

```
#include <18f452.h>
#include <float.h>
#define ADC=8 // resolución A/D 8 bits
#define fuses HS,NOWDT,NOPROTECT,NOLVP
#define use delay(clock=20000000)
#include <lcd.c>

void main() {

    int ad0; // variable del potenciómetro para control de PWM
    float duty;
    lcd_init();
    lcd_putc('\f');
    delay_us(15);

    setup_adc(ADC_CLOCK_INTERNAL); // habilita modulo A/D
    // y especifica clock interno

    setup_adc_ports(AN0);
    setup_timer_2(T2_DIV_BY_16,255,16); // genera una frecuencia de 1.225 KHz
    // valor de PR2 es de 255
    // POSTSCALER 16
    // PRESCALER T2_DIV_BY_16
    setup_ccp1(CCP_PWM); // configura el modulo CCP como PWM

    while(TRUE) {
        set_adc_channel(0); // define RA1 como canal de lectura
        //analógico
        delay_us(15);
        ad0=read_adc(); // lee el conversor AN0 que maneja el PWM
        delay_us(15);
        set_pwm_duty(ad0); // determina el valor del PWM
        // duty = ad0/255.0)*100
        lcd_gotoxy(1,1);
        printf(lcd_putc,"valor PWM %3u",ad0);
        lcd_gotoxy(1,2);
        printf(lcd_putc,"duty cycle %3.2g",duty);
        output_high(PIN_D3);
        delay_ms(100);
        output_low(PIN_D3);
        delay_ms(100);
    }
}
```

Realice las siguientes tareas.

- 1.- Con un osciloscopio muestre la señal PWM.
- 2.- Agregue un led y una resistencia a la puerta configurada como PWM y observe el brillo en el led.
- 3.- Quite el led y deje la resistencia anterior conecte un buzzer, tenga cuidado de conectar correctamente la polaridad del mismo, calibre el potenciómetro hasta obtener un sonido adecuado en el buzzer.