



SVILUPPO DI UN SISTEMA DI RILIEVO E MEMORIZZAZIONE DI GRANDEZZE FISICHE

Introduzione

Era quasi un anno che ci provava. Aveva gli strumenti, ma non i materiali; aveva le conoscenze e le abilità, ma non riusciva ad applicarle.

Finalmente qualche mese fa si decise: mettere da parte la roba vecchia comprata di seconda mano e non funzionante ed acquistare dei pannelli solari nuovi, affidabili, per mettere fine a quella tortura mentale: la voglia di realizzare un sistema di riscaldamento dell'acqua che sfruttasse l'energia solare con l'impossibilità di portare avanti questo progetto.

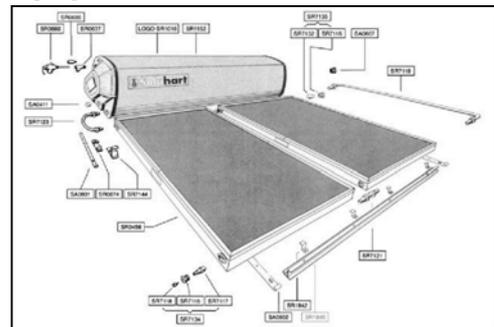
In un mese tutto fu pronto: i componenti ordinati, montati e funzionanti.

Nei giorni di sole adesso si può effettivamente disporre di acqua calda a sufficienza per soddisfare largamente i bisogni di una famiglia.

Le domande che immediatamente dopo si è posto mio padre erano: quale temperatura ha l'acqua all'interno del serbatoio? Come posso fare per conoscerla senza uscire all'aperto per rilevarla?

Grazie alle conoscenze acquisite durante il mio corso di studi, posso cercare di progettare (almeno nei principi di funzionamento fondamentali) un sistema che riesca a rispondere alle 2 domande poste.

Fig 1: pannelli solari



Analisi del problema

È dato un boiler contenente acqua ad una certa temperatura, riscaldata tramite pannelli che sfruttano l'energia solare.

Si chiede di sviluppare un sistema che rilevi la temperatura dell'acqua del tubo di uscita con una tolleranza di massimo 1°C; dopo l'acquisizione, il dato deve essere trasmesso alla distanza di qualche decina di metri per poi essere reso noto all'utente che lo deve leggere da un display. Dovrà esserci perciò una conversione del valore da analogico a digitale per poterlo visualizzare.

Può essere utile anche rilevare la temperatura esterna dell'aria per capire in che misura essa influisca sul sistema di pannelli.

Viene previsto inoltre un sistema informatico in grado di memorizzare automaticamente ad un'ora prefissata i dati delle temperature e di associarli con il valore dell'insolazione ricevuta durante il giorno, che può essere espressa in ore di sole oppure in percentuale che indichi le ore di sole sulle ore totali del dì.

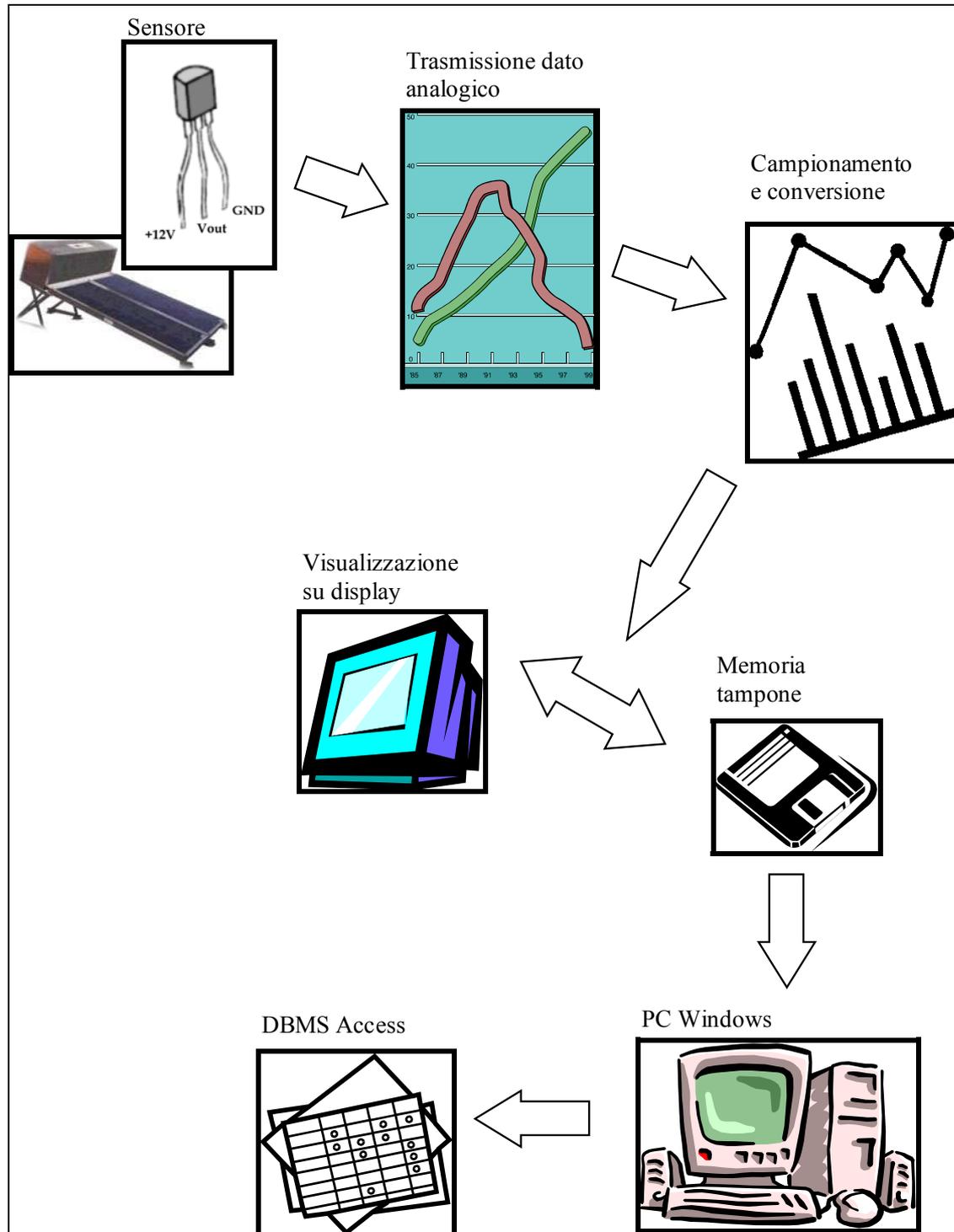
I dati così acquisiti devono essere scaricati su un PC che provveda a mantenere un Data Base che possa essere utilizzato dall'utente a fini statistici.

Con il lavoro che ho svolto non ho l'ambizione di sviluppare un sistema professionale che sia in grado di soddisfare perfettamente tutte le richieste, ma voglio provare ad analizzare alcuni dei problemi che posso incontrare in una progettazione che coinvolga le materie che ho studiato, cosciente che le soluzioni che troverò avranno dei limiti causati dalla mia limitata esperienza da studente.



Schema generale

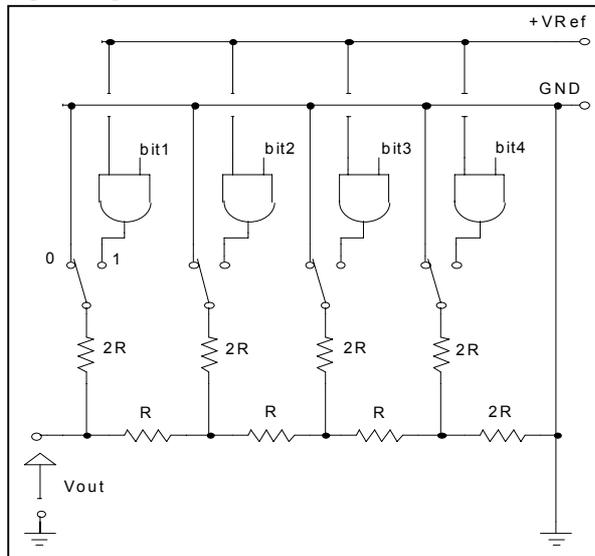
Per il rilevamento della temperatura dell'acqua. Lo stesso schema può valere per i rilievi di insolazione e temperatura esterna, solo si provvederà a dividere i dati in fase di memorizzazione con un Multiplexer.





CONVERSIONE A/D : la conversione del segnale da analogico a digitale serve per poter visualizzare il valore su un display digitale.

Fig 3: esempio D/A R-2R con 4 bit



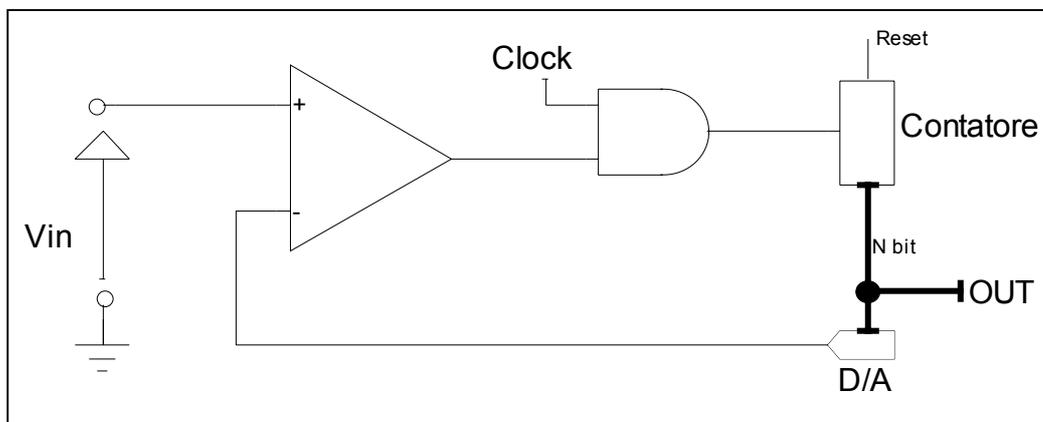
Anche qui ci sono alcune possibili soluzioni adottabili, per esempio potrebbe essere adottabile un A/D che compari la V_{in} con il valore convertito in analogico di un contatore resettabile: quando $V_{in} > V_{out}$, il contatore riporta il valore binario corrispondente all'ingresso.

Per costruire questo convertitore occorrono un contatore ed un generatore di clock.

Inoltre serve uno stadio di conversione da digitale ad analogico, che si può realizzare con un sistema R-2R che permette di utilizzare 2 soli valori di resistenza, uno il doppio dell'altro.

Il convertitore A/D che si serve del contatore, del clock e del D/A (chiamato anche A GRADINATA) ha un difetto considerevole per quanto riguarda il tempo di conversione del dato, molto variabile e più precisamente proporzionale all'ampiezza del valore da convertire.

Fig 4: convertitore A/D a gradinata



Si potrebbe anche optare per un A/D che converte i bit in PARALLELO (chiamato anche A/D FLASH), che offre una digitalizzazione del dato in un tempo prestabilito, con la necessità, però, di una conversione delle uscite in un numero binario tramite una rete combinatoria, difficilmente realizzabile per la sua complessità. In questo caso converrebbe comperare un A/D completo in un integrato.

Bisogna però tener conto di una considerazione molto semplice: la costanza nel periodo di conversione non è un problema fondamentale in un sistema di misura di questo tipo perché: la temperatura non varia così velocemente, perciò 2 campioni successivi avranno circa lo stesso tempo di conversione, che dipenderà dalla temperatura ma sarà pressoché fisso nel breve periodo. Inoltre questa non è un'applicazione per l'audio o per trasmissione radio, il che mi esonera dal ricercare la precisione nella risoluzione di questo problema.

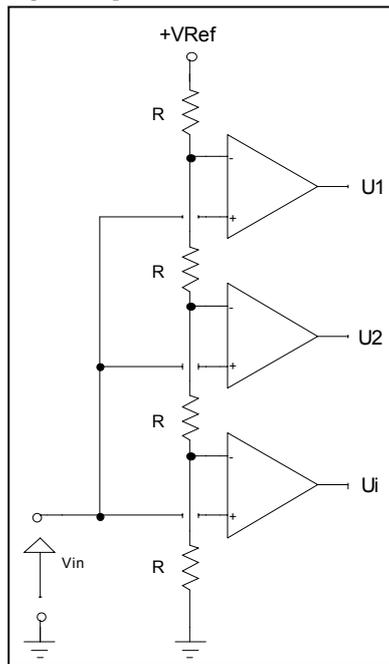


Scelgo quindi di adottare un A/D a gradinata, che inoltre è più semplice sia come concetto di funzionamento che come implementazione.

Per poter gestire l'escursione di temperatura dei sensori, supponendo di sfruttarne tutto il potenziale, occorre che il valore digitale copra il range prefissato di 150°C con almeno 150 valori, in modo da ottenere una tolleranza accettabile di 1°C nella misurazione.

Per avere questo risultato bisogna utilizzare una risoluzione di 8 bit, che fornisce $2^8 = 256$ valori, assicurando al rilevamento della temperatura una tolleranza di $\frac{150}{256} \cong 0,5859^{\circ}\text{C}$.

Fig 5: esempio A/D flash



È necessario quindi un circuito capace di digitalizzare la tensione prelevata dai sensori con 8 bit di risoluzione.

Un numero inferiore di bit porterebbe ad avere tolleranze che superano 1°C , infatti già con 7 bit avrei il range di 150°C coperto da 128 valori.

La tolleranza risultante sarebbe di $\frac{150}{128} \cong 1,1719^{\circ}\text{C}$, troppo alta per l'esigenza già spiegata.

Prima ancora di arrivare al circuito convertitore, il dato analogico deve passare per lo stadio Sample/Hold (letteralmente: mantenimento del campione), un quadripolo che ha la funzione di bloccare il segnale variabile, tra un campionamento e l'altro, ad un certo livello appena acquisito per permettere all'A/D di lavorare con un valore di tensione fisso per tutta la durata della conversione.

Il circuito quadripolare in questione deve poi essere isolato in impedenza dal circuito integrato A/D, che lavora a bassa impedenza di ingresso, e dal sensore che, invece, dà un'uscita ad alta impedenza.

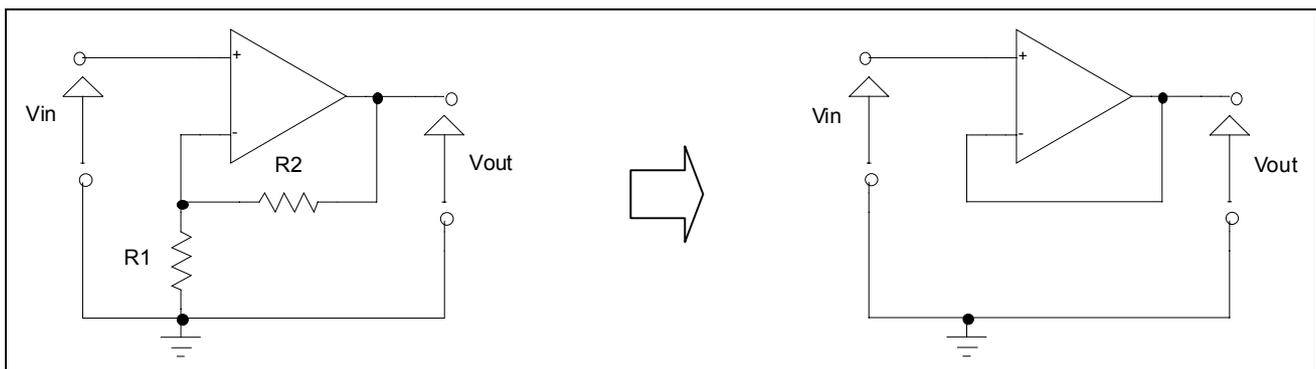
Gli amplificatori operazionali bene si adattano a svolgere questo compito di isolamento, infatti tra le loro principali caratteristiche ci sono:

- Alta impedenza di ingresso
- Bassa impedenza di uscita

La configurazione in cui verranno utilizzati sarà quella di BUFFER NON INVERTENTE (chiamato anche "voltage follower", cioè "inseguitore"), che ha la particolarità di avere un cortocircuito come retroazione negativa e di essere senza resistenza all'ingresso +.

In questo modo, secondo teoria: $- R_2 = R_u = 0$
 $- R_1 = R_i = \infty$

Fig 6: buffer non invertente

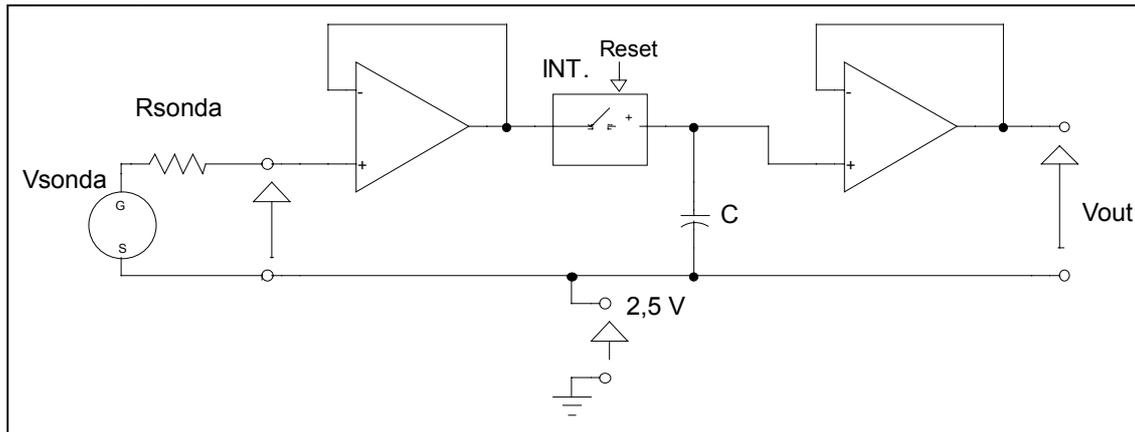




Con questi valori delle resistenze, nella configurazione non invertente ho un guadagno $A = 1 + \frac{R_2}{R_1} = 1 + 0 = 1$, cioè il segnale non viene amplificato, ma rimane il vantaggio intrinseco dell'operazionale sulle impedenze di ingresso e di uscita.

Quindi il circuito risultante per realizzare il S/H è il seguente:

Fig 7: quadripolo S/H



Il blocco denominato "INT." È un interruttore elettronico, che verrà comandato dal convertitore A/D: per tutta la durata della conversione deve rimanere aperto, mentre si chiuderà per prelevare il campione che verrà memorizzato sotto forma di tensione caricando il condensatore al livello di ingresso.

Nonostante il S/H nella conversione del dato di una temperatura non sia indispensabile, in quanto è una grandezza che non varia molto velocemente, credo sia utile inserire questo componente per una maggior affidabilità nel campionamento, ma soprattutto perché è interessante studiarne il comportamento.

La capacità inserita nel circuito deve essere:

- *abbastanza grande* per avere un tempo di tenuta della tensione sufficientemente lungo;
- *abbastanza piccola* per avere un basso tempo di conversione.

In questo caso il tempo di conversione può essere scelto a piacere perché non occorre che il rilevamento della temperatura sia aggiornato molto frequentemente, per 2 motivi:

1- *FREQUENZA* - anche se, per ipotesi assurda, la temperatura variasse significativamente almeno 20 volte al secondo ed il rilevamento seguisse tali variazioni, l'occhio umano non riuscirebbe a percepirle dal display perché l'immagine della retina ha una persistenza di circa 1/16 di secondo e per tale fatto non si ha più la percezione di fotogrammi separati per immagini che variano con frequenza $>$ di 16 Hz.

È su questo principio, inoltre, che si basano le tecnologie di cinema e televisione;

2- *VELOCITÀ* - la tolleranza di $0,6^{\circ}\text{C}$ con 8 bit di risoluzione sta a significare che prima che la temperatura visualizzata sul display cambi di valore, quella effettiva deve subire una variazione pari o maggiore alla tolleranza.

Anche in una giornata estiva di sole cocente, il tempo in cui 2 pannelli solari possono far compiere un'escursione termica di $0,6$ gradi a circa 300l di acqua sarà sicuramente di almeno alcuni minuti!



Devo scegliere il valore della capacità in base alla frequenza di campionamento, che potrà anche non essere molto alta per i motivi già citati. Scelgo 500Hz , che sebbene sia alta rispetto ai 16Hz percepiti dall'occhio, mi permette di prendere il condensatore con un valore di capacità non troppo grande. Esso dovrà avere la costante di tempo T , che influisce sul transitorio di carica, che sia almeno 10 volte minore del tempo di sampling, in quanto il componente è carico pressoché totalmente dopo un tempo di circa 5-6 volte T .

Suppongo che la metà del periodo risultante dalla frequenza di campionamento sia riservata al S/H e l'altra metà al campionamento (del massimo valore) vero e proprio.

Avrò quindi un tempo di sampling: $t = \frac{1}{2} \cdot \frac{1}{500\text{Hz}} = 1\text{ms}$ perciò: $T = \frac{t}{10} = 0,1\text{ms} = 100\mu\text{s}$

Se applico una tensione E ai capi del S/H, quindi del condensatore, la tensione ai capi di C è data dalla relazione: $V_c = E \cdot (1 - e^{-\frac{t}{T}})$.

Dopo 1ms (10 volte T) il transitorio di carica si sarebbe praticamente esaurito, portando il condensatore alla tensione E : $V_c = E \cdot (1 - e^{-10}) = E \cdot (1 - 45 \cdot 10^{-6}) \cong E$

Come si può vedere, quindi, con queste considerazioni lo scarto di V_c da E è trascurabile.

Considerando la R_u dell'OP-AMP di poche decine di ohm (30Ω ca.), la formula per calcolare la capacità occorrente è: $T = R \cdot C \rightarrow C = \frac{T}{R} = \frac{0,1\text{ms}}{30\Omega} = 33\mu\text{F}$

- **VISUALIZZAZIONE** : consiste nel rendere disponibile all'utente tramite un display il dato rilevato.

La visualizzazione della temperatura potrebbe essere realizzata in 2 modi: con un display a led a 7 segmenti oppure con un display a cristalli liquidi.

Nel primo caso bisogna reperire o realizzare un circuito logico combinatorio che gestisca l'accensione / lo spegnimento dei segmenti dei led in funzione del numero binario da rappresentare.

Un esempio è riportato nella prossima figura con 3 led 7-seg. che forniscono una precisione al decimo di grado.

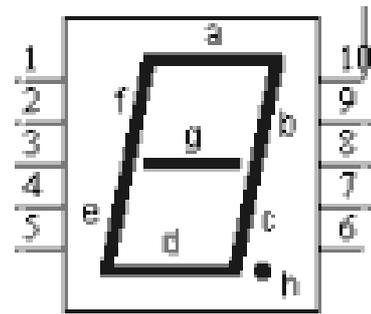
TABELLA ENTRATE/USCITE:

volendo realizzare il display con i led a 7 segmenti ed ipotizzando che le temperature possano variare tra $-9,6^\circ\text{C}$ e $99,6^\circ\text{C}$, si può costruire la tabella delle entrate ed uscite del circuito combinatorio. Essa rappresenta ciò che mi aspetto in uscita con ogni determinato ingresso. Per valori di temperatura che fuoriescono da questi due limiti, si può eliminare la cifra decimale per mostrare le temperature a 3 cifre o quelle sotto lo 0, ma a 2 cifre.

binario								led n°2				led n°1				led n°0															
b7	b6	b5	b4	b3	b2	b1	b0	a2	b2	c2	d2	e2	f2	g2	h2	a1	b1	c1	d1	e1	f1	g1	h1	a0	b0	c0	d0	e0	f0	g0	h0
...
0	0	1	1	0	1	1	1	0	0	0	0	0	0	1	0	1	1	1	0	0	0	0	1	1	1	0	1	1	0	1	0
...
0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	1	1	1	1	1	0	1	1	0	1	1	1	1	0	0
0	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	1	1	0	1	1	1	1	0	0
...
1	0	0	0	1	1	0	0	0	1	1	0	0	0	0	0	0	1	1	0	0	1	1	1	1	1	1	1	1	1	1	0
...



decimale	temp. (°C)	led 2	led 1	led 0
...
55	-7,2	-	7.	2
...
66	-0,6	-	0.	6
67	0		0.	0
68	0,6		0.	6
...
140	43,8	4	3.	8
...



Nell'esempio il valore di temperatura 0°C corrisponde al n° 67 nella scala di conversione ed al numero binario "0100 0011".

Non è consigliabile avere un riferimento sul valore di fondoscala di -40°C perché in questo caso ci si troverebbe sfasati rispetto ai 0°C, nel senso che non si avrebbe un valore binario precisamente corrispondente a questa temperatura che in generale viene presa come riferimento. Perciò è preferibile prendere come riferimento questo valore (è la temperatura alla quale il ghiaccio comincia a fondere).

Di conseguenza cambieranno i valori minimo e massimo rilevabili dal circuito rispetto a quelli del sensore:

- *Minimo* → binario: "0000 0000" ; -39,26°C circa
- *Massimo* → binario: "1111 1101" ; 109,57°C circa

Ricordo che la temperatura massima rilevabile dal sensore è 110°C, mentre la massima digitalizzabile ("1111 1111") sarebbe 110,74°C. Per causa del sensore considero come 'massimo' il primo valore di temperatura minore del limite del sensore stesso.

La seconda soluzione prevede l'utilizzo di un display a cristalli liquidi, accompagnato da un circuito integrato che ne gestisca la logica prelevando il dato digitale ed, ovviamente, la tensione di alimentazione. Questa soluzione, per me è meno adatta per il problema di scelta dei due componenti.

ALIMENTAZIONE : in questa sezione mi occupo di come fornire l'alimentazione ai vari componenti, che richiedono valori di tensione di:

- 12 V: sensori ed operazionali;
- 5 V: display;
- 2,5 V: tensione di riferimento >0; permette alla sonda di riportare misure di temperatura negative.

In ogni caso la tensione di uscita richiesta agli operazionali non sarà mai superiore ai 3,6 V, perché il sensore incrementa la sua V_o di 0,01V ogni °C, perciò tenendo conto della tensione di riferimento di 2,5V:

$$2,5 + (110 \cdot 0,01) = 3,6V \quad (\text{tensione rif.} + \text{°C}_{\text{MAX}} * \text{sensibilità} = \text{tensione Max})$$

1. 12V

Per avere una tensione di 20V_{eff} (poi spiegherò perché non direttamente 12V) partendo dai 220V_{eff} della tensione di rete fornita dall'ENEL ci vuole un trasformatore.

A questo punto la tensione deve essere resa continua.



Il primo stadio si può realizzare con un *raddrizzatore di Graetz*, chiamato anche *ponte a diodi*, che dalla tensione alternata dà in uscita una tensione fatta solo con semionde positive.

Fig 01: raddrizzatore di Graetz

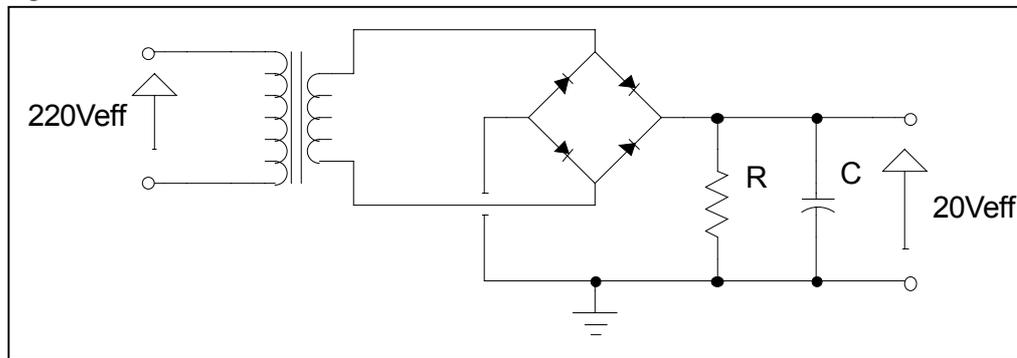


Fig 8: ingresso del Ponte a Diodi

Vi:

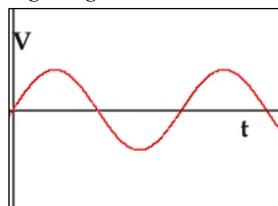


Fig 9: uscita del Ponte a Diodi

Vod:

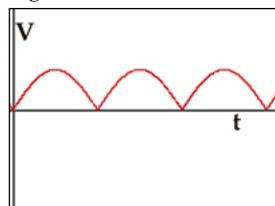
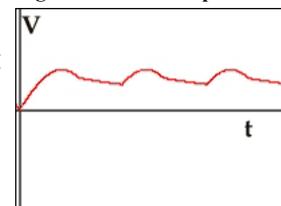


Fig 10: uscita con capacità

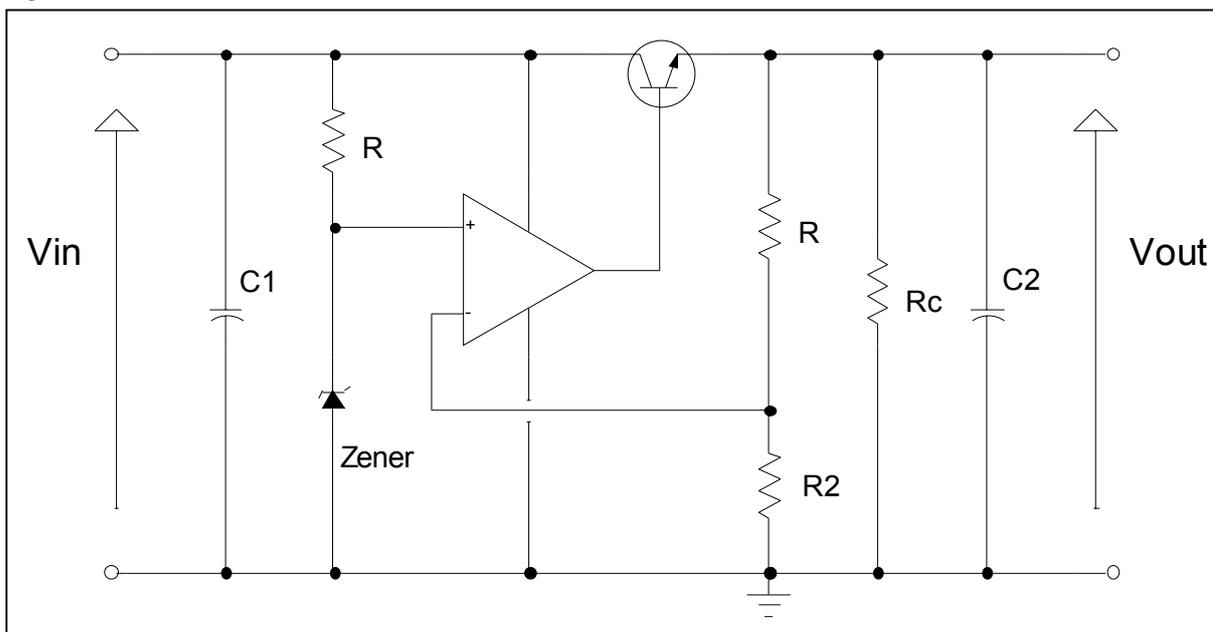
Vo1:



Un condensatore inserito tra l'uscita del raddrizzatore e la massa darà una compensazione nei momenti in cui la tensione di ingresso diminuisce, fornendo un'uscita che si avvicina alla continuità.

Nel prossimo stadio esaminerò come la tensione possa essere stabilizzata. Si può utilizzare un piccolo integrato che svolge questa funzione.

Fig 12: stabilizzatore di tensione



Il principio di funzionamento dello stadio stabilizzatore può essere schematizzato tramite un circuito che sfrutta le caratteristiche dell'amplificatore operazionale e del diodo.



Questo circuito modifica il proprio comportamento nel tempo in modo da compensare le variazioni che subisce la tensione di uscita.

Prendiamo come esempio un ipotetico aumento della tensione ai capi di R_c : di conseguenza si alza anche la tensione di R_2 , cioè quella dell'ingresso invertente dell'OP-AMP.

Sul terminale positivo, invece, c'è la tensione quasi costante del diodo zener. Se consideriamo che l'OP-AMP amplifica la differenza dei suoi 2 ingressi, si capisce come in uscita da esso si abbia un segnale minore all'aumento di V_c . Questa diminuzione va ad influire sulla tensione di R_c perché comporta una diminuzione del segnale di base del BJT, alla quale consegue una riduzione della corrente di collettore che va in parte a determinare la caduta su R_c .

La stabilizzazione viene alla fine spiegata tramite la II legge di Ohm: $V = R \cdot I$.

Con l'aumento della V_{Rc} ho un calo della corrente che scorre in essa, perciò un successivo ritocco della V_{Rc} verso il basso.

Il discorso risulta invertito se V_{Rc} diminuisce.

Per cercare di dissipare le variazioni verso massa, inoltre, posso inserire i condensatori in entrata ed in uscita del circuito, come si vede nella figura.

È da considerare il fatto che in un circuito così strutturato la V_i dovrà essere un po' maggiore della V_o per compensare le perdite causate da:

- 1- le caratteristiche dell'OP-AMP;
- 2- la polarizzazione del BJT;
- 3- l'adozione di un condensatore dopo il ponte a diodi che fornisce una tensione che cala col tempo.

Si può scegliere un valore di V_i di $20V$, che può bastare ad assicurare un buon margine di indifferenza a tali perdite.

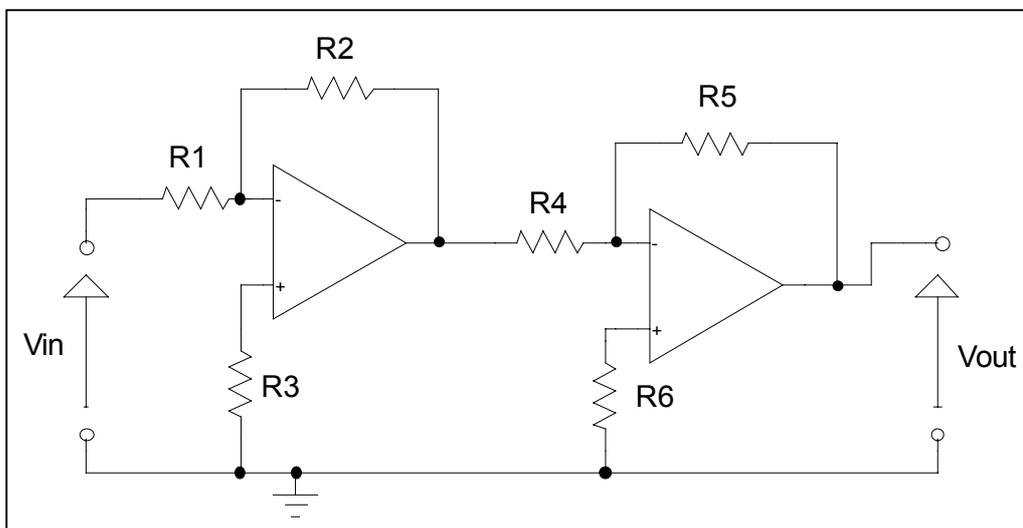
2. 5V

Per disporre della tensione di $5V$ si può pensare di utilizzare un partitore sempre partendo dai $12V$ ottenuti. Ho però dei problemi nel trovare le resistenze che mi occorreranno pescando dalla serie E12, inoltre per il partitore occorrerebbero valori molto precisi, con poca dispersione.

Considerati i difetti di un partitore, in questo caso scelgo l'opzione OP-AMP su cui fisso il

guadagno a $\frac{5}{12} \cong 0,42$.

Fig 13: OP-AMP invertente dove fisso il guadagno; rendo positiv





Dimensionamento: secondo le formule del guadagno, il circuito amplifica V_{in} secondo la formula $V_{out} = V_{in} \cdot \left(-\frac{R2}{R1}\right) \cdot \left(-\frac{R5}{R4}\right)$, ma siccome il secondo OP-AMP viene usato solo per riportare positiva la tensione, invertita dal primo, assumo $R5=R4$ fissando il guadagno a -1 per il secondo stadio. Quindi $G = \frac{R2}{R1} = \frac{5}{12}$

Sapendo che il rapporto tra $R2$ ed $R1$ è di $5/12$, le resistenze sono facili da dimensionare.

N.B.: $R3, R6$ hanno lo scopo di limitare le correnti di offset degli OP-AMP dove sono montate.

$$R1 = 120K\Omega$$

$$R2 = 50K\Omega \quad R3 = R1 \text{ par. } R2 \cong 35K\Omega$$

$$R4 = 39K\Omega$$

$$R5 = 39K\Omega \quad R6 = R4 \text{ par. } R5 = 18K\Omega$$

Standardizzando le resistenze alla serie E12, si può fissare $R3 = 33K\Omega$ mentre $R2$ verrà fisicamente implementata non da una resistenza da $47K\Omega$, che offrirebbe un valore simile a quello teorico, ma dal parallelo tra due resistenze di $100K\Omega$, per mantenersi il più possibile vicino al rapporto di $5/12$.

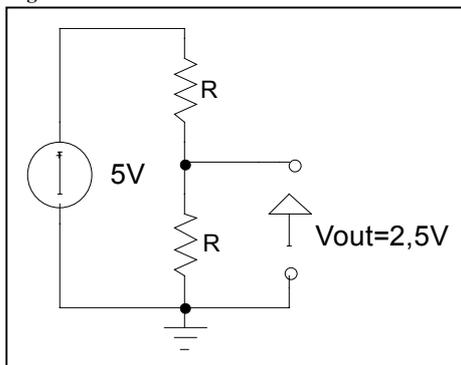
Non si è potuto utilizzare semplicemente un OP-AMP in configurazione non-invertente perché non permette di ottenere un guadagno inferiore ad 1 in quanto la formula è

$$G = 1 + \frac{R2}{R1} \text{ e le resistenze non possono assumere valori negativi.}$$

3. 2,5V

Per ottenere la tensione di riferimento di $2,5V$, invece, si può realisticamente pensare ad un partitore con 2 resistenze, per esempio da $1K\Omega$, ma ad alta precisione (tolleranza molto bassa).

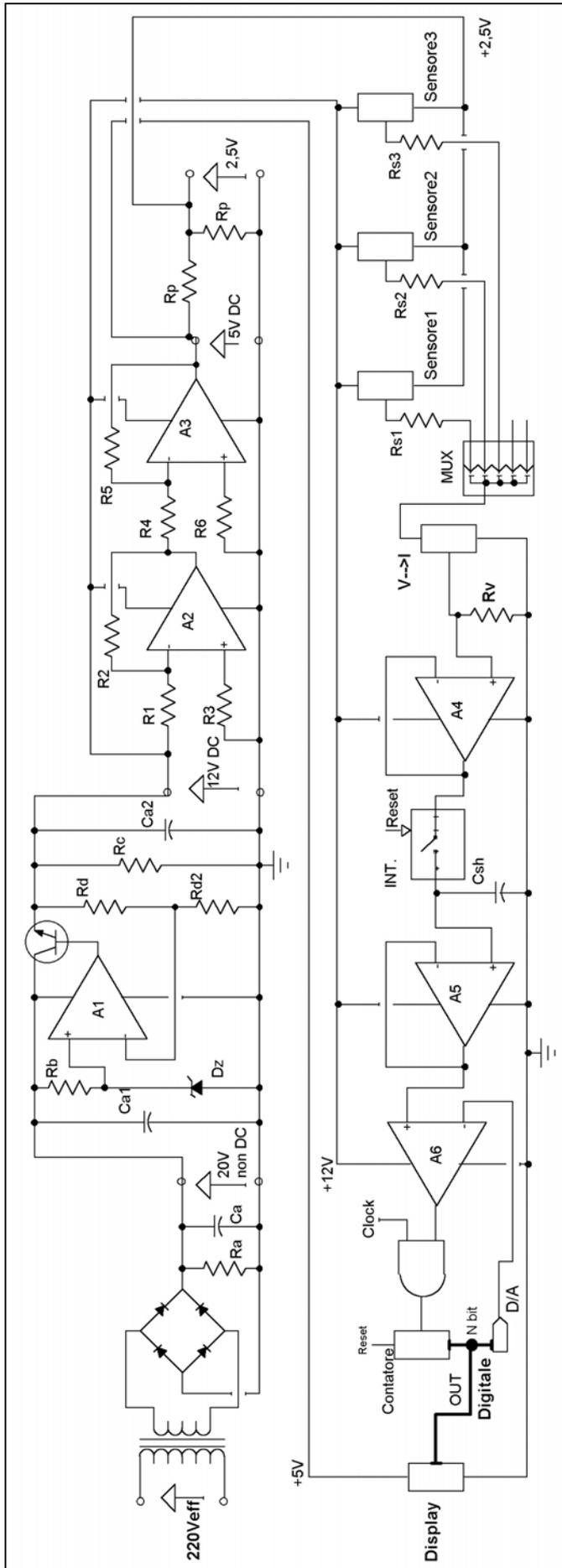
Fig 14: Partitore con 2 resistenze



Se messe in serie, con una tensione di $5V$ da dividersi (vedi Fig.13), 2 resistenze uguali daranno ognuna una ddp dimezzata rispetto a quella totale.

$$R = 1000\Omega \pm 1\%$$

La tolleranza minima delle resistenze sarà dell' 1%. Per far questo non basterà la serie E12 ma ne occorrerà una maggiore.



Ecco come appare il circuito completo di tutti i suoi componenti principali.



Sviluppo parte informatica

Per lo sviluppo dello stadio di memorizzazione, posso dedicarmi alla costruzione di un DataBase, forma di archiviazione dati che abbiamo studiato quest'anno a scuola.

Suppongo di aver disponibile un programma in C++ che permette di dialogare con un piccolo dispositivo esterno tramite la porta seriale del PC.

Questa periferica è in grado di memorizzare automaticamente, di sera, ad un'ora prestabilita, i dati delle temperature interna ed esterna e quello dell'insolazione, in una memoria interna che il programma realizzato può leggere. Le informazioni così lette saranno trasferite in un file.

Un operatore avrà il compito di trascrivere i valori dal file ad un DataBase.

Si richiede quindi un DB che possa contenere record formati dalle 3 grandezze, associate alla loro data di rilevazione.

Questo DB è riservato ad un uso domestico e quindi è futile ed antieconomico richiedere un sistema molto sofisticato che possa agire da applicazione professionale.

Scelgo come piattaforma di sviluppo *Access 2000* perché è un programma molto diffuso, fornito con il pacchetto *Office*, ma principalmente perché è stato studiato durante l'anno scolastico sia in teoria che con applicazioni pratiche.

Un'alternativa ad *Access* potrebbe essere *Excel*, ma esaminandone le potenzialità, non fornisce la stessa flessibilità e funzionalità di *Access*.

Il sistema operativo che gestirà l'interazione tra utente e programmi può essere scelto tra i pacchetti *Windows*, dalla versione *98* in poi. Essendo anch'essi prodotti Microsoft, hanno come loro punto di forza semplicità di utilizzo e la compatibilità con periferiche ed applicativi.

Come si può intuire anche dall'esiguo numero di dati da raccogliere, il DB non sarà particolarmente grande né in termini di tabelle né di query.

Inoltre prevedo che nel periodo di un anno di registrazioni la memoria utilizzata dal DB sarà di qualche decina di KB, poiché in un giorno le informazioni raccolte saranno mediamente una cinquantina di byte, se consideriamo che l'utente può inserire qualche nota testuale in un campo apposito.

Ora passiamo ad esaminare le funzionalità richieste dal problema:

- I. scelta della maschera da utilizzare, perciò è richiesta una maschera principale;
- II. inserimento dei dati;
- III. visualizzazione dei dati;
- IV. modifica o cancellazione di un record (per riparare eventuali errori dell'operatore);
- V. sviluppo dei modelli del DataBase
- VI. interrogazione del DB per la quale possono essere formulate alcune domande:
 - MEDIA VALORI DI UN CERTO PERIODO?
 - VALORI MAX E MIN IN UN CERTO PERIODO?
 - QUANTI GIORNI SOPRA AD UNA CERTA TEMPERATURA?



I. MASCHERA PRINCIPALE

È la prima maschera che appare all'avvio del programma. Permette di scegliere tramite un click del mouse una tra le opzioni disponibili.

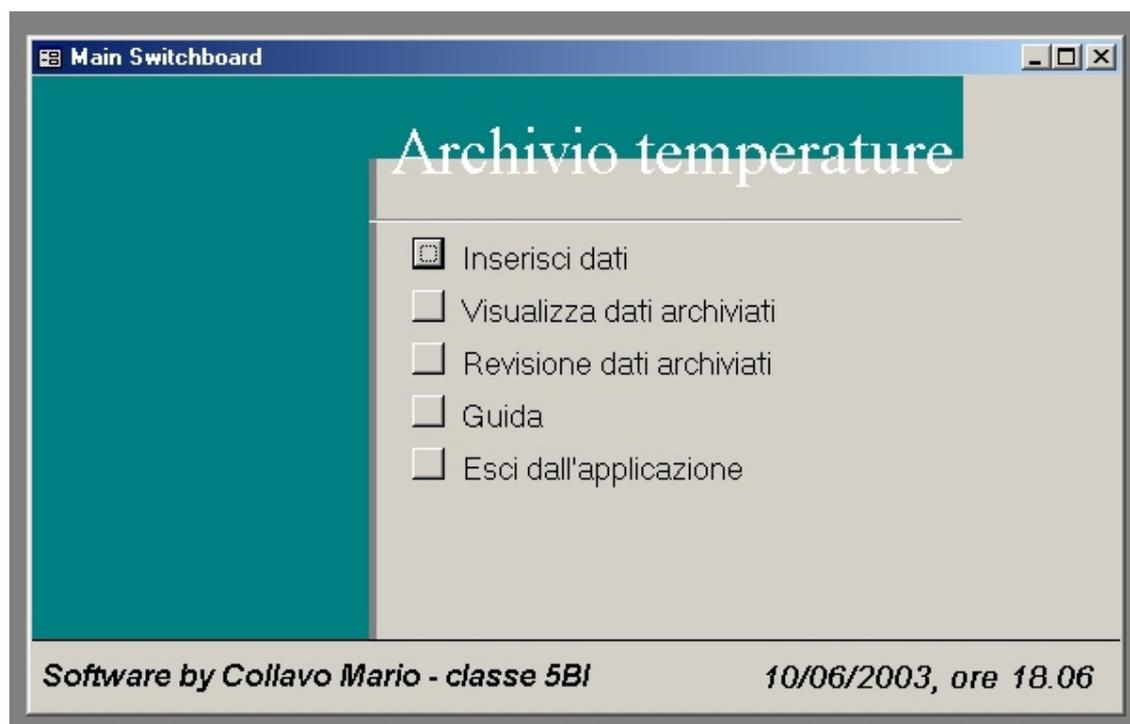
Il metodo più veloce per creare questa maschera è l'utilità di creazione di una Switchboard.

Il pulsante "GUIDA" può essere associato all'apertura di un documento o, meglio, un file della guida, che contiene le informazioni essenziali per l'utilizzo del programma. Questa opzione non verrà approfondita oltre.

Premendo "ESCI" si termina l'esecuzione del programma.

Gli altri pulsanti permettono di accedere alle altre funzioni del programma.

Fig 05: Maschera principale



II. INSERIMENTO DEI DATI

Per l'inserimento ci sarà una maschera che riporta tutti i campi vuoti in modo che possano essere riempiti.

Il lavoro dell'operatore viene facilitato dalla presenza delle caselle di riepilogo per riempire i campi delle 3 grandezze fisiche.

Ci vorrà l'adozione di 2 tabelle esterne nel modello relazionale del DB, riempite precedentemente con tutti i valori possibili per l'insolazione e le temperature. Queste ultime saranno memorizzate con una tolleranza di 1°C, per non appesantire la ricerca del valore esatto nella casella di riepilogo.

Dopo aver inserito i 3 campi fondamentali e la data, è data la possibilità di aggiungere delle note sotto forma di testo.

Il pulsante "AGGIUNGI" memorizza i dati appena inseriti e predispone la maschera ad un nuovo inserimento.

Si può implementare questa funzione in Access semplicemente modificando l'aspetto del

pulsante  già implementato in Access con la funzione di aggiunta di un record.

Il comando con la scritta "ESCI" fa ritornare alla maschera principale.



Non sono presenti i classici pulsanti per muoversi tra i record perché la maschera è dedicata esclusivamente all'azione di inserimento.

III. VISUALIZZAZIONE DEI DATI

Con questa maschera si vuole permettere all'utente di prendere visione di tutti i dati contenuti nel DataBase.

Data	Temp Int	Temp Ext	Insolazione	Note
21/05/2003	57	29	80%	
22/05/2003	40	10	0%	Oggi ha piovuto
23/05/2003	47	22	50%	
20/05/2003	59	18	60%	
25/05/2003	47	23	90%	
19/05/2003	40	15	30%	
27/05/2003	42	20	90%	
16/05/2003	56	20	60%	
26/05/2003	66	24	50%	



Le informazioni verranno presentate sotto forma di tabella: sono presenti le barre di scorrimento nel caso i dati cercati non compaiano nella prima schermata.

Premendo il pulsante "STAMPA" si avranno i dati della tabella in output su supporto cartaceo.

Il pulsante "RICERCA" apre la maschera dove si possono specificare i criteri per la visualizzazione della tabella. Per esempio si possono cercare tutte le informazioni relative al mese di marzo, specificando delle temperature massime e minime.

ARCHIVIO TEMPERATURE: Ricerca Dati

Ricerca Dati

Data da: a:

Temp interna da: a:

Temp esterna da: a:

Insolazione da: a:

Contenuto in Note:

Ricerca

Esci

Software by Collavo Mario - classe 5BI 10/06/2003, ore 19.21

I criteri di ricerca comprendono tutti i campi. Se non viene definita la clausola di un campo (cioè viene lasciato vuoto uno spazio), non verrà applicato nessun filtro a quell'attributo.

Sono presenti due pulsanti: uno per avviare la ricerca e l'altro per uscire.

Premendo "RICERCA", viene avviata l'operazione e viene aperta una finestra quasi uguale a quella che presenta tutti i dati, con disponibili tutti i pulsanti e le funzioni ad essi legate, ma la differenza è che la tabella riporterà solo i dati che corrispondono ai criteri di ricerca.

Il pulsante "ESCI" riporta l'utente alla maschera precedente.

Tornando alla maschera di visualizzazione, il pulsante "MODIFICA" di fatto apre un'altra maschera, molto simile a quella di origine, ma dedicata allo specifico compito di permettere, appunto, la modifica di dati già inseriti nel DB.



IV. MODIFICA E CANCELLAZIONE

La maschera appare simile a quella di visualizzazione. Sono cambiati però i pulsanti nella parte bassa della finestra ed è stata aggiunta all'utente la possibilità di revisionare i dati.

Selezionando il campo di una tupla, esso verrà evidenziato perché l'utente possa modificarlo tramite tastiera o con il mouse agendo sulle caselle di riepilogo.

Se invece si clicca sul pulsante "CANCELLA", i dati della tupla selezionata verranno eliminati.

Data	Temp Int	Temp Ext	Insolazione	Note
16/05/2003	56	20	60%	
19/05/2003	40	15	30%	
20/05/2003	59	18	60%	
21/05/2003	57	29	80%	
22/05/2003	40	10	0%	Oggi ha piovuto
23/05/2003	47	22	50%	
25/05/2003	47	23	90%	
26/05/2003	66	24	50%	
27/05/2003	42	20	90%	

Cancella **Indietro** **Esci**

Software by Collavo Mario - classe 5BI 10/06/2003, ore 19.41

Record: 5 di 9

"INDIETRO" permette di retrocedere alla maschera di visualizzazione ed il pulsante "ESCI", invece, ci riporta direttamente alla maschera principale.

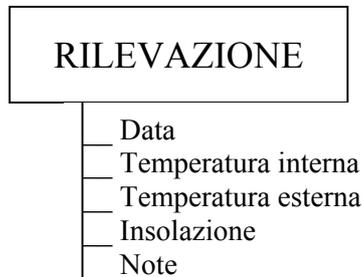


V. MODELLI

Per l'effettiva costruzione del DataBase e delle sue tabelle, ci serviamo dei modelli.

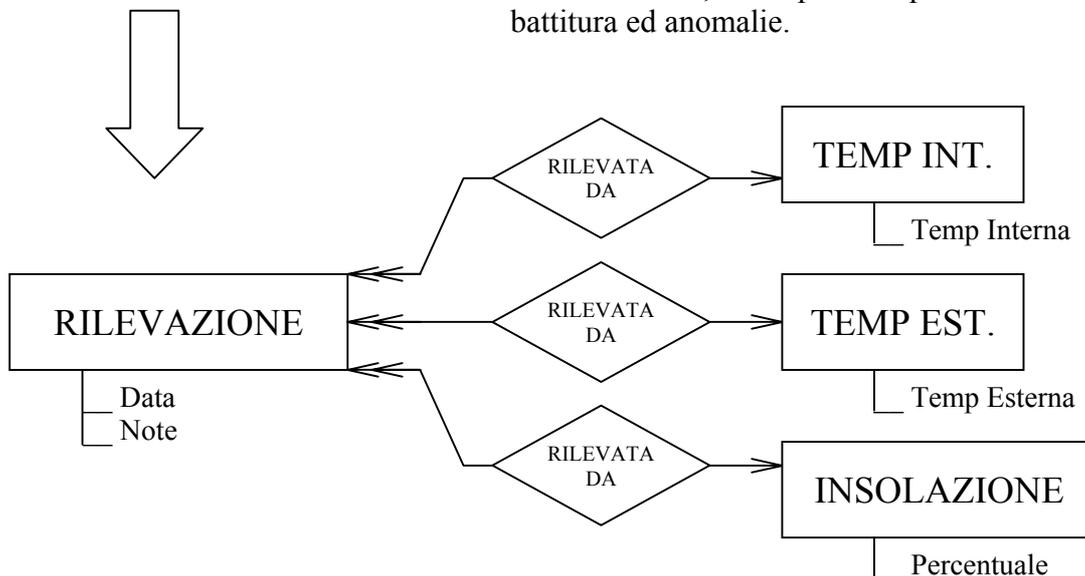
Il primo passo è quello di realizzare il *MODELLO E/R* che si basa sull'adozione di entità e relazioni. Un'entità corrisponderà poi ad una tabella ed una relazione ad un'associazione.

Il più semplice dei modelli E/R realizzabili presenta una sola entità senza relazioni.

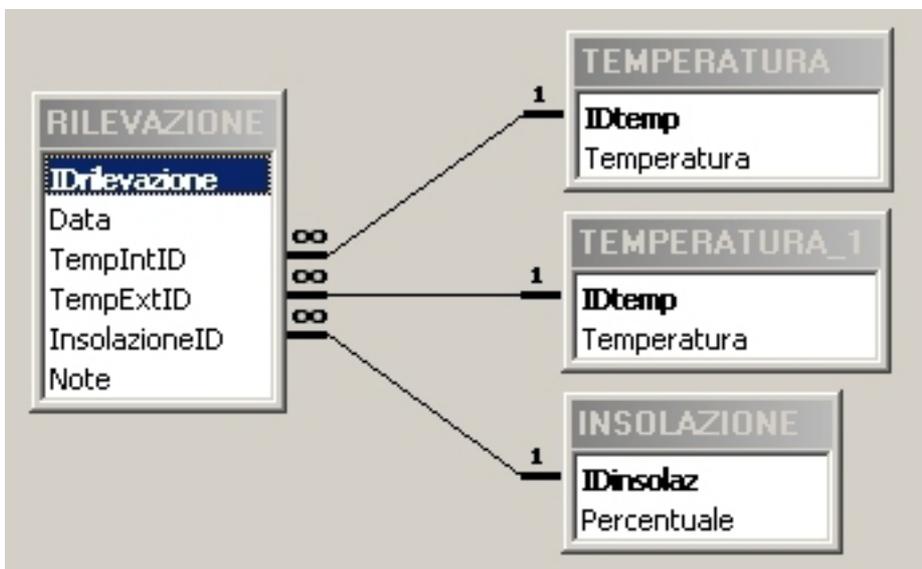


Gli attributi relativi alle 3 grandezze fisiche possono essere portati fuori con chiavi esterne, per formare altre tre entità, allo scopo di:

- 1- facilitare l'inserimento dei dati da parte dell'operatore, in quanto si potranno utilizzare caselle di riepilogo;
- 2- evitare l'operazione di inserire le informazioni tramite tastiera, il che può comportare errori di battitura ed anomalie.



Dopo lo sviluppo del modello E/R si può passare alla definizione del *MODELLO RELAZIONALE*.



È stata inserita una chiave primaria artificiale per ogni tabella. I campi TempIntID, TempExtID, InsolazioneID sono le chiavi esterne per associare la tabella RILEVAZIONE con le altre 3 tabelle. Tutti i valori di "Temperatura" e "Percentuale" verranno inseriti prima di cominciare l'attività di inserimento.



VI. INTERROGAZIONE DATABASE

Lo strumento che ci permette di ottenere informazioni elaborate e precise partendo dai dati contenuti nel DB sono le *QUERY*.

In classe abbiamo sviluppato principalmente l'abilità nel programmare con il linguaggio *SQL* (Structured Query Language) e un po' meno quella di utilizzare il *QBE* (Query By Example) di Access.

Per costruire delle query, bisogna prima porsi delle domande in riferimento alle informazioni che si desidera cercare.

Le domande più significative che l'operatore può porsi e le loro relative risoluzioni in SQL sono:

1- *Qual è la media dei valori in un certo periodo?*

L'operatore dovrà inserire le due date che definiscono il periodo in cui cercare. La media verrà fornita utilizzando la funzione *AVG()* di Access.

In questo caso l'opzione migliore è quella di realizzare 3 query separate, una ogni attributo.

Temperatura Esterna:

```
SELECT Avg(Temperatura) AS T_ext
FROM TEMPERATURA inner join RILEVAZIONE ON
(TEMPERATURA.IDtemp=RILEVAZIONE.TempExtID)
WHERE ((Data>=[da che data?]) And (Data<=[a che data?]));
```

Temperatura Interna:

```
SELECT Avg(Temperatura) AS T_int
FROM TEMPERATURA inner join RILEVAZIONE ON
(TEMPERATURA.IDtemp=RILEVAZIONE.TempIntID)
WHERE ((Data>=[da che data?]) And (Data<=[a che data?]));
```

Insolazione:

```
SELECT Avg(Percentuale) AS INS
FROM INSOLAZIONE inner join RILEVAZIONE ON
(INSOLAZIONE.IDinsolaz=RILEVAZIONE.InsolazioneID)
WHERE ((Data>=[da che data?]) And (Data<=[a che data?]));
```

2- *Quali sono i valori massimo e minimo di un certo intervallo di tempo?*

Sempre dopo l'inserimento delle due date, i valori max e min potranno essere trovati esaminando le tuple rispettivamente attraverso le funzioni *MAX()* e *MIN()*.

Anche in questo caso è molto conveniente separare le query, che saranno strutturate in modo molto simile alle tre del quesito 1.

Temperatura Esterna:

```
SELECT Max(Temperatura) AS T_ext_Max, Min(Temperatura) AS T_ext_Min
FROM TEMPERATURA inner join RILEVAZIONE ON
(TEMPERATURA.IDtemp=RILEVAZIONE.TempExtID)
WHERE ((Data>=[da che data?]) And (Data<=[a che data?]));
```



Temperatura Interna:

```
SELECT Max(Temperatura) AS T_int_Max, Min(Temperatura) AS T_int_Min
FROM TEMPERATURA inner join RILEVAZIONE ON
(TEMPERATURA.IDtemp=RILEVAZIONE.TempIntID)
WHERE ((Data>=[da che data?]) And (Data<=[a che data?]));
```

Temperatura Esterna:

```
SELECT Max(Percentuale) AS Ins_Max, Min(Percentuale) AS Ins_Min
FROM INSOLAZIONE inner join RILEVAZIONE ON
(INSOLAZIONE.IDinsolaz=RILEVAZIONE.InsolazioneID)
WHERE ((Data>=[da che data?]) And (Data<=[a che data?]));
```

3- *Quanti giorni, in un periodo definito, la temperatura è rimasta sopra ad un limite?*

Scegliendo un valore di temperatura, verrà visualizzato il n° di giorni che, in un certo periodo, hanno visto una temperatura maggiore di quella data. Verrà utilizzata la funzione *COUNT(*)*.

Sempre separando le query:

Temperatura Interna:

```
SELECT Count(*) AS N
FROM TEMPERATURA inner join RILEVAZIONE ON
(TEMPERATURA.IDtemp=RILEVAZIONE.TempIntID)
WHERE (Temperatura>=[Temperatura minima?]);
```

Temperatura Esterna:

```
SELECT Count(*) AS N
FROM TEMPERATURA INNER JOIN RILEVAZIONE ON
(TEMPERATURA.IDtemp=RILEVAZIONE.TempExtID)
WHERE (Temperatura>=[Temperatura minima?]);
```

**BIBLIOGRAFIA:**

- "ELETTROROTECINICA ELETTRONICA TELECOMUNICAZIONI",
R. Giometti - F. Frascari, Calderini editore, vol.1 e 2;

- "INFORMATICA, Database con applicazioni in Access",
A. Garavaglia - N. Mazzucchielli, Zanichelli editore, vol.3;

- "NUOVA ELETTRONICA",
Rivista mensile, aprile 2003, pagg 22-35;

- Appunti personali di elettronica ed informatica presi durante l'anno scolastico.