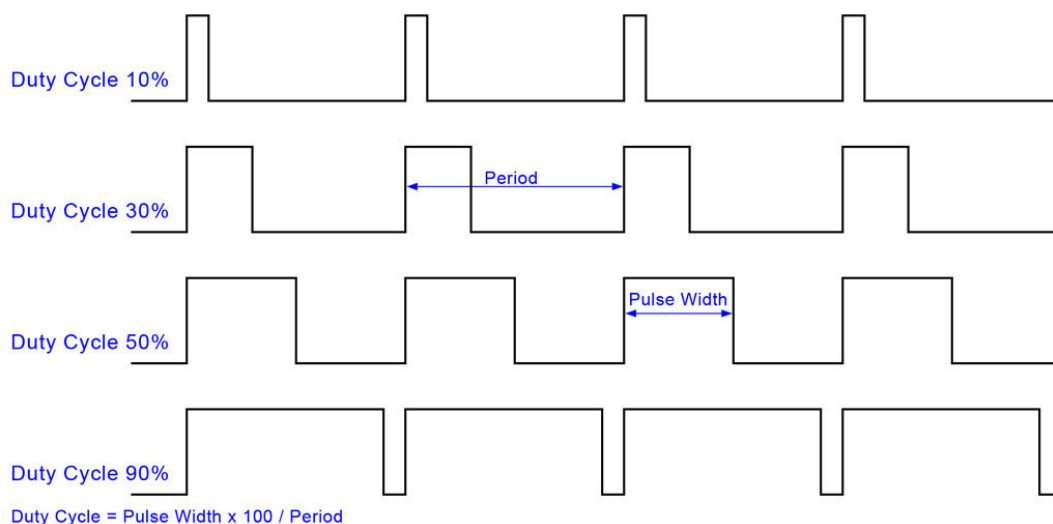


### Laboratorio 3: Uso di PWM per regolare la luminosità del LED

La modulazione di larghezza di impulso (pulse width modulation, PWM) può essere usata per generare una tensione “pseudo-analogica” a partire da un’uscita digitale. Qui useremo un segnale PWM per pilotare il led sulla scheda di sviluppo, e regolarne quindi la luminosità. (Nota: l’uscita PWM sarà realizzata “a mano” sul pin di GPIO P1.6, collegato al led. Non bisogna invece considerare le uscite PWM già comprese nel microcontrollore, dato che non sono collegate al led).

Il PWM è un segnale digitale con periodo (frequenza) fissato, e duty cycle variabile. Il “duty cycle” del PWM è la frazione del periodo in cui l’uscita è 1, e può essere compresa tra 0% (led sempre spento) e 100% (led sempre acceso), passando per tutti i valori intermedi. Se la frequenza del PWM è abbastanza alta, ovvero più alta della massima frequenza riconoscibile dall’occhio umano (quindi >100 Hz), una commutazione veloce tra stato “1” (led acceso) e stato “0” (led spento) e viceversa dà un effetto equivalente ad una tensione intermedia, in cui il led risulta acceso con luminosità ridotta. Ad esempio, un duty cycle 50% corrisponde al caso in cui il led è acceso per metà del periodo del PWM (ovvero, il PWM dà in uscita un’onda quadra simmetrica). Il duty cycle del PWM corrisponde quindi direttamente alla luminosità del LED.



Si chiede di realizzare un sistema che abbia le seguenti funzionalità:

- Il LED sia controllato da un PWM, in modo che la sua luminosità sia impostabile in modo continuo con risoluzione 8 bit da 0% a 100%.
- La pressione del pulsante P3.7 abbia una duplice funzione: se premuto e immediatamente rilasciato, deve commutare lo stato del LED (da acceso a spento o viceversa). Se tenuto premuto per più di 1 secondo, si entri nella “modalità di configurazione” (vedi prossimo punto).
- Quando il sistema è in modalità di configurazione, il LED deve essere acceso, e finché il pulsante viene tenuto premuto la luminosità del LED deve variare da 0% a 100% e poi da 100% a 0%, eccetera, in modo continuo e relativamente lento (pochi secondi). Il rilascio del pulsante fa uscire dalla modalità di configurazione, tornando alla normale operazione con led acceso al nuovo valore di luminosità selezionato.

Si impongono queste scelte di implementazione:

- Il valore della luminosità sia immagazzinato in una variabile globale a 8 bit (unsigned char) "Lumi", o altro nome a piacere.
- Il task che gestisce il PWM sia basato su un timer a 8 bit (T0 o T1), e gestito unicamente via interrupt. La routine di interrupt del timer deve accendere/spegnere il led, leggere la variabile "Lumi" e impostare i valori di reload di conseguenza. Per questo task, la variabile "Lumi" è di sola lettura.
- Il task che gestisce l'interazione con l'utente (gestione del pulsante, misura della durata della pressione, velocità di variazione di luminosità in modalità configurazione, ...) sia basato su un solo timer a 16 bit, con "tick" impostato 10 ms. Questo task può leggere e scrivere il valore della variabile "Lumi". Anche qui si richiede che tutto sia gestito in interrupt, ma in modo non bloccante: nella routine di interrupt non devono essere presenti cicli.
- In conseguenza delle condizioni precedenti, il "main" deve contenere solo l'inizializzazione, seguita da un ciclo vuoto "while(1);".

**Al termine dell'esperienza, si chiede di consegnare via mail ai docenti il progetto (con codice commentato).**