



**UNIVERSITAT POLITÈCNICA DE CATALUNYA**  
**BARCELONATECH**

---

**Facultat d'Informàtica de Barcelona**

Enunciat de la pràctica de laboratori

---

# **PWM**

**(Modulació per amplada de pols)**

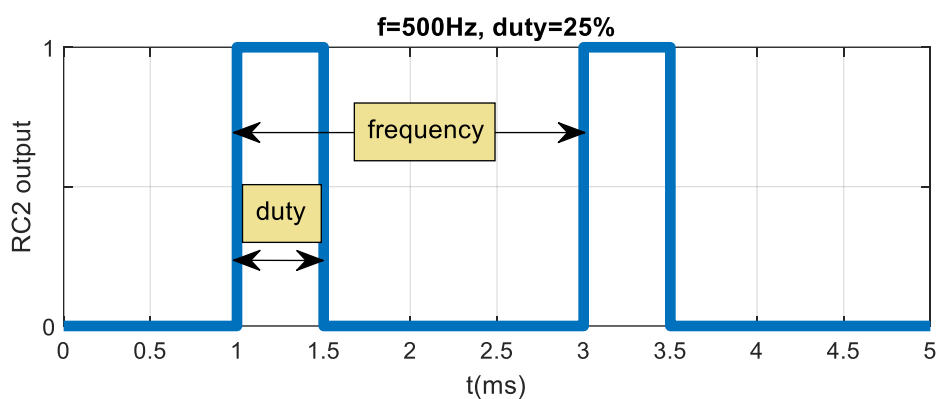
---

# 1. Objectius

L'objectiu d'aquesta pràctica és capacitar a l'alumne en l'ús d'un *Interval Timer* com a dispositiu generador de senyals digitals modulats en amplada de pols (PWM, sigles de *Pulse Width Modulation*).

Per fer-ho farem un programa que serveixi per desenvolupar un carregador de bateries. Es tracta d'un programa simplificat que varia el duty-cycle o cicle de treball de tal manera que les bateries d'ió de liti es carreguin adequadament. En els dispositius reals, el pin on està configurat el PWM es connecta a un transistor que carrega la bateria més o menys segons el duty-cycle. En el nostre programa ho simplifiquem i connectarem un led al pin on configurem el PWM per variar la intensitat del led segons el duty-cycle.

Recordeu que les dues característiques principals d'un PWM són la freqüència i el duty-cycle. La freqüència és la inversa del temps en què la ona es repeteix, i el duty és la relació de temps en què la senyal està a ON respecte al temps total.



## 2. Coneixements previs de l'alumne

L'alumne ja ha de dominar els següents conceptes:

- L'arquitectura del PIC18F45K22
- El simulador PROTEUS
- La programació del PIC en C
- El funcionament dels ports d'E/S del PIC
- Programació de les interrupcions
- El funcionament del controlador del GLCD
- Els *Timers* del PIC18F45K22 com a unitats de mesura de temps

## 3. Pràctica base

L'objectiu és ser capaços de generar un senyal digital de període 2 msec (freqüència 500Hz) i que podem controlar el seu duty-cycle amb un perfil de càrrega donat que vagi variant al llarg del temps.

Per a tal fi, s'utilitzarà la unitat **CCP1** associada al *timer* **TMR2** per generar el senyal PWM mitjançant el maquinari del micro. El resultat haurà de ser visible en el LED RC2 observant que la intensitat de la seva llum és proporcional al % de duty-cycle. Aquest efecte de variació de la intensitat es pot observar en un LED real (placa EasyPIC), però malauradament el LED del Proteus no mostra aquest efecte (anirà oscil·lant entre apagat i encès, però no mostra canvis progressius en la intensitat).

És recomanable que structureu la vostra feina en varies fases: una fase preliminar en la que aprendreu a configurar el CCP1 per a generar un PWM amb un duty-cycle fix. Una segona fase en la que el duty-cycle vagi oscil·lant segons uns paràmetres. Una tercera fase per pintar la informació per la pantalla GLCD. I una última fase opcional on es calcula l'estat de càrrega de la bateria.

### 3.1. Fase 1: duty-cycle fix

En aquesta primera part, genereu un senyal de PWM amb període (2 ms) i el duty-cycle fix al 50%. Feu servir el timer2 i els registres associats per configurar tot adequadament.

Connecteu un led i una resistència adequada al pin RC2. Com s'ha comentat abans, el simulador Proteus no pot simular l'efecte d'intensitat de llum del led, però un cop al laboratori, es podrà apreciar l'efecte sobre la placa EasyPIC.

Afegiu un oscil·loscopi al vostre esquemàtic i comproveu que la senyal obtinguda al pin RC2 és de 500Hz = 2ms.

## 3.2. Fase 2: duty-cycle variable

Un cop hagueu comprovat que sabeu generar un PWM amb un duty-cycle fix, afegiu la següent funcionalitat: el duty-cycle ha de canviar segons la línia blava de la següent gràfica:

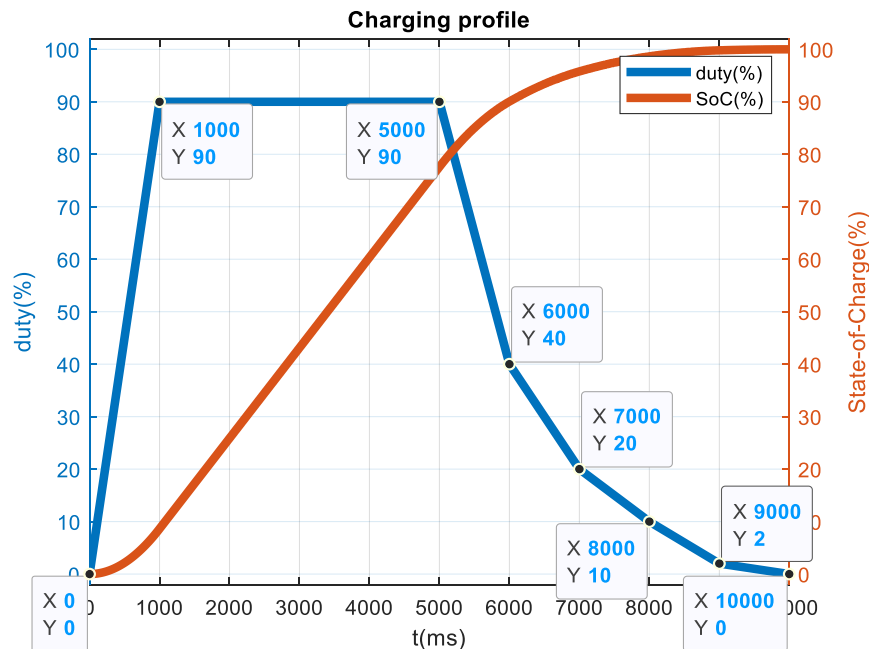


Fig 1. Procés de càrrega típica de carregadors de bateries d'ió de liti.

En aquesta gràfica apareix en color blau el duty-cycle en tant per cent al llarg del temps. El procés de càrrega dura 10000ms=10s. Al començament del procés de càrrega el duty augmenta ràpidament fins arribar al 90%. Després segueix una part de càrrega màxima entre el segon 1 i el 5. Posteriorment, un cop la bateria s'ha carregat substancialment, el duty-cycle baixa ràpidament fins al 40% al segon 6, fins al 20% al segon 7, fins al 10% al segon 8, fins al 2% al segon 9 i fins a 0% al final del procés de càrrega.

Per veure que el duty-cycle varia de forma adequada, s'ha de fer servir alguna eina que permeti convertir una senyal quadrada amb duty variable en una senyal analògica. Això ho farem amb un filtre passa-baixos. Aquest filtre elimina les components d'alta freqüència i es queda amb les de baixa freqüència.

El circuit de la figura 2 fa aquesta funció. Aquest circuit només serveix per visualitzar, per res més. L'entrada (esquerra) del filtre es connecta al pin RC2, la sortida (dreta) es connecta a un altre l'oscil·loscopi per visualitzar el resultat.

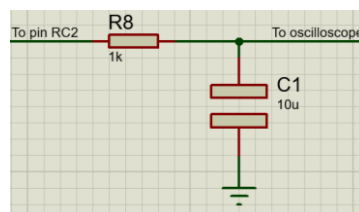


Fig 2. Filtre passo-baixos per filtrar el valor

El resultat de la senyal del PIN RC2 al ser filtrada amb el circuit de la Fig.2 es pot veure a la Fig.3.

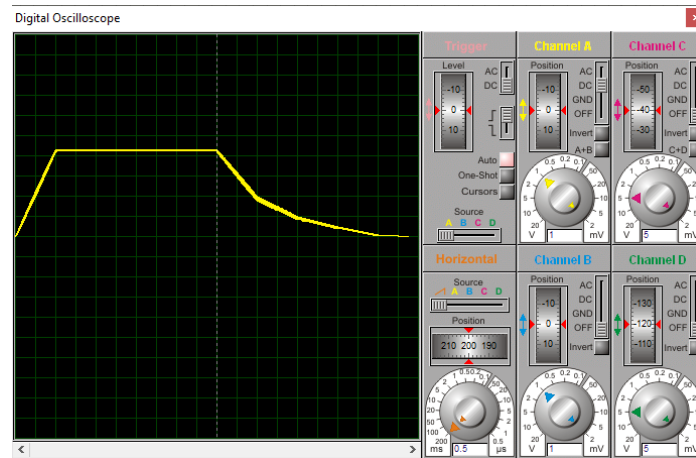


Fig 3. Senyal filtrada del PIN RC2 amb el filtre passa-baixos.

### 3.3. Fase 3: Pintar a la GLCD

La informació del temps transcorregut i del duty-cycle s'ha de pintar per la pantalla GLCD tal i com es pot veure a la següent figura:

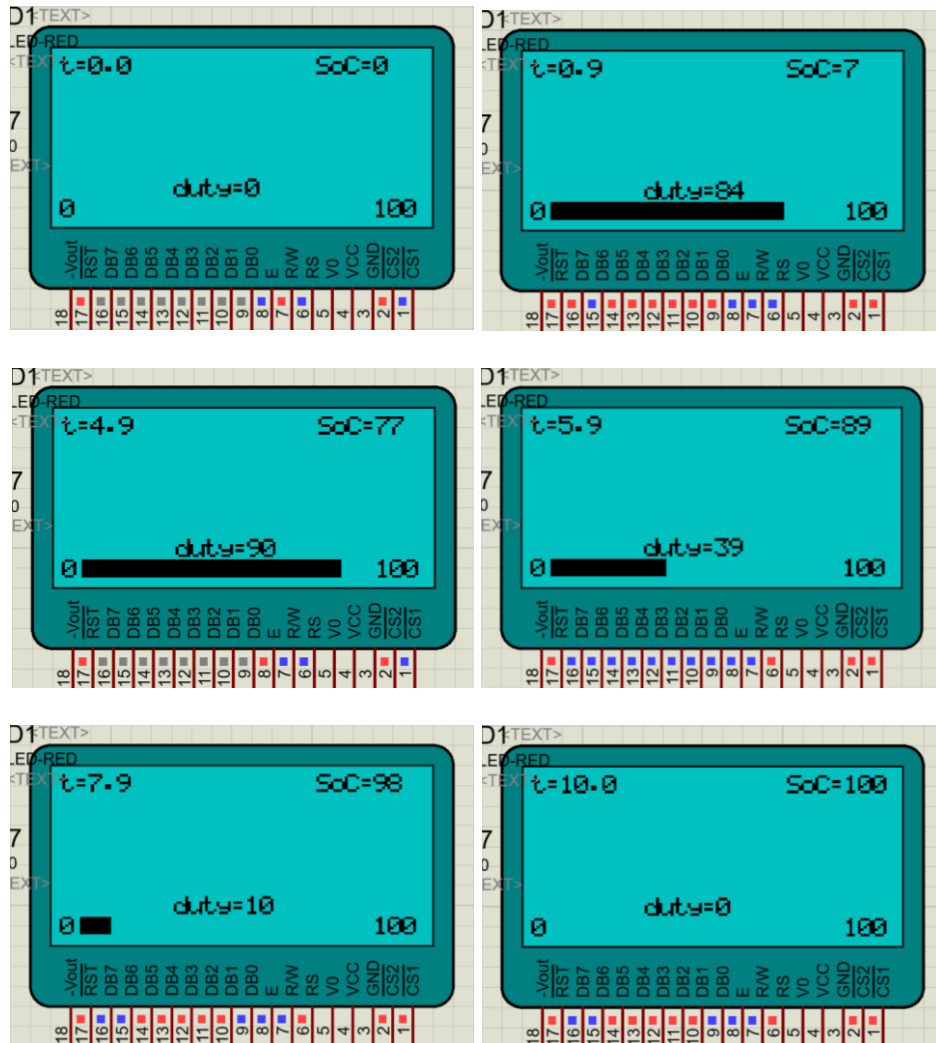


Fig 4. Captures de pantalla amb la informació del procés de càrrega.

El temps transcorregut desde l'inici de la càrrega es veu a la part superior esquerra de la pantalla.

El duty-cycle es mostra com una barra de progres entre el 0 i el 100% a la part inferior de la pantalla.

### 3.4. Fase 4: Estimació de l'State-of-Charge (opcional)

Opcionalment, es pot fer una quarta fase en la qual es farà una estimació de l'estat de càrrega (SoC State-of-Charge). Aquest apartat no és obligatori, però servirà per millorar la nota del previ.

A la figura 1 es pot veure en color taronja com evoluciona el valor de SoC al llarg del procés de càrrega.

Aquest càlcul es fa com:

$$\text{SoC}(\%) = k \int_0^t \text{duty} dt$$

Haureu de trobar com s'implementa aquesta equació dintre del microcontrolador i pintar-lo en la part superior dreta de la pantalla GLCD, on posa SoC=xx.

## 4. Treball previ

- **Lectura detallada de les seccions 14.0 i 14.3** (PWM mode) del *DataSheet* del PIC18F45K22 de Microchip. Consultar en general el capítol 14 (CCP module) per a trobar la informació que us calgui.
- **Entregar el projecte de Proteus en un arxiu \*.zip amb el full de respostes**
- Es valorarà la correcta programació dels registres i els comentaris al codi sobre les seleccions efectuades. També es tindrà en compte el bon ús dels recursos del microcontrolador, la modularitat del codi i una correcta estructura del programa. L'apartat opcional del SoC servirà per compensar errors en les fases anteriors i per millorar la nota.

## 5. Pràctica en el laboratori

El treball a realitzar en el laboratori consta dels següents apartats:

- 1) Mostrar el correcte funcionament del programa proposat sobre PROTEUS.
- 2) Comprovar el funcionament dels programes realitzats sobre la placa EASYPIC.
- 3) Comprovar amb l'oscil·loscopi que la freqüència base del senyal PWM és de 500Hz.
- 4) Realitzar l'ampliació proposada pel professor a classe.



## Qüestionari Pràctica PWM

Grup	Noms i Cognoms
13	Bernat Borràs Civil i Miquel Torner Viñals

1) Quin són el registres amb el que es pot controlar la freqüència?

T2CON (Prescaler)

CCPR1L (MSB)

CCP1CON (LSM)

2) Indica els càlculs realitzats per configurar el PWM amb una freqüència de 500Hz.

PWM Period = 1/500 s      Duty Cycle Ratio = 50%

PWM Period = [PR2+1]\*PRE\*4\*1/8000000 = 1/500

PR2 = 8000000/(500\*4\*PRE)

PRE = 1:      PR2 = 8000000/(500\*4) = 3999 > 2<sup>8</sup>

PRE = 16:      PR2 = 8000000/(500\*4\*16) = 249 < 2<sup>8</sup>

Duty Cycle Ratio = (CCPR1L:CCP1CON<5:4>)/(4\*(249+1)) = 0.5

CCPR1L:CCP1CON<5:4> = 1000\*0.5 = 500 = 0b0111110100

MSB = 0b01111101

LSM = 0b00

3) Quin és el registre amb el que es pot controlar el duty-cycle?

El registre PR2

4) Quants valors diferents de PWM es poden generar en aquesta aplicació?

10 bits diferents -> 2<sup>10</sup> = 1024 valors diferents

5) Per tal de que el carregador de bateries no faci soroll, és convenient que la freqüència sigui de l'ordre de  $20\text{kHz}=50\mu\text{s}$  que està per sobre de l'espectre auditiu de les persones. Repeteix els càlculs i les modificacions del PIC18F per aquesta freqüència? Creus que hi haurien problemes amb aquesta configuració a la EasyPIC? Quins problemes se t'acudeixen? Alguna solució?

$$\text{PWM Period} = 50 \cdot 10^{-6} \text{ s} \quad \text{Duty Cycle Ratio} = 50\%$$

$$\text{PWM Period} = [\text{PR2}+1] \cdot \text{PRE} \cdot 4 \cdot 1/8000000 = 50 \cdot 10^{-6}$$

$$\text{PR2} = 8000000 / (50 \cdot 10^{-6} \cdot 4 \cdot \text{PRE})$$

$$\text{PRE} = 1: \quad \text{PR2} = 8000000 / (50 \cdot 10^{-6} \cdot 4) = 0,04 > 2^8$$

El principal problema és que el pic no pot produir una freqüència tan petita. El timer hauria de fer overflow 25 vegades a cada tick.

Per arreglar-ho es podria fer una espècie de prescaler on fos 1:1/32, que dividís els ticks. Probablement hauríem d'incrementar la freqüència d'oscilació del pic per poder-ho fer.