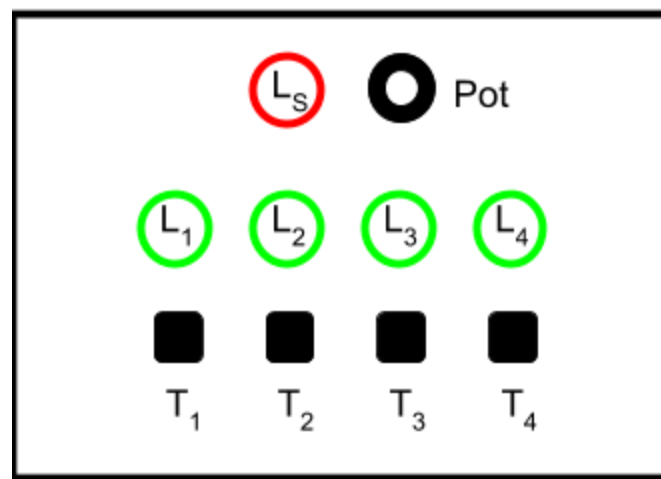


Progetto #1 - *Track the Led Fly*

Si vuole realizzare un sistema embedded che implementi un gioco che chiamiamo *Track the Led Fly*.

Descrizione di massima

La board del gioco è costituita da 4 led verdi L_1 , L_2 , L_3 , L_4 un led rosso L_S , quattro interruttori tattili T_1 , T_2 , T_3 , T_4 e un potenziometro Pot, disposti come rappresentato in figura:



I led verdi rappresentano quattro posizioni diverse in cui può comparire una mosca virtuale (led fly). Il led acceso indica che la mosca si trova in quella posizione. Durante il gioco, la mosca virtuale si sposta da una posizione all'altra in modo casuale (muovendosi sempre in una posizione adiacente), con tempi casuali. Scopo del gioco è tracciare il percorso della mosca mediante i pulsanti tattili, per cui quando la mosca si trova in posizione p ($1 \leq p \leq 4$) - ovvero si accende il led L_p - il player deve riuscire a premere il pulsante in posizione corrispondente T_p in tempo, prima che la mosca cambi posizione. E' definito un tempo minimo $TMIN$ (che dipende dal livello del gioco) per cui la mosca rimane in una posizione. La mosca può rimanere nella posizione in un tempo casuale compreso fra $TMIN$ e $K \cdot TMIN$, dove K è un parametro costante da definire nel gioco. Se l'utente preme in tempo, il gioco prosegue, diminuendo il tempo $TMIN$ minimo; se non preme in tempo o preme un pulsante non corretto, il gioco termina. Il potenziometro serve per definire il livello di difficoltà del gioco, determinato dal tempo $TMIN$ di partenza e dal fattore con cui man mano $TMIN$ diminuisce.

Comportamento dettagliato del gioco

Nello stato iniziale del gioco, tutti i led sono spenti tranne il led L_s si pulsa ad un ritmo regolare, in attesa che il player dia inizio al gioco.. Sulla seriale viene inviato il messaggio “Welcome to the *Track to Led Fly Game*. Press Key T1 to Start”.

Alla pressione del pulsante T_1 , parte il gioco. Tutti i led si spengono, viene inviato sulla seriale il messaggio “Go!”. Inizia il ciclo del gioco per cui:

- Viene scelta casualmente una posizione fra le tre ove posizionare la mosca e si accende il led corrispondente L_{pos} (con pos da 1 a 4).
- La mosca rimane in quella posizione per un tempo casuale compreso fra TMIN e $K \cdot TMIN$. Entro questo tempo il player deve riuscire a premere il pulsante tattile corrispondente T_{pos}
- Se il pulsante viene premuto in tempo, la mosca si sposta in una posizione casuale adiacente (ovvero, si spegne il led dov'era e si accende il led nella nuova posizione). Se è in una posizione di bordo (pos = 1 o pos = 4), la posizione di bordo include anche quella dalla parte opposta (esempio: posizione adiacente per pos = 1 sono pos = 2 e pos = 4).
- Ogni volta che si preme in tempo, in uscita viene inviato il messaggio: “Tracking the fly: pos X”, dove X indica la posizione corrente tracciata.
- Nel caso in cui il player non riesca in tempo a premere il pulsante o preme il pulsante sbagliato, il gioco termina, si accende L_s per 2 secondi e in uscita - concorrentemente - viene inviato il messaggio: “Game Over - Score: X”, dove X è la lunghezza del percorso tracciato. Quindi si torna allo stato iniziale del gioco.
- Ogni volta che la mosca si sposta in una nuova posizione, il tempo TMIN viene ridotto di un fattore pari a $\frac{1}{8}$ del valore corrente (ovvero $TMIN_{nuovo} = \frac{7}{8} TMIN_{precedente}$).
- Quando ci troviamo nello stato iniziale del gioco, mediante Pot deve essere possibile regolare il livello di difficoltà del gioco L, che si presuppone essere di 8 livelli - da livello 1 (più semplice) a livello 8 (più difficile). Il valore iniziale di TMIN deve essere quindi modulato opportunamente secondo il fattore L (per cui livello 1 implica valore TMIN iniziale più elevato mentre livello 8 implica valore DT iniziale più basso).

Realizzare il sistema su Arduino, implementando il programma in C + wiring.

Il gioco deve essere implementato con architettura di controllo super loop. Per i parametri non specificati, utilizzare valori che si ritengono sensati a partire da valutazioni empiriche. Per tutti gli aspetti non specificati, fare le scelte che si credono più opportune.

Il materiale da consegnare deve includere:

- Schema Fritzing o TinkerCad del sistema
- Sorgenti del progetto - indicando il nome dei componenti del gruppo nel file *.ino principale