



**UNIVERSITA' DEGLI STUDI DI TRIESTE
DIPARTIMENTO DI ELETTROTECNICA, ELETTRONICA E
INFORMATICA
ANNO ACCADEMICO 2002/2003**

TESINA DI ELETTRONICA

**TERMOMETRO DIGITALE GESTITO DA MICROCONTROLLORE
PIC 16F84**

Professore:
Marsi Stefano

Studenti:
Bert Alessandro
Bottos Enrico

INDICE

Introduzione

Pic 16F84

Programmatore Ludipipo

Circuito

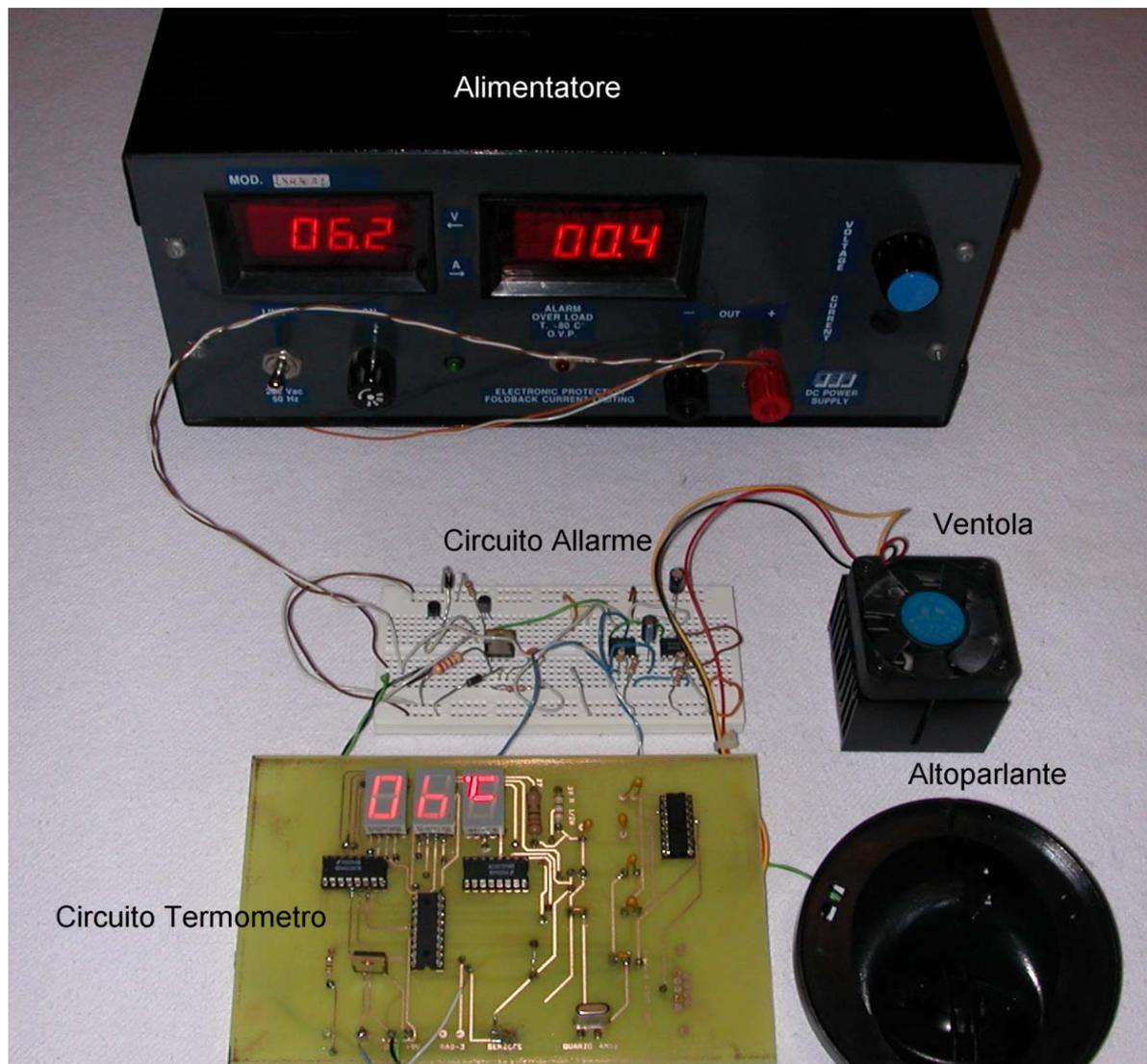
Programmazione

Disegno del circuito per lo stampaggio

Conclusioni

Appendice: Sistema di Controllo e Allarme di Temperatura

Bibliografia



INTRODUZIONE

Lo scopo di questo studio è quello di realizzare un termometro digitale gestito da un microcontrollore. La misura della temperatura viene effettuata tramite un sensore digitale. Tale misura viene acquisita da un microcontrollore e, dopo essere adeguatamente convertita e controllata, viene inviata tramite porta seriale ad un computer remoto e ad un display locale. Come microcontrollore viene utilizzato un Pic 16F84 della Microchip e come sensore di temperatura digitale un Dallas Ds 1820. La parte fondamentale di questo studio è quella di implementare, in linguaggio Assembler, un codice per gestire l'acquisizione della temperatura e quindi di convertirla in formato idoneo per poi essere inviata ad un computer tramite porta seriale.

Oltre al circuito descritto di seguito è stato assemblato un programmatore Ludipipo apposito per programmare il microcontrollore Pic 16F84 prima di poter essere utilizzabile dal circuito.

Il fine di tale studio è di progettare e costruire un circuito digitale indipendente ed in grado di comunicare con un computer remoto.

La parte fondamentale dello studio è la programmazione del microcontrollore che è il cuore pulsante del circuito.

Nella realizzazione tecnica e nelle scelte operate per arrivare a conclusione si è puntualizzato sul fatto che questo studio è puramente scolastico, quindi per mezzo del quale applicare conoscenze studiate durante i corsi teorici e per imparare a familiarizzare con gli strumenti da laboratorio.

Nella scelta dei componenti è stato un fattore importante il costo dei vari componenti, infatti si è puntato sui componenti meno costosi a parità di prestazioni, perché lo scopo è di arrivare ad un termometro digitale semplice, utile e poco costoso, cioè in linea con i termometri digitali attualmente in commercio.

In appendice è stata sviluppata una possibile realizzazione pratica di gestione della temperatura attraverso l'utilizzo di una ventola di raffreddamento e di un allarme sonoro (sirena bitonale).

Gli schemi del circuito riportati di seguito fanno riferimento alla realizzazione completa di termometro digitale e sistema di allarme. Nel caso in cui si voglia realizzare solo il termometro digitale si possono escludere le parti relative al sistema di allarme.

PIC 16F84¹

Il microcontrollore Pic 16F84 della Microchip è a basso costo, grandi prestazioni, tecnologia CMOS, completamente statico e a 8 bit. Ha un'architettura RISC. Ha bus separati per dati e per le istruzioni, 14 bit per le istruzioni e 8 bit per il bus di dati. Avendo due linee di pipeline è in grado di eseguire tutte le istruzioni in un singolo ciclo di clock, eccetto per alcune istruzioni che ne abbisognano due. Per la programmazione si usa un linguaggio semplificato composto da solo 35 istruzioni. La frequenza massima di operatività è di 10 MHz, quindi 400 ns per un ciclo. Ha 15 registri, modo diretto e indiretto per l'indirizzamento. Ha 13 porte di input/output programmabili a scelta, ha un contatore programmabile a scelta. E' costituito da 68 byte di RAM e 64 byte di EEPROM. E' un microcontrollore che consuma relativamente poco ed il range di esercizio va da 2 a 6 V; inoltre è dotato di una funzione, SLEEP, che permette di risparmiare potenza, utilizzandolo come fosse in stand by. E' un microcontrollore molto versatile e permette di fare 1000 scritture nella EEPROM, 1000000 cicli di scrittura di dati nella EEPROM ed inoltre può vivere anche più di 40 anni. La piedinatura del Pic 16F84 è riportata nel diagramma sopra.

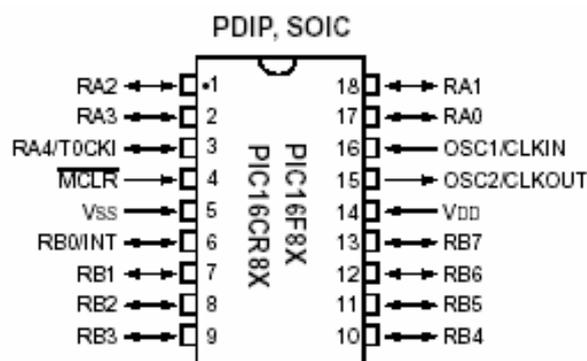


Fig. 1: Diagramma del Pic 16F84.

¹ Rif. [Data Sheet Pic 16F84]

I pin RA0-RA4 sono le linee della porta A configurabili sia come ingresso che come uscita.

I pin RB0-RB7 sono le linee della porta B configurabili sia come ingresso che come uscita.

MCLR è il master reset, utile per resettare il Pic a livello Hardware (normalmente a livello +Vcc, si porta a livello 0 per il reset).

Vss alimentazione positiva da 2 a 6 V.

Vdd massa

RB0 è il pin che in programmazione può essere utilizzato per gestire gli interrupt esterni.

OSC1 e Osc2 pin utilizzati per il clock tramite quarzo oppure come rete R_c.

CLKIN può essere utilizzato per inviare un clock esterno

RB6 e RB7 vengono utilizzati in programmazione rispettivamente come data e come clock.

La scelta del microcontrollore è ricaduta sul pic 16F84 perché è un microcontrollore a basso costo, reperibile, facilmente implementabile e con un numero di ingressi e uscite adeguate per il progetto. E' molto versatile ed inoltre il Pic 16F84 ha un basso consumo e potrebbe essere alimentato da una batteria esterna.

PROGRAMMATORE LUDIPIPO

Il Ludipipo è un semplice programmatore del Pic. Ludipipo è l'acronimo di LUDI's Pic PrOmmer, Ludi dal nome del suo inventore, Ludwig Catta. Lo schema è facilmente reperibile su internet². I Pic sono predisposti per l'accesso ai dati in memoria in modo seriale. Un pin è necessario per il clock e un pin per il transito dei dati, che possono essere bidirezionali. Il programma compilato è normalmente disponibile nel PC come file ed è sufficiente inviarlo al Pic nell'apposita area di memoria per eseguire la programmazione. Per semplificare il processo si usa la porta seriale che dispone dei segnali di comunicazione seriale, l'unica difficoltà è che il Pic funziona a + 5V, la seriale a ± 12 V. Questo problema è risolto inserendo alcune resistenze ed alcuni Zener per limitare i segnali al dovuto livello. Per la connessione al PC si utilizza un cavo seriale maschio femmina PinToPin. Il programmatore agisce unicamente sui Pin: Vcc, Vdd, Mclr, Data ed è quindi idoneo per tutti i Pic con programmazione InCircuit, purchè vengano rispettate le connessioni elettriche.

Di seguito vengono riportati lo schema del circuito e lo schema PCB, oltre alla lista dei componenti.

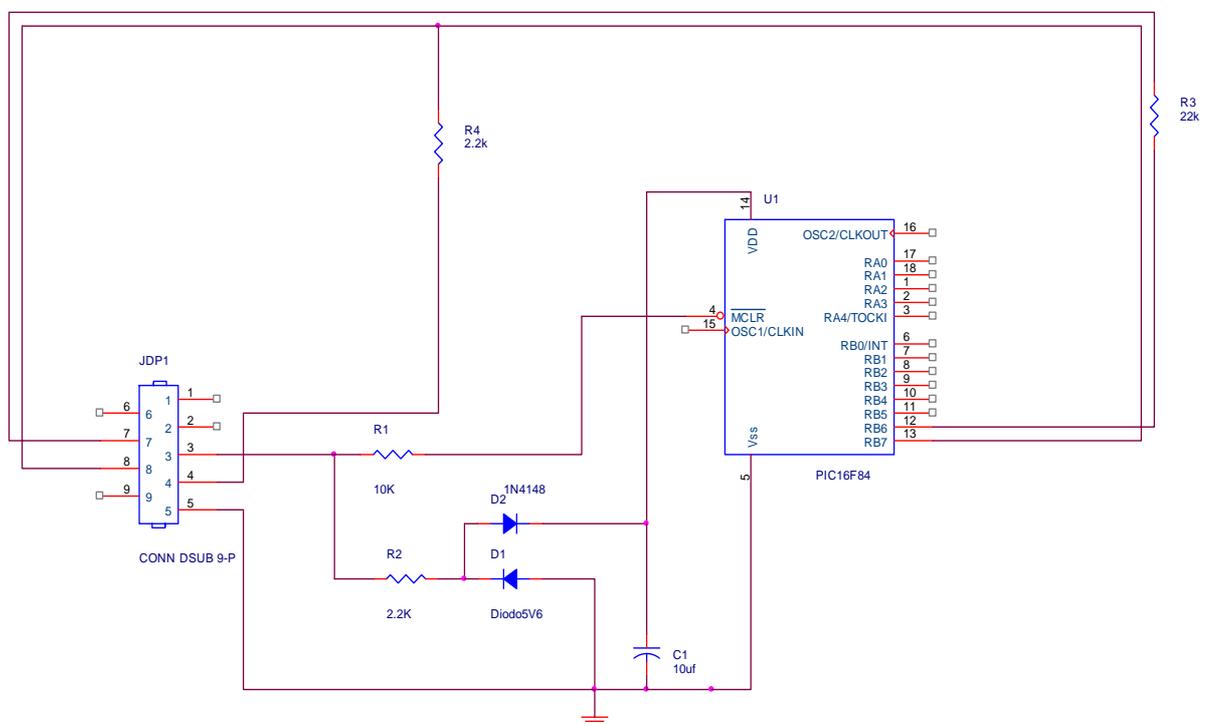


Fig. 2: Schema del programmatore Ludipipo.

² Rif [programmatore Ludipipo]

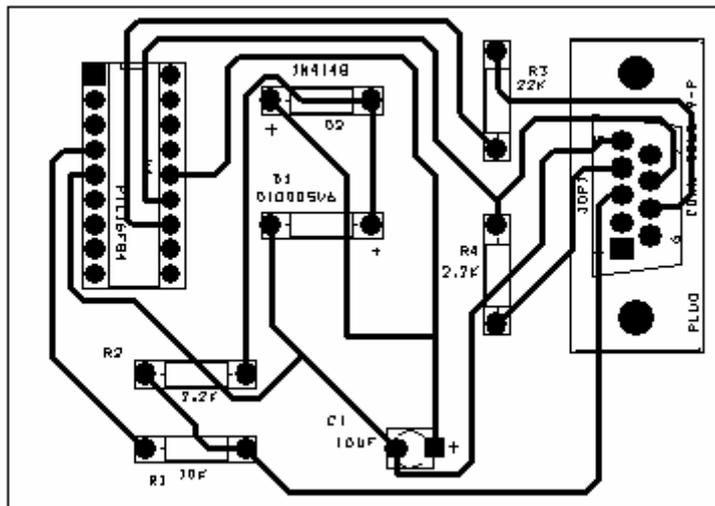


Fig. 3: Schema PCB up del programmatore Ludipipo

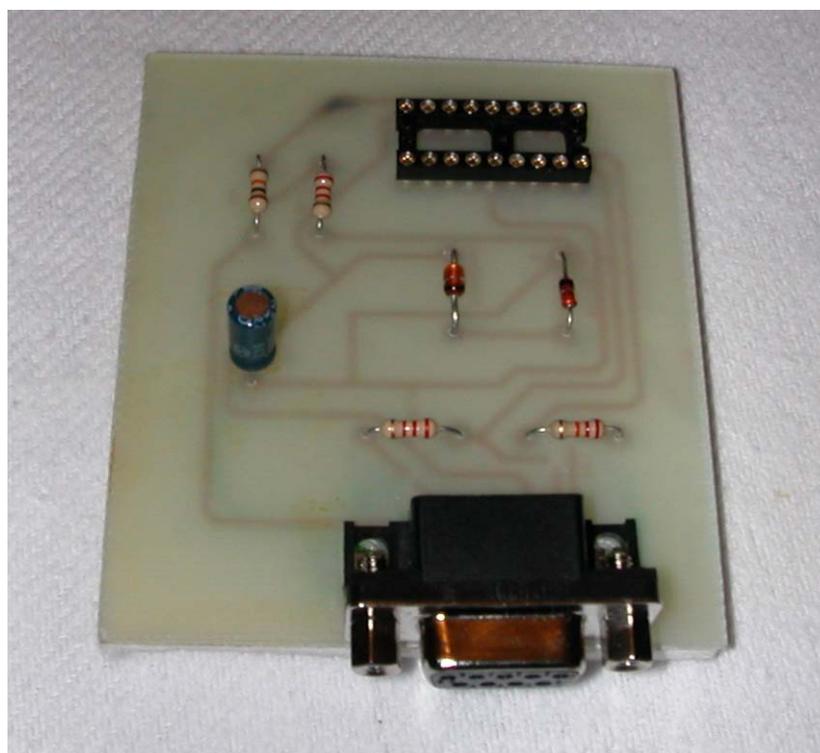


Fig. 4: Fotografia del programmatore Ludipipo realizzato.

I componenti utilizzati sono i seguenti:

R1	2.2K
R2	22K
R3	2.2K
R4	10K
C1	10 μ F
D1	1N4148
DZ1	5.6V Zener
J1	SUBD-9 Femmina
U1	18-pin socket for the PIC

Il software utilizzato per la programmazione del Pic è ICPROG³. E' un software gratuito, reperibile nella rete e facilmente intuibile nell' uso. Il software è dotato di alcune utility per la verifica del programmatore. Il software funziona con i sistemi operativi Win200/NT/XP; si deve ricopiare nella stessa cartella del Pic il driver icprog.sys nelle impostazioni del Pic; su Misc si deve attivare Enable NT/2000/XP Driver e le funzionalità sono operative solo accedendo come amministratore. Una volta avviato il programma si settano alcune funzioni: tipo di Pic, oscillatore, porta seriale. Dal menù File si apre il listato in formato .HEX, dal menù setting si sceglie Program All e si apre una finestra in cui si vede progressivamente lo stato di programmazione; a questo punto il Pic è programmato e pronto all'uso.

³ Rif [software programmatore]

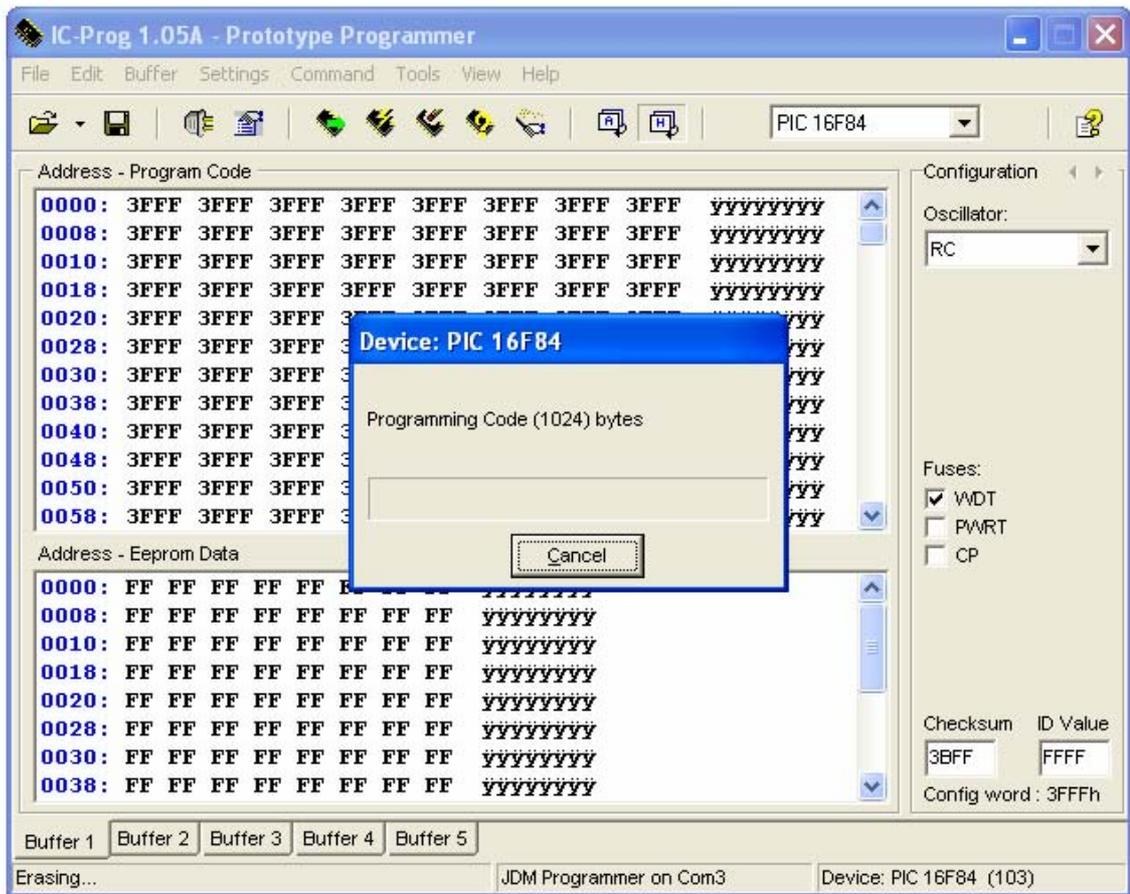


Fig. 5: Stato di programmazione di ICPROG.

microcontrollore e non ha bisogno di driver o connessioni particolari con il Pic16F84.

Per la connessione seriale tra Pic e Computer si è optato per un collegamento tramite porta RS232. Il segnale di uscita dal Pic 16F84 è un segnale di tipo TTL e deve essere trasformato da TTL a segnale utilizzabile dalla porta seriale RS232. Come Driver tra microcontrollore e porta seriale è stato scelto il Drivers Receivers Max232 della Maxim⁵, o qualsiasi altro Max232 generico. Tale semiconduttore integrato a 16 Pin è alimentato a +5 V, di basso consumo ed è adatto ad un sistema alimentato da batterie. La sezione ricevente del MAX232 è costituita da due porte invertenti che accettano in ingresso una tensione di $\pm 12V$ (o altra tensione compatibile allo standard RS232) ed in uscita hanno un segnale TTL compatibile. Per ottenere la tensione positiva e negative necessarie per il funzionamento dell'integrato è usata una configurazione a pompa di carica, costituito da circuiti interni all'integrato e quattro condensatori esterni da 1 uF. La capacità effettiva dipende dal tipo di integrato e dalla relativa frequenza di commutazione; a volte i condensatori sono presenti all'interno dell'integrato stesso.

Il limite dei circuiti a pompa di carica è la limitata quantità di corrente disponibile: infatti se si preleva corrente da C7 questo tende a scaricarsi, facendo scendere la tensione; la corrente generata da un circuito integrato tipo Max232 è generalmente tutta utilizzata per il solo funzionamento del driver e quindi non è disponibile per altri circuiti.

La sezione trasmittente ha due driver invertenti con in ingresso una logica TTL compatibile e capaci di erogare a vuoto una tensione di poco meno di $\pm 10V$, compatibile con lo standard RS232.

La scelta è ricaduta su tale componente perché facilmente reperibile, largamente usato ed inoltre non abbisogna di driver specifici per il microcontrollore.

Per dotare il circuito di un visualizzatore della temperatura è stato scelto un display a 7 segmenti a catodo comune a singola cifra. Due display vengono utilizzati per visualizzare la temperatura misurata dal sensore in doppia cifra, cioè le unità e le decine. Il terzo display è utilizzato per visualizzare la scala di misurazione cioè °C; per ottenere ciò il display è collegato all'alimentazione tramite una resistenza di valore opportuno per raggiungere un livello di luminosità paragonabile a quella degli altri due display. Sono stati collegati solo i pin necessari per ottenere il simbolo desiderato. I display sono pilotati dalla tensione che viene fornita in uscita dai piedini del 4511. La scelta è ricaduta su display SC 56-11 EWA⁶ perché reperibili in commercio e perché poco costosi. Per interfacciare i display a sette cifre con il microcontrollore è

⁵ Rif. [data sheet Maxim Max232]

⁶ Rif. [data sheet SC56-11EWA]

stato necessario utilizzare un driver. La scelta è ricaduta sul 4511 latch/decoder/driver della Philips⁷. Tale integrato necessita di solo quattro ingressi digitali per poter pilotare un display a sette segmenti. Il 4511 evita di usare molte porte del Pic per accendere il display e rende più facile la programmazione in quanto esso ha al suo interno un codice che accetta in ingresso 4 pin di 0 o 1 e in uscita dà una sequenza con relative impedenze per far accendere il display. La scelta è ricaduta su tale integrato perché il numero delle uscite del Pic 16F84 a disposizione è limitato ed inoltre perché a basso costo e facilmente reperibile.

Il Pic ha una frequenza di 4 MHz fornita esternamente da un quarzo a 4MHz. Il Pic 16F84 comunica con la porta seriale con un protocollo a 9600 baud, 8 bit più un o di start e parità nulla.

La tensione del circuito è di +5 V, ottenuta grazie all'impiego di un integrato della famiglia LM78XX, in questo caso è stato scelto un LM7805 data la necessità di pilotare il circuito alla tensione di 5 V. La tensione di alimentazione può variare da un minimo di 6 V ad un massimo di 12 V.

lista dei componenti del circuito

1	R1	270	¼ W
1	R2	10 K	¼ W
1	R4	4,7 K	¼ W
1	R5	39	½ W
1	R6	27	1 W
5	C3-C4-C5-C6-C8	1µF	Elettrolitici
1	C7	10 µF	Elettrolitici
2	C1-C2	22pF	
1	Y1	4Mhz	Cristallo
1	PIC16F84		
2	4511		
3	SC56-11 EWA		Display
1	MAX232		
1	SUB-D connector 9 pins		Femmina
1	DS1820		Sensore
2	J1-J3	CON 2	
1	LM7805		
1	Zoccolo 18 pin		
3	Zoccolo 16 pin		

PROGRAMMAZIONE

⁷ Rif. [data sheet 4511]

La fase più importante del progetto e più onerosa nel senso della tempistica è la fase di programmazione. Essa consiste nel programmare il microprocessore affinché gestisca funzionalmente il sensore digitale e la comunicazione seriale. Il linguaggio di programmazione utilizzato è l'assembler proprio del Pic 16F84, composto da 35 istruzioni. Il software utilizzato per la programmazione è il MPLAB IDE⁸ della Microchip. Tale programma è scaricabile gratuitamente da internet ed è facilmente utilizzabile. E' dotato di un editor per la scrittura. Il file deve essere salvato in formato .ASM. Il debugger compila il listato e crea un file di caratteri esadecimali .HEX, il quale sarà trasferito al microcontrollore. Inoltre il debugger genera un altro file dove vengono visualizzati gli errori durante la compilazione e gli eventuali warning. Una volta compilato il file e scelto il tipo di simulatore desiderato si può testare il listato facendolo simulare. Si possono vedere i passi da una riga all'altra, i registri utilizzati del Pic, le locazioni di memoria, le istruzioni utilizzate e le varie funzioni in serie. In alternativa a questo software si può scrivere il codice con un semplice editor di testo e salvare il file in formato .ASM oppure .TXT. Si può utilizzare il software MPASM⁸ della Microchip per generare il codice in formato .HEX. Dal listato in formato .HEX si utilizza il software ICPROG e si programma il microcontrollore. Di seguito è riportato un semplice schema a blocchi del programma realizzato.

⁸ Rif. [software programmazione Pic]

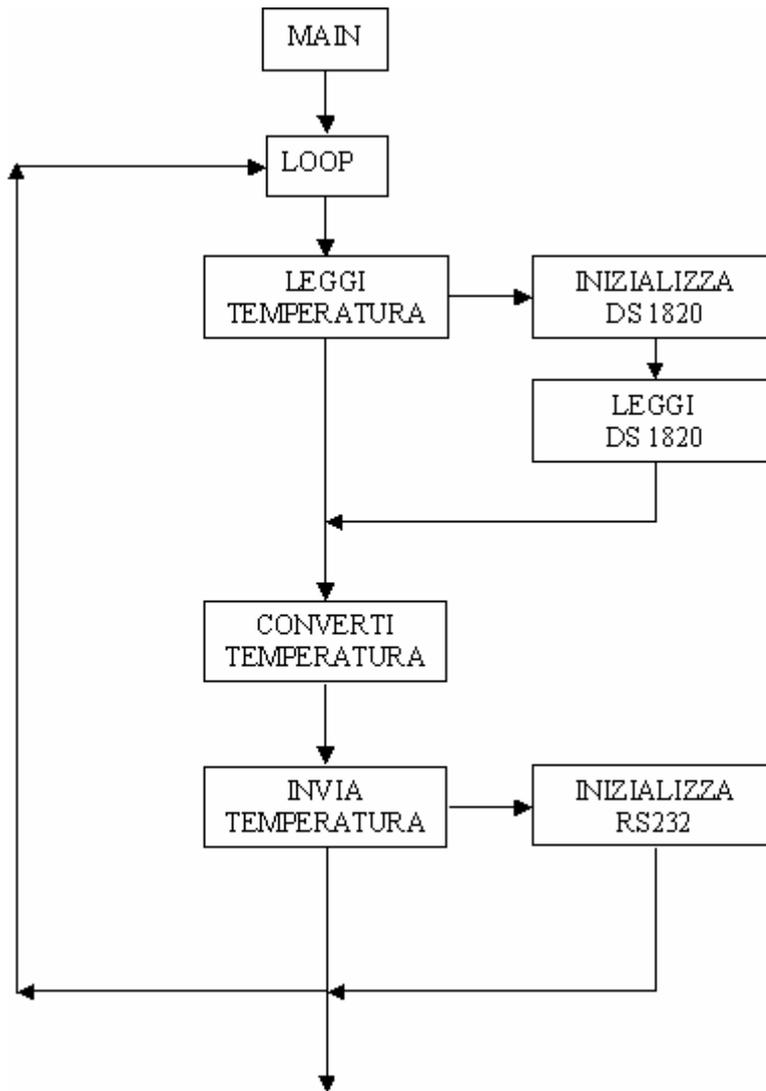


Fig. 7:Schema di programmazione.

La routine main ha il compito di inizializzare il Pic 16F84, adeguando le varie porte di input ed output e settando i vari registri utilizzati. Si deve inizializzare anche la frequenza del microcontrollore e il baud rate per la trasmissione con la porta seriale. La procedura Loop è il corpo centrale del programma; infatti il termometro digitale ogni 1 secondo misura la temperatura, la converte in formato ASCII e la invia tramite porta seriale al computer remoto. Tale procedura ha il compito di gestire il ciclo continuo di misura della temperatura, conversione e invio attraverso la RS232 chiamando di volta in volta la procedura idonea. Prima di poter leggere la temperatura, il Pic deve inizializzare il sensore di temperatura, quindi utilizzare il pin desiderato come ingresso e scrivere nel registro corretto la serie di bit che compongono la temperatura. Un problema sorto in questo caso è che il sensore di temperatura lavora a 9 bit, mentre il Pic lavora a 8 bit. Usando un comando proprio di conversione (0x44) del sensore DS 1820 durante l'inizializzazione, si può convertire la temperatura in formato a 16 bit: 8 per la misura in gradi centigradi e 8 per il segno. Tale conversione

avviene nel sensore stesso e viene inviata al microcontrollore la temperatura in formato di 16 bit. Nel Pic i bit per la misura vengono memorizzati in un registro e quelli per il segno in un altro. A questo punto si deve fare una conversione successiva di tipo software per dividere la misura in segno, centinaia, decine, unità e frazione decimale. Una volta fatto ciò si invia al computer remoto ed al display locale. La trasmissione attraverso la porta seriale avviene a 9600 simboli al secondo, con una risoluzione di 8 bit di dati, 1 bit di start e controllo di parità nullo. La routine di invio attraverso la porta seriale avviene inizializzando la porta, cioè specificando il protocollo utilizzato. Dopo il bit start si inviano gli 8 bit che devono essere trasmessi. Con il Pic 16F84 è possibile inviare anche dei caratteri direttamente in formato ASCII, permettendo quindi una ampia scelta di opzioni.

Le routine di attesa (delay) sono state implementate tenendo conto dei cicli di clock che ha ogni istruzione. Infatti, contando il numero di istruzioni e sapendo che ogni ciclo di clock necessita di 1 μ s, si possono creare dei tempi di attesa desiderati. In modo molto semplice si può creare una routine di attesa mettendo nel registro di lavoro W un numero utile e facendo un ciclo di decrementi si riesce a creare un tempo adeguato. Per visualizzare meglio la temperatura è stata fatta una routine di scrittura: prima di ogni misurazione viene visualizzata la dicitura "TEMPERATURA: " implementata direttamente in carattere ASCII. Nel computer remoto è possibile vedere la temperatura misurata con un qualsiasi programma di comunicazione. In questo studio è stato utilizzato Hiper Terminal⁹. Dal menù di avvio del sistema operativo si lancia Hiper Terminal dalla cartella comunicazioni. Dopo aver inserito un nome di connessione, si deve selezionare la porta di comunicazione utilizzata (generalmente COM1) e si devono inserire i parametri necessari, cioè 9600 bit al secondo, 1 bit di start e parità nulla; inoltre il controllo di flusso può essere selezionato "nessuno". A questo punto la connessione si attiva e nell' editor appaiono la stringa di caratteri inviati dal microcontrollore. Possono essere utilizzate anche alcune funzioni di Hiper Terminal quali il salvataggio in file esterno dei dati o l'invio diretto dei caratteri ad una stampante remota.

Di seguito è riportata una schermata tipo di Hiper Terminal durante la trasmissione con il microcontrollore.

⁹ Software incluso nel sistema operativo Microsoft 9x/XP.

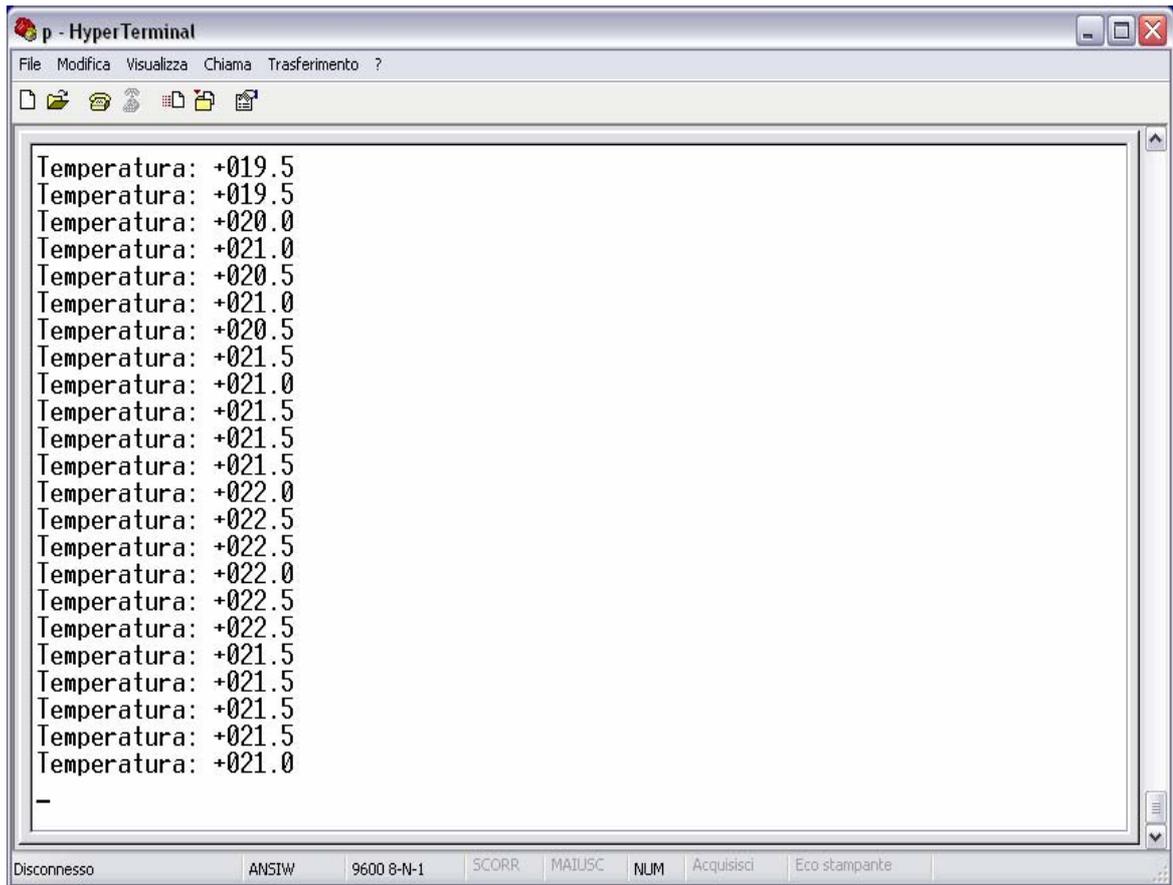


Fig. 8: Videata di trasmissione Hiper Terminal.

Schemi PCB del circuito

Di seguito sono riportati gli schemi PCB realizzati con il software ORCAD Release 9.0.

Prima di iniziare la realizzazione del circuito stampato occorre disegnare lo schema. Uno dei programmi che permette di fare ciò è OrCad Capture; in teoria potrebbe essere utilizzato qualunque altro programma che supporti uno dei formati richiesti da OrCad Layout ma, in pratica, questa è la scelta più razionale sia come costi (il pacchetto Capture + Layout ha sostanzialmente lo stesso costo del solo Layout) che come compatibilità. È anche possibile l'utilizzo di OrCad Express.

Una volta disegnato il circuito, dopo aver caricato la Netlist¹⁰ si utilizza il Capture Layout per disegnare lo schema PCB.

Nella realizzazione di questo circuito si devono rispettare alcune misure relative alla dimensione delle piste, di solito misurate in mills, e della distanza tra le piste stesse. Inoltre vanno aumentate le dimensioni dei piedini dei vari componenti. Questo è stato necessario in quanto il circuito è stato realizzato con tecnologia artigianale.

¹⁰ Rif. [Tutorial Orcad]

Di seguito sono riportati gli schemi PCB del circuito realizzato. Non è stato possibile realizzare un circuito a singola faccia dato il numero elevato di collegamenti, quindi è stata scelta l'opzione a doppia faccia. Il primo è uno schema completo e gli altri due sono il Top ed il Bottom. Il Top rappresenta la parte inferiore del PCB, mentre il Bottom la parte superiore. Gli schemi PCB sono stati disegnati con la parte relativa all'appendice. Nel caso del solo termometro digitale si tralasciano le parti eccedenti.

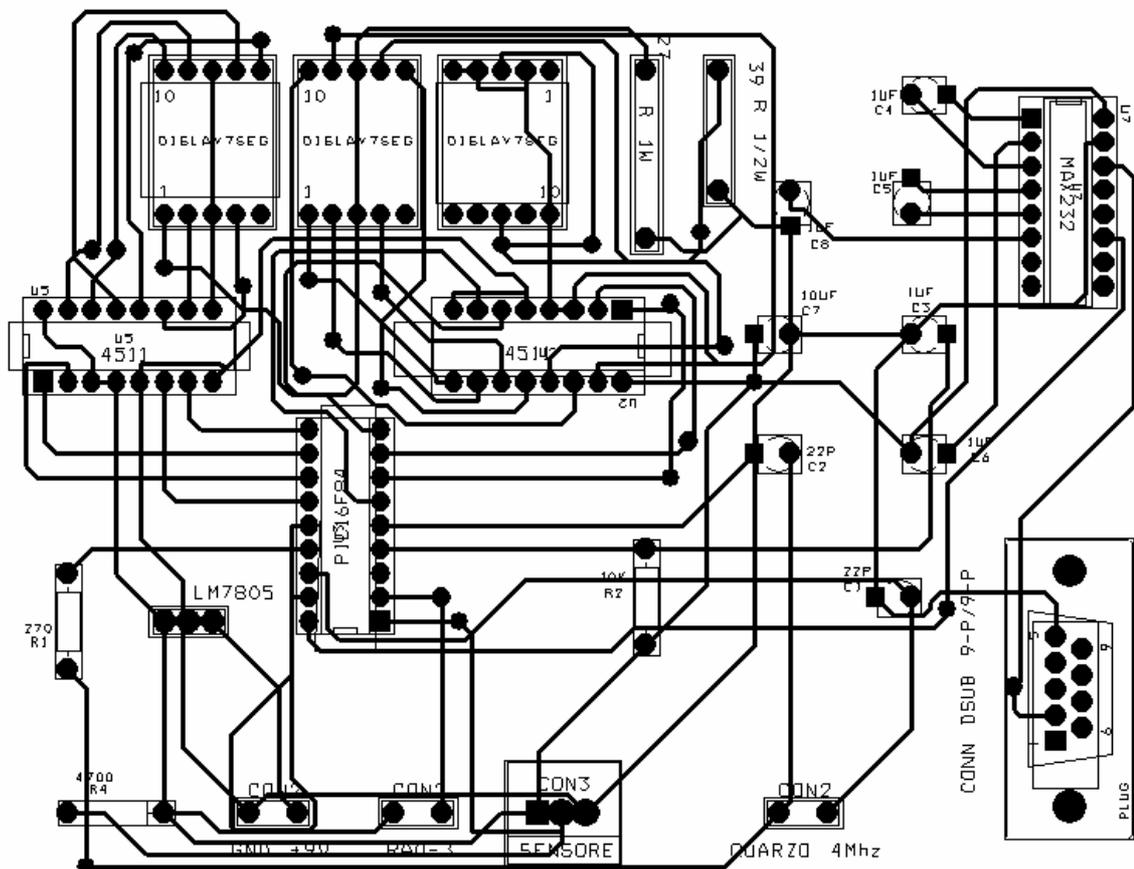


Fig. 9: Schema PCB doppia faccia

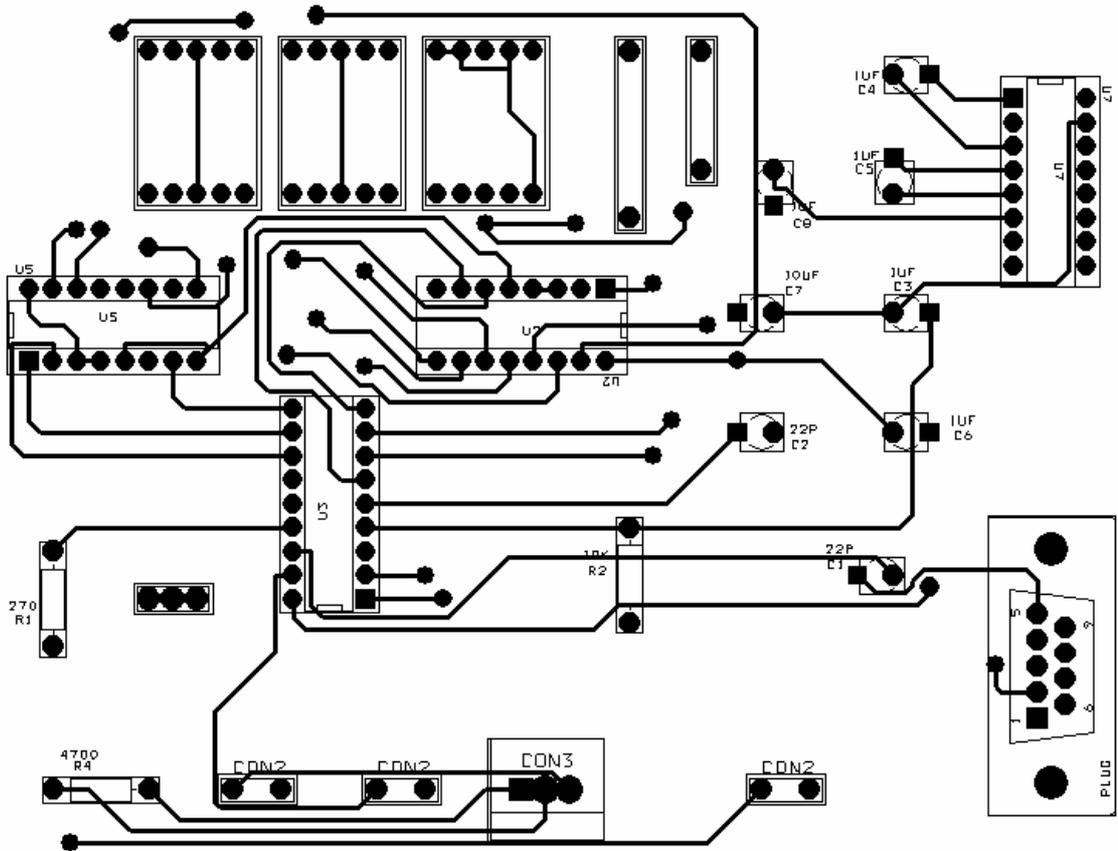


Fig. 10: Schema PCB Top

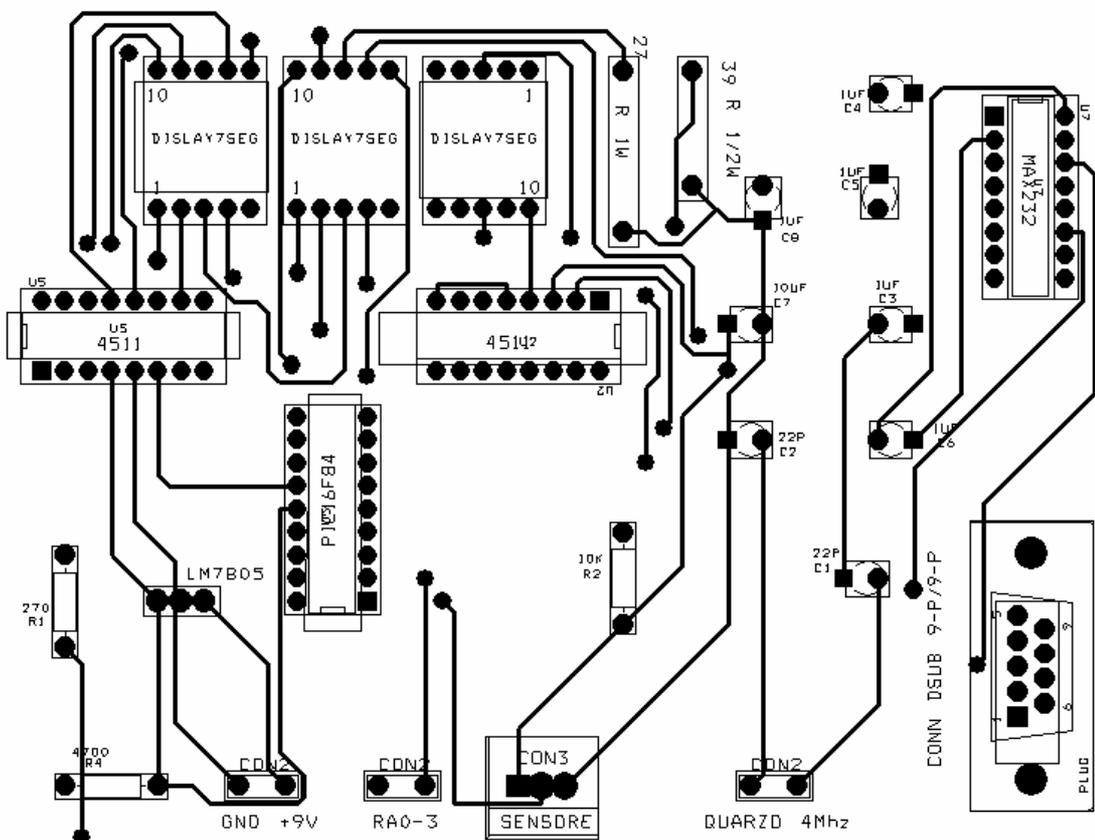


Fig. 11: Schema PCB Bottom

CONCLUSIONI

Lo studio è stato caratterizzato dalla difficoltà di progettare il circuito perfettamente funzionante. Dapprima si è dovuto scegliere i vari componenti in base ai presupposti dichiarati ed alla compatibilità degli stessi. In seguito è risultata impegnativa la programmazione del microcontrollore in quanto la programmazione Assembler richiede la conoscenza delle singole istruzioni e dei registri interessati da tali istruzioni.

Lo studio ha richiesto molto tempo nella fase di programmazione e nella fase di verifica e test del circuito.

Lo studio è stato ampiamente sviluppato e risolto. Tale studio potrebbe essere un punto di partenza per un ulteriore progetto: tale termometro potrebbe essere utilizzato per la misura della temperatura in un posto remoto, tipo le centraline di rilevazione delle società metereologiche in comunicazione con la centrale attraverso un modem o con telefoni cellulari. Oltre al termometro potrebbe essere aggiunto al sensore di temperatura anche un sensore di umidità, un igrometro inviando i vari dati ad una centrale remota tramite lo standard RS232.

Il termometro digitale potrebbe essere utilizzato come termostato per un ambiente o per un macchinario particolarmente delicato; infatti si potrebbe programmare il microcontrollore in modo tale che ad una certa temperatura venga emesso un suono tipo allarme, oppure venga pilotata una ventola di raffreddamento; le applicazioni di un termometro sono notevoli.

A corollario di questo studio si riportano i costi sostenuti per realizzare il circuito.

Componenti e costi:

1	R1	270 $\frac{1}{4}$ W	0,24 €
1	R2	10 K $\frac{1}{4}$ W	0,24 €
1	R4	4,7 K $\frac{1}{4}$ W	0,24 €
1	R5	39 $\frac{1}{2}$ W	0,35 €
1	R6	27 1 W	0,35 €
5	C3-C4-C5-C6-C8	1 μ F	0,18 € X1
1	C7	10 μ F	0,34 €
2	C1-C2	22pF	0,23 € X1
1	Y1	4Mhz	1,44 €
1	PIC16F84		10 €
2	4511		0,71 €
3	SC56-11 EWA		1.29 €
1	MAX232		1,85 €
1	SUB-D connector 9 pins		2,55 €
1	DS1820		11, 52 €
2	J1-J3	CON 2	0.54 €
1	LM7805		0,60 €
1	Zoccolo 18 pin		0,33 €
3	Zoccolo 16 pin		0,26 €
1	R1	2.2K $\frac{1}{4}$ W	0,24 €
1	R2	22K $\frac{1}{4}$ W	0,24 €
1	R3	2.2K $\frac{1}{4}$ W	0,24 €
1	R4	10K $\frac{1}{4}$ W	0,24 €
1	C1	10 μ F	0,34 €
1	D1	1N4148	0,05 €
1	DZ1	5.6V Zener	1,50 €
1	J1	SUBD-9	2,55 €
1	U1	Zoccolo 18 pin	0,33 €
		Totale	26,77 €

APPENDICE: SISTEMA DI CONTROLLO E ALLARME DI TEMPERATURA.

Per completezza di questo studio si è voluto sviluppare un esempio dei possibili impieghi di questo rilevatore e visualizzatore di temperatura, tra diversi progetti è stato preso in considerazione un sistema di controllo e allarme di temperatura.

Questo consta nell' utilizzo del termometro come oggetto che tiene sotto controllo una temperatura prefissata dal programmatore, se per qualche motivo la temperatura da mantenere entro certi limiti varia, il dispositivo mette in funzione una ventola che deve riportare la temperatura entro i livelli predefiniti.

A questo punto si è pensato di inserire un'ulteriore precauzione nel sistema che nel caso in cui la ventola per diversi motivi non riporti la temperatura ai livelli desiderati si inneschi un sistema di allarme sonoro creato attraverso una sirena bitonale.

Le modifiche per ottenere questo dispositivo partendo dal termometro sono quelle di avere disponibili due ingressi I/O per pilotare la ventola e la sirena d'allarme.

Naturalmente non potendo comandare questi dispositivi direttamente con le uscite TTL dobbiamo usare degli amplificatori che faranno da intermediari tra i due.

Quindi abbiamo scelto di usare due BC337 come stadio di amplificazione, e li faremo lavorare come interruttori sfruttando il fatto di mandarli attraverso una rete di polarizzazione adatta dall' interdizione alla saturazione, inoltre osserviamo che l'alimentazione dei dispositivi aggiunti è di 9 volt, da cui si è scelto di usare LM7805 per alimentare il termometro in modo da avere un'unica alimentazione per il circuito complessivo.

Per quanto riguarda la ventola non c'è molto da dire si è usata una semplice ventola per personal computer che lavora alla tensione di 9 volt, e non richiede un'eccessiva corrente di spunto per essere pilotata.

Mentre per quanto riguarda la realizzazione dell' allarme sono stati impiegati per il circuito due comunissimi integrati NE.555, usati come multivibratori.

Il primo NE.555 oscilla ad una frequenza di pochi Hertz; questa frequenza viene usata per pilotare il secondo integrato che oscilla facendo i calcoli a circa 5000 Hz.

In questo modo si ottiene un segnale modulato, molto efficace come segnalatore acustico bitonale.

Volendo modificare la frequenza della modulazione, si potrà variare la capacità del condensatore elettrolitico C1, tenendo presente che riducendo il suo valore aumenterà la frequenza.

Il segnale modulato che esce dal piedino 3 del secondo NE.555 viene applicato sulla base del transistor di tipo darlington BDX.53C, che, amplificandolo, permetterà all'altoparlante collegato al suo collettore di emettere un suono di adeguata potenza.

In fine come altoparlante si è utilizzato un tweeter per auto senza connettere il condensatore di solito usato per eliminare le basse frequenze, così otteniamo un suono più potente e assordante, naturalmente può essere utilizzato un qualsiasi altro altoparlante.

Come già accennato in precedenza le modifiche da effettuare per ottenere questo interessante dispositivo non riguardano gli schemi del termometro perché si è provveduto in fase di progetto, quindi basta creare un altro circuito seguendo gli schemi riportati di seguito.

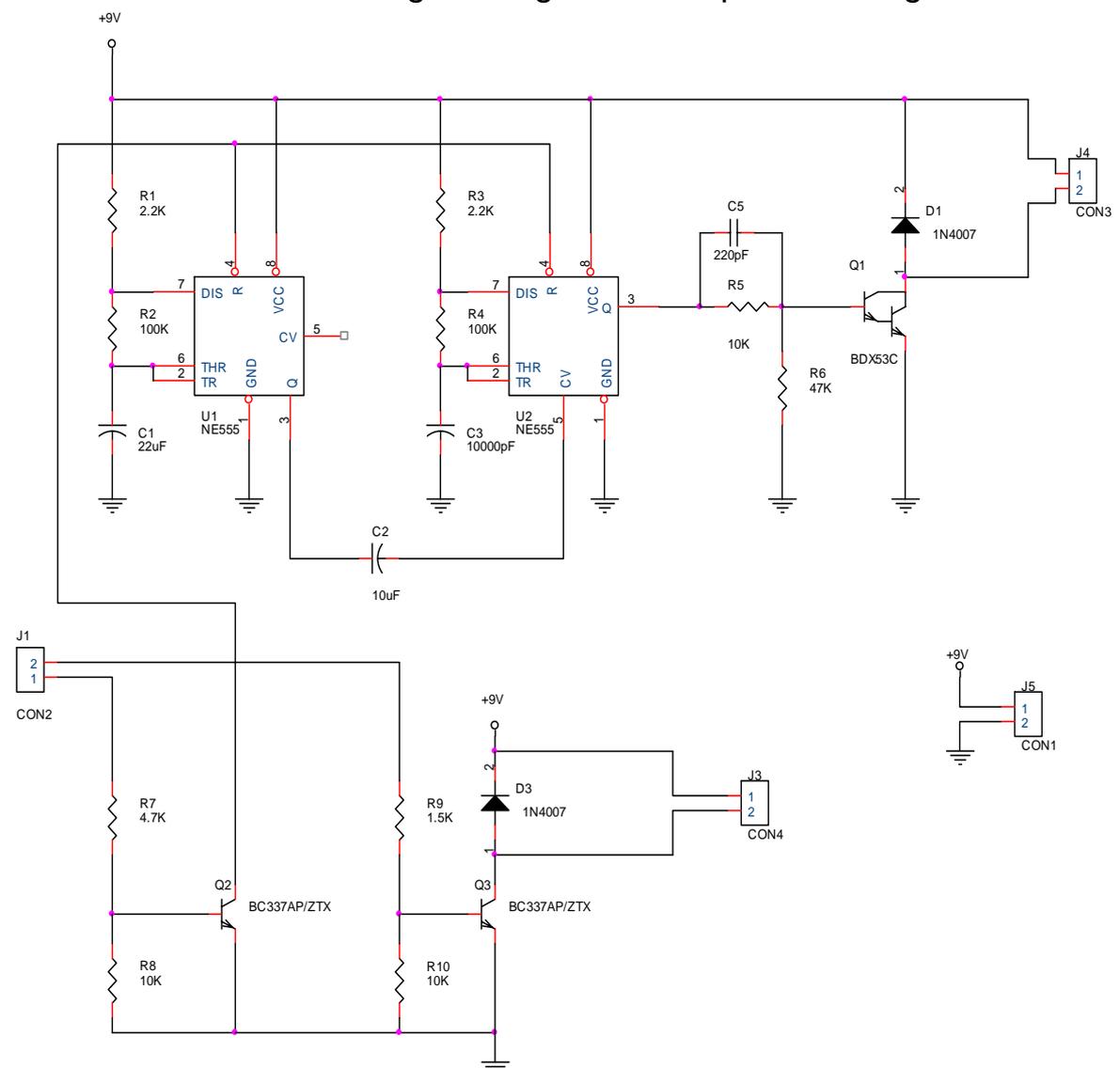


Fig. 12: Schema circuito Allarme.

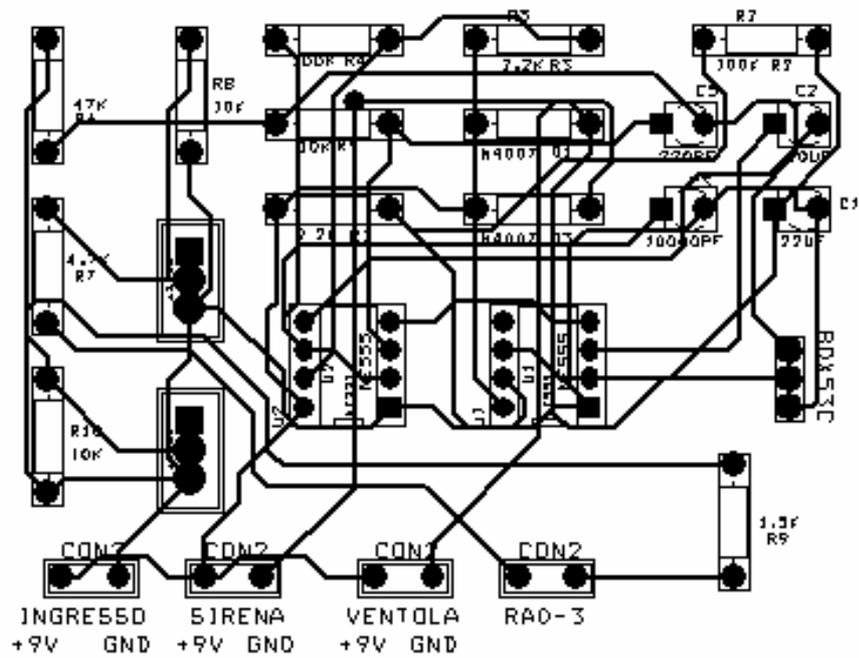


Fig.13 : Schema PCB Allarme.

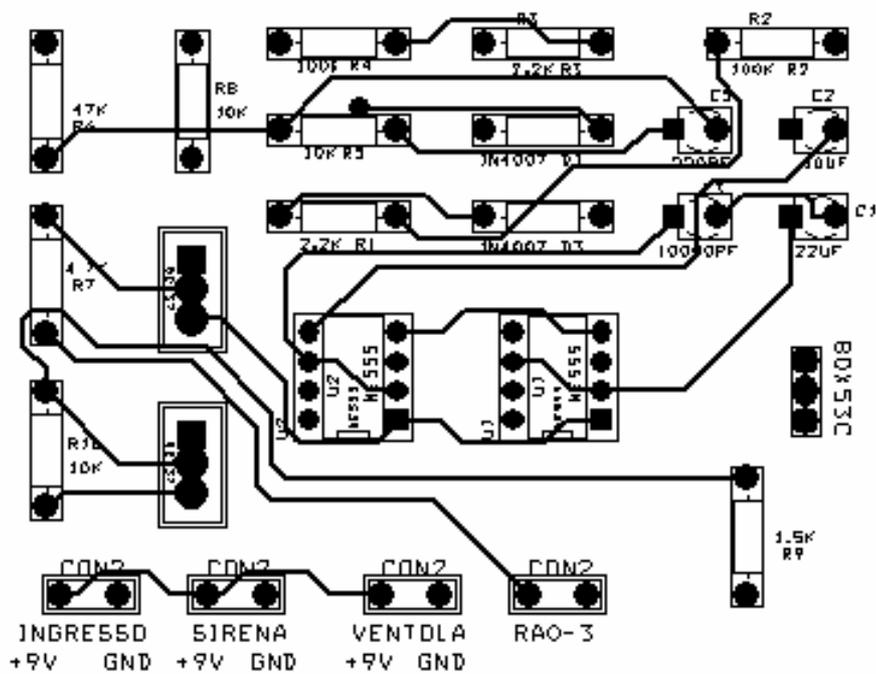


Fig. 14:Schema PCB Allarme Top

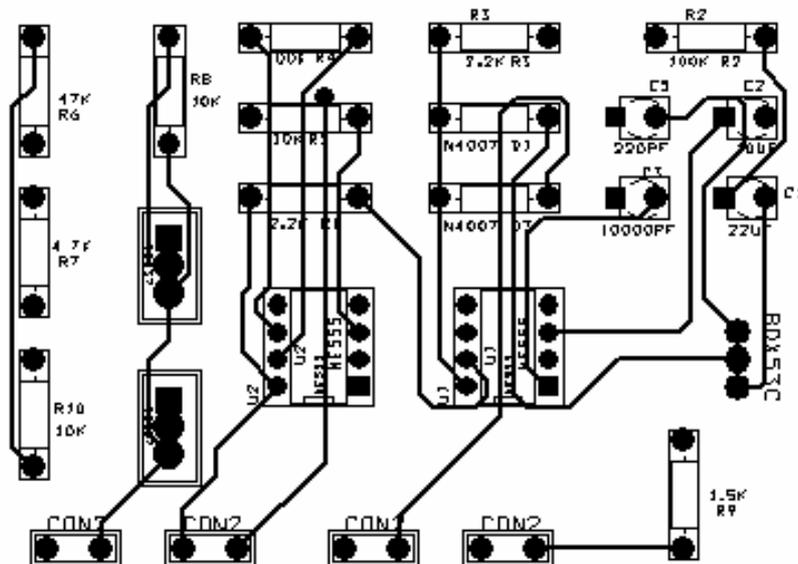


Fig. 15: Schema PCB Allarme Bottom

Elenco componenti.

2	R1-R3	2.2K	¼ W	0.24 €
2	R2-R4	100K	¼ W	0.24 €
1	R5	1K	¼ W	0.24 €
2	R8-R10	10K	¼ W	0.24 €
1	R6	47K	¼ W	0.24 €
1	R7	4.7K	¼ W	0.24 €
1	C1	22mF	Elettrolitico 16V	0.25 €
1	C2	10mF	Elettrolitico 16V	0.23 €
1	C3	10.000 pF	Poliestere	0.44 €
1	C5	220 pF		0.18 €
2	D3-D1	1N4007		0.05 €
1	BDX.53C		Darlington	0.63 €
2	U1-U2	NE.555		0.49 €
4	Connettori	2 Ing.		0.53 €
			Totale	4.72 €

BIBLIOGRAFIA

Driver decoder 4511

<http://www.semiconductors.philips.com/pip/HEF4511BP.html#datasheet>

Display led 7 segmenti

<http://www.kingbright-led.com/data/spec/SC56-11EWA.pdf>

Sensore DS 1820

<http://pdfserv.maxim-ic.com/en/ds/DS1820-DS1820S.pdf>

MAX 232

<http://pdfserv.maxim-ic.com/en/ds/MAX220-MAX249.pdf>

PIC

<http://www.microchip.com/download/lit/pline/picmicro/families/16f8x/30430c.pdf>

Software MPASM

<http://www.microchip.com/1010/pline/tools/archive/other/61/index.htm>

Manuale MPASM

<http://www.microchip.com/download/tools/picmicro/code/mpasm/33014g.pdf>

<http://www.microchip.com/download/tools/picmicro/code/mpasm/30400f.pdf>

Software MPLAB

<http://www.microchip.com/1010/pline/tools/picmicro/devenv/mplabi/mplab6/index.htm>

Manuale MPLAB

<http://www.microchip.com/download/tools/picmicro/devenv/v6xx/v620/51281c.pdf>

ICPROG

<http://www.ic-prog.com/download.html>
download/icprog.chm (Manuale in italiano di ICPROG)

Rs232

<http://www.tanzilli.com>

<http://www.vincenzov.net>

Corso sui pic

http://www.elettronicashop.com/he/PicSheets/contributi_ext/picstartup/index.htm

<http://www.picpoint.com/?cookie=1>

<http://digilander.libero.it/mircose/pic/pic.htm>

Tutorial su Orcad

<http://www.vincenzov.net>