

Relazione primo assegnamento – Led Pong

Autori: Aprigliano Cristiano, Caldara Mariano, Scucchia Matteo

Implementazione:

L'applicazione è stata implementata seguendo le regole discusse a lezione, tuttavia ci sembra doveroso discutere alcune scelte implementative da noi realizzate.

Scambi:

gli scambi finali sono conteggiati a partire dalla prima pressione di uno tra i bottoni T1 e T2, finché la sessione di gioco non è terminata. Il Reaction Time invece viene fatto diminuire di $1/8$ a partire dalla seconda pressione di uno tra i due bottoni T1 e T2 ossia non viene conteggiato come scambio completo il primo, quando la pallina viene generata dal led centrale, ma si inizia a modificare RT nel momento in cui è stato fatto un primo scambio completo.

Interruzioni:

le interruzioni sono state associate mediante i pin preimpostati 2 e 3 rispettivamente ai bottoni T2 e T1 ossia quelli governati dai due avversari. Per cercare di rendere il gioco più attinente alle specifiche si è impostata la chiamata delle interruzioni durante il FALLING dei bottoni, in quanto così facendo se un giocatore continua a premere il suo bottone, esso non viene recepito come premuto e una volta esaurito il Reaction Time egli avrà perso. Le interruzioni associate ai bottoni richiamano ciascuna una funzione, rispettivamente `react_A()` per il pulsante T1 e `react_B()` per il pulsante T2. Tali funzioni gestiscono il contatore degli scambi e un booleano verso che indica la direzione della pallina.

Velocità:

la velocità di gioco assume solo due valori, 1 o 2, a seconda del valore assunto dal potenziometro. Velocità 1 corrisponde a valori minori di 512, velocità 2 corrisponde a valori maggiori di 512, ossia il valore mediano.

Timer:

i timer per comodità ed esperienza del gruppo, sono stati gestiti tutti internamente senza l'utilizzo di librerie esterne mediante le funzioni `millis()` o `micros()` e con variabili interne.

Rimbalzi:

per la gestione dei rimbalzi è stato implementato un timer che attende 50 millisecondi per cercare di minimizzarli.

Loop:

il loop viene utilizzato come “reset” una volta terminata la sessione di gioco. Esso infatti non viene mai fatto scorrere, ma vengono implementati dei cicli while nel loop. E' stata una scelta implementativa che ci sembra possa apportare maggior efficienza al progetto. Una volta terminata la sessione, re-inizializza le variabili di gioco e fa ripartire un'altra sessione.