

Blind labyrinth application note

Relazione di: Mattia Riola s280169

PROGETTO SVOLTO IN COLLABORAZIONE CON:

Mattia Riola S280169, Marco Smorti S281554, Gabriele Scaffidi Militone S288200

Per sviluppare il progetto dell'extrapoint2 si è usato il progetto 12_sample_GLCD_touchpanel già presente nel portale aggiungendo una funzione alla libreria GLCD per la stampa di triangoli (che raffigurano il robot) e una per la stampa di rettangoli. Abbiamo inoltre importato la libreria del RIT e la libreria dei joystick dal progetto 09_sample_JOYSTICK ampliando le loro funzionalità gestendo le direzioni del joystick e modificando il timer del RIT da 50ms a 25ms.

- Nel sample.c si inizializzano i timer (RIT e Timer0) per il polling del joystick e per la ricezione dell'input del touchscreen (in particolare per la gestione del tocco sui bottoni di restart e clear e sul pannello del labirinto).
- Negli handler del RIT viene gestito il funzionamento delle funzioni del joystick controllando il FIOPIN relativo a ogni direzione.

NOTA: Per questo progetto si decide di tenere le direzioni intermedie disabilitate e resettano il conteggio del secondo usato per il movimento (variabili south, north, east, west).

- Il cuore dell'applicazione è nella libreria labyrinth che è stata presa dall'extrapoints1 e modificata per questa nuova applicazione.
 - Le funzioni draw disegnano a schermo i vari "oggetti" del labirinto
 - I colori sono definiti all'inizio del .h per avere maggiore pulizia e controllo del codice.
 - La clear labyrinth implementa la funzionalità del pulsante clear ridisegnando il pannello del labirinto e il robot nella sua posizione precedente.
 - La start labyrinth inizializza la partita impostando correttamente le variabili utilizzate nella libreria per il funzionamento del gioco nella situazione di partenza, inoltre riattiva le funzionalità del joystick riabilitando il polling.
 - L'init_labyrinth crea l'interfaccia utente che rimarrà sempre stampata sullo schermo
 - La restart_labyrinth implementa la funzionalità di restart permettendo all'utente di iniziare una nuova partita
 - La obstacle_scan esegue il controllo degli ostacoli e viene chiamata dalla robot move e dalla robot explore dopo ogni movimento e ogni rotazione.
 - La check victory controlla se la posizione del robot è la casella della vittoria (quest'ultima non viene mostrata sullo schermo quando rilevata dai sensori del robot in quanto non è richiesto dalle specifiche) e stampa a video la scritta della vittoria disabilitando il joystick (agendo sul timer del polling) e il tasto clear (mediante la variabile globale victory).
 - Per visualizzare lo spostamento e la rotazione del robot si usa la cancel robot nella posizione iniziale e la draw robot con la nuova direzione aggiornata. Il controllo del colore del robot viene effettuato nella draw robot con un check della variabile globale mode.
 - Le coordinate delle posizioni del labirinto vengono trasformate dalla coord_transform per ottenere la posizione dei pixel e poter disegnare correttamente le figure sullo schermo