

# Embedded System Design with MCU and FPGA

LAB05 0858706 黃品嫻

## Goal

Let us know more about PWM and ADC.

## Problems

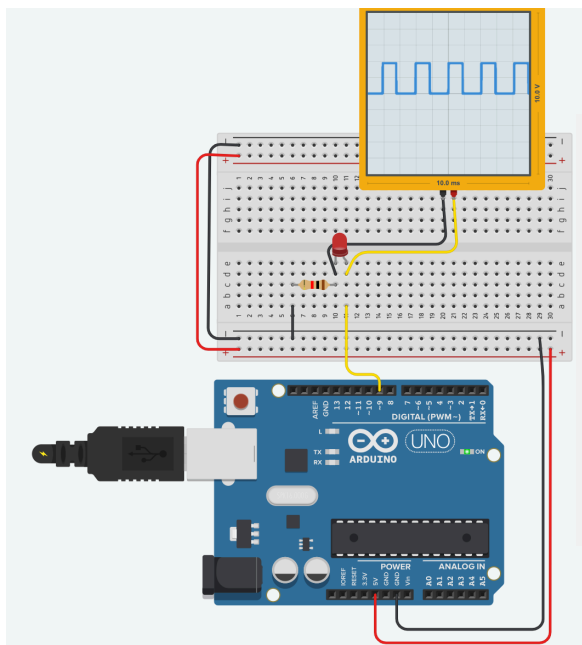
### 1. Is there any DAC (digital-to-analog converter) in ATmega328?

- There is no DAC in the ATmega328.
- The ATmega328 has ADC inputs but no DAC outputs. Although the internal ADC contains a 10 bit DAC, this DAC cannot be used stand alone.
- Although there is no DAC, we could use PWM pin with a lowpass filter to create an analog signals.
- Or, we could just use the external DAC chip.

### 2. List at least 5 applications of PWM signals.

- In the class, we learned there are several uses of PWM:

#### A. Dimming an LED

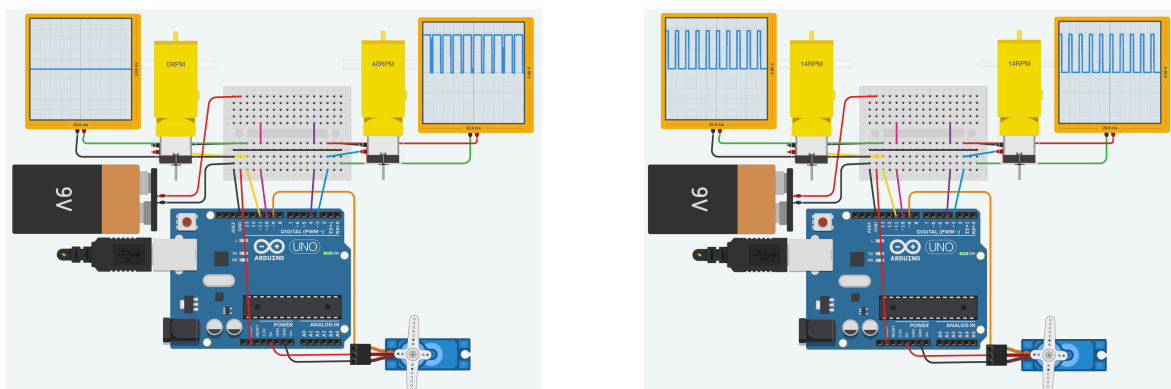


```
int brightness = 0;

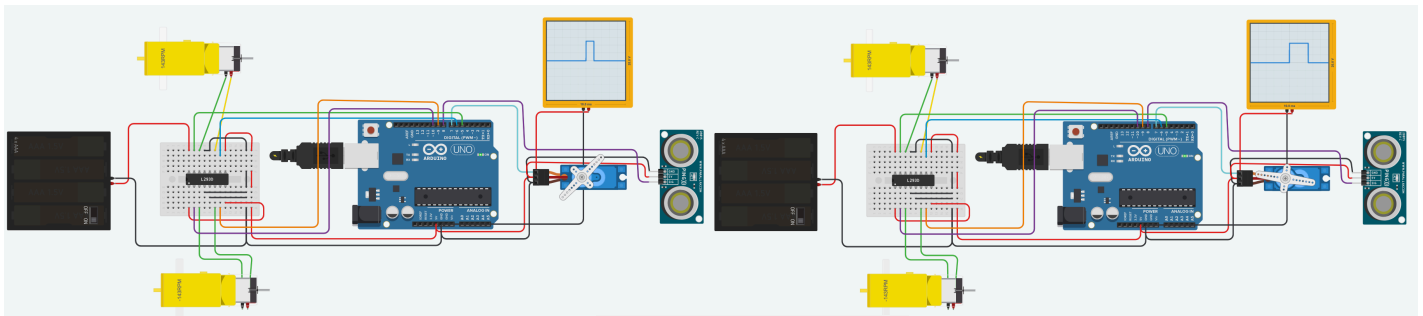
void setup()
{
  pinMode(9, OUTPUT);
}

void loop()
{
  for (brightness = 0; brightness <= 255; brightness += 5) {
    analogWrite(9, brightness);
    delay(30); // Wait for 30 millisecond(s)
  }
  for (brightness = 255; brightness >= 0; brightness -= 5) {
    analogWrite(9, brightness);
    delay(30); // Wait for 30 millisecond(s)
  }
}
```

#### B. Providing variable speed controls for motors



### C. Providing variable angle controls for motors



#### D. Providing an analog output [3]

### E. Generating audio signals

- ### F. Generating a modulated signal

```
void loop()
{
    digitalWrite(13, HIGH);
    delayMicroseconds(100); // Approximately 10% duty cycle @ 1KHz
    digitalWrite(13, LOW);
    delayMicroseconds(1000 - 100);
}
```

3. What are the frequencies of the PWM signals generated by Arduino UNO's original settings?

- There are six PWM pins in the Arduino UNO, pin 3, 5, 6, 9, 10, and 11.
- There are two frequencies of the PWM signals: [1]
  - A. 490 Hz (pin 3, 9, 10, and 11)
    - the result comes from:  $16 \text{ MHz} / 64 / 255 / 2 = 490.196\text{Hz}$
    - 16 MHz is the CPU clock
    - 64 is default prescaler value
    - 255 is the timer repeatedly counts
    - 2 is because the timer runs both up and down
  - B. 980 Hz (pin 5 and 6)
    - the result comes from:  $16 \text{ MHz} / 64 / 255 = 980.392\text{Hz}$
    - 16 MHz is the CPU clock
    - 64 is default prescaler value
    - 255 is the timer repeatedly counts
- We could change the value of register to change the frequencies of the PWM signals.

4. In Arduino UNO board, there are 6 analog input pins. Inside the Atmega328, there is only one ADC.

Please explain how it works.

- In Arduino UNO board, the ADC Multiplexer let 6 analog input pins could connect to only one ADC.
- Multiplexer is a device that selects one of several analog/digital signals and forwards the selected input into a single line.
- There are Input Pins, Output Pin and Control Signal:
  - Input Pins: These are the available signal pins from which one has to be selected. These signals can either be a digital signal or an analog signal.
  - Output Pin: A multiplexer will always have only one output pin. The selected input pin signal will be provided by the output pin.
  - Control/Selection Pin: The Control Pins are used to select the input pin signal. The number of Control pins on a Multiplexer depends on the number of input pins.

5. When the performance of ADC doesn't meet the project's requirement, what can you do?

- We could use external ADC chip to conquer this problem.
- In Arduino UNO, there are 6 ADC of 10 bits. The ADS1115 provides 4 16-bit ADCs, 15 for the measurement and one last for the sign. The ADS1115 is connected by I2C, so it is easy to read. [2]
- So, the ADS1115 could have higher resolution and is easier to read.

## Reference

[1] <https://www.arduino.cc/en/Tutorial/SecretsOfArduinoPWM>

[2] <http://www.electrooobs.com/>

[eng\\_arduino\\_tut83.php#targetText=The%20ADS1115%20is%20an%20external,use%20the%20Arduino%20analog%20inputs.](http://www.electrooobs.com/eng_arduino_tut83.php#targetText=The%20ADS1115%20is%20an%20external,use%20the%20Arduino%20analog%20inputs)

[3] <https://www.arduino.cc/reference/en/language/functions/analog-io/analogwrite/>