

PROGETTAZIONE E REALIZZAZIONE DI UN TERMOMETRO DIGITALE

Matteo Zauli

Classe 5^aA Corso Elettronica e Telecomunicazioni

Istituto Tecnico Industriale "Nullo Baldini" Ravenna

Anno Scolastico 2007/2008

INTRODUZIONE

Il progetto che vi sto per illustrare è la realizzazione di un termometro digitale a microcontrollore con display a cristalli liquidi. La mia scelta è stata indirizzata dal voler mettere in pratica vari argomenti di diverse materie, dimostrando che l'istruzione degli ultimi tre anni di specializzazione mi ha reso in grado di progettare e sviluppare un progetto interamente sviluppato da me, con il massimo livello di autonomia.

Questo progetto nacque come approfondimento estivo del quarto anno dopo aver introdotto i dispositivi ADC (convertitori analogici – digitali) con un gruppetto di studenti della mia classe e portato avanti in particolar modo da me. La sua realizzazione si basa su uno degli argomenti principali del corso di quinta elettronica: l'acquisizione e la elaborazione dati.

Il termometro digitale realizzato è caratterizzato da una elevata precisione e velocità grazie a cui trova applicazione sia in campo domestico dove si potrebbe tenere sotto controllo la temperatura di un ambiente, sia in campo industriale dove potrebbe rilevare la temperatura di un magazzino o di un processo produttivo con notevole accuratezza. In una industria potrebbe anche costituire l'elemento di un controllo automatico occupandosi del comando di vari dispositivi in base alle rilevazioni effettuate.

PREFAZIONE

Inizialmente il progetto era costituito da un circuito molto semplice con soli due componenti: una sonda di temperatura e un convertitore ADC ad 8 bit, che era collegato attraverso una porta parallela a un computer, dove un programma che avevo scritto in linguaggio Assembler si occupava di acquisire e visualizzare in schermata DOS la temperatura rilevata.

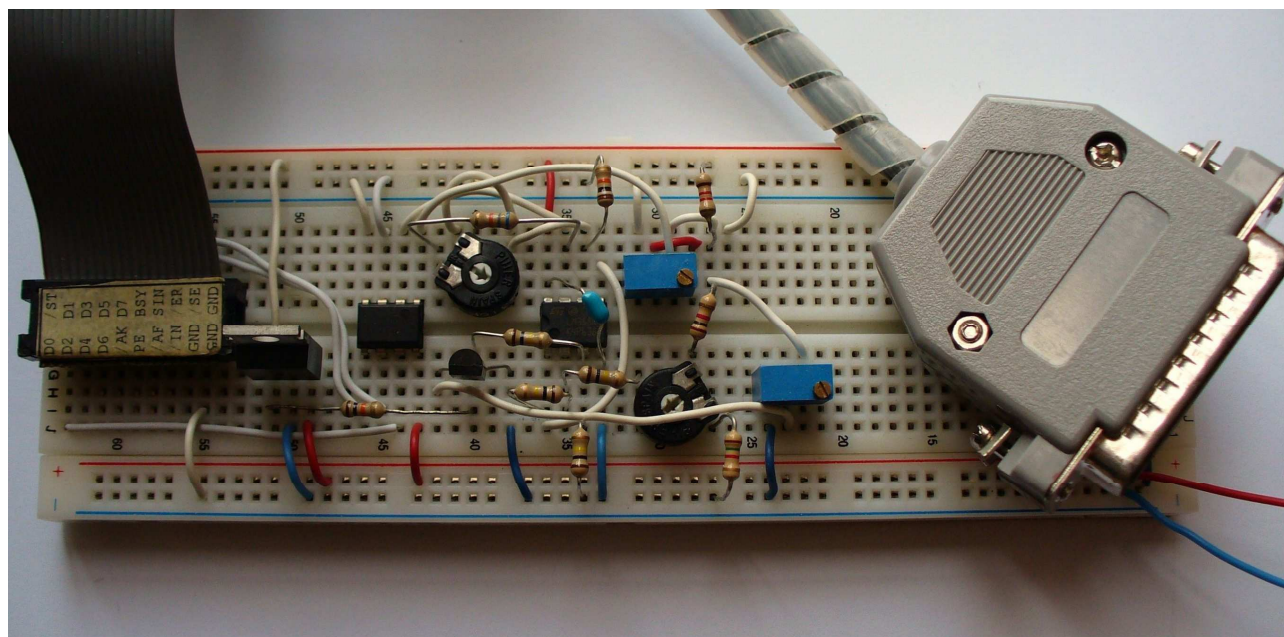


Fig1 Progetto termometro dopo i primi sviluppi

Questa prima realizzazione aveva un problema di stabilità nella rilevazione della temperatura, che si riscontrava nel momento in cui essa veniva visualizzata sul monitor del

computer adibito alla sua elaborazione. La temperatura visualizzata subiva delle variazioni oltre al grado centigrado e il malfunzionamento era causato da alcuni limiti di costruzione del dispositivo ADC (TLC548) riguardanti l'impostazione delle tensioni di riferimento del convertitore e da una bassa definizione della conversione a soli 8 bit. Questo era solo un primo prototipo. Successivamente, con l'inizio del quinto anno scolastico, in classe abbiamo incominciato a trattare nuovi argomenti come il microcontrollore e i display LCD (liquid crystal display). Con questi nuovi elementi a disposizione sono stato in grado di realizzare un progetto completamente rivoluzionato: la parte di acquisizione è stata assegnata ad un nuovo ADC (TLC1543), nella elaborazione il computer è stato sostituito da un microcontrollore della Atmel (AT89C51) e per la visualizzazione della temperatura è stato impiegato un display LCD (16x2 HD44780). Nella acquisizione i miglioramenti sono notevoli grazie all'utilizzo del nuovo ADC, che ci mette a disposizione nuove caratteristiche: una risoluzione più elevata (10 bit) e non abbiamo più un solo ingresso per l'acquisizione come nel convertitore precedente ma undici, grazie ai quali in futuro tramite il solo aggiornamento del software potrebbero essere collegati nuovi elementi, attraverso i quali eseguire altre rilevazioni oltre la temperatura, come ad esempio l'umidità o la pressione attraverso i relativi sensori. Con l'introduzione del microcontrollore ho reso più indipendente e flessibile il progetto in modo da non dover sottostare alla presenza di un computer, che oltretutto sarebbe dovuto rimanere costantemente in funzione. Non avendo più a disposizione un computer e quindi il relativo monitor la visualizzazione del risultato della elaborazione ora è assegnata ad un display LCD, componente dalle dimensioni ridotte, economico e dai consumi contenuti, che ci offre due righe di testo con ognuna sedici caratteri, sufficienti a visualizzare le informazioni che ci interessano. L'unico componente mantenuto dal progetto iniziale è la sonda di temperatura (LM335) non avendo trovato complessità nel suo utilizzo, ma solo una difficoltà di interfacciamento dovuta a un problema del ADC non della sonda e risolto con un piccolo circuito composto da operazionali di cui parlerò in maniera più approfondita in seguito. Vi ho illustrato i componenti principali che formano il mio termometro digitale, proseguendo nella tesina troverete maggiori informazioni sul loro utilizzo e l'intero sviluppo sull'interfacciamento tra essi.

Da diversi anni le classi quinte del corso di elettronica e telecomunicazioni stanno portato avanti un impegnativo progetto: realizzare una centralina meteorologica. All'inizio dell'anno scolastico insieme alla mia classe sono stato impiegato nel progetto e gli schemi dei vari moduli erano già definiti, alcuni erano stati costruiti, mentre altri dovevano ancora essere realizzati. Il progetto che più ha attirato la mia attenzione è stato quello riguardante il modulo di acquisizione, nel quale ho trovato delle possibili migliorie da apportare sfruttando alcuni elementi del termometro digitale che stavo già sviluppando. La scheda di acquisizione era adibita unicamente alla rilevazione di due temperature tramite due ADC singolo canale, 8 bit, mentre le altre grandezze fisiche (pressione, umidità, velocità vento, ecc..), di cui la centralina meteo avrebbe dovuto occuparsi non erano ancora state definite nel metodo di acquisizione. A questo punto proposi di utilizzare lo stesso ADC del progetto della mia tesina, con cui potei progettare insieme al professor Maurizio Montanari una nuova scheda, che si sarebbe occupata di tutti i rilevamenti necessari con semplicità e maggior risoluzione nelle conversioni analogiche digitali. Attualmente la realizzazione del nuovo modulo è stata portata a termine, ma non esegue tutte le misure che la centralina meteorologica deve effettuare come precedentemente ideato, ma questo compito è stato ripartito con un'altra scheda molto simile, la quale ha subito solo alcune modifiche per l'interfacciamento con alcune sonde che necessitano di accorgimenti particolari.

1.2 TLC1543 convertitore analogico-digitale (ADC) ad approssimazioni successive

I convertitori analogici-digitali, indicati anche come ADC (Analog to Digital Converter), sono dispositivi che convertono un segnale analogico in uno digitale. Essi compiono essenzialmente due operazioni: la quantizzazione del segnale analogico e la codifica dei valori ottenuti. Esistono vari tipi di ADC che si distinguono principalmente per la tecnologia di costruzione, tempi di conversione e risoluzione.

Nel progetto utilizzo un ADC ad approssimazioni successive della Texas Instruments: il TLC1543. Questo integrato ha come caratteristiche principali una risoluzione a 10 bit, 11 ingressi, un errore di 1 LSB e un circuito di Sample and Hold interno. La mia scelta in questo modello di convertitore è stata influenzata principalmente dalla risoluzione, che risulta elevata per la mia applicazione e dalla presenza di molti ingressi analogici, che potrebbero essere sfruttati per una espansione futura del progetto. In fig1.4 è illustrata la piedinatura del convertitore dove sono indicati i numerosi ingressi (da A0 a A10), i terminali di controllo e i pin sui quali fornire le tensioni di riferimento per la conversione (REF-, REF+).

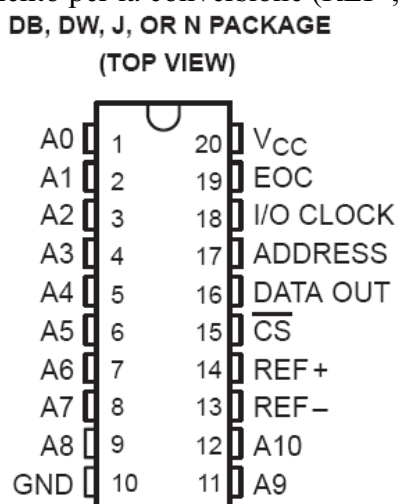


Fig1.4 ADC TLC1543

I convertitori analogici-digitali ad approssimazioni successive sono i più diffusi e sono caratterizzati da una elevata risoluzione e tempi di conversione bassi. Essi sono composti da un comparatore, un registro ad approssimazioni successive detto SAR (Successive Approximation Register), un convertitore digitale-analogico (DAC), un circuito di controllo e un buffer a latch.

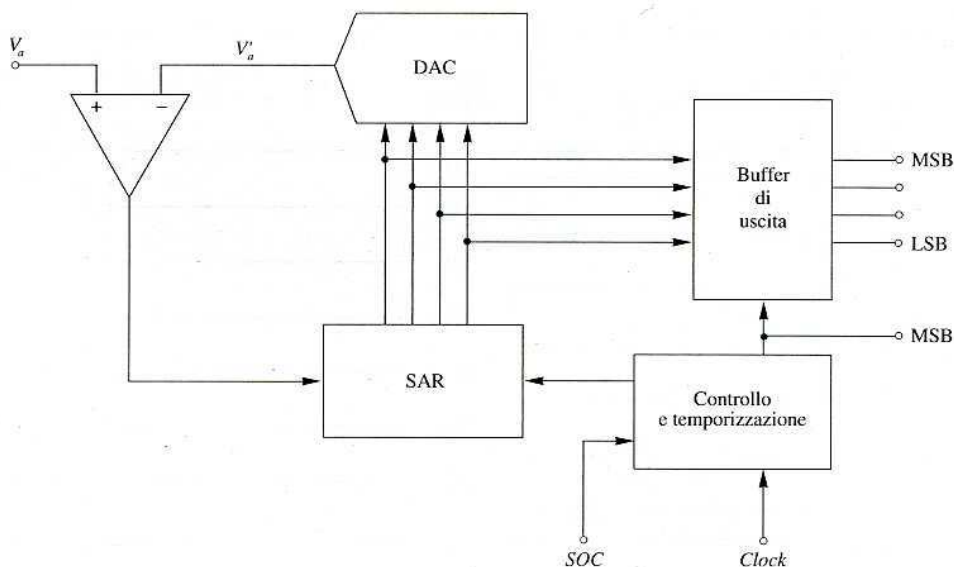


Fig1.5 Schema interno di un ADC ad approssimazioni successive

Il principio di funzionamento è il seguente. La tensione analogica da convertire è applicata all'ingresso non invertente del comparatore, la conversione ha inizio quando l'impulso SC (start of convention) abilita il circuito di controllo, determinando l'inizio della conversione con l'inizializzazione del SAR. Per scandire il processo è utilizzato il clock al cui primo impulso, attraverso il blocco di controllo, azionerà il SAR che porterà il bit più significativo al livello uno. L'uscita del registro è collegata al DAC, che si occupa di convertire il dato binario proveniente da esso per applicare la tensione risultante all'ingresso invertente del comparatore. Il comparatore confronta quest'ultima tensione con quella in ingresso all'ADC e da in uscita un uno nel caso in cui la tensione proveniente dal DAC sia a un livello più basso della tensione da convertire, mentre da uno zero nel caso contrario. Questi due livelli incidono sul funzionamento del SAR: nel caso in cui gli venga trasmesso un uno il registro mantiene il bit settato, nel caso contrario azzerà il bit. Con gli impulsi di clock seguenti il SAR setta o azzerà i bit successivi secondo il procedimento precedentemente descritto per il primo bit. La conversione è ultimata quando anche il bit meno significativo è stato valutato così viene mandato l'impulso EOC (end of convention) al buffer, che pone in uscita il risultato della conversione. Il convertitore ha n bit (n risoluzione), il tempo necessario alla conversione è uguale ad n periodi di clock, quindi per il caso del TLC1543 avendo una risoluzione a dieci bit bisogna fornire dieci impulsi di clock.

1.3 LM358 doppio amplificatore operazionale

L'amplificatore operazionale è un amplificatore in continua ad elevato guadagno ed il suo nome deriva dal fatto che con esso è possibile realizzare circuiti elettronici in grado di effettuare operazioni matematiche: somma, sottrazione, derivata e integrale. È costituito da due ingressi: uno invertente indicato con il segno meno, uno non invertente indicato col segno più e da una uscita. I due ingressi vengono distinti per la loro funzione: se applico una tensione (positiva o negativa) all'ingresso non invertente in uscita ho una tensione con la stessa polarità, invece se applico una tensione (positiva o negativa) all'ingresso invertente in uscita ho una tensione con polarità opposta. Gli amplificatori operazionali sono contraddistinti da alcune caratteristiche che nel caso di un componente ideale sono: resistenza di ingresso infinita, resistenza di uscita nulla, guadagno ad anello aperto infinito e banda passante infinita. Queste caratteristiche lo rendono un perfetto amplificatore di tensione.

Per questo progetto ho scelto l'integrato LM358 che al suo interno include due amplificatori operazionali (come visibile in fig1.6). La mia scelta è stata orientata proprio da

questo, avendo bisogno di due operazionali invece di utilizzare due singoli integrati ho deciso di utilizzarne uno solo che ne integrasse due, così da rendere più compatta e semplice la circuiteria del termometro.

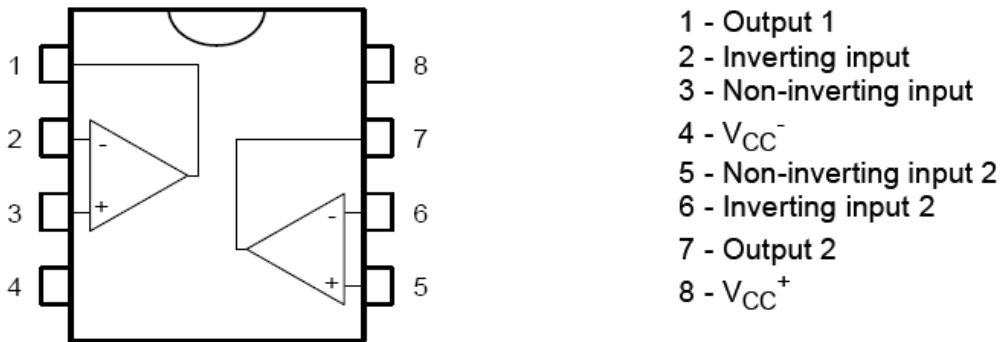


Fig1.6 Amplificatore operazionale LM358

1.4 LM336 diodo di riferimento di tensione

I diodi di riferimento di tensione sono dei componenti che vengono utilizzati per generare delle tensioni di riferimento con alta precisione e stabilità. Come visibile in fig1.7 questo tipo di diodo ha tre terminali: al due deve essere collegata l'alimentazione tramite una resistenza che garantisca al componente almeno 10 mA, al tre deve essere collegata la massa e al primo terminale per eseguire la calibratura del diodo deve essere collegato il centrale di un trimmer da 10 Kohm con gli altri due terminali collegati uno alla alimentazione e uno a massa.

In questo progetto è necessario inserire un diodo di riferimento di tensione per stabilizzare la tensione di riferimento positiva dell'ADC. LM336 compie questo compito in maniera perfetta ed è particolarmente indicato nell'utilizzo in un termometro, perché la sua particolarità più importante è quella di mantenere altamente stabile il suo funzionamento al variare della temperatura ambiente, in questo modo la tensione ai suoi capi rimane costante con precisione. Esistono vari tipi di LM336 contraddistinti dal valore stabilizzato di tensione che forniscono.

TO-92



1. Adj 2.+ 3. -

Fig1.7 LM336

1.5 AT89C51 microcontrollore Atmel

Un microcontrollore è un dispositivo che raggruppa il necessario per realizzare un sistema a microprocessore. I principali elementi che integra sono: una CPU, una memoria RAM, una memoria ROM, porte I/O, contatori, interfacciamento per porte di comunicazione (seriale, USB) e in alcuni casi convertitori analogici digitali. I microcontrollori sono utilizzati per controllare altri dispositivi elettronici o per realizzare dispositivi con funzioni specifiche. La circuiteria interna è suddivisa in tre moduli: uno dedicato alla elaborazione del programma, uno per l'interfacciamento con dispositivi esterni e uno per l'impiego dei segnali esterni. Questi dispositivi hanno una elevata flessibilità essendo componenti programmabili, che eseguono delle istruzioni determinate in base alla funzione da eseguire. Il linguaggio utilizzato per programmare questi componenti è l'Assembler ed esso è costituito da un set di istruzioni specifiche per ogni microcontrollore.

Il centro di controllo del termometro digitale che vi sto illustrando è basato su un microcontrollore della famiglia 8051 e si tratta dell'AT89C51 della Atmel. Le caratteristiche di

questo componente sono: CPU a 8 bit, 128 byte di memoria RAM, 4 kilobyte di memoria ROM, 32 porte I/O, due timer (contatori) a 16 bit, cinque interrupts (interruzioni), massima memoria esterna per programmi gestibile di 64 Kbyte e massima memoria esterna gestibile per dati di 64 Kbyte. Per questa applicazione abbiamo a disposizione un grande quantitativo di memoria interna RAM e ROM e le porte I/O sono numerose, queste due caratteristiche si riveleranno utili per espansioni future, così che ai nuovi progetti collegabili gli si possano dedicare un buon livello di risorse. La scelta nell'AT89C51 è stata influenzata anche dalla sua larga disponibilità e dalla presenza di diversi modelli compatibili tra di loro, che differiscono per alcune caratteristiche come la quantità di memoria dati e programmi e la velocità, così si potrebbero modificare le prestazioni del termometro cambiando il solo centro di elaborazione costituito dal microcontrollore.

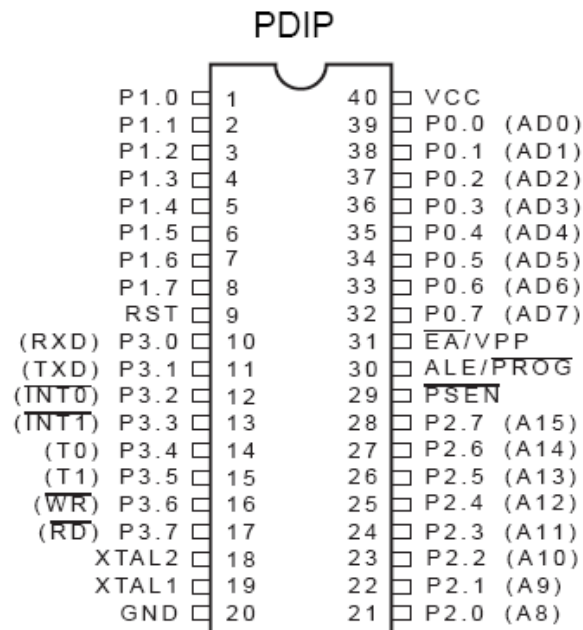


Fig1.8 Microcontrollore AT89C51

1.6 WM-C1602M display a cristalli liquidi

I display a cristalli liquidi, detti anche display LCD (liquid crystal display), sono utilizzati per generare caratteri alfanumerici e simboli. I display LCD non sono emettitori di luce e un'immagine è visibile solo se vi è una fonte luminosa e a seconda della sua provenienza si distinguono tre tipi di LCD: trasmissivo, riflessivo e transriflessivo. I display trasmissivi utilizzano la retroilluminazione, che consiste nel posizionamento di una luce sul retro dello schermo, sono utili in ambienti con scarsa illuminazione e presentano un aumento del consumo di corrente rispetto ai riflessivi a causa della fonte di luce artificiale che devono alimentare. I riflessivi utilizzano l'illuminazione ambientale, hanno consumi molto ridotti non dovendo alimentare nessuna fonte di luce, ma la loro leggibilità diminuisce con la luminosità esterna. Infine i transriflessivi combinano le due tipologie precedenti, sfruttando sia la retroilluminazione sia la luce ambientale si ha un display con buona leggibilità in tutte le condizioni di luce.

Alla base del principio di funzionamento degli LCD ci sono i cristalli liquidi, che sono delle sostanze con le caratteristiche dei solidi e dei liquidi e sono posizionate all'interno dei display tra due superfici vetrose con numerosi contatti elettrici (4), con i quali si può applicare un campo elettrico ai cristalli. Ogni contatto elettrico comanda una piccola porzione di pannello detta pixel.

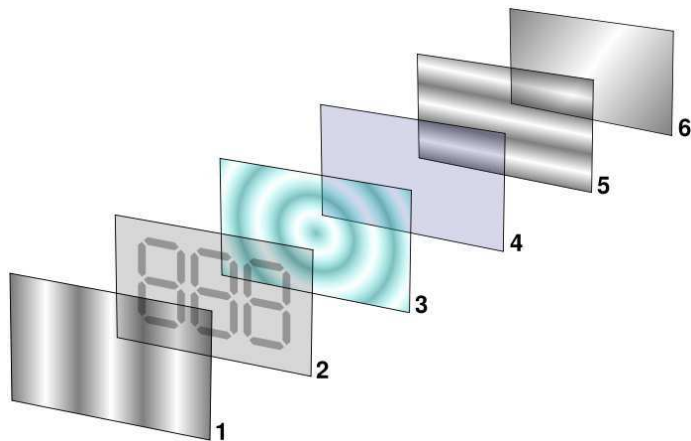


Fig1.9 Componenti di un display LCD

- 1 Polarizzazione verticale
- 2 Schermo di vetro con maschera delle zone scure
- 3 Strato con i cristalli liquidi
- 4 Strato di vetro con elettrodi
- 5 Polarizzazione orizzontale
- 6 Superficie riflettente

Prima dell'applicazione del campo elettrico la luce passa attraverso tutta la struttura (fig.1.9) venendo riflessa dalla superficie riflettente (6) alla base dello schermo, mentre quando si applica un campo elettrico si modifica l'orientamento delle molecole dei cristalli liquidi. La luce passa attraverso lo strato di polarizzazione verticale (1), le molecole si orientano verticalmente per il campo elettrico, quindi la luce viene polarizzata perpendicolarmente allo strato di polarizzazione orizzontale (5) così da essere bloccata facendo apparire il pixel spento. Controllando l'orientamento delle molecole dei cristalli liquidi si può controllare quanta luce far passare e con la combinazione dei vari pixel otteniamo la visualizzazione di un carattere.

In questo progetto utilizzo il display LCD WM-C1602M, a cui è affidata la visualizzazione della temperatura rilevata dal termometro. Questo schermo è di tipo 16x02, in pratica può visualizzare trentadue caratteri in due righe e fa parte degli schermi transriflessivi, quindi abbiamo a disposizione la retroilluminazione collegabile alla alimentazione opzionalmente, potendo utilizzare il display anche solo come riflessivo. Il motivo per cui ho scelto questo modello di display è la facilità di interfacciamento e di gestione con un microcontrollore e questo è reso possibile dal fatto che lo schermo integra un suo specifico microcontrollore, che in questo caso si tratta di un HD44780 della Hitachi attraverso il quale si esegue un controllo molto semplice. La scelta in questo tipo di display rende possibile la sua facile sostituzione dato che gli LCD provvisti di microcontrollore HD44780 sono i più diffusi, hanno tutti la stessa piedinatura e metodo di controllo, quindi ho una applicazione compatibile con una vasta gamma di schermi.

1.7 NE555 timer

I timer sono circuiti di temporizzazione in grado di generare precisi ritardi di tempo. Uno dei più diffusi è l'NE555 in grado di produrre temporizzazioni da alcuni microsecondi ad alcune decine di minuti e la sua applicazione più importante è come multivibratore monostabile o astabile. È composto essenzialmente da un partitore resistivo costituito da tre resistenze uguali da 5 Kohm, da due comparatori, da un latch RS, da una porta NOT e da un transistor (schema visibile in fig1.10). Il partitore resistivo fa in modo che la tensione all'ingresso invertente del primo comparatore (CP1), sia una tensione di $\frac{2}{3} V_{cc}$, mentre sull'ingresso non invertente del secondo comparatore (CP2), sia di $\frac{V_{cc}}{3}$. Quando la tensione di Trigger (TRG) scende al di sotto di $\frac{V_{cc}}{3}$, il comparatore CP2 va a livello alto, il latch ha in ingresso S=1 R=0, per cui l'uscita negata è a zero, il transistor è interdetto e l'uscita dell'NE555 è alta. Quando la tensione di Threshold (THR) supera i $\frac{2}{3} V_{cc}$, l'uscita di CP1 passa al livello alto, il latch ha in ingresso S=0 R=1, per cui l'uscita negata è a uno, il transistor è in conduzione e l'uscita dell'NE555 è bassa.

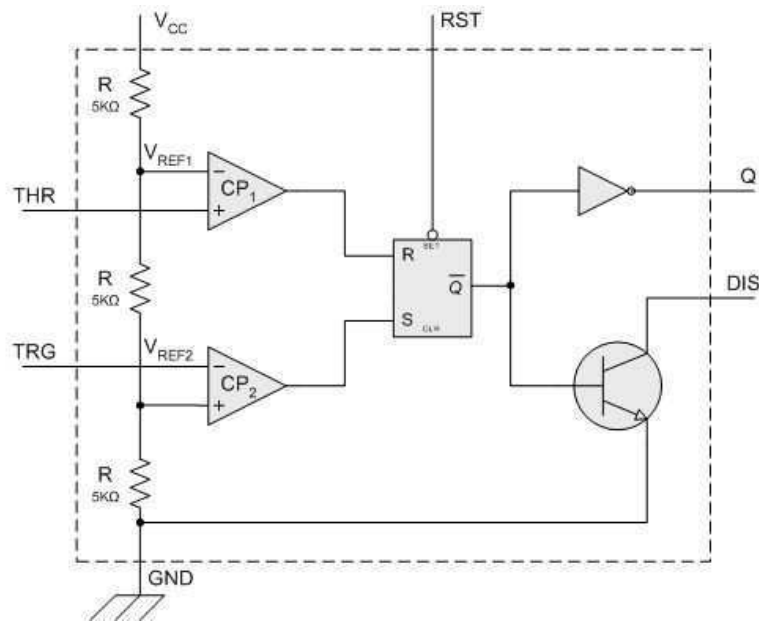


Fig1.10 Schema circuito interno NE555

L'NE555 è adoperato nel termometro per costituire un circuito di watch dog. In pratica il suo compito è quello di verificare che il programma del microcontrollore sia in esecuzione e non abbia blocchi. Per questo compito utilizzo un timer perché nel caso in cui il software si arresti l'NE555 genera un impulso di reset di una certa durata sul microcontrollore, in modo che esso possa ripartire con l'esecuzione del programma dal principio superando il blocco. Ho scelto di utilizzare un NE555 per la sua facile reperibilità e semplicità di utilizzo. Il circuito di watch dog realizzato è spiegato in maniera più approfondita nel capitolo due sul funzionamento del circuito del termometro.

1.8 78xx stabilizzatori di tensione monolitici

Gli stabilizzatori di tensione monolitici sono degli integrati che forniscono in uscita un livello di tensione fisso, oppure per alcuni tipi c'è la possibilità di regolare il livello di uscita di tensione. I più comuni hanno tre terminali (come visibile in fig1.11): il primo terminale è l'ingresso, a cui deve essere applicata una tensione maggiore di qualche volt del livello che deve essere stabilizzato per ottenere un buon funzionamento, il secondo deve essere collegato a massa e il terzo è l'uscita. Per ottenere un ottimo funzionamento da questi stabilizzatori bisogna inserire un condensatore sull'ingresso per stabilizzare il funzionamento del componente e uno sull'uscita per minimizzare il ripple. Con soli tre componenti grazie agli stabilizzatori monolitici si può realizzare un alimentatore stabilizzato di alta qualità.



Fig 1.11 Stabilizzatore monolitico

Proprio per l'estrema facilità nel realizzare un alimentatore con questi integrati ho utilizzato degli stabilizzatori monolitici della serie 78xx, in cui le ultime due cifre indicano il livello di tensione stabilizzato che forniscono in uscita. I 78xx sono in grado di erogare fino a un ampere di corrente e quindi sono appropriati nell'utilizzo del mio progetto, avendo un assorbimento molto inferiore, nell'ordine dei milliamperes. Sono provvisti di contenitore TO-220

quindi possono essere collegati a un dissipatore per la dissipazione della potenza del componente. Due caratteristiche molto importanti dei 78xx è la presenza di un limitatore di corrente che impedisce di superare il valore massimo di corrente fornito in uscita (nel caso dei 78xx 1.5A) e di una protezione termica contro i cortocircuito, la quale interviene quando la temperatura del chip raggiunge livelli pericolosi per la sua integrità e quindi interrompe l'erogazione di corrente. Nel progetto utilizzo un 7815 (15V) nell'alimentatore esterno, mentre montato sul termometro utilizzo un 7812 e un 7805 per stabilizzare ulteriormente la tensione proveniente dall'alimentatore a 12V e 5V necessarie al funzionamento del circuito.

2 FUNZIONAMENTO CIRCUITO

Il termometro digitale che vi sto illustrando è composto principalmente da un sistema di acquisizione e elaborazione dati costituito da un convertitore analogico digitale e da un microcontrollore. La circuiteria che circonda questi due componenti non è di minor rilevanza, ma è indispensabile al funzionamento del termometro e alla sua applicazione. La funzione globale del circuito è posta all'acquisizione di una rilevazione da parte di un trasduttore, alla elaborazione del dato ottenuto e infine alla sua visualizzazione. In questo caso il trasduttore è una sonda di temperatura, che è interfacciata da un circuito di adattamento con un ADC, il quale è gestito da un microcontrollore.

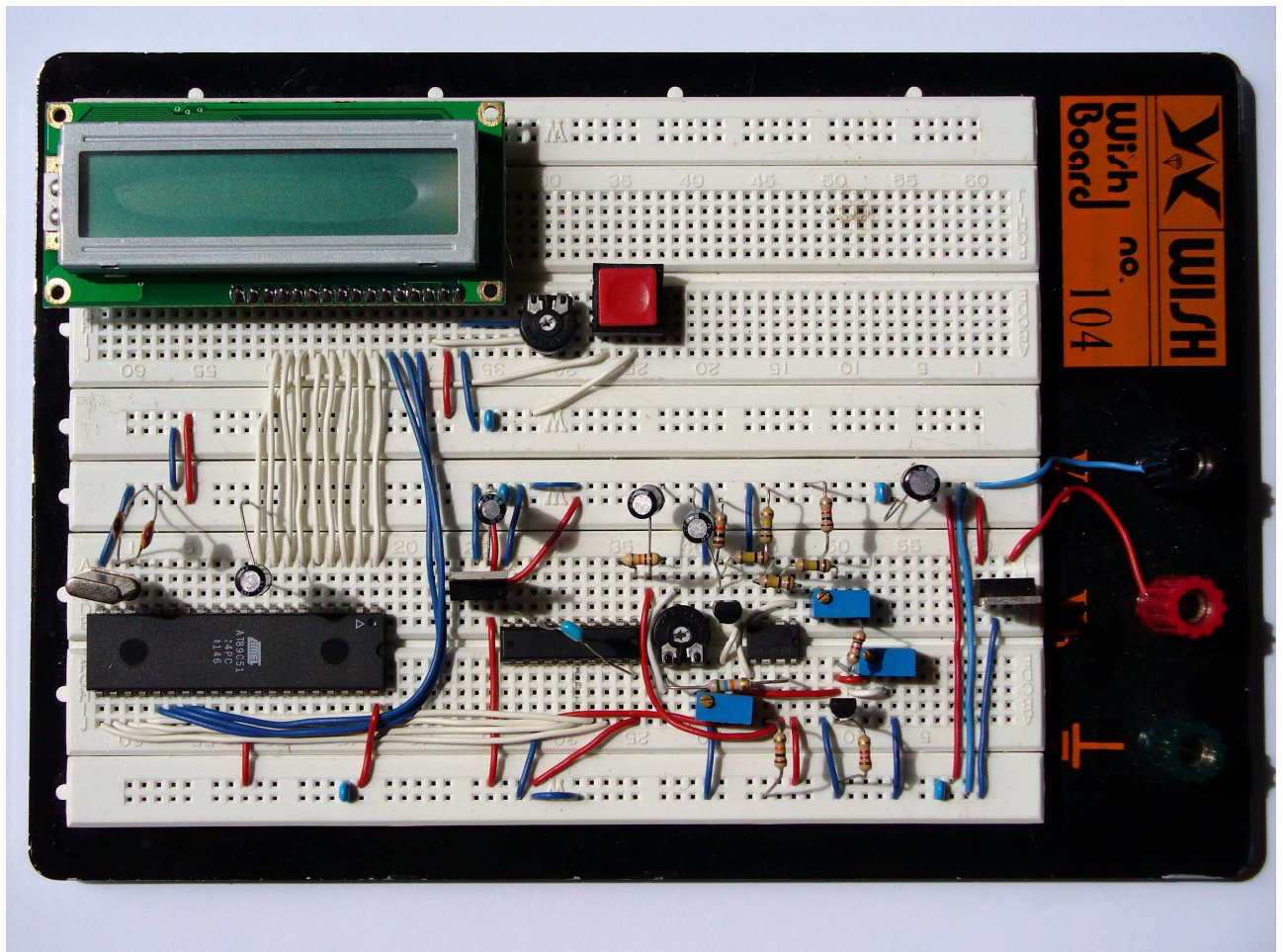


Fig2.1 Breadboard per prova termometro

Le prime prove sul funzionamento del termometro sono state svolte su di una breadboard (fig2.1), in modo da poter applicare liberamente variazioni per testarne e migliorarne le prestazioni.

La realizzazione del circuito stampato del termometro è stata suddivisa in tre parti: scheda principale, scheda di visualizzazione e alimentatore. La scheda principale racchiude tutto il sistema di funzionamento del termometro, quella di visualizzazione comprende solo la disposizione per il montaggio del display LCD e l'alimentazione è data da un alimentatore stabilizzato esterno. La scheda principale e quella di visualizzazione sono collegate tra di loro tramite due connettori, che permettono ai due moduli di essere montati orizzontalmente uno sopra l'altro, montaggio detto a "sandwich".

Le linee di gestione del display sono portate alla scheda di visualizzazione tramite i connettori, che sono collegati anche alle porte libere del microcontrollore e gli ingressi liberi dell'ADC. La disponibilità di queste linee inutilizzate e i tipi di connettori usati rende semplice l'espansione del progetto con l'inserimento di nuovi moduli, che andrebbero ad essere montati tra la scheda principale e quella di visualizzazione, con alcuni semplici accorgimenti nella disposizione dei connettori sul circuito stampato.

Per fornire una descrizione dettagliata del funzionamento del termometro passo a descrivere ogni singola parte del circuito, in modo che alla fine si possa avere una visione complessiva su come è stato ideato il progetto. Le figure sono tratte dagli schemi elettrici disegnati che sono allegati a questa tesina.

2.1 Descrizione circuiteria

2.1.1 Scheda principale

Sonda

La sonda di temperatura (LM335) utilizzata necessita di un piccolo circuito per la sua alimentazione e calibratura (fig2.1).

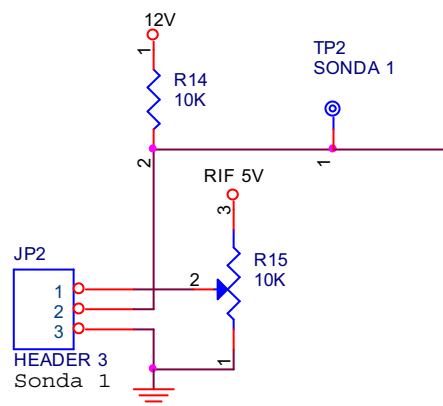


Fig2.2 Circuito per sonda LM335

La sonda LM335 è collegata al termometro tramite un connettore a 3 pin (header 3), a cui il trasduttore deve essere montato con il taglio del contenitore verso l'alto, in modo che il primo terminale (adjuster) sia collegato al primo pin del connettore, il secondo alla alimentazione e il terzo a massa. La resistenza R14 di 10Kohm è stata scelta per alimentare la sonda con circa 1mA, sufficiente al suo funzionamento. Il trimmer R15 come indicato dal datasheet del componente è di 10Kohm ed è utilizzato per eseguire la calibratura della sonda, potendo modificare il valore di tensione in uscita dal trasduttore variando il trimmer. Il test pointer (TP2) è utilizzato per misurare la tensione di uscita dalla sonda ed è utile durante la calibratura.

Riferimento di tensione

Per fornire un riferimento di tensione molto stabile all'ADC utilizzo il diodo di riferimento di tensione LM336, che mi fornisce 5V e per il suo funzionamento necessita di un circuito di alimentazione e calibratura.

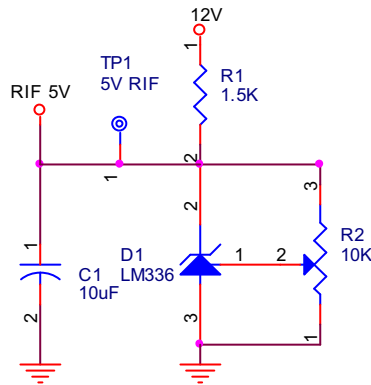


Fig2.3 Circuito per LM336

Il diodo di riferimento di tensione è alimentato tramite la resistenza R1 di 1.5Kohm e la calibratura del componente è effettuata tramite il trimmer R2, con il quale regoliamo una tensione di uscita di 5V, misurabili tramite il test pointer (TP1). Il condensatore C1 procura un'ulteriore stabilità al riferimento di tensione. La particolarità di questa tensione di mantenere una estrema stabilità e precisione la rende adatta nell'utilizzo di impostazioni di altre tensioni come nel partitore del circuito successivo sull'adattatore sonda ADC.

Adattatore sonda ADC

Per un corretto funzionamento del convertitore analogico digitale, come indicato dal datasheet bisogna fornire due riferimenti di tensione, che abbiano un divario di almeno 1V e proprio da questo limite nasce la necessità di un circuito di adattamento tra sonda e ADC. Con questo termometro misuriamo una temperatura che va dai -10°C ai 50°C e con la caratteristica della sonda di variare la tensione di uscita di 10mV per grado, con una tensione a 0°C di 2.73V, abbiamo in uscita dal trasduttore una tensione che può variare dai 2.63V ai 3.23V. Quindi all'ADC avrei dovuto fornire un riferimento minimo di 2.63V e un riferimento massimo di 3.23V con un divario fra i due di 600mV, valore insufficiente per il corretto funzionamento del convertitore. Per ovviare da questo problema ho realizzato un circuito di adattamento, che mi fa variare la tensione proveniente dalla sonda da 0V (-10°C) a 5V (50°C), così da dover fornire al convertitore analogico digitale questi due riferimenti, che mi garantiscono una corretta conversione.

L'adattatore è formato da due operazionali: il primo è configurato come differenziale, il secondo come amplificatore non invertente. L'ingresso invertente del differenziale è collegato a un partitore di tensione a cui tramite R3 arrivano i 5V stabilizzati dal LM336 e con il trimmer R4 si regola la tensione minima che la sonda deve raggiungere e che in questo caso è di 2.63V corrispondente a -10°C . All'ingresso non invertente del differenziale è collegata la sonda di temperatura tramite la resistenza R8, così l'operazionale può eseguire la differenza fra la tensione regolata tramite il trimmer e quella proveniente dal trasduttore. Il risultato viene amplificato dall'amplificatore non invertente con un guadagno corrispondente al rapporto tra la massima variazione di cui si vuole disporre (5V) e quella effettiva della sonda (600mV), quindi il guadagno è di 8.33 ($5\text{V}/600\text{mV}=8.33$) ed è impostabile tramite il trimmer R11.

Ad esempio se la sonda rileva 50°C , quindi ha in uscita una tensione di 3.23V, a questa si sottraggono i 2.63V impostati con il partitore sull'ingresso invertente dell'operazionale. A questo punto abbiamo in uscita dal differenziale 600mV, che saranno amplificati dall'amplificatore non invertente, ottenendo ($600\text{mV}\times 8.33=5\text{V}$) il livello di tensione massimo corrispondente alla massima temperatura rilevabile.

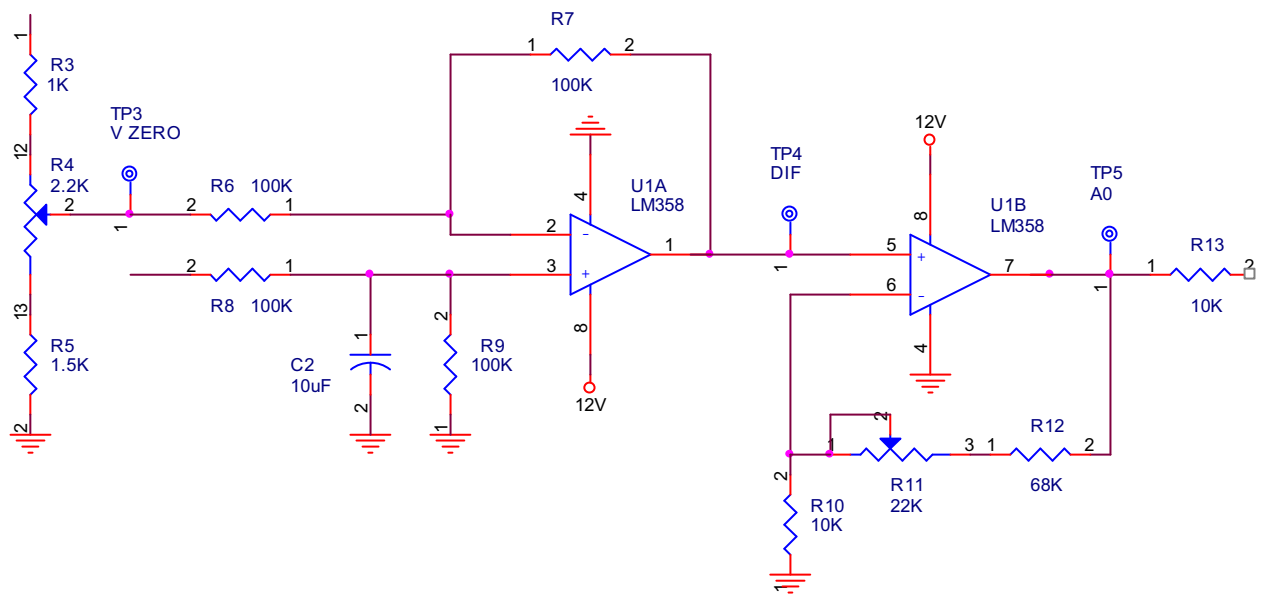


Fig2.4 Circuito di adattamento sonda ADC

Le resistenze R3, R5 con il trimmer R4 formano il partitore di tensione sull'ingresso invertente dell'operazionale. Le resistenze R6, R8, R7 e R9 costituiscono un differenziale a guadagno unitario. Le resistenze R10, R12 e il trimmer R12 formano l'amplificatore non invertente con guadagno definito precedentemente. Il condensatore C2 stabilizza e filtra eventuali disturbi provenienti dalla sonda di temperatura. La resistenza R13 di 10Kohm è a protezione dell'ingresso dell'ADC salvaguardandolo da tensioni maggiori di 5V, che potrebbero risultare da un malfunzionamento del circuito di adattamento. I tre test pointer utilizzati mi permettono di regolare il circuito e di testarne il funzionamento: TP3 tensione minima raggiungibile dalla sonda, TP4 funzionamento differenziale e TP5 regolazione guadagno amplificatore non invertente.

ADC e microcontrollore

L'ADC è gestito interamente dal microcontrollore, il quale mette a disposizione una parte della porta P2 per fornire i vari segnali di controllo: clock (CLK), chip select (CS), impostazione indirizzo ingresso di conversione (ADIN) e lettura dato della conversione (DOOUT). Il resto della porta P2 si occupa della gestione del display, per il quale è utilizzata anche la porta P1, che si occupa di mandare i dati da visualizzare allo schermo. Le porte P0, P3 e gli ingressi del convertitore da A1 ad A10 sono tutte linee libere e le ho collegate a dei connettori per poterle utilizzare su altri moduli aggiungibili. L'unico ingresso utilizzato dell'ADC è A0 ed è collegato alla sonda di temperatura tramite il circuito di adattamento precedentemente descritto.

Il microcontrollore per funzionare ha bisogno di un clock che è generato da un oscillatore interno, a cui deve essere solamente collegato un quarzo, che può essere di varie frequenze fino ad un massimo indicato dal datasheet, in questo caso per l'AT89C51 è di 24Mhz, mentre io ho scelto il valore più utilizzato di 11Mhz. I due condensatori C18 e C19 completano la parte dell'oscillatore esterno.

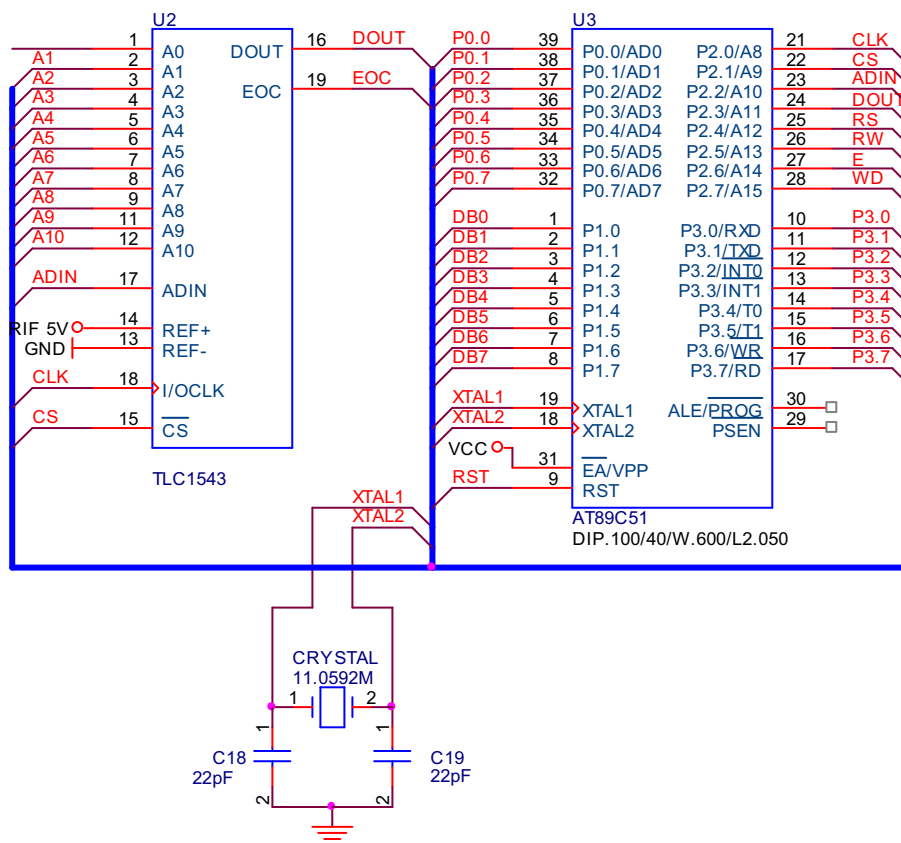


Fig2.5 Circuito interfacciamento ADC microcontrollore

Connettori

In fig2.5 sono visibili i due connettori a 30 pin utilizzati per montare la scheda principale e la scheda di visualizzazione una sopra l'altra. Si possono notare le linee di gestione del display sui primi undici terminali del connettore JP4, come sono state collegate le numerose linee libere (P0, P3, A1-A10) e come sono state disposte le alimentazioni alla base di JP3.

Alla base del voler aggiungere questi due connettori c'è la voglia di voler rendere questo progetto espandibile tramite nuovi moduli collegabili ad essi e a cui rendere disponibili le varie linee non utilizzate dell'ADC e del microcontrollore. Ho optato per due connettori da trenta terminali con un totale di sessanta connessioni, anche se le linee che utilizzo sono in numero minore, per dare maggiore flessibilità al progetto, nel caso in cui si vogliono aggiungere dei moduli che vogliono comunicare tra di loro ci sono venti collegamenti disponibili.

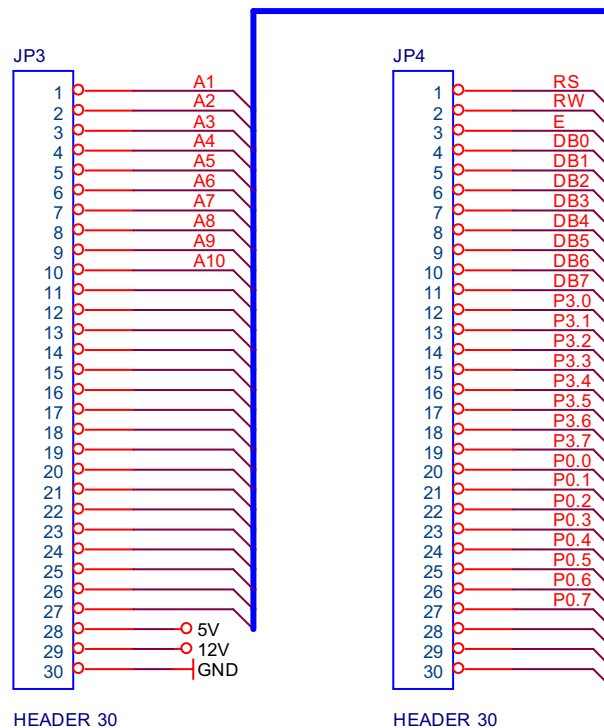


Fig2.6 Connettori

Circuito di reset e watch dog

Il circuito descritto in questo paragrafo si occupa di fornire il reset al microcontrollore tramite due modi. Il primo e più semplice consiste nell'utilizzo di un pulsante (SW1) collegato a Vcc, che al momento della sua pressione da un impulso di reset. Il condensatore C20 è utilizzato come antirimbalzo e si scarica tramite la resistenza di pull down interna al terminale di reset del microcontrollore. Questo sistema è un metodo manuale ed è utile durante le prime prove del software del termometro. Il secondo metodo è quello definitivo, che ha un funzionamento autonomo ed è utilizzato al termine della programmazione per evitare blocchi del software in maniera automatica senza alcun intervento manuale. Questo metodo utilizza un circuito detto di watch dog, il quale è collegato a una linea del microcontrollore, che viene continuamente complementata tramite il software in modo che il condensatore C16 sia mantenuto scarico, essendo cortocircuitato a massa dal transistor Q2 (BC547) mantenuto in conduzione dai picchi positivi generati dal derivatore (C17, C21) per la complementazione della linea WD. Il diodo D3 (1N4148) elimina i picchi negativi generati dal derivatore. Fino a quando il condensatore C16 è scarico secondo il funzionamento del NE555 (capitolo 1.7) la tensione di Trigger (pin 2 NE555) del componente è mantenuta al di sotto di $V_{cc}/3$ e quindi l'uscita è al livello alto. L'uscita è collegata tramite la resistenza R18 al transistor Q1 (BC328), il quale quando ha un livello alto in base è interdetto e quindi l'impulso di reset è mantenuto basso dalla resistenza di pull down R20, mentre quando ha in base un livello basso è in conduzione e quindi il terminale di reset viene collegato a Vcc, causando il reset del microcontrollore. Per ottenere il livello basso in uscita dal NE555 la linea del microcontrollore non deve essere complementata, così il transistor Q2 è interdetto non ricevendo più impulsi dal derivatore, il condensatore C16 si carica e quindi la tensione di Threshold (pin 6 NE555) è maggiore di $2/3 V_{cc}$ e l'uscita del timer è al livello basso. Il fatto che la linea del watch dog (WD) non venga più complementata dal programma vuol dire che esso si trova in una situazione di blocco e quindi è necessario eseguire un reset come appunto generato in automatico dal circuito descritto. Il connettore J3 è un selezionatore con un jumper, che mi permette di scegliere se collegare alla linea di reset del microcontrollore il pulsante (SW1) o il circuito di watch dog.

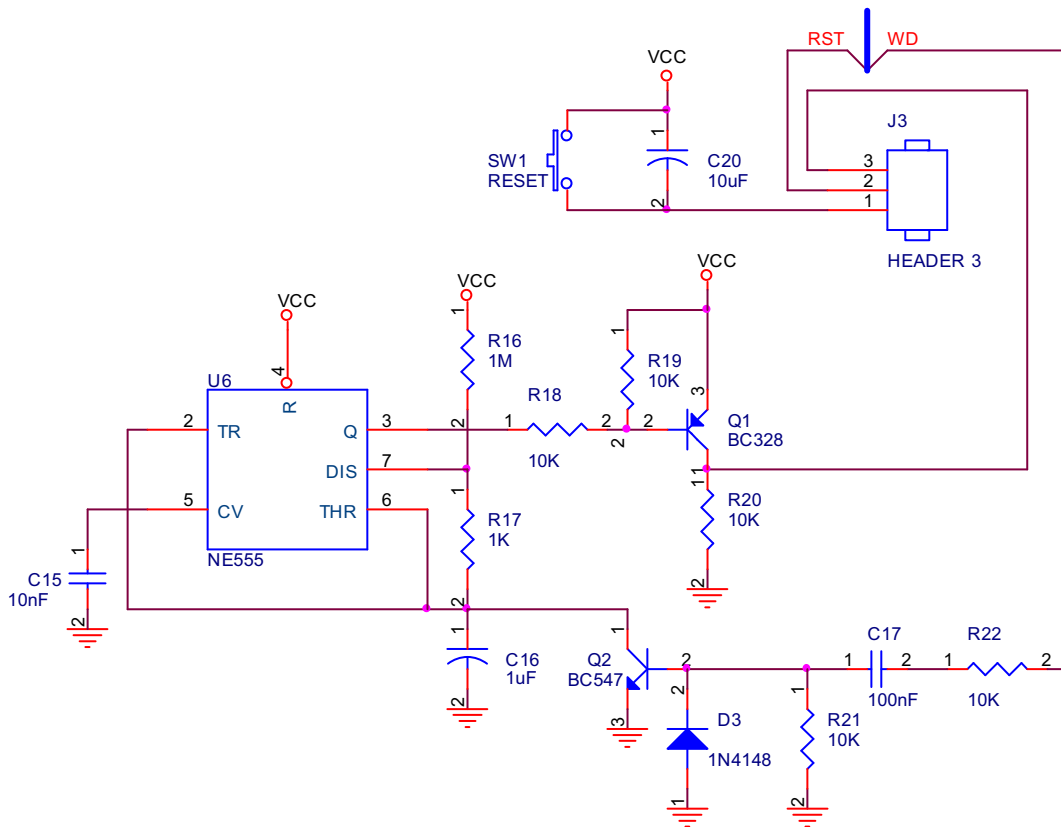


Fig2.7 Circuito di reset e watch dog

Alimentatore

Il termometro deve essere alimentato a 12V e 5V per le diverse alimentazioni degli integrati.

Per non realizzare un alimentatore esterno con un connettore con tre contatti invece di uno comune a due ho scelto di integrare sulla scheda principale un circuito di alimentazione, per ottenere le due diverse tensioni direttamente sul circuito del termometro.

Il circuito di alimentazione è realizzato con due stabilizzatori della serie 78xx: un 7812 per ottenere i 12V e un 7805 per ottenere i 5V. I condensatori sono necessari per la stabilità del funzionamento degli stabilizzatori monolitici e il diodo D2 protegge l'intero termometro da una inversione di polarità, che potrebbe verificarsi per un errato collegamento dell'alimentatore esterno al connettore JP1, a cui devono essere forniti almeno 15V per il buon funzionamento degli stabilizzatori, ai quali deve essere fornita una tensione maggiore di circa due volt la tensione stabilizzata per garantire il loro corretto funzionamento.

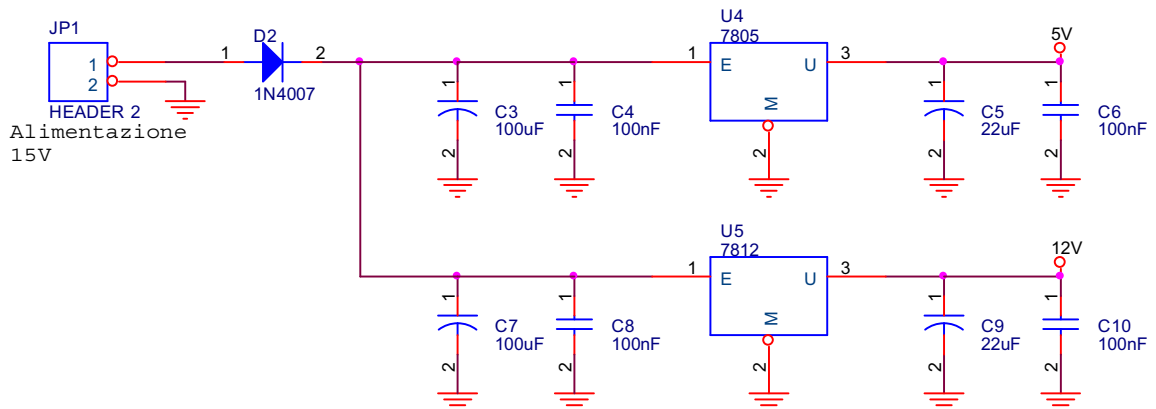


Fig2.8 Circuito di alimentazione

Parlando sempre di alimentazione per gli integrati in tecnologia CMOS è necessario inserire il più vicino possibile ad ognuno di essi dei condensatori da 100nF (ceramici o poliestere) tra l'alimentazione e massa per eliminare i picchi di tensione, comuni in questi componenti durante la loro commutazione.

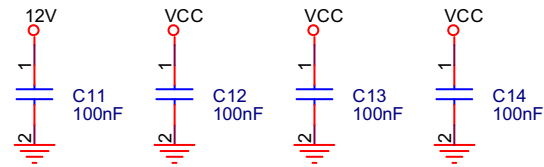


Fig2.9 Condensatori per eliminare i picchi di tensione

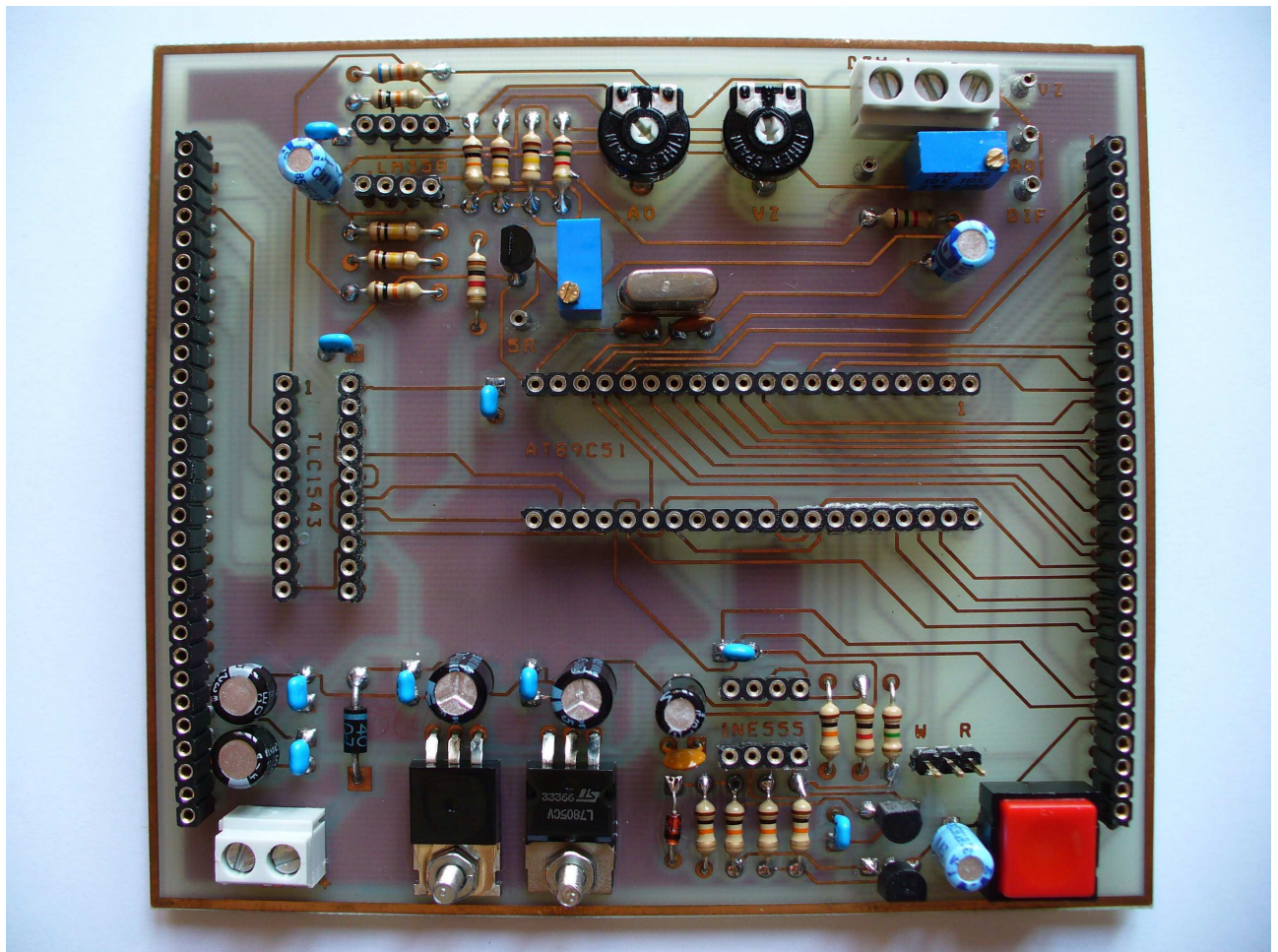


Fig2.10 Scheda principale ultimata

2.1.2 Scheda di visualizzazione

La scheda di visualizzazione è molto semplice siccome il display ha già integrato sulla sua struttura tutta la circuiteria necessaria al suo interfacciamento con il microcontrollore.

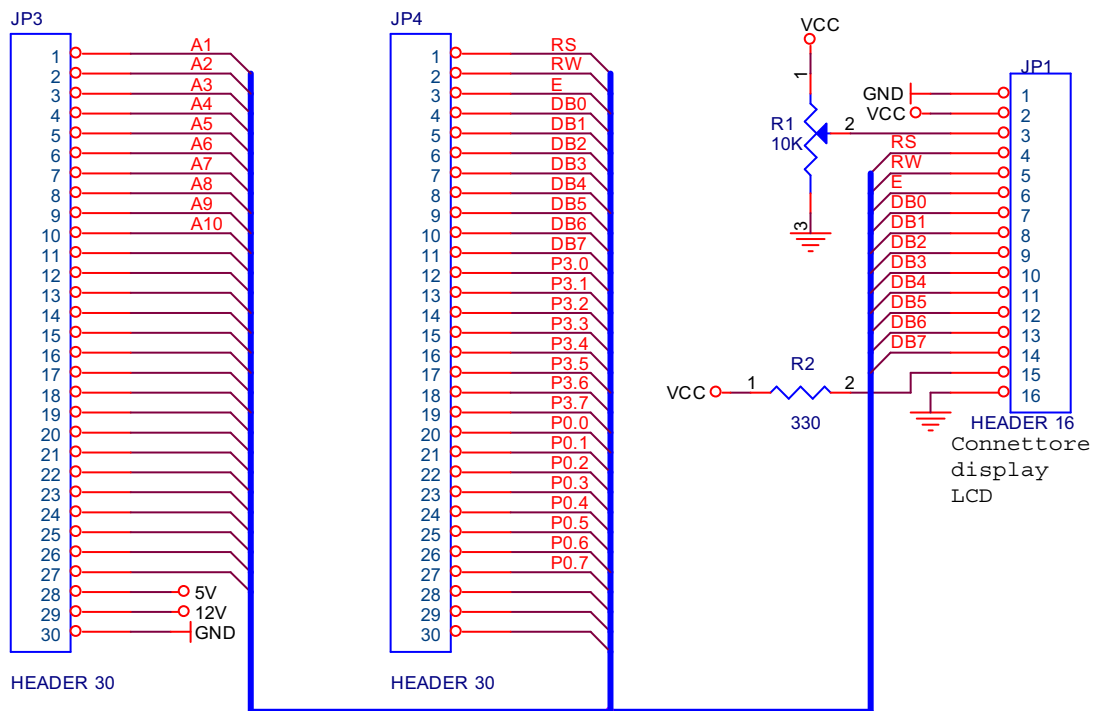


Fig2.11 Scheda di visualizzazione

Le uniche cose di cui necessita sono: il trimmer R1 per la regolazione del contrasto dello schermo e la resistenza R2 per l'alimentazione della retroilluminazione.

Al connettore JP1 (header 16) oltre a poter collegare il display che ho scelto è possibile utilizzare un qualsiasi schermo, che monti un microcontrollore HD44780, avendo tutti lo stesso metodo di gestione e piedinatura, come precedentemente detto nel capitolo sulla descrizione dell'hardware.

In fig2.8 sono visibili anche i connettori JP3 e JP4 per il collegamento alla scheda principale, dalla quale provengono i segnali di controllo del display.

2.1.3 Alimentatore

Con l'utilizzo di un 7815 e di un trasformatore da 1.5VA ho realizzato un alimentatore stabilizzato da 15V che fornisce fino a 80mA di corrente.

Il trasformatore usato ha un doppio secondario da 15V AC che ho collegato in parallelo per avere più corrente e questa tensione alternata è raddrizzata da un ponte di Graetz. I condensatori C1 e C2 sono utilizzati per eliminare il ripple dalla tensione raddrizzata, mentre C3 e C4 danno stabilità alla tensione in uscita. La resistenza R1 di 10Kohm alimenta con un corrente bassissima (1mA) il led, che indica l'accensione dell'alimentatore. In protezione all'alimentatore e al termometro da un eccessivo assorbimento di corrente utilizzo un fusibile posto sul primario del trasformatore, che deve essere di 100mA, poiché l'intero progetto ha un assorbimento massimo di 40mA.

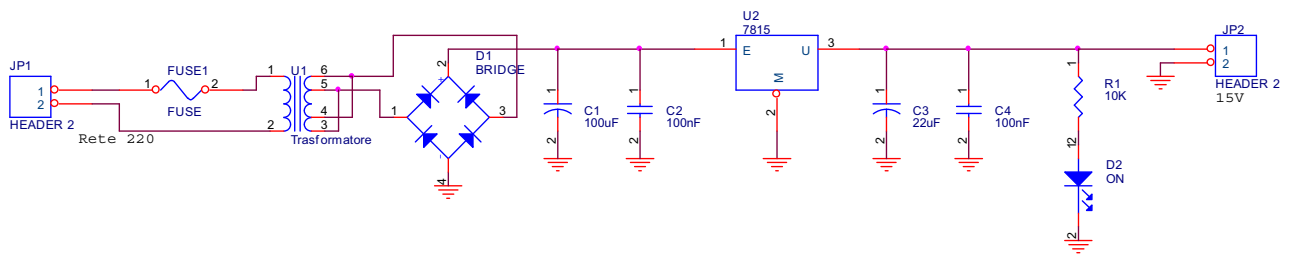


Fig2.12 Alimentatore

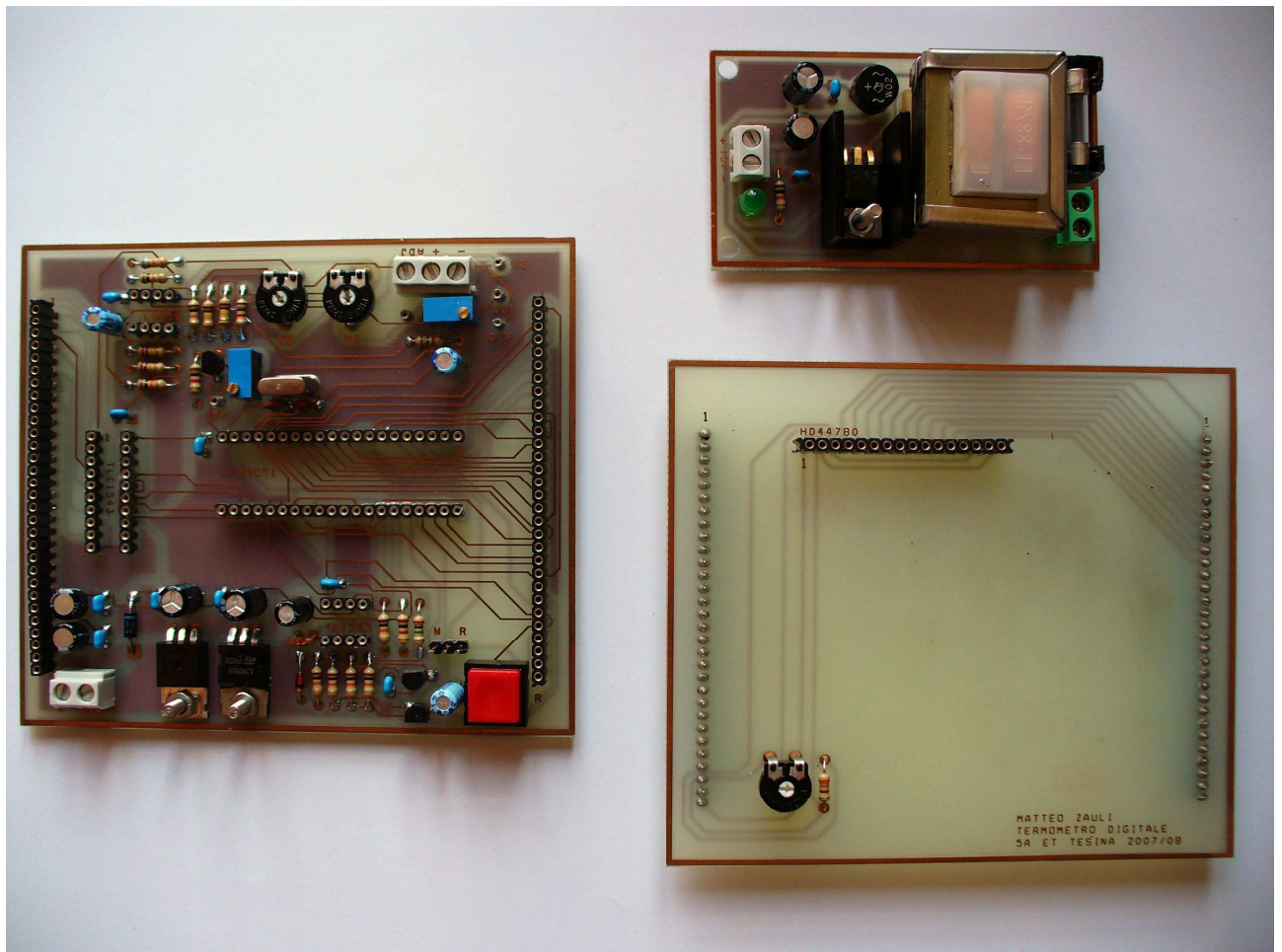


Fig2.13 Le tre schede ultimate di cui è composto il termometro

2.2 Componenti utilizzati

Scheda principale

	Valore	Quantità
Resistenze		
R3, R17	1Kohm	2
R1, R5	1.5Kohm	2
R10, R13, R14, R18, R19, R20, R21, R22	10Kohm	8
R12	68Kohm	1
R6, R7, R8, R9	100Kohm	4
R16	1Mohm	1
Trimmer multi giri verticali		
R2, R15	10Kohm	2
Trimmer orizzontali		
R4	2.2Kohm	1
R11	22Kohm	1
Condensatori		
C18, C19	22pF	2
C15	10nF	1
C4, C6, C8, C10, C11, C12, C13, C14, C17	100nF	9
C16	1uF	1
C1, C2, C20	10uF	3
C5, C9	22uF	2
C3, C7	100uF	2
Diodi		
D2	1N4007	1
D3	1N4148	1
Transistor		
Q1	BC328	1
Q2	BC547	1
Connettori		
JP1	Morsetti 2 vie	1
JP2	Morsetti 3 vie	1
JP3, JP4	30 pin	2
J3	Selezionatore 3 vie	1
Integrati		
U1	LM358	1

U2	TLC1543	1
U3	AT89C51	1
U4	7805	1
U5	7812	1
U6	NE555	1
D1	LM336	1
Interruttori		
SW1	Pulsante	1
Quarzi		
11.0592M	Quarzo 11Mhz	1

Scheda di visualizzazione

	Valore	Quantità
Resistenze		
R2	330ohm	1
Trimmer orizzontali		
R1	10Kohm	1
Connettori		
JP3, JP4	30 pin	2
JP1	16 pin	1

Alimentatore

	Valore	Quantità
Resistenze		
R1	10Kohm	1
Condensatori		
C2, C4	100nF	2
C3	22uF	1
C1	100uF	1
Diodi		
D1	Ponte Graez W02	1
D2	Led	1
Connettori		
JP1, JP2	Morsetti 2 vie	
Integrati		

U2	7815	1
Portafusibili		
FUSE1	5cm	1
Trasformatori		
U1	1.5VA 15AC	1

2.3 Procedura di collaudo

Per eseguire rapidamente il collaudo sono stati inseriti nel disegno degli schemi dei test pointer (TP), di cui si parla anche precedentemente quando appaiono nei vari circuiti descritti all'inizio del capitolo due. Questi test pointer sono identificati da etichette, che sono state riportate sul circuito stampato, in modo da poter identificare i punti dove è necessario apportare delle regolazioni o eseguire delle verifiche. Nei layout riportati è possibile vedere i vari test pointer con la relativa etichetta.

Espongo la procedura di collaudo solo per la scheda principale perché essa è l'unica che richiede alcuni accorgimenti particolari.

Procedura scheda principale

- Tarare il trimmer R2 per ottenere 5V sul TP 5R (layout).
- Tarare il trimmer R4 (VZ) per ottenere 2.63V sul TP VZ (layout).
- Isolare la sonda di temperatura poi metterla a contatto con del ghiaccio, il trimmer R15 dovrà essere tarato per ottenere 2.73V sul TP Sonda 1 (schema elettrico).
- Misurare la tensione sul TP DIF (layout), moltiplicarla per il guadagno che deve avere l'amplificatore non invertente (8.33) e tarare il trimmer R11 (A0) per ottenere sul TP A0 (layout) il risultato della operazione eseguita (tensione che può variare dai 0 ai 5V).

3 DESCRIZIONE SOFTWARE

Il software è l'elemento base di tutto il termometro, esso regge il funzionamento di tutto il progetto. Il suo compito è quello di gestire il convertitore analogico digitale, dal quale acquisisce le rilevazioni della sonda, che poi converte in una cifra corrispondente alla temperatura misurata. Questa elaborazione è definita dal programmatore e in questo caso ho scelto di realizzare un termometro che misurasse una temperatura tra i -10°C e i 50°C , quindi ho a disposizione sessanta valori, volendo aggiungere maggior precisione indico anche le decine, quindi ho un complessivo di seicento valori. Il fatto di essere a conoscenza del range di valori rilevabili dal termometro mi permette di realizzare la conversione del dato acquisito dall'ADC in un valore di temperatura attraverso la proporzione: $600 : 1023 = \text{temperatura} : \text{valore letto dall'ADC}$. In questo modo si ottiene in maniera molto semplice la temperatura; utilizzando la massima temperatura rilevabile (600), il valore massimo ottenibile dalla conversione dell'ADC (1023) e il dato letto dalla conversione. Infine per ottenere le temperature negative bisogna semplicemente sottrarre il risultato di una quantità corrispondente al numero di valori negativi, in questo caso arrivando ad un minimo di -10°C si deve sottrarre cento. Prima di eseguire la stampa sul display LCD eseguo nuovamente per diverse volte le operazioni precedentemente descritte, in modo da poter fare una media tra le acquisizioni, rendendo la visualizzazione della temperatura più stabile e immune da possibili acquisizioni errate, che andrebbero corrette dalla media di numerosi valori.

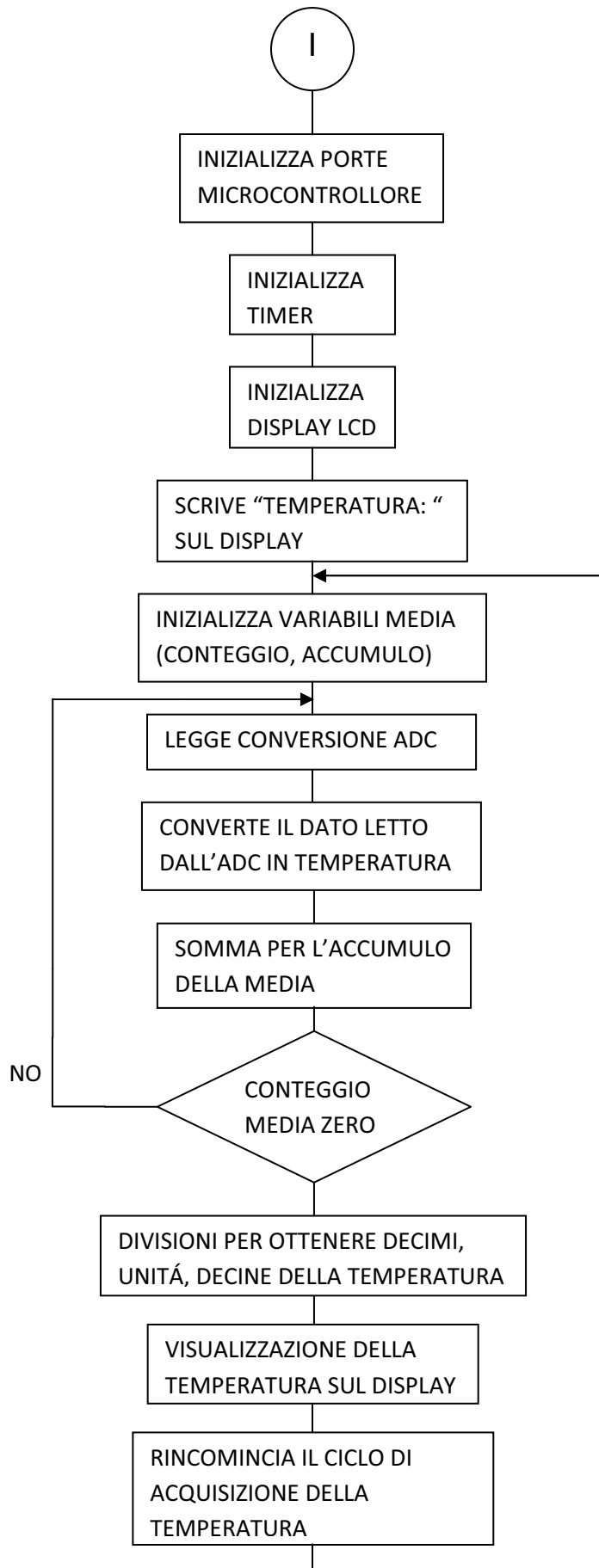
Lo schermo, anch'esso gestito tramite software, necessita di alcuni accorgimenti come una subroutine di inizializzazione, attraverso cui si possono impostare diverse opzioni come il tipo di carattere e il metodo di scrittura. La comunicazione con lo schermo avviene tramite un bus dati, che in questo caso consiste nella porta P1 del microcontrollore e con la quale possiamo inviare dati o istruzioni, leggere o scrivere in base allo stato dei tre segnali di comando: RS (1 dato, 0 istruzione), RW (1 legge, 0 scrive) e E (1 abilitato, 0 disabilitato).

L'esecuzione di questi processi è schematizzata nel diagramma a blocchi nel paragrafo 3.1 e ogni istruzione è indicata e commentata nel listato al paragrafo 3.2.

L'unica difficoltà incontrata nella realizzazione del software è dovuta al fatto che il microcontrollore dispone di una CPU, registri e celle di memoria a 8 bit, quindi nel momento in cui devo gestire un ADC a 10 bit bisogna trattare ogni singolo dato come uno a 16 bit, andando a occupare due celle di memoria invece che una e dovendo utilizzare diverse istruzioni per eseguire semplici operazioni matematiche normalmente risolvibili con un passaggio (vedi ad esempio capitolo 3.2 subroutine programma: div16, conversione).

3.1 Diagramma a blocchi software termometro

In questo capitolo è illustrato lo schema a blocchi del programma realizzato per il termometro digitale e si possono notare tutti i passaggi principali.



3.2 Listato software termometro

Di seguito è riportato il listato del programma realizzato in linguaggio Assembler, con il quale si è potuto programmare il microcontrollore. All'inizio dichiaro l'utilizzo della porta P2 e le variabili create per rendere più intuitiva la realizzazione del software.

Il programma è strutturato da una parte principale detta main, da cui sono chiamati i diversi sottoprogrammi adibiti a funzioni diverse.

Le scritte in blu sono i commenti, in rosso le etichette e in nero le istruzioni.

```
;Settaggio utilizzo porte
CLK      = P2.0           ;Clock ADC
CS       = P2.1           ;Chip select ADC
ADDR     = P2.2           ;Dato ADC
DO       = P2.3           ;Indirizzo ingresso ADC
RS       = P2.4           ;Dato - istruzione, display
RW       = P2.5           ;Legge - scrive, display
EN       = P2.6           ;Abilita display
WD       = P2.7           ;Watch Dog
BDATI    = P1             ;Bus dati display

;+++++
+++
;Disposizione memoria interna
org 0030h
deci      db 0            ;Decine temperatura
uni       db 0            ;Unità temperatura
decimi    db 0            ;Decimi temperatura
neg       db 0            ;Carattere meno per la scrittura sul display
tl        db 0            ;Risultato traslazione, parte bassa
th        db 0            ;Risultato traslazione, parte alta
dl        db 0            ;Dato acquisito, parte bassa
dh        db 0            ;Dato acquisito, parte alta
Cmedia    db 0            ;Contatore media
Aml       db 0            ;Accumulo media, parte bassa
Amh       db 0            ;Accumulo media, parte alta
CmediaK   db 0            ;Costante conteggio media
Test      db 0            ;Cella di memoria sulla quale eseguire il test
per verificare se il programma è partito dall'inizio

mess1     db 'Temperatura: ',0    ;Messaggio da visualizzare sul display
prima del valore della temperatura

;+++++
+++
;MAIN
;Programma principale
org 0h
    jmp start           ;Va all'inizio del programma saltando le celle di
memoria delle variabili
org 0100h
start:
    mov Test,#18        ;Assegna valore alla cella di test
    mov sp,#60h         ;Sposta stack pointer a 60h
    mov CmediaK,#30h    ;Valore conteggio media
    setb DO             ;Prepara la porta per l'acquisizione dei dati dalla
conversione dell'ADC
    setb CS             ;Disabilita l'ADC
```

```

    clr CLK                ;Azzera il clock
    clr ADDR              ;Utilizzo solo il primo ingresso dell'ADC quindi
mantengo a zero l'indirizzo degli ingressi del convertitore, variabile utile
in caso di variazioni future al programma
    call inittmr         ;Subroutine: inizializza timer
    call initlcd        ;Subroutine: inizializza LCD
    mov dptr,#mess1     ;Inizializza la subroutine di scrittura mettendo nel
dptr la cella di memoria da cui inizia il testo da scrivere
    call scritta        ;Subroutine: prepara la scritta "Temperatura: " sul
display LCD a cui seguirà la misura della temperatura
acqui:
    mov r0,Test         ;Per eseguire cjne sposta il contenuto della cella
di memoria in R0
    cjne r0,#18,start   ;Esegue il test, se non è uguale salta all'inizio
del programma (start)
    clr WD              ;Azzera p2.7 a cui è collegato il watch dog
    mov Aml,#0         ;Azzera accumulo media, parte bassa
    mov Amh,#0         ;Azzera accumulo media, parte alta
    mov Cmedia,CmediaK ;Inizializza contatore media
media:
    call lettura        ;Subroutine: lettura conversione ADC 10 bit
    call conversione   ;Subroutine: conversione dato letto da ADC
attraverso una proporzione per ottenere il valore della temperatura
    call traslazione    ;Subroutine: sposta il valore acquisito di -100 per
ottenere un valore di temperatura fino a -10°C
    call sommamed      ;Subroutine: somma per l'accumulo della media

    djnz Cmedia,media  ;Decrementa contatore media

    setb WD            ;Setta p2.7 a cui è collegato il watch dog

    mov r0,Aml         ;Parte bassa dividendo
    mov r1,Amh         ;Parte alta dividendo
    mov r2,#30h        ;Parte bassa divisore
    mov r3,#0          ;Parte alta divisore
    call div16         ;Subroutine: divisione

;Prepara divisione per scomporre la temperatura in decimi, unità, decine,
per eseguire la stampa dei vari caratteri singolarmente
    mov r0,tl          ;Parte bassa dividendo
    mov r1,th          ;Parte alta dividendo
    mov r2,#10         ;Parte bassa divisore
    mov r3,#0          ;Parte alta divisore
    call div16         ;Subroutine: divisione

    mov a,r0           ;Il resto della divisione precedente (r0) consiste
nei decimi della temperatura
    add a,#30h         ;Somma 30h per ottenere il valore corrispondente
della tabella ASCII utile al display LCD per la stampa
    mov decimi,a      ;Il resto della divisione precedente consiste nei
decimi della temperatura

    mov r0,tl          ;Parte bassa dividendo
    mov r1,th          ;Parte alta dividendo
    mov r2,#10         ;Parte bassa divisore
    mov r3,#0          ;Parte alta divisore
    call div16         ;Subroutine: divisione

    mov a,r0           ;Il resto della divisione precedente (r0) consiste
nelle unità della temperatura
    add a,#30h         ;Somma 30h per ottenere il valore corrispondente
della tabella ASCII utile al display LCD per la stampa

```

```

    mov uni,a          ;Il resto della divisione precedente consiste nelle
unità della temperatura
    mov a,r4          ;Il risultato della divisione precedente (r4)
consiste nelle decine della temperatura
    add a,#30h        ;Somma 30h per ottenere il valore corrispondente
della tabella ASCII utile al display LCD per la stampa
    mov deci,a        ;Il risultato della divisione precedente consiste
nelle decine della temperatura

    call lcd          ;Subroutine: scrittura temperatura sul display LCD
    jmp acqui         ;Ricomincia il processo di acquisizione
;Il programma si ripete sempre per eseguire sempre nuove rilevazioni di
temperatura

;+++++
+++
;Lettura seriale ADC 10bit
lettura:
    mov dl,#0         ;Azzerà parte bassa acquisizione
    mov dh,#0         ;Azzerà parte alta acquisizione
    mov r0,#10        ;Inizializza il contatore, 10 come il numero di bit
da acquisire
    clr CS            ;Abilita l'ADC
dato:
    mov c,DO          ;Utilizza il carry per salvare il bit ricevuto dalla
conversione
    mov a,dl          ;Mette nell'accumulatore dl
    rlc a             ;Rotate con il carry per spostare nell'accumulatore
il bit appena acquisito
    mov dl,a          ;Salva il risultato in dl
    mov a,dh          ;Mette nell'accumulatore dh
    rlc a             ;Quando vado oltre gli 8 bit di dl sposterà il bit
eccedente nel carry, per non perderlo lo sposto nell'accumulatore
    mov dh,a          ;Salva il risultato in dh, dove avrò i due bit più
significativi dei 10 bit acquisiti
    setb CLK          ;Impulso di clock
    clr CLK           ;Azzerà il clock
    djnz r0,dato      ;Contatore acquisizione, r0 volte
    setb CS           ;Disabilita l'ADC
ret
;+++++
+++
;Conversione lettura ADC in temperatura
;Proporzione tra il valore acquisito e la temperatura, 600(valore massimo
temperatura):1023(massimo valore acquisizione)=temperatura:valore acquisito
(dl,dh)
conversione:
;Moltiplicazione 16 bit tra il valore acquisito e 600 (valore massimo
temperatura)
;          R6 R7
; x        R4 R5
; = R0 R1 R2 R3

;Preparazione moltiplicazione
    mov r7,dl         ;Mette al (parte bassa acquisizione) nella parte
bassa moltiplicando
    mov r6,dh         ;Mette ah (parte alta acquisizione) nella parte alta
moltiplicando
    mov r5,#58h       ;Mette 600 (valore massimo temperatura) nella parte
bassa moltiplicatore
    mov r4,#2h        ;Mette 600 (valore massimo temperatura) nella parte
alta moltiplicatore

```

```

;Moltiplica R5 con R7
    mov a,r5           ;Mette r5 nell'accumulatore
    mov b,r7           ;Mette r7 in b
    mul ab             ;Moltiplica r7 con r5
    mov r2,b           ;Mette b (la parte alta risultato moltiplicazione)
in r2
    mov r3,a           ;Mette a (la parte bassa risultato moltiplicazione)
in r3

;Moltiplica R5 con R6
    mov a,r5           ;Mette r5 nell'accumulatore
    mov b,r6           ;Mette r6 in b
    mul ab             ;Moltiplica r5 con r6
    add a,r2           ;Somma la parte bassa del risultato a r2
    mov r2,a           ;Mette il risultato in r2
    mov a,b           ;Mette la parte alta del risultato della
moltiplicazione (b) nell'accumulatore
    addc a,#00h        ;Somma il carry all'accumulatore
    mov r1,a           ;Mette il risultato in r1
    mov a,#00h         ;Azzera l'accumulatore
    addc a,#00h        ;Somma il carry all'accumulatore
    mov r0,a           ;Mette il risultato in r0

;Moltiplica R4 con R7
    mov a,r4           ;Mette r4 nell'accumulatore
    mov b,r7           ;Mette r7 in b
    mul ab             ;Moltiplica r4 con r7
    add a,r2           ;Somma la parte bassa del risultato a r2
    mov r2,a           ;Mette il risultato in r2
    mov a,b           ;Mette la parte alta del risultato della
moltiplicazione (b) nell'accumulatore
    addc a,r1          ;Somma all'accumulatore r1 più il carry
    mov r1,a           ;Mette il risultato in r1
    mov a,#00h         ;Azzera l'accumulatore
    addc a,r0          ;Somma all'accumulatore r0 più il carry
    mov r0,a           ;Mette il risultato in r0

;Moltiplica R4 con R6
    mov a,r4           ;Mette r4 nell'accumulatore
    mov b,r6           ;Mette r6 in b
    mul ab             ;Moltiplica r4 con r6
    add a,r1           ;Somma la parte bassa del risultato a r1
    mov r1,a           ;Mette il risultato in r1
    mov a,b           ;Mette la parte alta del risultato della
moltiplicazione (b) nell'accumulatore
    addc a,r0          ;Somma all'accumulatore r0 più il carry
    mov r0,a           ;Mette il risultato in r0

;Fine moltiplicazione 16 bit, il risultato è in: R0, R1, R2, R3

    clr c              ;Azzera il carry, mi servirà nella divisione per
1023
    mov a,r2           ;Deve dividere per 1023 quindi rotate di 10bit,
elimino R3 corrisponente a un rotate di 8bit, mi mancano ancora 2 bit quindi
devo fare due rotate con il carry per non perdermi il riporto della parte
alta del risultato
    push acc           ;Salva stato accumulatore dove c'è r2
    mov a,r1           ;Mette nell'accumulatore r1
    rrc a              ;Ruota con il carry di 1 bit
    mov r1,a           ;Salva il risultato in r1

```

```

    pop acc                ;Scarica lo stato salvato precedentemente
dell'accumulatore quindi riprendo r2
    rrc a                  ;Rotate con il carry (riporto di r1) di 1 bit
    push acc               ;Salva risultato nell'accumulatore
    clr c                  ;Azzerà il carry dove c'era il riporto del rotate di
r2
    mov a,r1               ;Mette r1 nell'accumulatore
    rrc a                  ;Ruota con il carry di 1 bit
    mov r1,a               ;Completata la divisione per 1023 di r1
    pop acc                ;Scarica lo stato salvato precedentemente
dell'accumulatore quindi riprendo r2
    rrc a                  ;Ruta con il carry (riporto di r1) di 1 bit
    mov r2,a               ;Completata la divisione per 1023 di r2, il
risultato definitivo è in r1 (parte alta), r2 (parte bassa)
    clr c                  ;Non utilizzo più il carry lo azzerò
ret
;+++++
+++
;Somma 16 bit per la media
;   R2  R1
;+  th  tl
;=  Amh Aml
;           !!MEDIA FINO A 64 (40h) VALORI PRIMA DELL'OVERFLOW DI Aml Amh!!
sommamed:
    mov r1,tl
    mov r2,th
;passaggio 1
    mov a,r1               ;Sposta nell'accumulatore parte bassa
    add a,Aml              ;Somma all'accumulatore la parte bassa
    mov Aml,a              ;Sposta il risultato nell'accumulo della media, parte
bassa
;passaggio 2
    mov a,r2               ;Sposta nell'accumulatore parte alta
    addc a,Amh             ;Somma all'accumulatore la parte alta, più il carry.
    mov Amh,a              ;Sposta il risultato nell'accumulo della media, parte
alta
;Finito, il risultato è in Aml e Amh.
ret
;+++++
+++
;Traslazione valore temperatura secondo il valore negativo che devo
raggiungere in questo caso -10°C quindi -100
traslazione:
;Per eseguire la sottrazione a 16 bit eseguo una operazione in colonna
;   R1 R2
; - R3 R4
; = th tl
    mov r4,#100           ;Imposta la parte bassa del sottraendo
    mov r3,#0              ;Imposta la parte alta del sottraendo
    mov a,r2               ;Mette la parte bassa del minuendo nell'accumulatore
per preparare la sottrazione
    clr c                  ;Azzerà il carry
    subb a,r4              ;Sottrae le due parti basse del minuendo e del
sottraendo
    mov tl,a               ;Mette il risultato in tl
    mov a,r1               ;Mette la parte alta del minuendo nell'accumulatore
per preparare la sottrazione
    subb a,r3              ;Sottrae le due parti alte del minuendo e del
sottraendo

```

```

    mov th,a          ;Mette il risultato in th, operazione completata il
risultato totale è in th (parte alta), t1 (parte bassa)
    jnc pos          ;Nel caso in cui il risultato sia positivo salta le
istruzioni seguenti per la trattazione di un dato negativo
    ;Se si ha un valore negativo il risultato è in overflow quindi per ottenere
il suo reciproco valore deve essere sottratto a zero
    clr c           ;Azzerà il carry, non deve condizionare la
sottrazione successiva
    mov a,#0        ;Azzerà l'accumulatore
    subb a,t1       ;Sottrae a zero la parte bassa della traslazione
(t1)
    mov t1,a        ;Salva il risultato in t1
    mov a,#0        ;Azzerà l'accumulatore
    subb a,th       ;Sottrae a zero la parte alta della traslazione (th)
    mov th,a        ;Salva il risultato in th
    mov neg,#2Dh    ;Cella di memoria in cui salva il carattere
corrispondente al meno, utilizzata per la stampa
pos:
ret
;+++++
+++
;Divisione 16 bit
div16:
;R1 (parte alta),R0 (parte bassa) dividendo
;R3 (parte alta),R2 (parte bassa) divisore
;RISULTATO riportato in th (parte alta),t1 (parte bassa)
;RESTO R1 (parte alta),R0 (parte bassa)

    clr c           ;Azzerà il carry
    mov r4,#0       ;Azzerà R4
    mov r5,#0       ;Azzerà R5
    mov b,#0        ;Azzerà B in quanto dovrà contare il numero degli
spostamenti a sinistra
div1:
    inc b           ;Incrementa il contatore per ogni spostamento a
sinistra
    mov a,r2        ;Parte bassa del divisore nell'accumulatore
    rlc a           ;Sposta la parte bassa a sinistra con il carry
    mov r2,a        ;Salva il risultato
    mov a,r3        ;Parte alta del divisore nell'accumulatore
    rlc a           ;Sposta la parte alta a sinistra
    mov r3,a        ;Salva il risultato
    jnc div1        ;Ripete div1 finché il carry non è settato
div2:
    mov a,r3        ;Sposta a destra il divisore
    rrc a           ;Ruota la parte alta del divisore a destra con il
carry
    mov r3,a        ;Salva il risultato
    mov a,r2        ;Sposta la parte bassa del divisore
nell'accumulatore
    rrc a           ;Ruota la parte bassa del divisore a destra con il
carry della parte alta
    mov r2,a        ;Salva il risultato
    clr c           ;Cancella il carry, non ci servirà più
    mov 07h,r1      ;Fa una copia di sicurezza della parte alta del
dividendo
    mov 06h,r0      ;Fa una copia di sicurezza della parte bassa del
dividendo
    mov a,r0        ;Sposta la parte bassa del dividendo
nell'accumulatore
    subb a,r2       ;Sottrae il divisore al dividendo
    mov r0,a        ;Salva il nuovo dividendo

```

```

    mov a,r1                ;Sposta la parte alta del dividendo
nell'accumulatore
    subb a,r3              ;Sottrae la parte alta del divisore
    mov r1,a              ;Salva la parte alta del divisore
    jnc div3              ;Se il carry non è settato il risultato è 1
    mov r1,07h           ;In caso contrario il risultato è 0, salviamo una
copia del divisore
    mov r0,06h
div3:
    cpl c                  ;Complementa il carry per essere messo nel risultato
    mov a,r4
    rlc a                  ;Sposta il carry nel risultato temporaneo
    mov r4,a
    mov a,r5
    rlc a
    mov r5,a
    djnz b,div2           ;Ripete da div2 fino ad azzerare b
    mov th,r5             ;Metto la parte alta del risultato in th
    mov tl,r4             ;Metto la parte bassa del risultato in tl
ret
;+++++
+++
;Subroutine per la scrittura della scritta "Temperatura: " sul display LCD
scritta:
    clr a
    movc a,@a+dptr
    jz fine
    call wrchar
    inc dptr
    jmp scritta
    fine:
ret
;+++++
+++
;Subroutine per la scrittura della temperatura sul display LCD
lcd:
    call setrigal
    mov a,neg
    jz posi
    call wrchar
    mov neg,#0
posi:
    mov a,deci
    call wrchar           ;Stampa decine
    mov a,uni
    call wrchar           ;Stampa unità
    mov a,#2Ch
    call wrchar           ;Stampa virgola per separare i decimi
    mov a,decimi
    call wrchar           ;Stampa decimi
    mov a,#0DFh
    call wrchar           ;Stampa carattere gradi "°"
    mov a,#43h
    call wrchar           ;Stampa la C di Celsius
    mov a,#20h
    call wrchar           ;Stampa spazio
    mov a,#20h
    call wrchar           ;Stampa spazio, con gli ultimi due spazi stampati
cancello i caratteri che rimarrebbero visualizzati quando la stampa si sposta
di un carattere a destra quando sulla sinistra si stampa il meno per i valori
negativi
ret

```

```

;+++++
+++
;Funzione per inviare al display LCD una istruzione
wrc:
    clr RS
    clr RW
    mov P1,a
    setb EN
    clr EN
ret
;+++++
+++
;Funzione per inviare al display LCD un dato
wrđ:
    setb RS
    clr RW
    mov P1,a
    setb EN
    clr EN
ret
;+++++
+++
;Test sul bf, flag che mi indica quando il display è pronto a eseguire una
nuova operazione
bf:
    mov P1,#0ffh
    clr RS
    setb RW
    setb EN
    jb P1.7,$
    clr EN
ret
;+++++
+++
;Ritardo di 100uS
del100u:
    mov t10,#256-92
    setb tr0
    jnb tf0,$
    clr tf0
    clr tr0
ret
;+++++
+++
;Subroutine con un ritardo base di 100uS ripetibile r1 volte per chiamare
ritardi più lunghi
delay:
    call del100u
    djnz r1,delay
ret
;+++++
+++
;Funzione per scrivere un carattere sul display LCD
wrchar:
    call bf
    mov P1,a
    setb RS
    clr RW
    setb EN
    clr EN
ret

```

```

;+++++
+++
;Funzione per cancellare il display LCD
canc:
    call bf
    clr RW
    clr RS
    mov P1,#01h
    setb EN
    clr EN
ret
;+++++
+++
;Imposta la scrittura del valore di temperatura nella seconda riga del
display dopo la scritta "Temperatura: "
setrigal:
    call bf
    clr RS
    clr RW
    mov P1,#0C0h
    setb EN
    clr EN
ret
;+++++
+++
;Inizializza timer
inittmr:
    mov a,tmod
    anl a,#0f0h
    orl a,#02h
    mov tmod,a
ret
;+++++
+++
;Inizializza LCD
initlcd:
    clr EN
    mov a,#00110000b    ;Inizializzazione fase 1
    call wrc
    mov r1,#50         ;Aspetta 5 ms
    call delay
    call wrc           ;Inizializzazione fase 1
    call del100u       ;Aspetta 100 us
    call wrc           ;Inizializzazione fase 2
    call bf
    mov a,#00111000b   ;Set interfaccia 8 bit
    call wrc
    call bf
    mov a,#00001100b   ;Display ON/OFF
    call wrc
    call bf
    mov a,#00000001b   ;Display home
    call wrc
    call bf
    mov P1,#00000110b  ;Display entry mode
    call wrc
ret
end

```

4 CONCLUSIONI

Attraverso questa esperienza ho potuto provare di persona le difficoltà e i vari modi su come può essere sviluppato un sistema autonomo, che gestisce una acquisizione e su come potrebbe essere utilizzato in un controllo automatico.

Il corso di quinta elettronica e telecomunicazioni mi ha fornito i mezzi per poter sviluppare il termometro digitale realizzato e i suggerimenti su come potrebbe essere migliorato. L'elettronica è stata fondamentale nell'impartirmi la teoria per l'utilizzo dei vari componenti. Attraverso sistemi ho appreso il linguaggio Assembler, con cui ho programmato il microcontrollore e appreso la teoria sui sistemi di controllo. Il lavoro svolto con TDP racchiude tutta la parte di disegno degli schemi elettrici e i relativi circuiti stampati. Il breve summary comprende l'utilizzo della lingua inglese sottolineandone l'importanza, essendo anche la lingua con cui sono scritti i datasheet, dai quali ho preso le caratteristiche dei componenti per effettuare la scelta su quali utilizzare. Purtroppo non ho potuto includere un'altra materia molto importante del mio corso come telecomunicazioni. Il suo utilizzo inizialmente era stato pensato per includere nel progetto un modulo radio, che avrebbe comunicato con una scheda indipendente, dalla quale eseguire altre rilevazioni tipo un secondo valore di temperatura.

Sono stato molto soddisfatto nell'aver portato a termine un simile progetto, che nella sua semplicità racchiude dei problemi pratici, che hanno saputo darmi una conoscenza più approfondita e accresciuto la mia pratica nella realizzazione di progetti elettronici. Per la mia poca esperienza è molto interessante la ricerca di continue migliorie da apportare nella realizzazione di un circuito elettronico.

Infine vorrei ringraziare il professor **Marco Ferrari** per l'aiuto e interesse mostrato per questa tesina, il professor **Tigani Sava**, il quale mi ha fornito l'idea di realizzare questo progetto e il tecnico di laboratorio **Roberto Bendandi** per la costruzione dei circuiti stampati non solo per questo progetto ma per tutto l'anno scolastico. In modo particolare vorrei ringraziare il professor **Maurizio Montanari**, senza la sua gentile collaborazione nel fornirmi i giusti consigli non avrei potuto ultimare il termometro digitale che vi ho illustrato e per la sua grande pazienza nel seguirmi nei miei numerosi esperimenti elettronici effettuati in questi anni di specializzazione. Inoltre vorrei aggiungere nei ringraziamenti anche il professor **Sergio Casuccio**, il quale anche se non fa più parte del corpo insegnanti dell'ITI, essendo andato in pensione l'anno scorso, durante il quarto anno scolastico ha saputo incoraggiarmi e sostenermi nella mia grande curiosità per il mondo dell'elettronica.

5 SUMMARY

This project is about a digital thermometer, a device that measures temperature in degree Celsius.

It has four important elements: temperature sensor, analog to digital converter (ADC), microcontroller and a liquid crystal display (LCD). Temperature sensor converts temperature to an output voltage which can be convert by ADC. The analog to digital converter is an electronic integrated circuit, that converts analog signal from temperature sensor to digital signal. The microcontroller is a component that including a processor (CPU), memory and input/output ports. I use a microcontroller to process the data from the ADC conversion and visualize them by the LCD. The LCD is a thin, flat display device used to visualize alphanumeric character and symbol.

This project offers several possibilities to expand it, because of the many free input/output ports in the microcontroller and the free analog inputs in the ADC, can control other devices and sensors.

The characteristics of this thermometer are: high precision because of the good resolution of the analog to digital converter and velocity in the visualization of the temperature on the display.

For this property this thermometer can be used in private places and in factories.

BIBLIOGRAFIA

Datasheet

- www.datasheetcatalog.com

Teoria componenti utilizzati

- www.wikipedia.it

- Biondo G., Sacchi E. “Manuale di elettronica e telecomunicazioni” Hoepli, quinta edizione

- Cuniberti E., De Lucchi L. “Elettronica analogica” Petrini Editore

Programmazione

- www.8052.com (operazioni 16 bit)

INDICE

Introduzione

Prefazione

1 Descrizione hardware

- 1.1 LM335 sonda di temperatura
- 1.2 TLC1543 convertitore analogico-digitale (ADC) ad approssimazioni successive
- 1.3 LM358 doppio amplificatore operazionale
- 1.4 LM336 diodo di riferimento di tensione
- 1.5 AT89C51 microcontrollore Atmel
- 1.6 WM-C1602M display a cristalli liquidi
- 1.7 NE555 timer
- 1.8 78xx stabilizzatori di tensione monolitici

2 Funzionamento circuito

- 2.1 Descrizione circuiteria
 - 2.1.1 Scheda principale
 - 2.1.2 Scheda di visualizzazione
 - 2.1.3 Alimentatore
- 2.2 Componenti utilizzati
- 2.3 Procedura di collaudo

3 Descrizione software

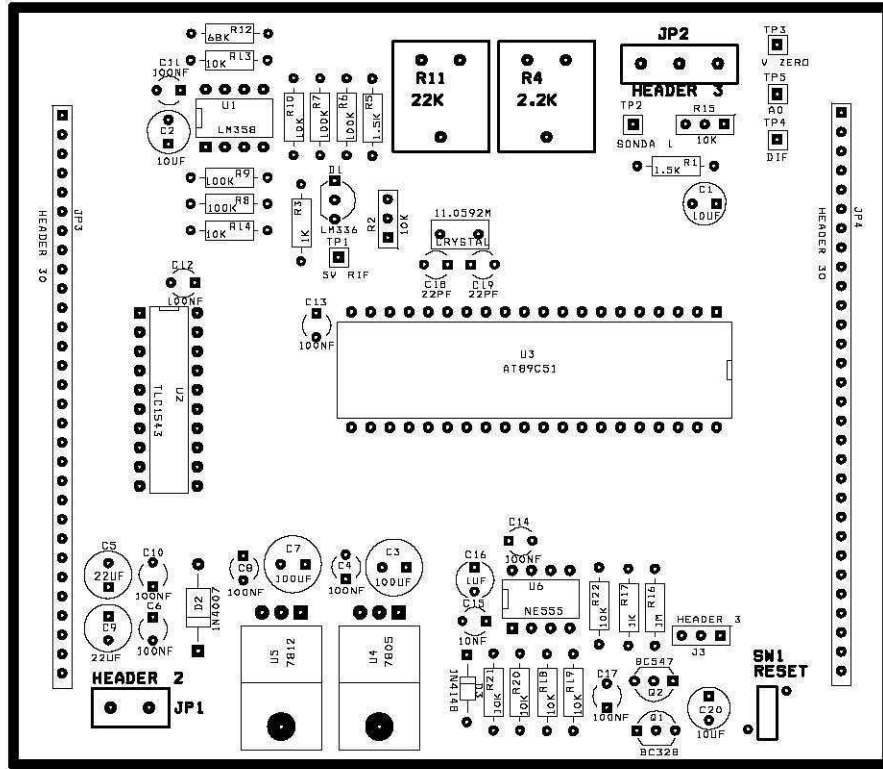
- 3.1 Diagramma a blocchi software termometro
- 3.2 Listato software termometro

4 Conclusioni

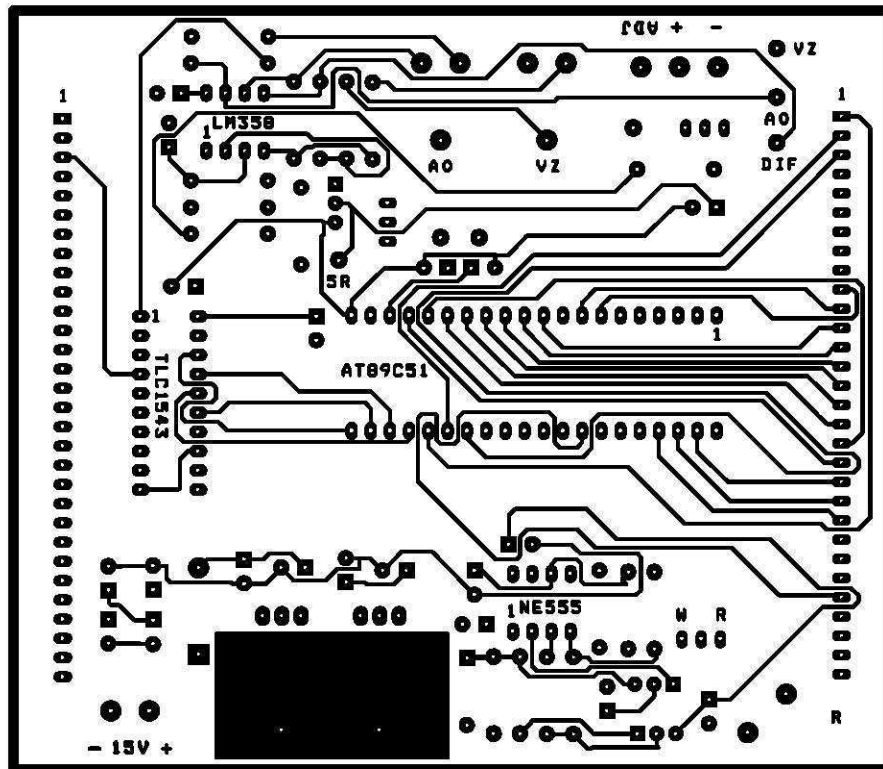
5 Summary

Bibliografia

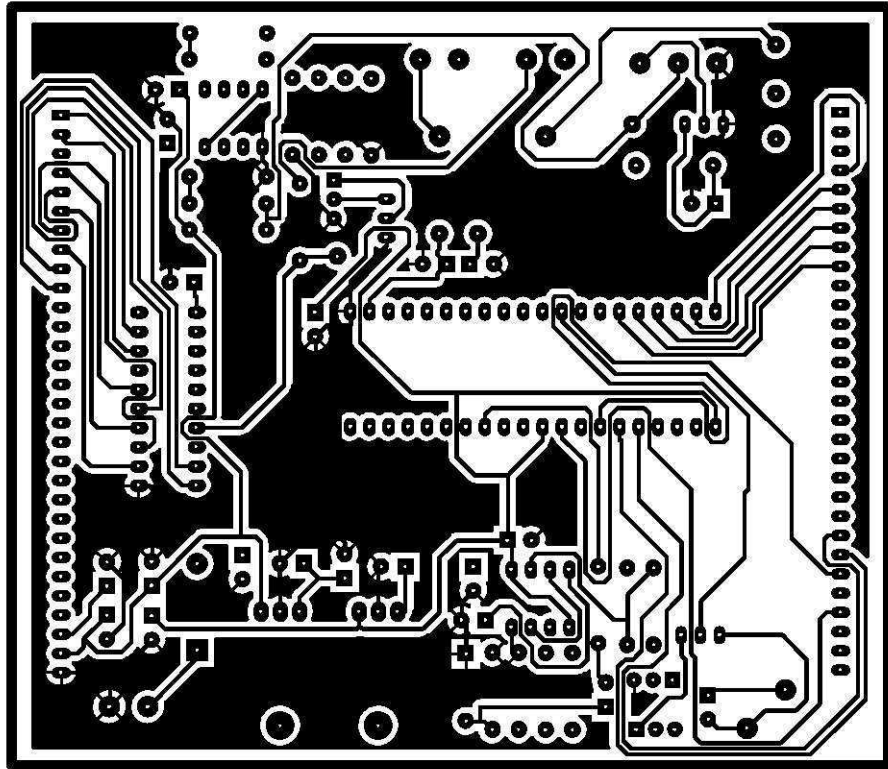
Allegati schemi elettrici e layout dei circuiti realizzati nel progetto



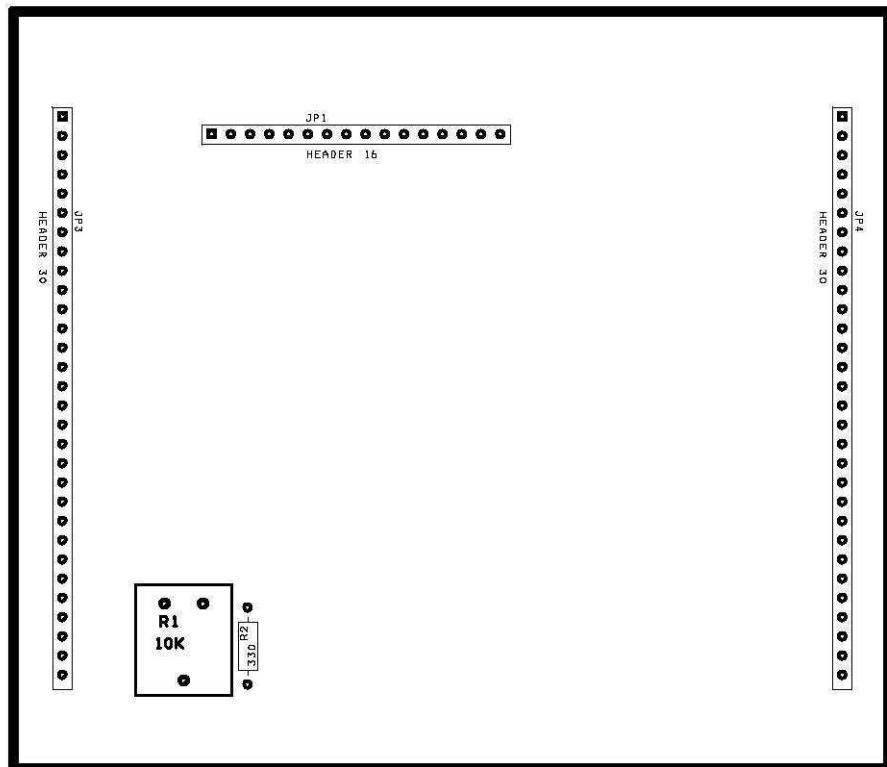
AST layout scheda principale



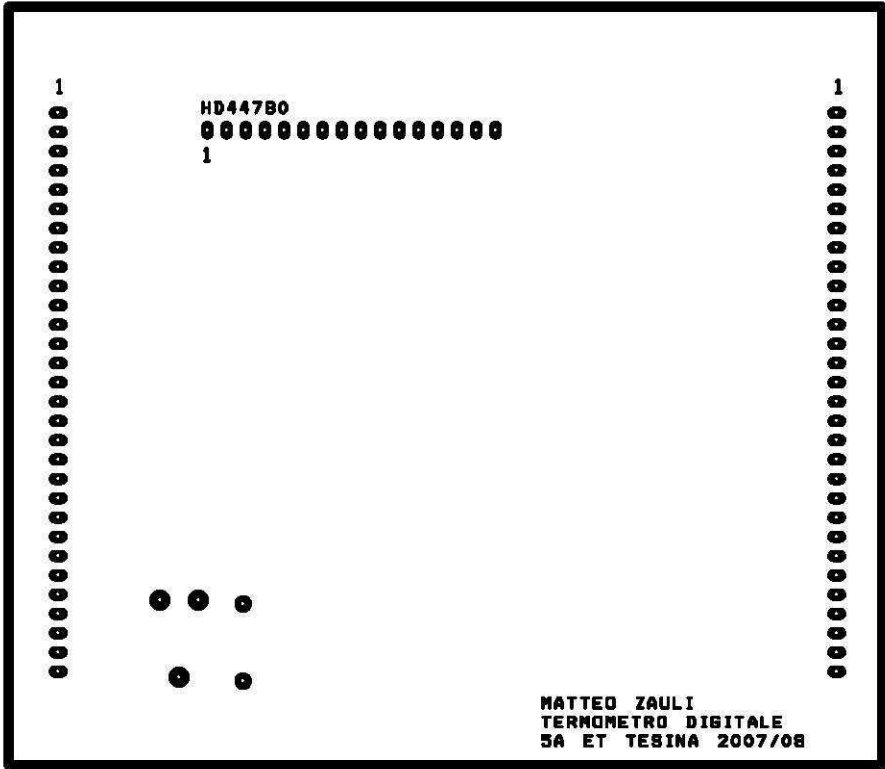
TOP layout scheda principale



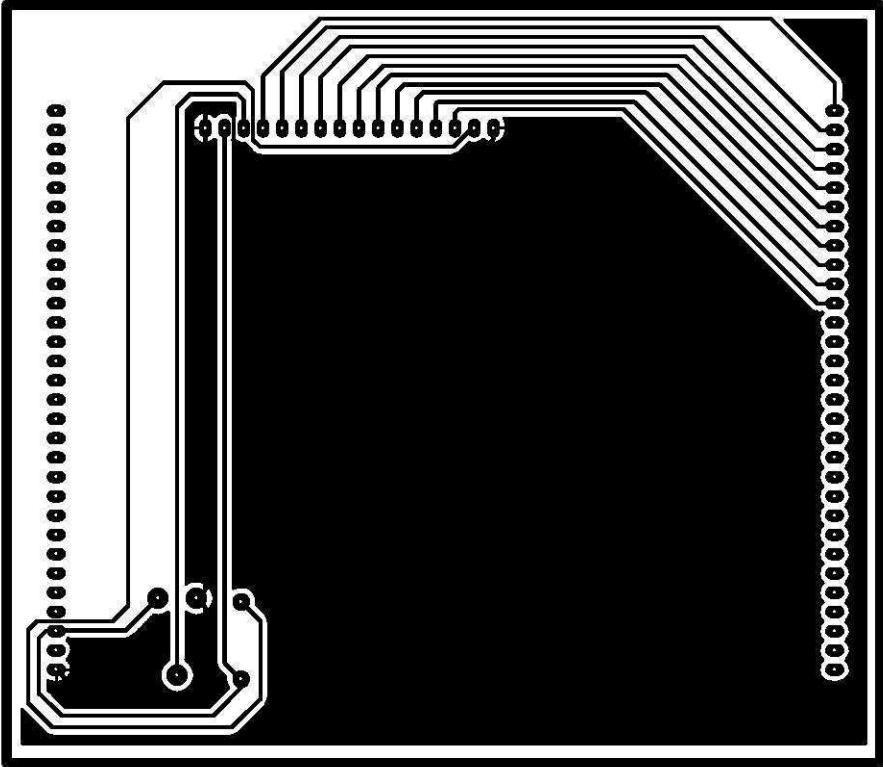
BOTTOM layout scheda principale



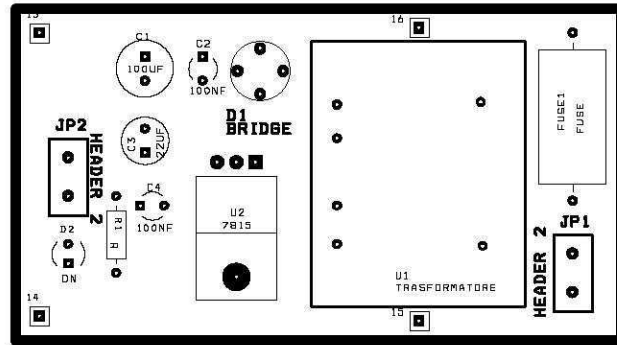
AST layout scheda di visualizzazione



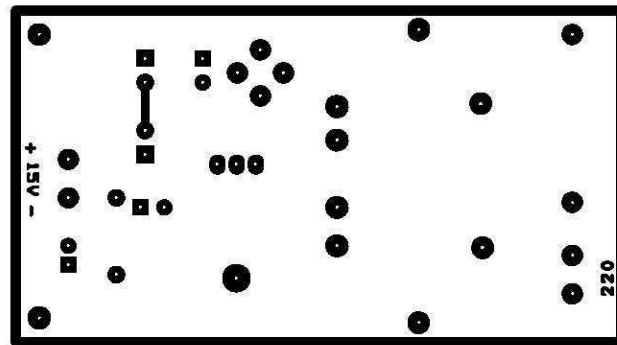
TOP layout scheda di visualizzazione



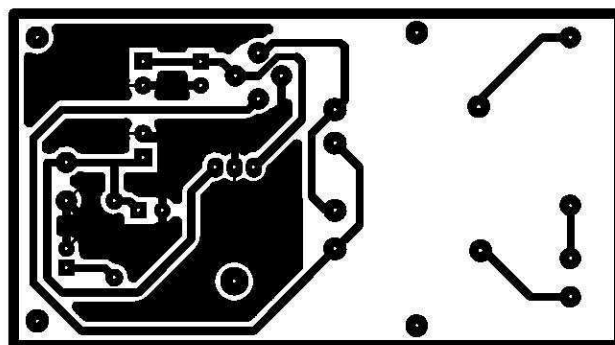
BOTTOM layout scheda di visualizzazione



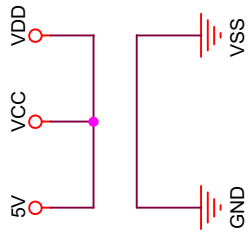
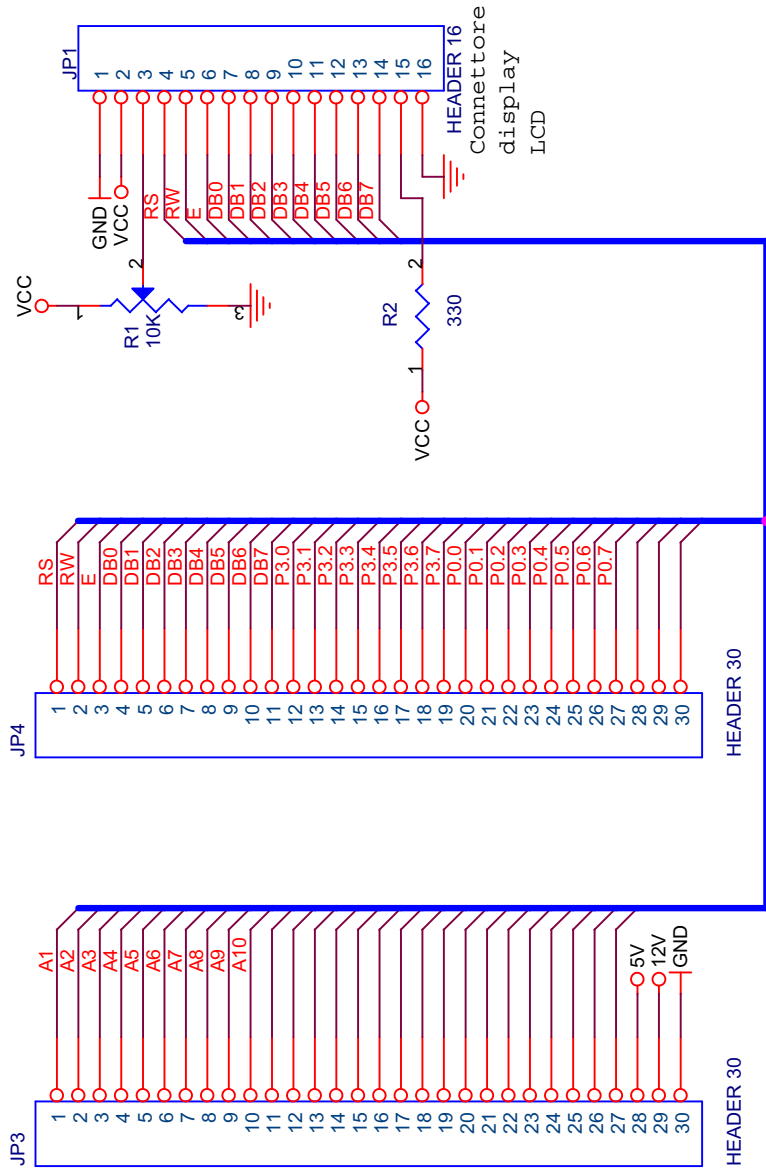
AST layout alimentatore



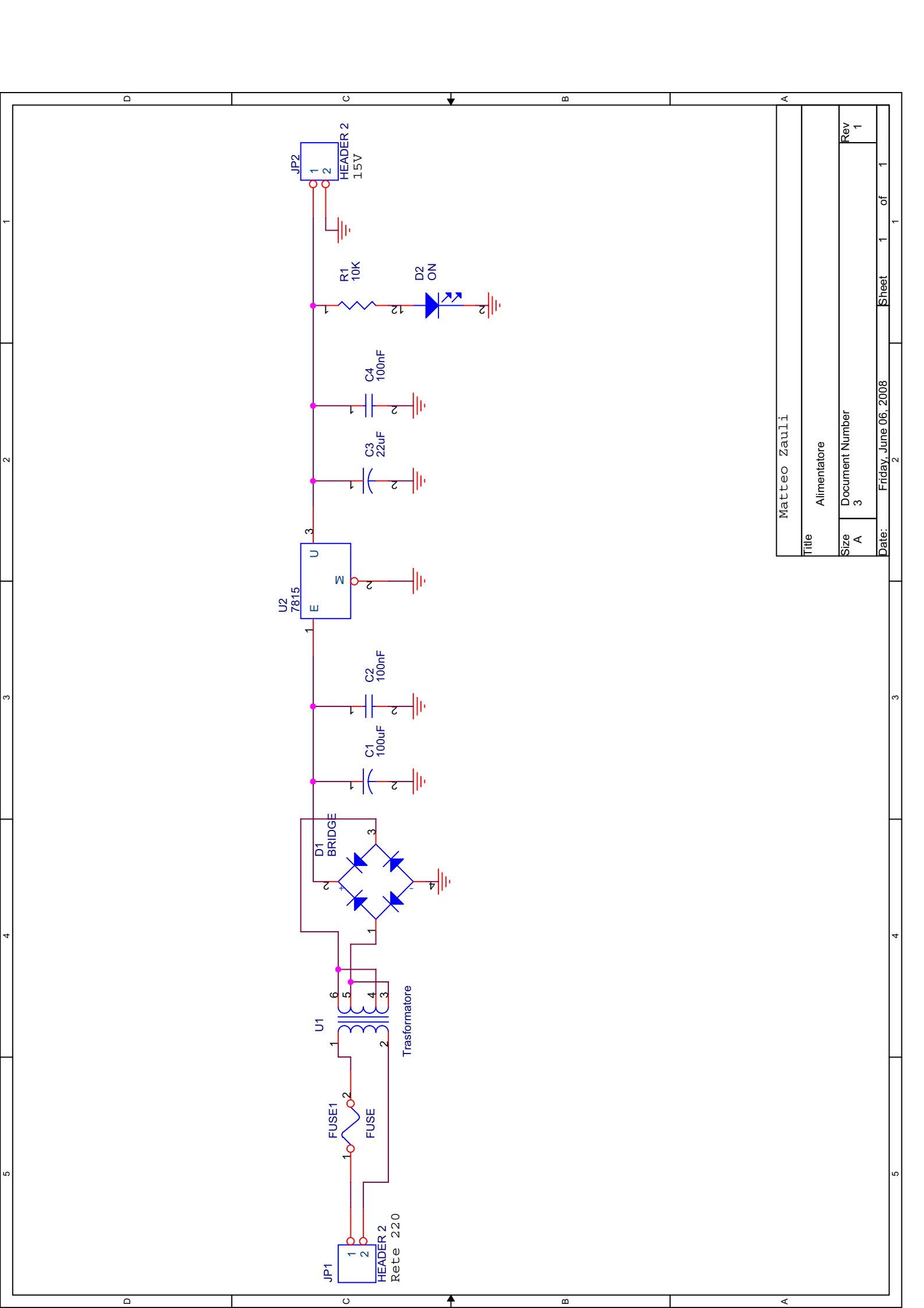
TOP layout alimentatore



BOTTOM layout alimentatore



Matteo Zauli	
Title	
Scheda di visualizzazione	
Size	Document Number
A	2
Rev	1
Date:	Friday, June 06, 2008
Sheet	1 of 1



Matteo Zauli	
Title Alimentatore	
Size A	Document Number 3
Date: Friday, June 06, 2008	
Rev 1	Sheet 1 of 1