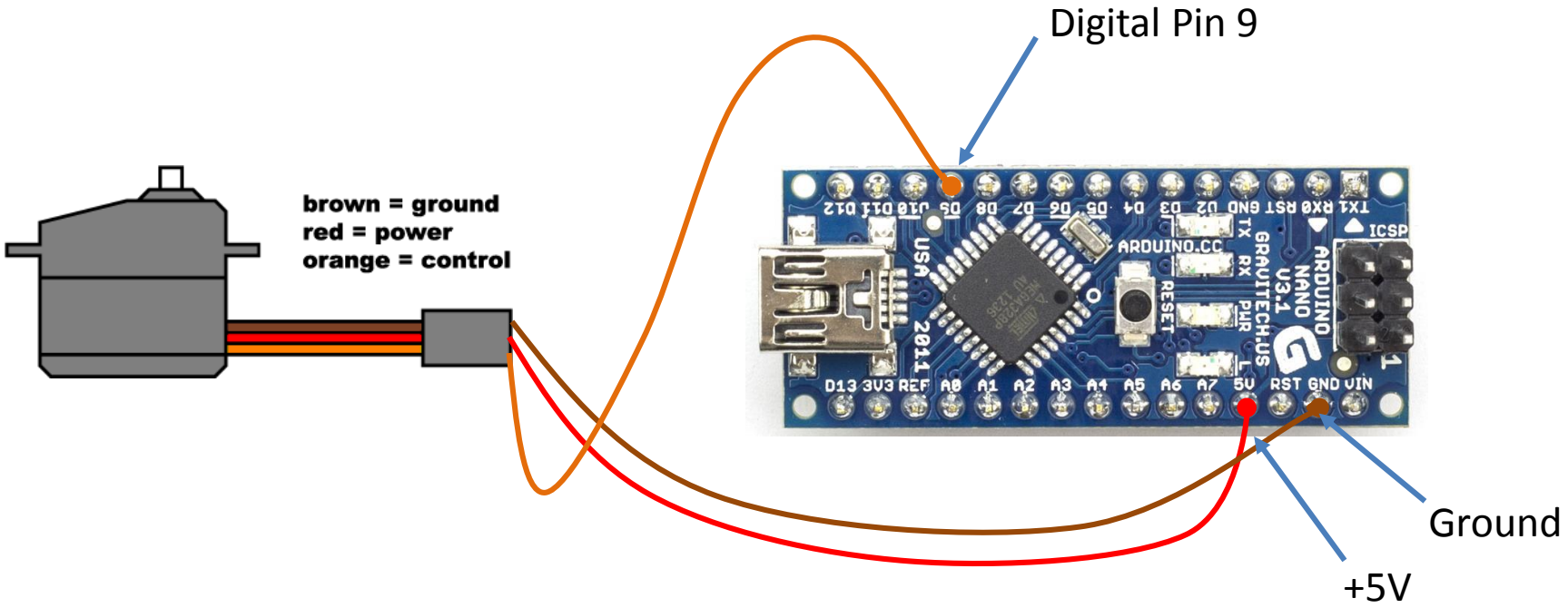
Arduino Nano Overview



Tips

- Power Supply: Mini-B USB, Pin 27 +5V
- 6-PWM Output: D3，5，6，9，10，11
- Test LED Pin D13: High-On, Low-Off

Arduino Servo Connection



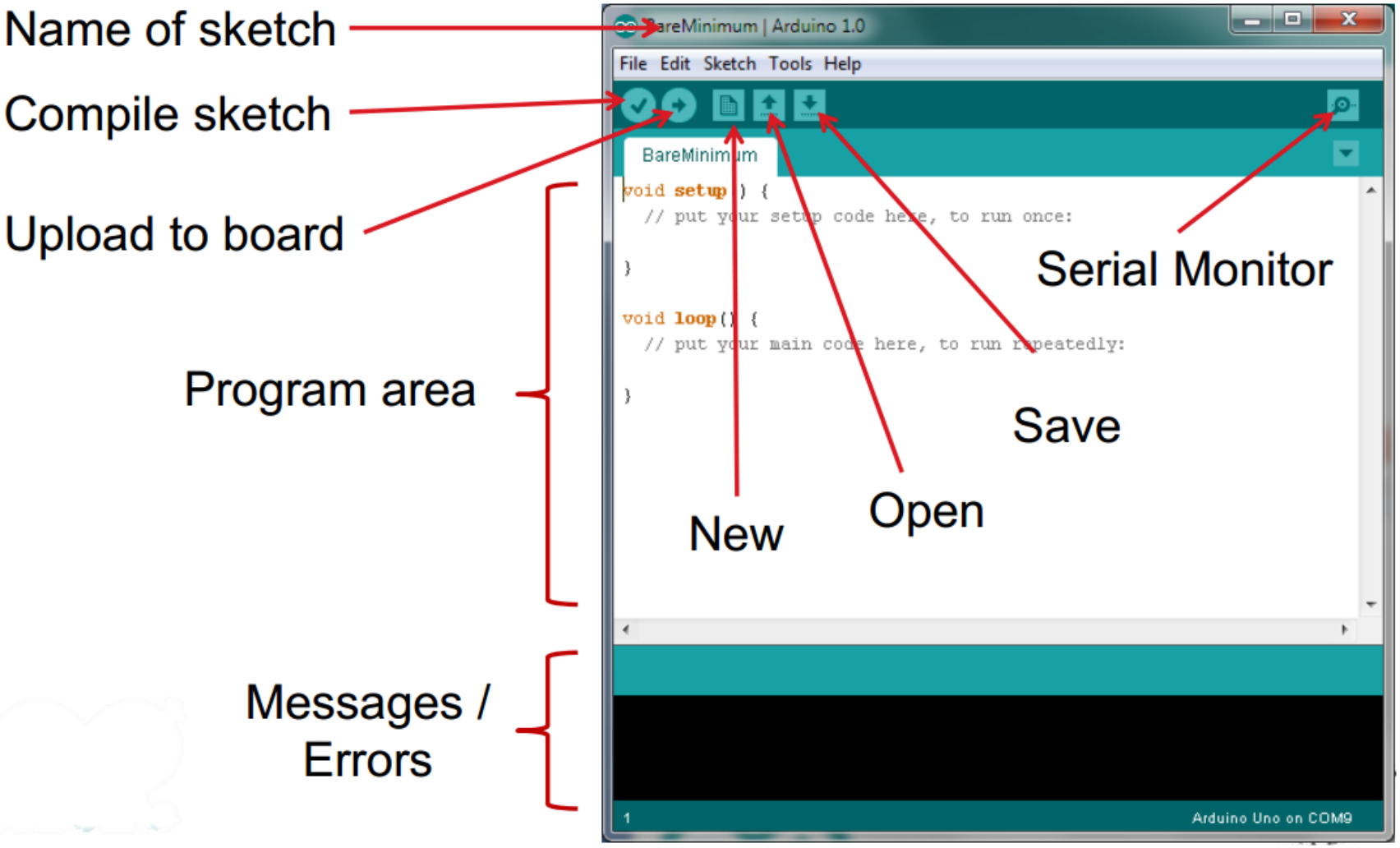
Download and Install

- Download Arduino compiler and development environment from:

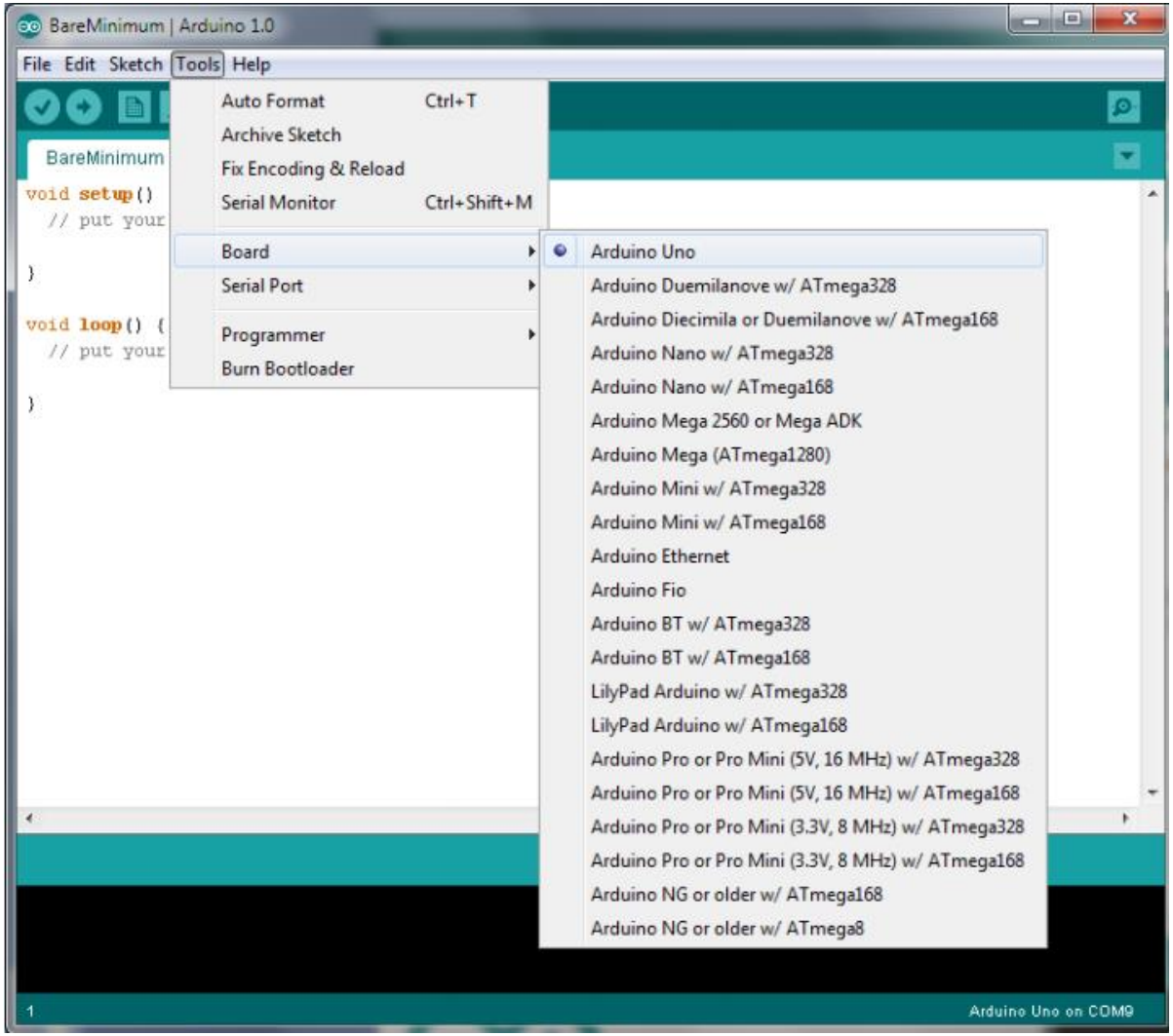
  https://www.arduino.cc/en/Main/Software?setlang=cn

- Current Version: 1.6.5
- Available for:
  - ✓ Windows
  - ✓ MacOX
  - ✓ Linux
- Before running Arduino, plug in your board using USB cable(external power is not necessary)
- When USB device is not recognized, navigate to and select the appropriate driver from the installation directory

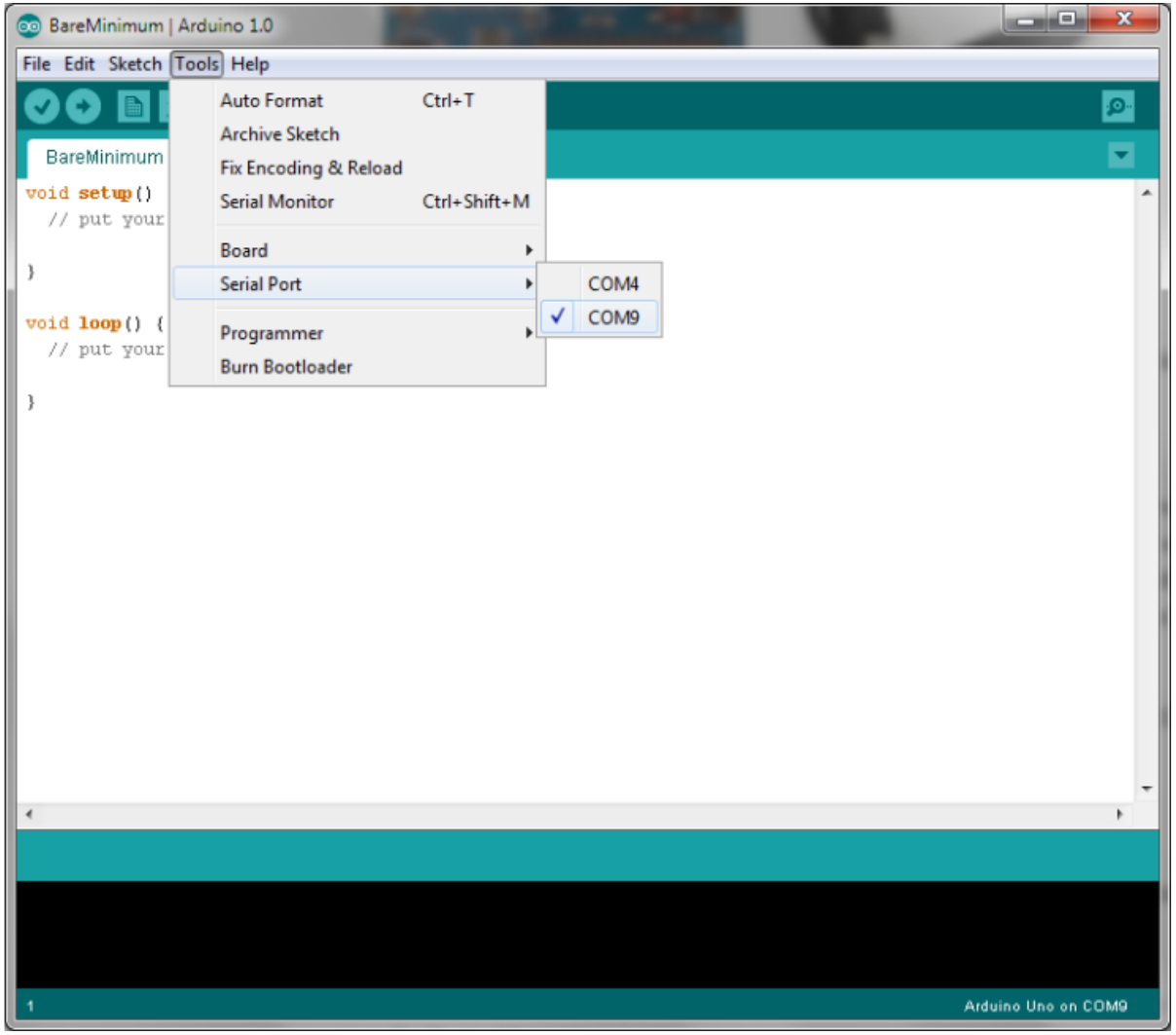
Using the Arduino IDE



Select your Arduino Nano Board



Select Serial Port



Arduino Sketch Structure

- #include <Servo.h>
  - Some specific libraries will be included in your Sketch, which will simplify your codes

- void setup()
  - Will be executed only when the program begins or reset button is pressed

- void loop()
  - Will be executed repeatedly

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Text that follows // is a comment (ignored by compiler)

Servo library-Servo.h

- *Functions*:
  - *Servo* myservo*;* *//*create servo object to control a servo
  - *myservo*.attach(pin) ;//Attach the Servo variable to a pin
  - *myservo*.write(angle);//Writes an angle value to the servo
  - *myservo*.writeMicroseconds(uS);//Writes an PWM value in microseconds (uS) to the servo
  - delay(ms);

Example

- Sweep
  - Sweeps the shaft of a servo motor back and forth across 180 degrees.

- Find out the stop position after servo modification
  - Find out the stop position by using *writeMicroseconds*

- Those codes will be first simulated by Proteus Software and then will be conducted on the Arduino board.

Practice

- Controlling a servo to sweep by using “Servo.h”
- Modify all the servos and test out the stop points, label the PWM value on the servos