RELAZIONE FINALE PROGETTO MICROPROCESSOR BASED INSTRUMENTATIONS

Gruppo 4:
Micheletti Jacopo 729452
Sandrini Pietro 728328

1 Descrizione del progetto

Il nostro progetto consiste nella realizzazione di un microcontrollore che ci permetta di far suonare un buzzer, più nel dettaglio la realizzazione si compone di due parti fondamentali:

1. Implementare una breve canzone a nostra scelta
2. Ricevere una nota a scelta tramite comunicazione seriale e riprodurla per una durata prefissata

Abbiamo ipotizzato un range di lavoro del buzzer tra 500 e 10000 Hz, in modo tale da sfruttare pienamente la caratteristica di risposta in frequenza del dispositivo.

2 Schema elettrico

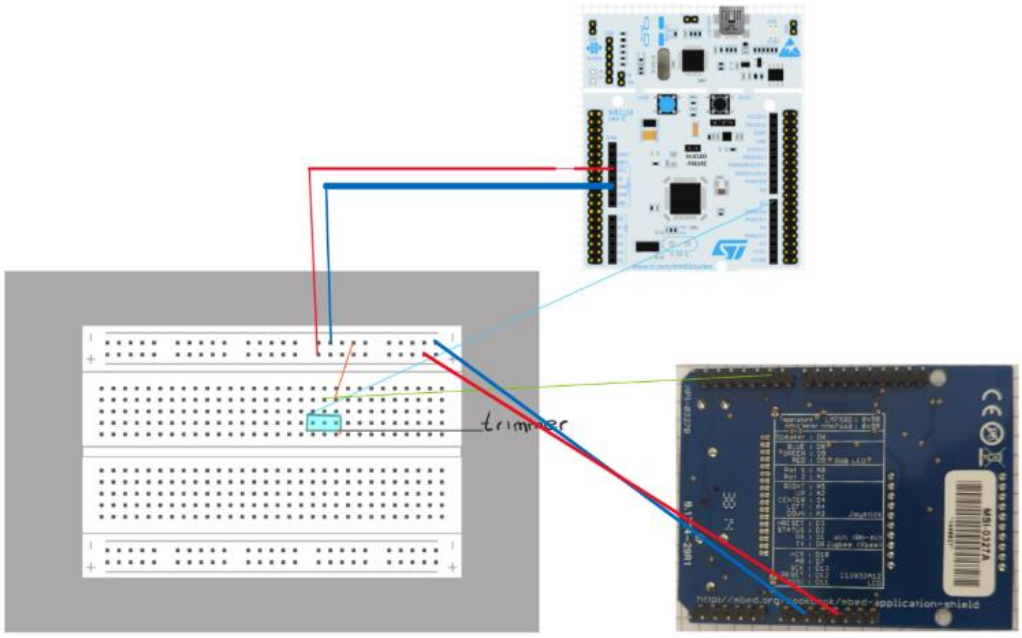

Figura 1 Schema circuitale

Lo schema è composto da due dispositivi, il microcontrollore STM32F401 e MBED application shield utilizzato per la riproduzione dei suoni. Il microcontrollore viene alimentato tramite usb, esso fornisce l'alimentazione di 3.3V alla breadboard e allo shield tramite i collegamenti blu e rossi. Nel circuito è presente un trimmer che permette di regolare il volume del buzzer.

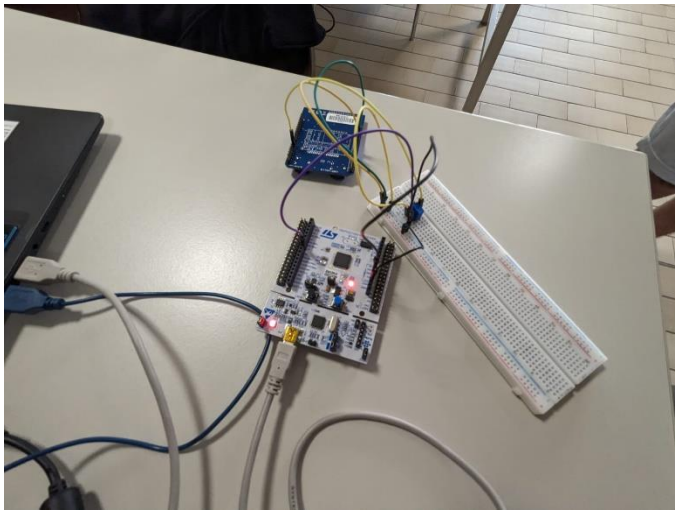

Figura 2 Circuito montato

3 Macchina a stati

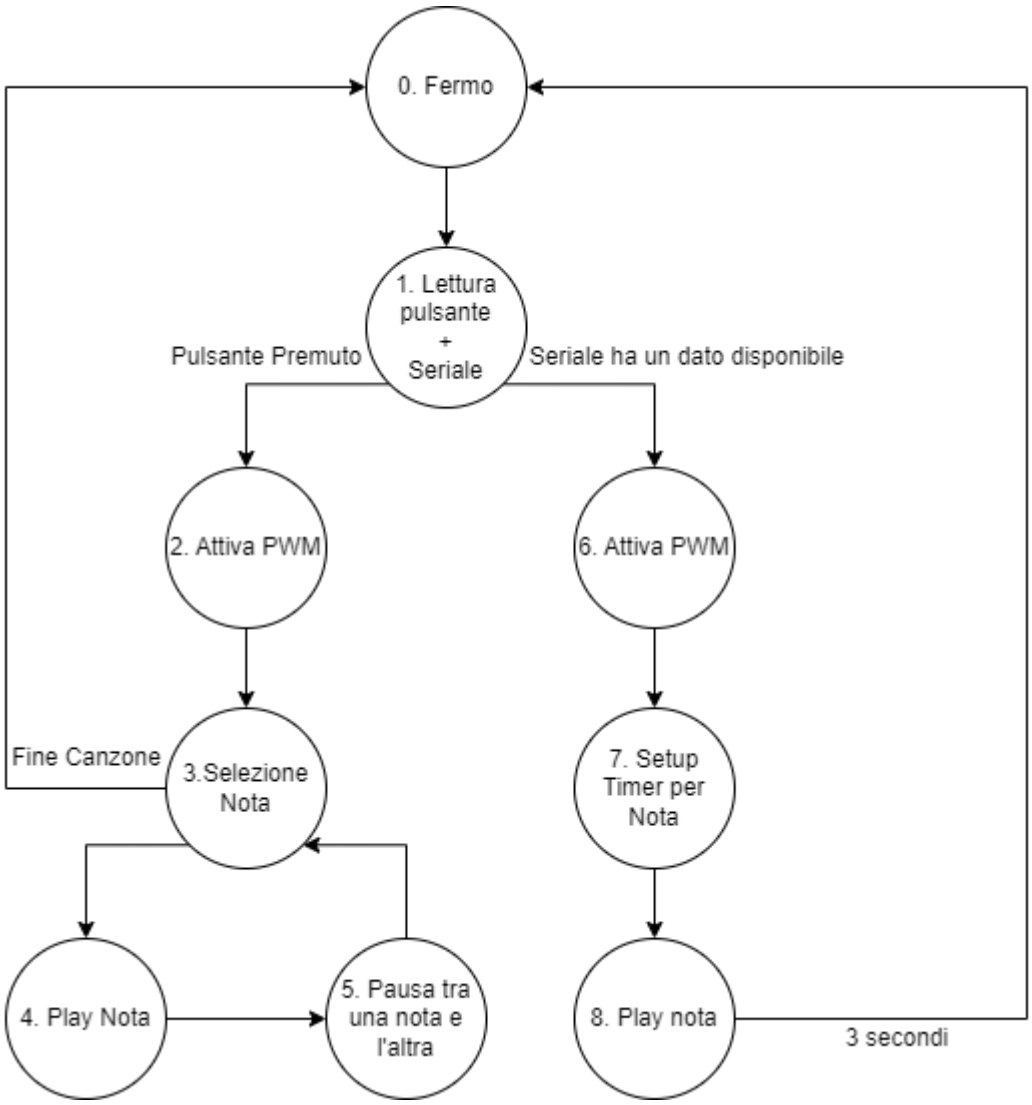

Figura 3 Macchina a stati

Per quanto riguarda le scelte applicate nell'implementazione del microcontrollore, abbiamo deciso di usare una macchina a stati composta da otto stati.

Dalla condizione iniziale di tutto fermo e spento, si passa allo stato uno di attesa di un input da parte dell'utente. Gli input possono essere di due tipi: la pressione del pulsante presente sulla scheda oppure la ricezione di un dato dall'interfaccia UART. Nel primo caso il programma passa allo stato due attivando il PWM ed entrando in un ciclo in cui avviene la riproduzione di una canzone, al termine della quale il sistema ritorna nello stato di fermo e aspetta nuovamente un input.

Nel secondo caso il programma riproduce la nota specificata dall'utente tramite la seriale. La nota viene riprodotta per un tempo arbitrariamente stabilito pari a tre secondi. Una volta finita la riproduzione della nota, il sistema torna ad aspettare nuovi input. Il dato da inviare via seriale deve essere un numero compreso tra 500 e 10000, che rappresenta il range di frequenze che abbiamo ipotizzato.

4 Scelte progettuali e codice

Abbiamo deciso di implementare il programma utilizzando una macchina a stati, per questo motivo nel while loop è presente uno switch case che consente si passare da uno stato ad un altro. Come già detto in precedenza ci sono due modalità implementate una in cui viene riprodotta una canzone già preimpostata e un'altra in cui viene inviato un dato dalla seriale e viene riprodotta la frequenza specificata. Abbiamo scelto di far partire la canzone solo quando viene premuto il pulsante on-board della scheda, mentre la riproduzione della frequenza su seriale avviene appena un dato è disponibile e il dispositivo non si trova in un'altra modalità. La canzone preimpostata è salvata sotto forma di un array chiamato melody, composto da una serie di valori in successione; vi è una sequenza di nota da riprodurre e tempo divisore (parametro che determina per quanto tempo si riproduce una nota). Es: melody[] = {nota, tempo, ……}

A ogni ciclo il programma legge gli input, seriale e pulsante e decide lo stato successivo a cui passare. Se il pulsante è premuto avviene la riproduzione della canzone seguendo i seguenti passi:

- Selezione della nota: per prima cosa si verifica che la canzone non sia conclusa, nel caso in cui la canzone fosse terminata si torna allo stato zero. Altrimenti viene selezionata la nota da riprodurre e il tempo per cui riprodurla, estraendo i valori dall'array denominato melody. La nota corrente viene identificata dalla variabile idNota. Vengono quindi calcolati i parametri prescaler, noteDuration e maxCount. Il prescaler è calcolato in modo da ottenere la frequenza desiderata in uscita al PWM, si utilizza la formula seguente formula:

$$PSC = \frac{Fck}{Fpwm \cdot (ARR + 1)} - 1$$

  La noteDuration viene calcolata in base alla durata di una nota intera e rappresenta la durata in secondi di riproduzione della nota. Infine, il maxCount viene calcolato in modo da ottenere una durata della nota pari a noteDuration, tramite la funzione calcolaConteggio. Viene quindi settato il dutyCycle del PWM al 50%, in modo che la nota possa essere riprodotta.
- Riproduzione della nota: il countCicli viene incrementato ogni volta che il timer raggiunge il suo conteggio massimo fino al raggiungimento del maxCount. Verificatasi questa condizione si resetta il countCicli in modo da essere utilizzato per il tempo di pausa e si setta il dutyCycle

del timer a 0%, in modo da non riprodurre alcun suono, infine si passa allo stato di pausa tra una nota e l'altra.

- Tempo di pausa tra una nota e la successiva: come nello stato precedente il countCucli viene incrementato fino al raggiungimento del maxCount. Raggiunta la condizione, si incrementa idNota in modo tale da selezionare la nuova nota da riprodurre. Si torna quindi allo stato di selezione nota.

Nel caso in cui il pulsante non fosse premuto si passa alla lettura dell'UART. Si va a verificare il contenuto del buffer tramite un ciclo for e si verifica se siano presenti i caratteri "\n\r", in tal caso si converte il dato in ingresso in un intero. Questo dato rappresenta la frequenza da riprodurre. Abbiamo deciso di riprodurre le note provenienti dalla seriale per tre secondi. Quindi si eseguono le seguenti operazioni:

- Calcolo di presclaer, noteDuration e maxCount: questi tre parametri sono necessari per la riproduzione della nota.
- Rirpoduzione della nota: come nel caso precedente il countCucli viene incrementato fino al raggiungimento del maxCount. Verificata la condizione si torna nello stato zero e infine alla lettura dei nuovi input.

5 Risultati

Il risultato ottenuto è la possibilità di suonare una breve canzone alla pressione del pulsante sul microcontrollore la possibilità di inviare al controllore tramite seriale una nota da riprodurre, per un tempo scelto di tre secondi.

I test sulle funzionalità del programma possono essere visionati tramite i video in allegato.

Nel primo video viene mostrata l'esecuzione della breve canzone conseguentemente alla pressione del pulsante sul microcontrollore. La canzone scelta è "Never gonna give you up" di Rick Astley.

Nel secondo video invece viene mostrata la comunicazione seriale, in particolare come inviando il valore in frequenza attraverso il programma RealTerm di una determinata nota questa venga riprodotta dal buzzer per un durata di 3 secondi, in particolar modo le frequenze usate per il video sono state: 1319 e 2349 Hz.