

La specifica 1 richiede di garantire l'interazione con il tamagotchi tramite TouchScreen. Alla pressione dello schermo nell'area dove è presente la figura del pet (di qui in avanti area di lavoro) deve essere avviata una animazione della durata di due secondi. Al termine dell'animazione deve aumentare di una tacca lo stato di HAPPINESS del pet.

Per poter assolvere alla specifica è necessario garantire una corretta interazione con il touch screen, per fare ciò è necessario alla inizializzazione della scheda procedere con l'inizializzazione anche dello schermo *LCD_Initialization()*; e contestuale inizializzazione del TouchScreen con la funzione *TP_Init()*. Per poter rimuovere le non idealità del touch panel è necessaria una taratura dello stesso tramite una funzione di calibrazione *TouchPanel_Calibrate()*. Maggiori dettagli sulle funzioni indicate e sul loro signature sono disponibili nella documentazione allegata. Il controllore del TP è l'ADS7843 di cui maggiori dettagli disponibili nel file **07_ads7843**, mentre l'algoritmo di calibrazione utilizza il metodo dei 3 punti ed è descritto in **07b_touch_screen_calibration_application_note**.

In fase di esecuzione del gioco ovvero quando la condizione `if(session.game_status_var == PLAY)` risulta vera viene ripetutamente effettuato del polling per poter valutare se una specifica area dello schermo sia stata oggetto di interazione tramite tocco dell'utente. Affinché sia eseguito il polling che identifica tale evento è configurato il TIMER 2 con MCR impostato a 0x3 ed il registro MR0 al valore di 125000. La frequenza di funzionamento del timer è di 12.5MHz. Ogni volta che il TC = MR0 ovvero ogni 10 ms il timer viene resettato, e viene generata una interruzione che si occupa di verificare la presenza un nuovo evento di tocco richiamando la funzione `void do_user_events_interrupt_touch_panel(void)` dove una funzione apposita `uint8_t getDisplayPoint(Coordinate * displayPtr, Coordinate * screenPtr, Matrix * matrixPtr)` verifica se sia presente un tocco del touch panel ed in caso positivo ritorna valore 1 e copia le coordinate del tocco in una variabile globale DISPLAY di tipo Coordinate (struct che descrive coordinate spaziali X ed Y). Attraverso una variabile globale SESSION definita nella libreria tamagotchi.h è possibile accedere ad una struttura display_map che mantiene l'informazione circa le dimensioni attuali della rappresentazione del pet ed altre coordinate con riferimento al dispaly. Confrontando la dimensione della figura con le coordinate nella variabile DISPLAY catturate al passo precedente è possibile verificare se il tocco sia avvenuto o meno nell'area di lavoro. In caso positivo viene generato un evento utente con una variabile di tipo user_event_t contenente nel campo di descrizione dell'evento `event_tmp.event = TOUCH_CUDDLES`. La variabile evento utente viene quindi inserita tramite push su una coda LIFO `events_buffer_staging_tp` che mantiene la serializzazione degli eventi di tocco, in particolare si verifica se esiste un precedente evento di classe *TOUCH_CUDDLES* in coda ed in caso positivo viene aggiornato il numero di volte in cui tale evento viene triggerato, in caso negativo invece l'evento viene inserito in coda. Estruendo un evento dalla staging area qualora l'evento risulti terminato ovvero non sono presenti ulteriori interazioni per la data classe di evento questo viene inserito in una seconda coda LIFO `events_buffer_done` area la quale mantiene gli eventi di differenti classi serializzati. Estruendo da questa seconda coda qualora l'evento risulti essere *TOUCH_CUDDLES* viene richiamata una funzione `tamagotchi_user_wrapper_cuddle(session_t * session)` la quale si occupa di settare l'animazione.

La funzione indicata si occupa di settare il timer 0 in modo tale che generi una interruzione ogni 2s/ numero di frame della animazione che nel mio caso sono 6. La stessa funzione si occupa di settare la melodia da riprodurre durante l'animazione di cui si fa riferimento alla documentazione sulla specifica suono. Il timer 0 è condiviso con la gestione della animazione EATING e per cui è necessaria una variabile di stato che identifichi la corretta animazione custom da mostrare. Ad ogni interruzione del timer 0 viene richiamata la funzione `void do_tamagotchi_custom_animation()`; la quale verificando che l'attuale animazione sia *CUDDLES* richiama la funzione di stampa del frame della animazione (il numero del frame viene incrementato ad ogni interruzione). Per poter rappresentare su schermo l'animazione è stato necessario partire da una gif animata, la quale processata tramite uno script Python, fornisce frame by frame e per ogni pixel il colore corrispondente. Tale codifica RGB passa per una funzione di compressione RGB565CONVERT in modo tale da garantire la rappresentazione del codice del colore su 16 bit. Le informazioni sulla colorazione di un pixel del frame sono raccolte all'interno di

una matrice che mantiene i frame processati. Il frame corrispondente è stampata su schermo tramite la funzione della libreria delle animazioni *void draw_cuddles(int start_frame, int end_frame);* conoscendo il frame di partenza e quello di destinazione è possibile stampare su schermo solamente le variazioni e non l'intera matrice del frame. Durante l'ultima interruzione viene richiamata la funzione *void tamagotchi_happiness_update (session_t* session , int update)* con *update=1* la quale incrementa (qualora sia possibile) di una tacca lo stato della felicità del pet.