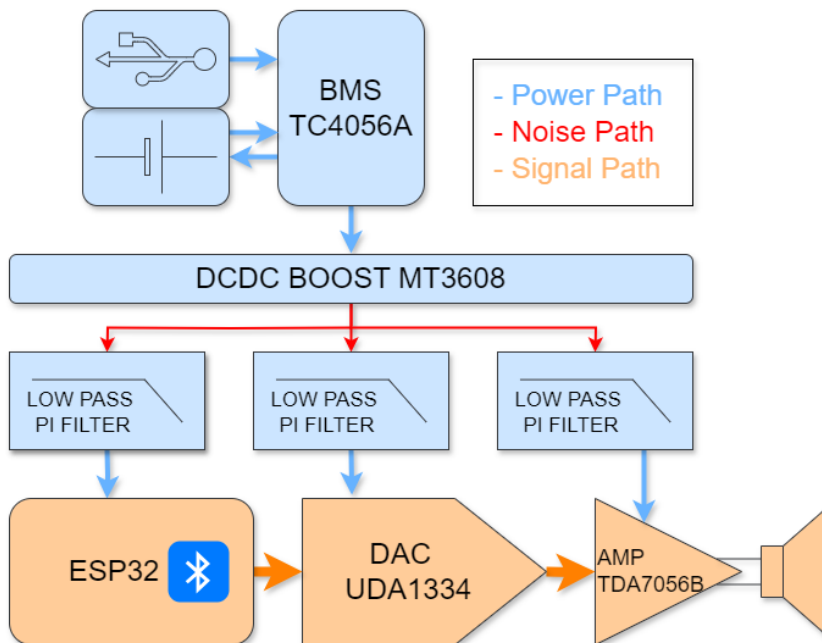


Sistemi e Tecnologie Elettroniche Avanzate

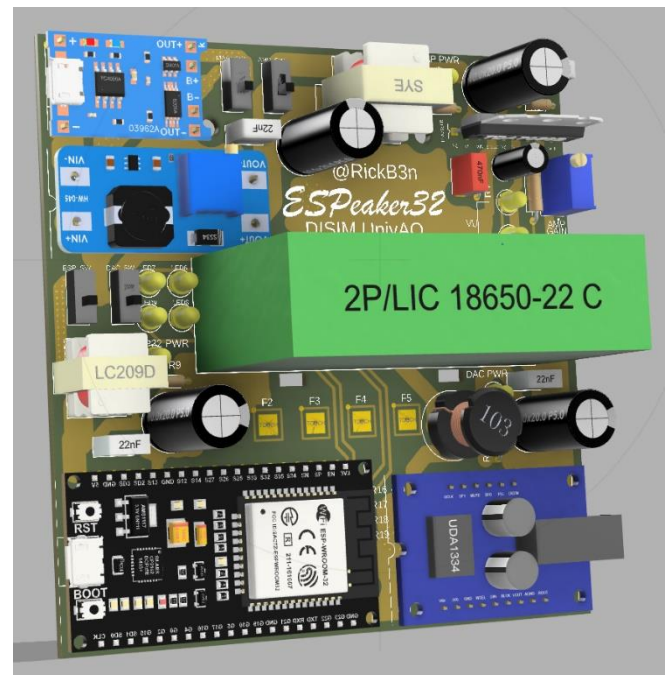
Tesina: "ESpeaker32"

Studente: Riccardo Beniamino 24 40 54

Schema a Blocchi:



Rendering 3D:



Descrizione breve del sistema:

Lo scopo del sistema è quello di replicare uno speaker bluetooth commerciale. Il dispositivo si alimenta da batteria agli ioni di litio e si carica da porta micro-usb. Il microcontrollore implementa un client bluetooth verso il quale è possibile inviare uno stream audio stereo (2 canali). L'audio in formato digitale viene convertito in analogico tramite il DAC, dopodiché viene amplificato e inviato al trasduttore, ossia all'altoparlante. **Le varie fasi saranno approfondite nei successivi paragrafi**

• Power Path (e Noise Path):

Il dispositivo per funzionare ovviamente ha bisogno di una sorgente di alimentazione. Per questo scopo viene utilizzata una batteria agli ioni di litio [VARTA 2P/LIC 1865-22 C](#) contenente due celle standard 18650 collegate in parallelo e un BMS PCM310 di sicurezza che serve a proteggere la batteria da scarica e carica eccessive e da sovracorrente. In scenari di potenza è possibile anche controllare la temperatura della batteria, tramite un sensore NTC incorporato (in questa applicazione è superfluo).

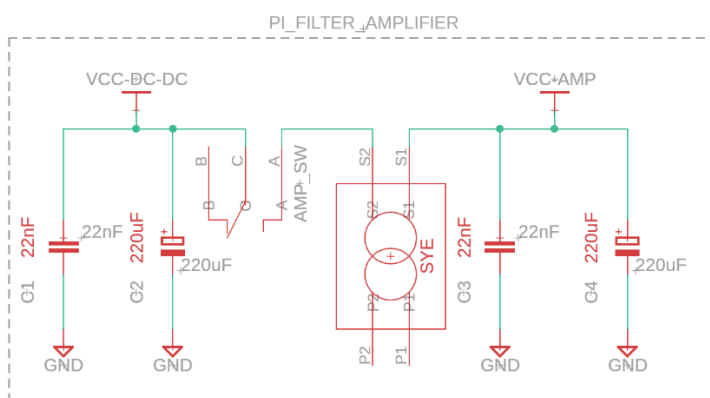
La batteria ha una capacità (quando nuova) di 4400 mAh più che sufficienti per lo scopo in quando il sistema ha un consumo medio sul rail 3,3V batteria pari a 300 mA quindi la durata teorica massima è pari $4400 \text{ mAh} / 300 \text{ mA} = 14 \text{ h}$. Ovviamente questo è un dato teorico e molto ottimistico, poi nella realtà dipende da come viene utilizzato effettivamente il dispositivo.

Per gestire opportunamente la carica e scarica della batteria viene utilizzato un secondo BMS [TC4056A](#) esterno che serve appunto a monitorare costantemente la tensione della batteria e la corrente in entrata e in uscita dalla batteria. Per esempio la corrente di carica viene mantenuta costante ad 1A per non stressare eccessivamente la batteria durante la carica. La corrente di 1A è compatibile con ogni alimentatore USB. Per una carica dal 0% al 100% saranno necessarie circa 4,5 ore.

La tensione della batteria agli ioni di litio carica tipicamente è pari a 4,2V tramite il convertitore DCDC Boost [MT3608](#), viene porta a 7 Volt, in questo modo è compatibile per l'alimentazione dell'amplificatore audio, mentre per quanto riguarda, il microcontrollore e il DAC, viene riportata a 3,3V tramite dei regolatori lineari.

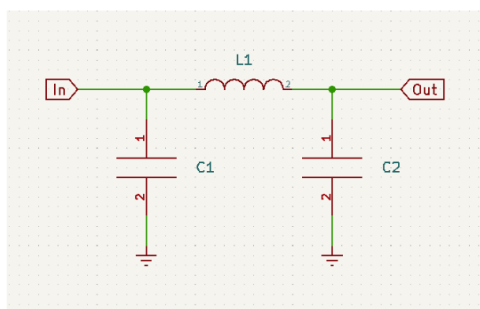
Il convertitore DCDC Boost [MT3608](#) operando a una frequenza di 1.2 MHz, introduce una quantità considerevole di rumore ad alta frequenza, nella fondamentale e nelle armoniche. Per risolvere questo problema vengono utilizzati tre filtri passa basso di tipo pi-greco (CLC).

Ad esempio questo è il filtro utilizzato per alimentare l'amplificatore audio:



In questo modo il filtro ha una banda passante (-3 dB) pari a 253 Hz, “ottima”.

A LPF consists of one series inductor (L1) and two identical shunt capacitors (C1=C2) as shown in the picture below.



Enter **L** and **C** values below to find the cut off.

Inductor L	Units
7.21	Millihenry
Capacitor (C)	Units
220	Microfarad (uF)
Cutoff Frequency	Units
252.74580915847184	Hz

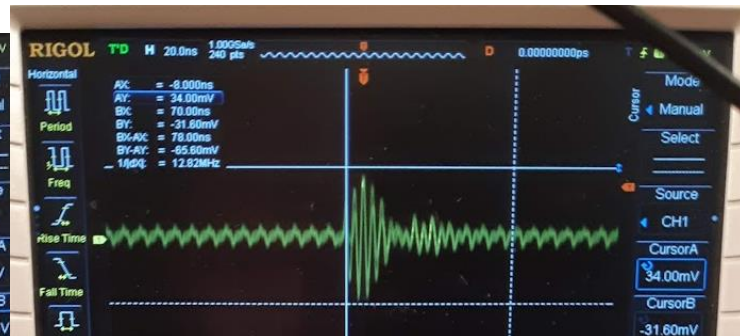
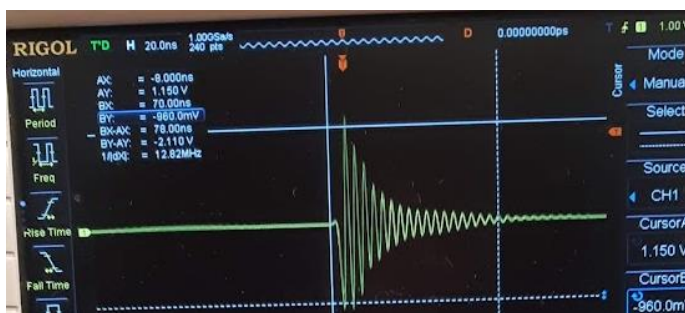
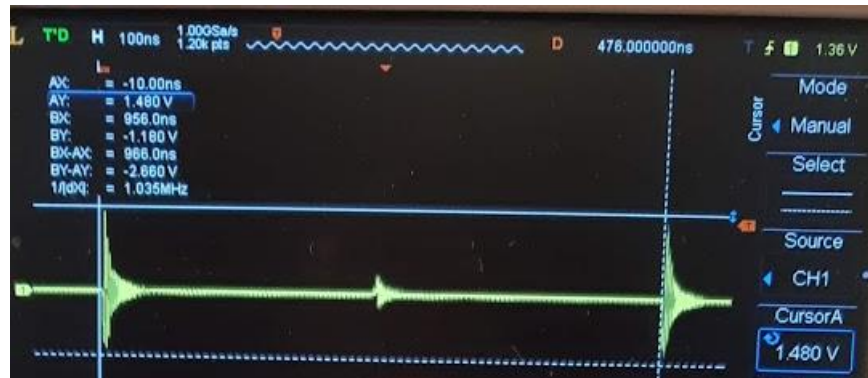
Formula

$$f_c = 1/(\pi * \sqrt{LC})$$

Invece se trascuro i condensatori elettrolitici notoriamente “lenti” ottengo una banda passante pari a 25 kHz, troppo grande.

Il parallelo degli elettrolitici e dei poliestere permette di avere un buon compromesso tra le due bande calcolate.

Misurazione del rumore prima e dopo il filtro:



Partendo dalla prima immagine in alto a sinistra vediamo la sonda con il contatto di massa opportuno per misure ad alta frequenza, il coccodrillo classico agirebbe da antenna falsando la misura.

Nella seconda immagine si vede la frequenza di switching pari a 1.035 MHz del convertitore DCDC Boost. Nella terza immagine in basso a sinistra abbiamo lo stesso segnale, zoomato nel tempo.

Quindi prima del filtro abbiamo dei transitori di rumore con una freq. pari a 12.82 MHz che si ripetono a 1.035 MHz e una ampiezza picco-picco massima pari a 2.110V.

Dopo il filtro abbiamo lo stesso segnale ma con una ampiezza pari a 65 mV in questo modo il rumore è stato attenuato di 32.5 volte. ($2.110V/X = 0.065V$, $X=32.5$)

$$\text{Attenuaz. in dB} = 20 \times \log_{10} \left(\frac{V_{out}}{V_{in}} \right) = 20 \times \log_{10} \left(\frac{0.065V}{2.110V} \right) = -30.23 \text{ dB}$$

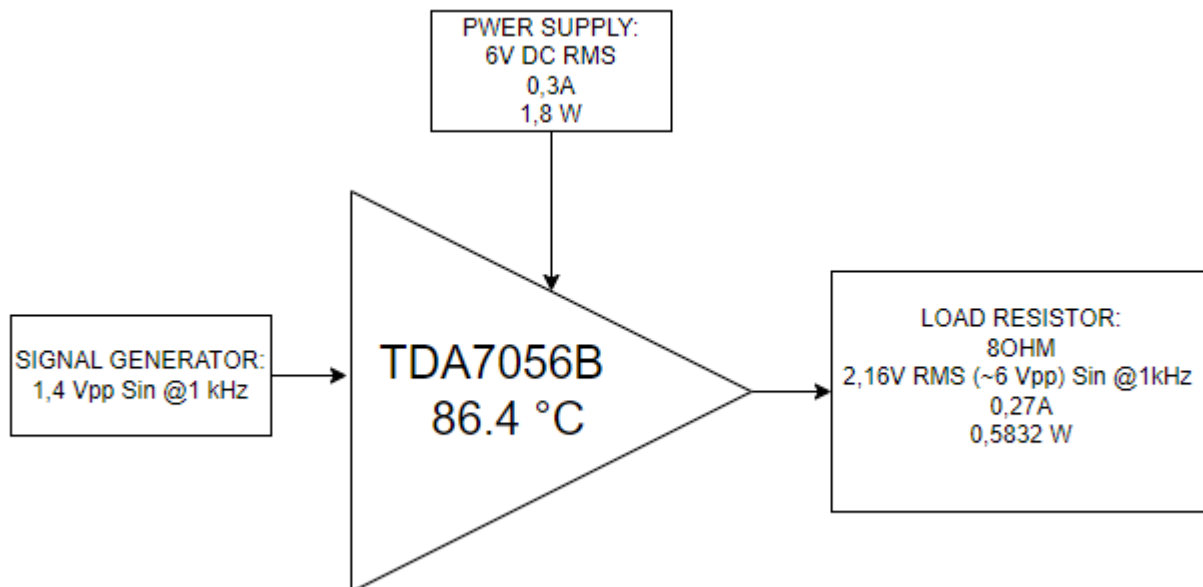
Questo dimostra l'efficacia del filtro.

Quindi nei paragrafi precedenti abbiamo affrontato le principali criticità del power e noise path, dal punto di vista elettrico.

- **Profilo termico:**

Per quanto riguarda la dissipazione del calore il componente più critico è ovviamente l'amplificatore [TDA7056B](#). Essendo un classe AB, configurazione BTL degli anni 90 è terribilmente inefficiente dal punto di vista energetico e quindi la maggior parte della potenza in ingresso, viene persa in calore.

Dalle misurazioni fatte in laboratorio, si evince il seguente comportamento:



In ingresso abbiamo 6V e 0,3A in DC, in uscita abbiamo una sinusoide a 1 kHz con ampiezza pari a 2,16V RMS, con un carico resistivo pari a 8 Ohm.

La potenza in entrata sarà: $P_{in} = V_{in} \times I_{in} = 6V \times 0,3A = 1,8W$

La potenza in uscita sarà: $P_{out} = R_{out} \times I_{out}^2 = 8\Omega \times \left(\frac{2,16V}{8\Omega}\right)^2 = 0,5832W$

Efficienza dell'amplificatore: $\eta = \frac{P_{out}}{P_{in}} = \frac{0,58W}{1,8W} = 0,32 \rightarrow 32\%$

La pot. dissipata in calore sarà: $P_{diss} = P_{in} - P_{out} = 1,8W - 0,5832W = 1,22W$

Dal [datasheet](#) vediamo:

THERMAL CHARACTERISTICS

SYMBOL	PARAMETER	VALUE	UNIT
$R_{th\ j-a}$	thermal resistance from junction to ambient in free air	55	K/W
$R_{th\ j-c}$	thermal resistance from junction to case	10	K/W

La resistenza termica $R_{th\ j-a}$ junction to air, dalla giunzione all'aria, ossia senza dissipatore è pari a 55 K/W

Non tenendo conto della temp. ambiente: $Temp_{TDA7056B} = P_{diss} \times R_{th\ j-a}$

Questo significa che: $55 \left[\frac{K}{W}\right] \rightarrow 1,22[W] \times 55 \left[\frac{K}{W}\right] = 67,1[K] \rightarrow 67,1[^\circ C]$

Andando a sommare la temp. ambiente otteniamo: $67[^\circ C] + 20[^\circ C] = 87[^\circ C]$

Il conto teorico, infatti, è in linea con la misurazione di temperatura del laboratorio.



L'amplificatore ha una T_{vj} virtual junction temperature pari a $150\text{ }^{\circ}\text{C}$

Quindi quella sarà la sua temperatura massima dalla quale stare lontani.

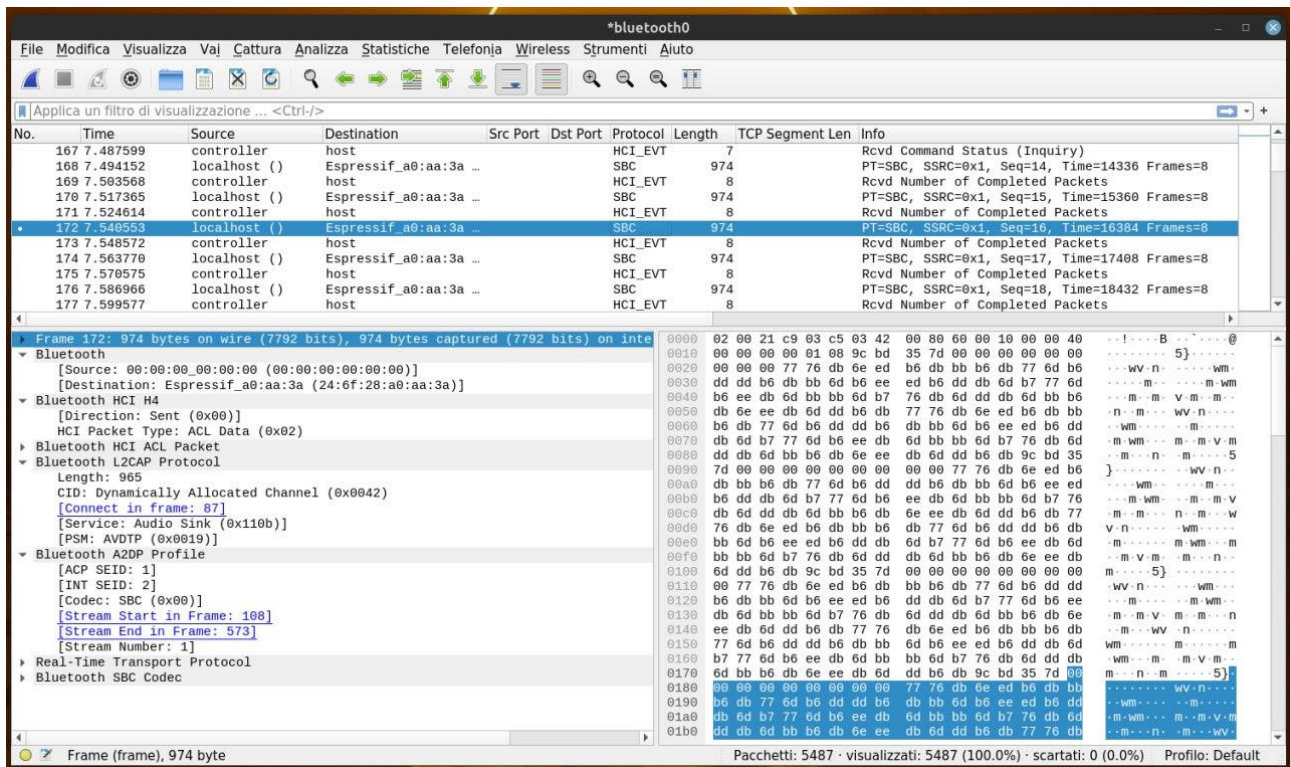
Nel circuito l'amplificatore lavora sempre a potenze inferiori rispetto a quelle della prova di laboratorio, quindi **il dissipatore non è necessario**. Anche se alimentato a 7V c'è l'induttore che non solo filtra il rumore hf ma fa anche da limitatore di corrente con i suoi parassiti, ad esempio la resistenza nel rame.



Qui si vede bene questo comportamento, il segnale giallo è un audio in ingresso all'amplificatore, il segnale blu è l'audio in uscita e il segnale viola è la tensione di alimentazione dell'amplificatore. Nei momenti di riposo è a 7V, mentre nei momenti di maggior richiesta di corrente scende a 4V. Quindi siamo ben lontani dai 6V della prova di laboratorio e quindi a temperature inferiori a $87\text{ }^{\circ}\text{C}$. Inoltre l'altoparlante non è una resistenza fissa a 80hm ma la sua impedenza cambia in continuazione in base alla frequenza, contribuendo a migliorare la situazione.

- **Signal Path**

Una volta acceso il microcontrollore [ESP32](#) implementa un client Bluetooth, verso il quale è possibile instaurare una connessione a pacchetti, e inviare lo stream di dati, contenenti frammenti di audio stereo (2 canali) codificati col profilo [A2DP](#) ad alta fedeltà.



Questi frammenti vengono memorizzati in un buffer e reindirizzati al DAC [UDA1334](#), il quale può essere configurato con una risoluzione verticale fino a 24 bit.

Il protocollo di comunicazione tra il microcontrollore e il DAC è il [I²S](#) Integrated Interchip Sound, standard comune nel mondo dell'audio digitale.

Interfaccia [I²S](#)

"[I²S](#)" in relazione con "[I²C](#)".

Esiste effettivamente una qualche relazione tra questi due protocolli. Entrambi sono stati inizialmente sviluppati da Philips Semiconductors (ora NXP) ed entrambi iniziano con "I²" perché sono destinati alla comunicazione tra circuiti integrati.

Tuttavia, [I²S](#) è stato rilasciato dopo [I²C](#) e, sebbene [I²C](#) sia un'interfaccia generica, [I²S](#) è progettata per trasportare dati audio: la "S" nel nome sta per "sound".

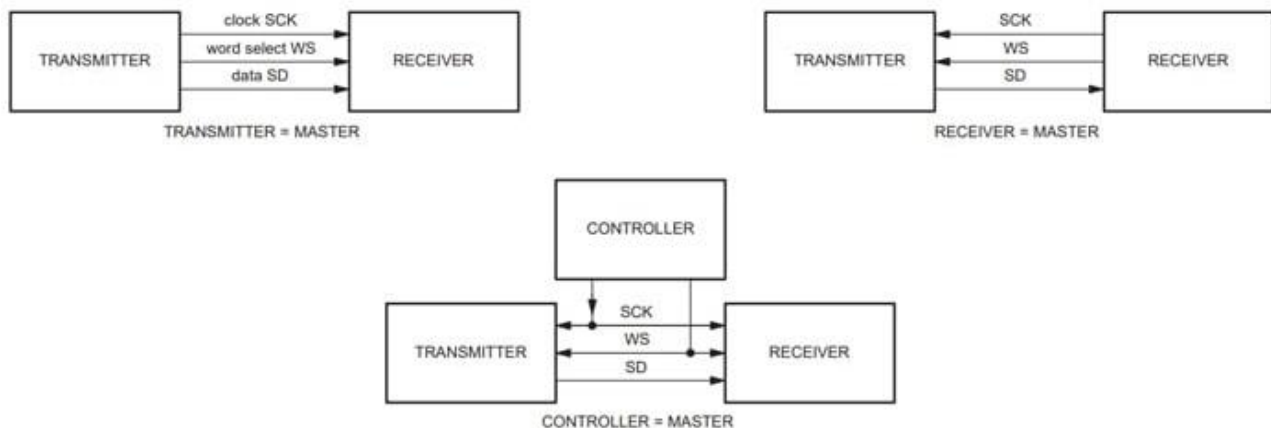
Il protocollo [I²S](#) è nato negli anni 80, quando il digitale stava iniziando a conquistare il mercato dell'audio consumer. Lo scopo dichiarato di [I²S](#)

è facilitare lo sviluppo dell'elettronica audio mediante un'interfaccia standardizzata per la trasmissione di dati digitali tra ADC, DAC, filtri digitali, processori di segnali digitali e altri tipi di circuiti integrati utilizzati nei sistemi audio. È intrinsecamente un protocollo a due canali, perché è stato progettato per il suono stereofonico.

Caratteristiche di [I2S](#)

Il diagramma seguente illustra le tre configurazioni supportate da [I2S](#)

Nel ns caso utilizziamo la prima configurazione in alto a sinistra, dove l'[ESP32](#) è il Transmitter (Master) e l'[UDA1334](#) è il Receiver (Slave)



I dati vengono trasportati sulla linea SD “Serial Data”, lo stato della linea WS “Word Select” corrisponde al canale audio (sinistro o destro) attualmente trasmesso e la linea del clock SCK “Serial Clock” trasporta appunto il clock seriale.

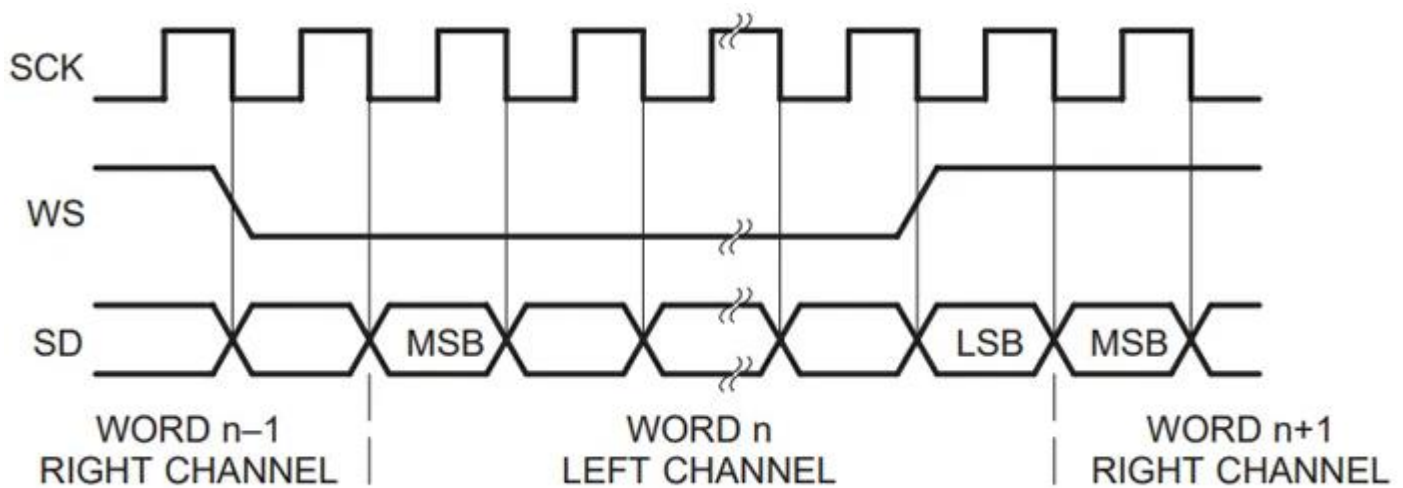
I segnali WS e SCK possono essere generati dal trasmettitore (nostro caso), dal ricevitore o da un componente controller esterno.

Serial Data SD:

- I valori digitali vengono trasmessi MSb first, ovvero bit più significativo prima.
- Non è necessario che il trasmettitore e il ricevitore abbiano una lunghezza di “parola” concordata. Il trasmettitore invia ciò che ha e il ricevitore prende ciò che può utilizzare.
- I nuovi bit di dati possono essere trasmessi sul fronte di salita o di discesa del clock. Tuttavia, devono essere acquisiti sul fronte di salita, quindi l’approccio più diretto in questo caso è l’organizzazione mostrata nel diagramma sottostante. Ovvero trasmettiamo i dati sul fronte discendente e li acquisiamo sul fronte di salita del segnale di clock.
- Il protocollo non include periodi di clock inutilizzati tra le “parole”. Il LSb di una parola è seguito immediatamente dall’MSb della “parola” successiva.

Word Select WS

- Un segnale logico basso su WS indica che la parola attualmente in trasferimento fa parte dello stream di dati per il canale audio sinistro; un segnale logico alto su WS indica l'audio del canale destro.
- Per agevolare la gestione dei dati sia sul lato trasmettitore che sul lato ricevitore, il segnale WS transita un periodo di clock prima del completamento di una “parola” di dati.



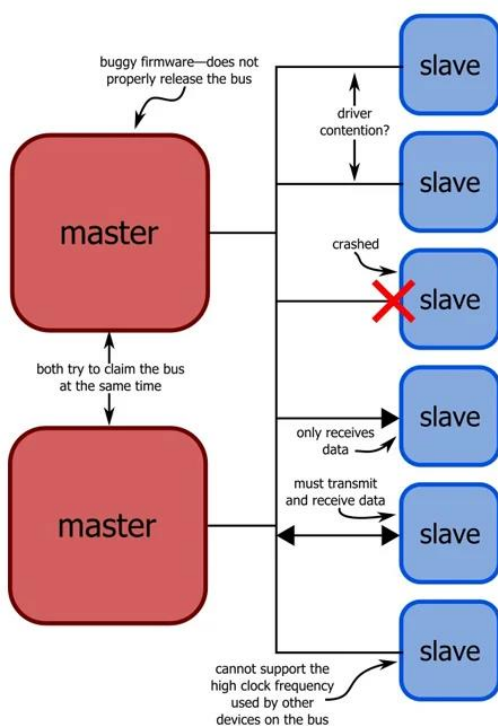
Clock

- Il protocollo non specifica una velocità massima di trasmissione dei dati. Il clock funziona continuamente.

I2C vs. I2S

Se sei familiare con il protocollo [I2C](#), potresti aver realizzato ora che [I2C](#) e [I2S](#) sono molto meno simili di quanto i loro nomi possano suggerire.

[I2C](#) non enfatizza alte velocità di trasmissione dati e coinvolge caratteristiche di handshake che consentono di funzionare in modo efficace e affidabile in reti (potenzialmente grandi) composte da vari tipi di circuiti integrati. Come indicato dal diagramma seguente, molte cose possono andare storte in questo tipo di ambiente di comunicazione, e la complessità del protocollo [I2C](#) riflette la complessità del compito per cui è destinato.



D'altra parte, [I2S](#) è progettato per spostare efficientemente un tipo specifico di dati digitali. La velocità di trasferimento è più importante, poiché la trasmissione serializzata in tempo reale di audio ad alta risoluzione a due canali richiede molto più larghezza di banda rispetto alle operazioni di comunicazione spesso eseguite tramite [I2C](#) (o [UART](#)).

La natura punto-punto della trasmissione [I2S](#) elimina la necessità dei resistori pull-up di [I2C](#) e l'uso di un terzo segnale per la sincronizzazione a livello di parola ci consente di evitare i dettagli del protocollo che aiutano [I2C](#) a mantenere l'organizzazione dei dati in un bus a due fili.

[I2S](#) assomiglia più a [SPI](#) che a [I2C](#). Infatti, un'implementazione [SPI](#) destinata alla trasmissione unidirezionale dei dati utilizza essenzialmente la

stessa configurazione: un segnale per il clock, uno per i dati e un terzo per la sincronizzazione a livello di parola.

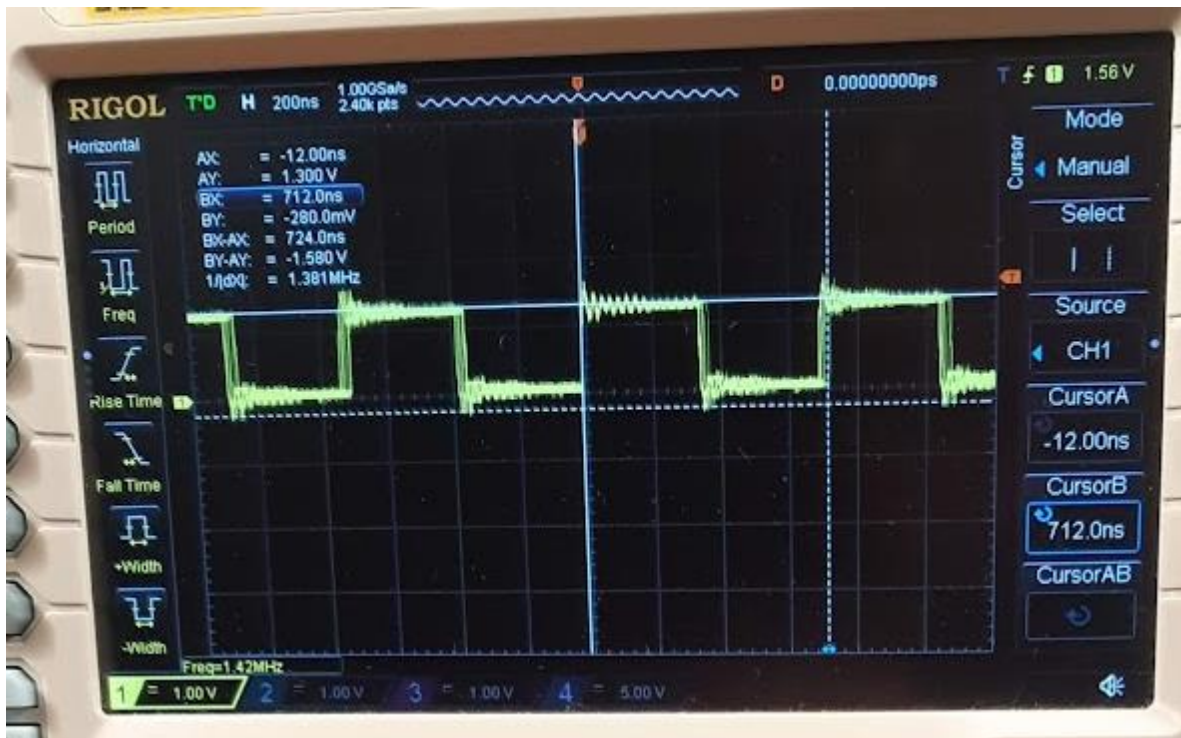
Conclusione: [I2S](#) è un protocollo di comunicazione seriale efficiente e diretto che è ottimo per l'audio digitalizzato. Tuttavia, non c'è alcuna legge che dica che sia limitato ai dati audio. [FONTE](#)

Segnale WORD SELECT



SEGNALE DATA IN





DAC [UDA1334](#)

- Risoluzione: Il DAC [UDA1334](#) supporta risoluzioni audio fino a 24 bit.
- Frequenza di campionamento: È in grado di gestire frequenze di campionamento fino a 192 kHz, consentendo la riproduzione di audio ad alta risoluzione.
- Rapporto segnale-rumore (SNR): Offre un alto rapporto segnale-rumore per una riproduzione audio pulita e fedele, tipicamente superiore a 100 dB.
- Distorsione armonica totale (THD): Garantisce bassa distorsione armonica totale per una qualità audio eccellente, solitamente inferiore allo 0,01%.
- Interfacce: Il DAC [UDA1334](#) dispone di interfacce digitali standard come I2S (Inter-IC Sound), che lo rendono compatibile con una vasta gamma di dispositivi digitali.
- Alimentazione: Può essere alimentato con una tensione di alimentazione tipica di 3,3 V, rendendolo adatto per applicazioni a bassa potenza.
- Consumo energetico: Grazie al suo design ottimizzato per il basso consumo energetico, il DAC [UDA1334](#) richiede una quantità minima di energia per le sue operazioni.
- Dimensioni e package: Le dimensioni e il package possono variare a seconda della versione specifica del chip e delle esigenze dell'applicazione, ma di solito sono progettati per essere compatibili con layout di circuito stampato standard e per l'integrazione in dispositivi audio di varie dimensioni.

Risoluzione: La risoluzione si riferisce alla precisione con cui il segnale audio può essere rappresentato digitalmente. Nel caso del DAC UDA1334, supporta risoluzioni fino a 24 bit. Questo significa che può convertire il segnale digitale in un'uscita analogica con una precisione fino a 2^{24} livelli distinti (di risoluzione verticale, pari a 16.777.216 scalini), il che permette una fedeltà sonora molto elevata.

Frequenza di campionamento: La frequenza di campionamento determina quanti campioni vengono usati per rappresentare un segnale audio analogico ogni secondo. Il DAC UDA1334 può gestire frequenze di campionamento fino a 192 kHz. Questo significa che può convertire segnali audio con una frequenza massima di 192.000 campioni al secondo.

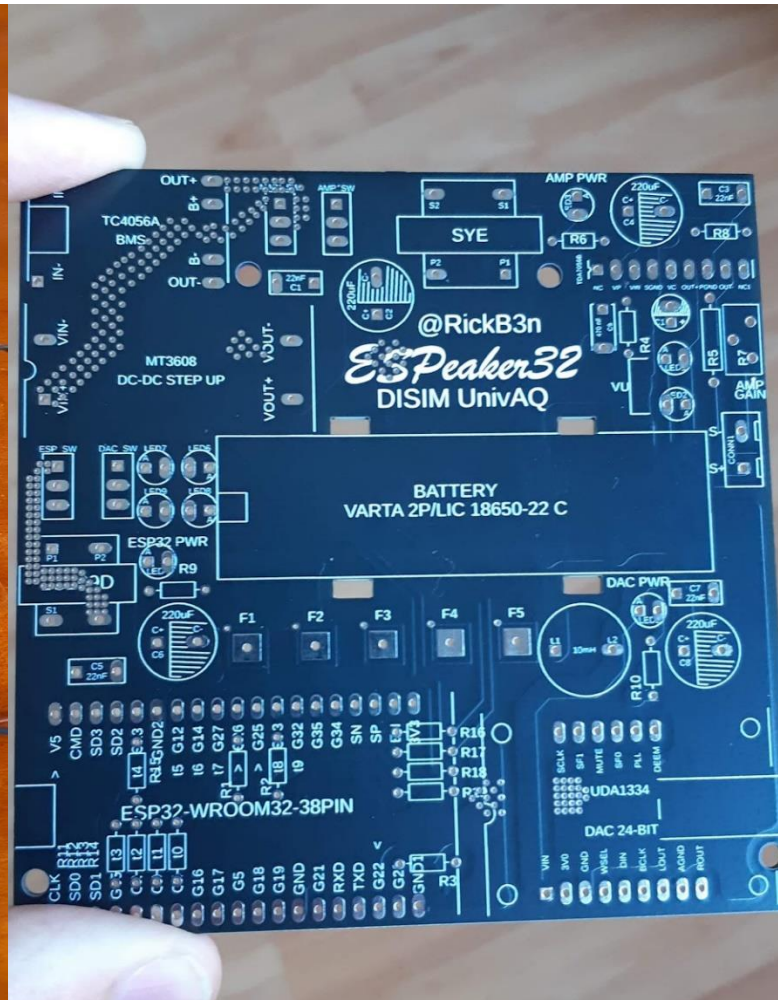
AMPLIFICATORE [TDA7056B](#)

Anche se molto inefficiente dal punto di vista energetico, ho scelto questo amplificatore poiché è un classe AB e quindi presenta delle buone caratteristiche di linearità e bassa distorsione, inoltre essendo in configurazione a ponte ossia con due amplificatori, si realizza un'unica uscita differenziale che permette di eliminare il condensatore di disaccoppiamento tra amplificatore e altoparlante.

Questo è un enorme vantaggio poiché non abbiamo problemi di sfasamenti indesiderati e fenomeni passa-alto, inoltre permette di pilotare molto meglio l'altoparlante.

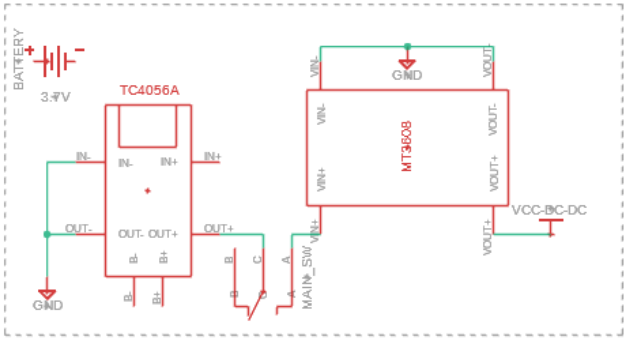
Il guadagno è regolato intorno a 15, ma può essere regolato tramite il segnale DC gain.

REALIZZAZIONE



SCHEMA ELETTRICO

POWER_MANAGMENT



Riccardo Beniamino

TITLE: ESPeaker32 Schematic v72

Document Number:

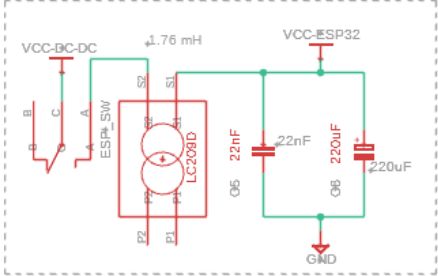
REV:

A

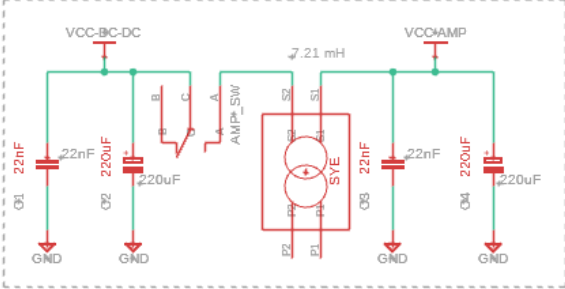
Date: 05/06/2024 18:16

Sheet: 1/1

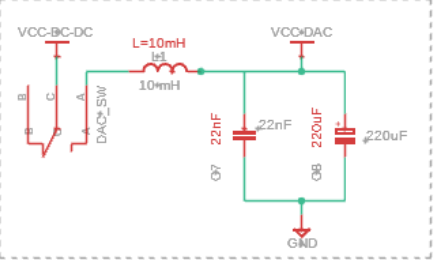
PI_FILTER_ESP32



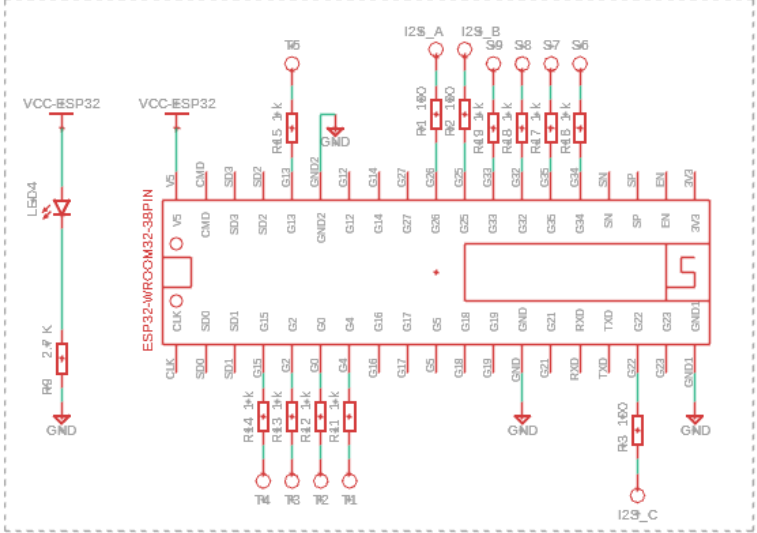
PI_FILTER_AMPLIFIER



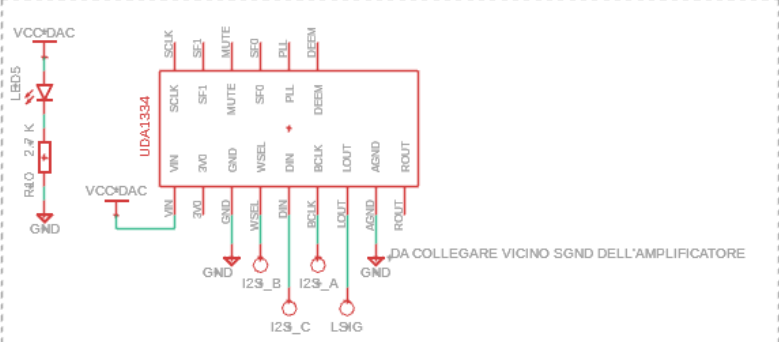
PI_FILTER_DAC



ESP32_MCU

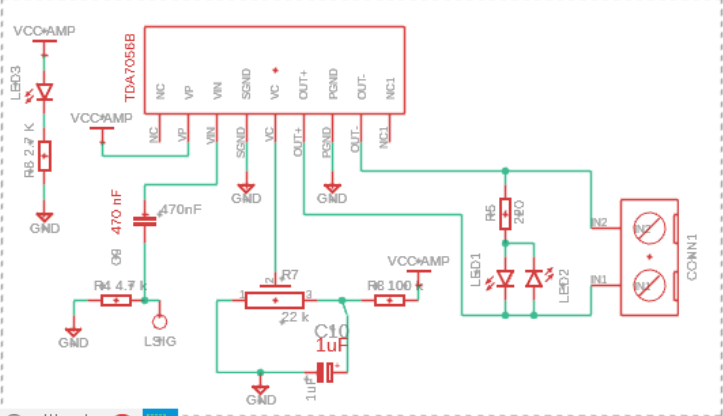


DAC_24_BIT



DA COLLEGARE VICINO SGND DELL'AMPLIFICATORE

AUDIO_AMPLIFIER



USER_INTERACTION

