

Report note - EXTRA points 2

Alessandra Comparetto 290207

Scelte implementative

Al fine di soddisfare le richieste della consegna le principali scelte implementative sono state:

- Inizializzare il **Timer 0** a $50\mu s$ per effettuare il polling sul display LCD e sui pin a cui è collegato il Joystick,
- Inizializzare il **Timer 1** per controllare la velocità del movimento del robot nel labirinto (1 *slot/s*),
- Definizione di una libreria **utilities.c** che riporta tutte le variabili e funzioni necessarie al corretto funzionamento del gioco

Librerie necessarie al progetto sono quelle del *TouchPanel*, *Joystick*, *GLCD* e *Timer*.

Step 1: Inizio della partita

Dopo aver mandato in esecuzione il codice ed aver eseguito la calibrazione del Touch Panel, il sistema provvederà alla preparazione della schermata iniziale di gioco (riga 23 di *sample.c*) tramite la chiamata alla funzione *preparaGioco()* definita in *utilities.c*.

Per iniziale la partita sarà sufficiente toccare l'area di gioco. Timer 0 è avviato dalla riga 26 di *sample.c* per effettuare polling. Dato che la partita non è ancora iniziata, il flag *partita* sarà uguale a 0, ma verrà portato ad 1 se il tocco percepito si trova nell'area di gioco (riga 37 di *IRQ_timer.c*). Si disegna il robot con la chiamata alla funzione *posizionaRobot()* di *utilities.c*.

Step 2: Modalità Move ed Explore

Riga 80 di *IRQ_timer.c*

Dopo aver avviato la partita il Joystick è abilitato (infatti le operazioni da effettuare quando uno dei pin associati al Joystick si attiva si trovano tutte sotto la riga 48 di *IRQ_timer.c*). Il cambio di modalità di gioco avviene con la pressione di Select: viene cambiato il flag *mode* (che vale 0 in modalità Move), si cambia colore al robot ridisegnandolo e viene scritto nello schermo in quale modalità si sta giocando.

Quando si attiva un pin compreso tra 26 e 29 (legati alle direzioni del Joystick), viene settata una nuova direzione. Quindi (riga 119) in base alla modalità di gioco, si ruota il robot o lo si fa muovere. Solo nel caso in cui la partita ancora non è stata vinta e ci si trovi in modalità Explore, si avvia il timer che controlla la velocità di movimento.

Il movimento è realizzato immediatamente per rendere quanto più differenti possibile le modalità Move ed Explore (altrimenti si potrebbe sempre esplorare nelle quattro direzioni rilasciando immediatamente il Joystick senza cambiare modalità).

Step 3: CLEAR button

Riga 64 di *IRQ_timer.c*

In ogni istante della partita è possibile cancellare tutti gli ostacoli individuati. Viene realizzato dalla funzione *clear()* di *utilities.c*. Questa disegna quattro rettangoli: uno che ricopre tutta l'area superiore alla posizione del robot, due per le aree laterali e uno per l'area sottostante. Durante questo processo, il testo del pulsante cambia colore in modo che sia possibile determinare il termine della procedura.

Step 4: Condizione di vittoria

Quando il robot si trova su una casella con posizione 2, il gioco termina. La condizione è verificata in *muovi()*, riga 271 di *utilities.c*. Viene stampato il testo di vittoria e settata la variabile flag *vittoria* ad 1. Infine vengono mostrati tutti gli ostacoli del labirinto e le uscite con la funzione *soluzione()*.

A questo punto del gioco l'unica azione consentita è l'inizio di una nuova partita, dunque la scritta del pulsante RESTART cambia colore (riga 275 di *utilities.c*).

Step 5: RESTART button

Riga 53 di *IRQ_timer.c*

Dopo la vittoria o in qualsiasi momento della partita è possibile ricominciare da capo premendo il pulsante di RESTART disegnato sullo schermo. La pressione di questo pulsante riporta le variabili flag *partita* e *vittoria* a 0 e ridisegna la schermata iniziale. Come per il pulsante di CLEAR, il colore della scritta ritorna viola a fine esecuzione.