

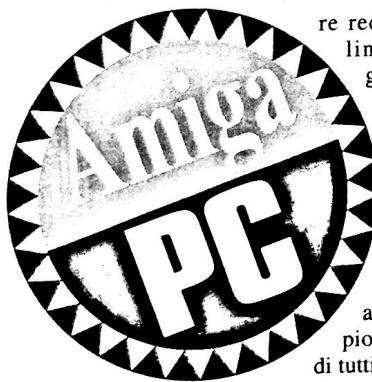
Ecoci per la seconda volta (si veda CG 2/95) a parlare di campionamento dei segnali analogici e della loro trasformazione in valori digitali. Questa volta parliamo di segnali in banda audio, ovvero quella compresa fra poche decine di Hertz e 20 KHz circa. Il circuito che vi sarà proposto è adatto al funzionamento sia su Amiga (per compensare gli "amighisti" un po' trascurati nello scorso articolo), sia su PC dotato di porta parallela di tipo bidirezionale. Questa volta vi forniremo un esempio di utilizzo della scheda che andremo a costruire.

trario, può assumere, nello stesso intervallo, solamente un certo numero di valori diversi, che dipende dal tipo di campionatore usato. Se non avete capito niente, fate riferimento alla Figura 1, dove trovate la rappresentazione di un tipico segnale analogico, e alla Figura 2, dov'è disegnato un tipico segnale digitale, ottenuto immaginando di campionare il segnale analogico di partenza in diversi punti (il segnale originario è rappresentato con una linea tratteggiata). La differenza dovrebbe essere evidente: il segnale campionato ha il tipico andamento "a gradino", dovuto di fatto alla bassa frequenza di

frequenza di campionamento è il numero di volte che il segnale sotto osservazione viene misurato (acquisito) nel corso di un secondo. Per esempio, campionare un segnale con una frequenza di campionamento di 44 KHz, significa misurare il valore istantaneo dello stesso segnale 44 mila volte ogni secondo. La risoluzione di quantizzazione, invece, ci dà un'idea della "qualità" con la quale vengono misurati i vari livelli raggiunti dal segnale originario. Una risoluzione di 8 bit ci permette di apprezzare 256 diversi livelli di segnale (2^8), mentre una risoluzione di 16 bit è sufficiente

Costruiamo un campionatore in banda audio

Con meno di 50 mila lire, potete costruirvi un campionatore audio di tutto rispetto per Amiga e PC con parallela bidirezionale

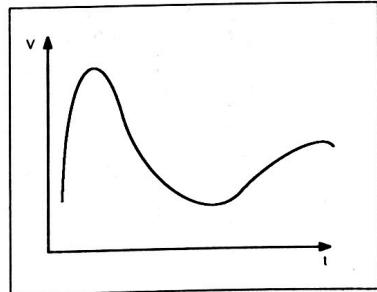


re redatto interamente in linguaggio C, per la gioia di tutti i puristi che odiano (a torto) il tanto bistrattato BASIC interpretato. Per iniziare, però, ritengo utile fornire qualche informazione generale sulla teoria che sta dietro all'operazione di campionamento, a beneficio di tutti coloro che non conoscono le basi di questa tecnica di acquisizione dei segnali utilizzata ormai in ogni campo. Per forza di cose, la trattazione sarà molto discorsiva e poco "tecnica", in modo da essere comprensibile a tutti i nostri lettori.

Segnali analogici e digitali

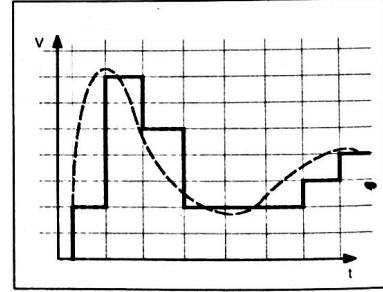
La prima nozione che bisogna apprendere è quella della differenza che c'è tra un segnale analogico e uno digitale. Per avere un esempio del primo possiamo pensare, per esempio, al segnale presente ai capi di ognuno degli altoparlanti del nostro stereo di casa. Tale segnale varia con continuità nel tempo, nel senso che può assumere, in istanti diversi, un valore qualsiasi fra quelli compresi in un certo intervallo. Un segnale di tipo digitale, al con-

campionamento e alla quantizzazione (cioè l'assegnazione del valore acquisito a un range limitato di valori) molto grezza. Una volta eseguita l'operazione di campionamento del segnale, lo stesso è disponibile in forma numerica



▲ Figura 1: una generica forma d'onda di tipo analogico

per distinguere fra 65.536 ampiezze diverse (nella Figura 2 il segnale originario è stato quantizzato con una risoluzione di 3 bit = 8 valori possibili; l'operazione di quantizzazione assegna semplicemente il valore misurato al più vicino fra quelli permessi). Ovviamente, la "qualità" dell'azione di campionamento è funzione diretta di questi due parametri. Esiste in merito un noto teorema, detto teorema di Shannon o del campionamento, il quale afferma più o meno che: per essere certi di poter ricostruire esattamente un segnale analogico partendo dalla sua forma



▲ Figura 2: lo stesso segnale della Figura 1 campionato e quantizzato

(per ricostruire il segnale sono sufficienti le singole ampiezze e gli istanti di campionamento), e come tale può essere trattato da qualsiasi calcolatore.

Il teorema del campionamento

A questo punto, è importante definire i due parametri principali che intervengono nel campionamento di un segnale analogico: frequenza di campionamento e risoluzione di quantizzazione. La

d'onda campionata, occorre utilizzare, durante l'acquisizione, una frequenza di campionamento pari almeno al doppio della massima frequenza di segnale presente nella forma d'onda originaria. In parole povere, il teorema afferma che se, per ipotesi, volessimo digitalizzare fedelmente un brano musicale in modo da ottenere la massima qualità udibile, dovremmo campionare lo stesso a una frequenza di almeno 40 KHz, in modo da ottenere una corretta ricostruzione delle frequenze fino a 20 KHz (40 KHz / 2) che sono poi quelle

udibili dal nostro orecchio. Se utilizzassimo frequenze di acquisizione inferiori, si otterrebbe il cosiddetto "effetto di aliasing" ovvero quel fastidioso "gracchiare" che si può udire ascoltando suoni campionati a frequenze troppo basse rispetto al loro contenuto armonico.

il circuito

Dopo tanta teoria, veniamo al circuito da costruire. Si tratta di uno schema che permette la realizzazione di un campionatore stereo in banda audio, facilmente modificabile però per acquisire anche segnali in continua. Lo schema si basa sul circuito integrato, prodotto dalla National, siglato ADC0808 (oppure ADC0809, versione un po' più economica con prestazioni leggermente minori), che integra al suo interno un campionatore a otto bit e un multiplexer a otto canali indipendenti. Il componente è stato pro-

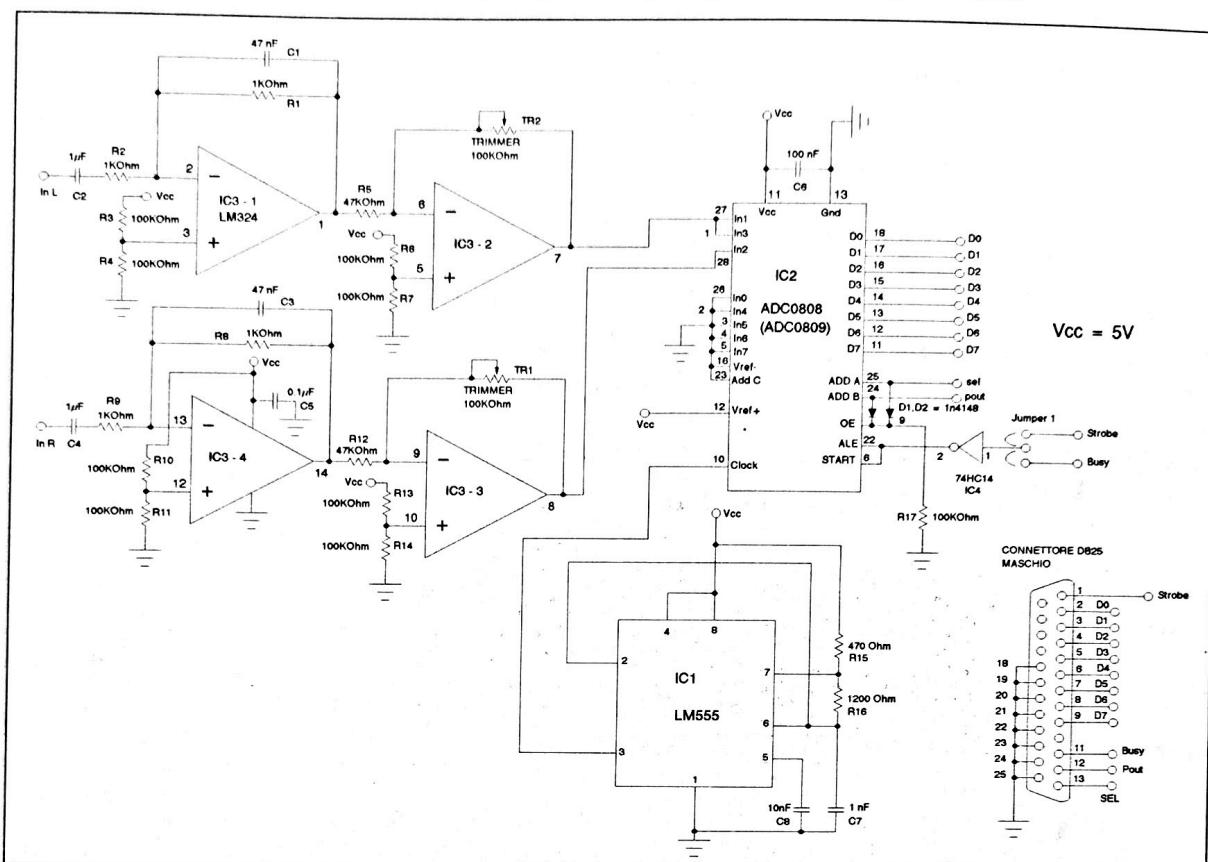
gettato è stato progettato appositamente per realizzare sistemi di acquisizione di basso costo e buona precisione, controllati direttamente da microprocessore. Nel nostro schema, utilizzeremo solo due (in realtà tre, come vedremo) dei canali disponibili, ma nulla vi vieta di servirvi dei canali inutilizzati per i vostri esperimenti personali. Il funzionamento è molto semplice: tramite i segnali SEL e POUT della porta parallela selezioniamo il canale di acquisizione voluto (Left oppure Right); il segnale BUSY fa partire la conversione (quando è a livello 0) e sui pin D0-D7 di IC2 compare il valore digitalizzato della tensione d'ingresso analogica presente sul canale o prescelto. A questo punto, basta leggere il valore degli otto bit forniti come risultato dal chip e il gioco è fatto!

Vediamo ora in dettaglio le varie funzioni dei singoli circuiti integrati presenti nello schema: il segnale da convertire è fornito a IC2 attraverso il circuito di amplificazione/filtraggio composto di due amplificatori operazionali per ogni canale (i due canali sono, come vedete, identici); IC3-1 è un filtro passa-banda tarato per eliminare le componenti di frequenza esterne all'intervallo 20 Hz-20 KHz, mentre IC3-2 (un più che classico amplificatore

invertente) serve ad amplificare o attenuare, a seconda della posizione di TR2, il livello del segnale d'ingresso per adattarlo ai nostri scopi, evitando distorsioni durante l'acquisizione. A questo punto, la forma d'onda così trattata è connessa agli ingressi di acquisizione di IC2 pronta per essere campionata. Avrete notato che il segnale proveniente da uno degli amplifica-

che POUT sia a 1 e che SEL sia a 0. Se tutti e due i segnali sono a livello logico basso, infine, viene acquisito il canale 0 che è collegato a massa. Tene-
te conto, se volete scrivere da soli dei programmi applicativi per la scheda,
del metodo di selezione dei canali che
vi ho appena illustrato.

Un'ultima nota riguarda il circuito per la generazione del segnale di clock.



▲ Lo schema del circuito de- nistro digita- lizzatore audio

ELenco dei componenti necessari

IC1	NE555
IC2	ADC0808 oppure ADC0809
IC3	LM324 o equivalente
IC4	74HC14 o equivalente
R1, R2, R8, R9	resistore 1K Ohm 1/4 watt
R3, R4, R6, R7,	
R10, R11, R13,	
R14, R17	resistore 100K Ohm 1/4 watt
R5, R12	resistore 47K Ohm 1/4 watt
R15	resistore 470 Ohm 1/4 watt
R16	resistore 1200 Ohm 1/4 watt
TR1, TR2	trimmer 100K Ohm
C1, C3	condensatore ceramico 47 nF
C2, C4	condensatore non polarizzato 1µF
C5, C6	condensatore ceramico 100 nF
C7	condensatore ceramico 1 nF
C8	condensatore ceramico 10 nF

Un connettore a vaschetta DB25 maschio

segna di STROBE fornito direttamente dalla porta parallela dell'Amiga (che è attivo a livello basso) per far partire la sequenza di campionamento (IC2 fa partire l'acquisizione quando rileva un livello logico alto sul suo pin 6; il segnale di STROBE va perciò invertito). Se utilizzate il circuito con un PC, oppure volete scrivervi del software applicativo, vi consiglio di non montare IC4 e di collegare, come

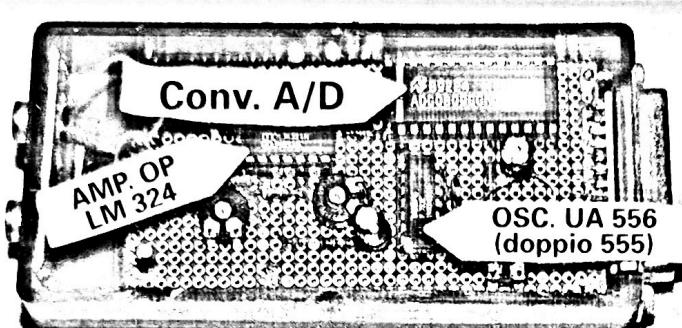
avete, potete ripiegare su una comune pila da 9 Volt formato radiolina, ricordandovi però, in questo caso, che lo stato di carica della pila influenza il comportamento del circuito (questo perché la tensione di riferimento del convertitore A/D è ricavata dalla tensione di alimentazione, e la tensione fornita da una pila a secco tende a scendere durante la scarica variando perciò quest'ultima).

lo, su CG 2/95) ed eliminate i condensatori C2 e C4. Ricordatevi però che il circuito così modificato può campionare solamente tensioni di segno positivo e quindi non è più adatto ad acquisire segnali di tipo audio (che, in generale, possono avere valori qualsiasi), se non opportunamente trattati.

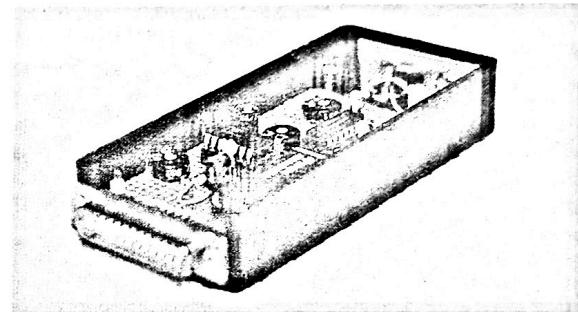
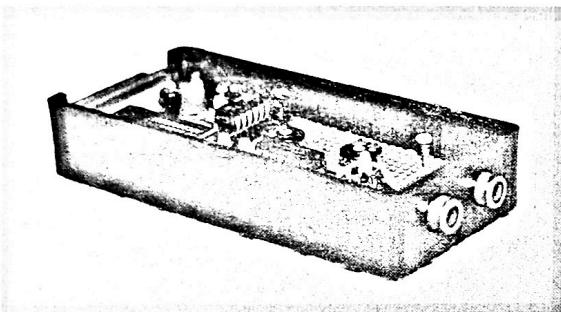
Il software

Anche questa volta vi proponiamo un esempio di software applicativo per l'uso personalizzato della scheda d'interfaccia. Il listato, in linguaggio C, fa riferimento all'Amiga (visto che la volta scorsa abbiamo privilegiato il PC), ma è utilizzabile con poche modifiche anche su macchine DOS dotate di porta parallela bidirezionale. Il programma si commenta da sé: vengono definite all'inizio quattro variabili puntatore che puntano agli indirizzi di controllo della porta parallela dell'Amiga (gestiti da due integrati CIA8520). Per

► Il campionatore una volta montato



► Si notino gli ingressi audio e il connettore DB25 maschio



mostrato, i pin 6 e 22 di IC2 al segnale BUSY della vostra porta parallela ricordandovi appunto che la conversione A/D viene eseguita dal canale prescelto ogni volta che forniamo al campionatore un impulso POSITIVO sul pin 6. Per alimentare il circuito, potete utilizzare un qualsiasi alimentatore stabilizzato capace di fornire una tensione continua di 5 Volt positivi. Se non lo

Se, infine, volete utilizzare lo schema per acquisire segnali a frequenze molto basse o tensioni continue, potete fare quanto segue: collegate i pin 3, 5, 10 e 12 di IC3 direttamente a massa, eliminando R3, R4, R6, R7, R10, R11, R13, R14; alimentate IC3 con una tensione duale di 5 volt positivi e 5 volt negativi (va bene l'alimentatore che vi abbiamo proposto nello scorso articolo).

acquisire un segnale attraverso la scheda occorre prima di tutto configurare i pin D0-D7 della porta parallela come INPUT e i pin BUSY, POUT e SEL come output: per fare questo, è sufficiente scrivere nei bit di controllo della porta il valore 1 per ogni pin che vogliamo sia configurato come INPUT, il valore 0 per ogni pin che vogliamo operi come OUTPUT. A questo punto, è sufficiente selezionare il canale desiderato pilotando opportunamente SEL e POUT e quindi porre a livello alto il segnale BUSY per far partire la procedura di conversione e sui pin D0-D7 compariranno quasi immediatamente i segnali corrispondenti alla conversione effettuata.

Il programma è tutto qui. Come vedete, è di una semplicità a dir poco sconvolgente. Naturalmente, può essere ampliato e modificato a piacere. L'unico limite è la vostra inventiva.

Come utilizzare i dati campionati

La tensione di riferimento fornita al convertitore A/D (pin 12) è, come si vede chiaramente, di 5 volt positivi, mentre il riferimento negativo (pin 16) è a massa. In questa configurazione, perciò, il chip ADC0808 può digitalizzare solamente segnali che siano positivi rispetto a massa, in quanto riconoscerebbe comunque come 0 volt qual-

NOTE SULL'INTEGRATO NE555

Il circuito integrato NE555 è definito come un "general purpose timer", cioè come un dispositivo progettato per realizzare facilmente circuiti generatori di forme d'onda quadre oppure temporizzati. Nella nostra applicazione viene usato alla stregua di un semplice generatore di clock (oscillatore stabile). Per calcolarne la frequenza di oscillazione, potete fare riferimento alle seguenti formule, ricavate dai *data-sheet* del componente stesso. Seguite lo schema elettrico e fate riferimento ai simboli dei componenti impiegati:

$$F = 1.44 / (R15 + 2 \cdot R16) \cdot C$$

dove F è la frequenza dell'onda quadra generata espressa in Hertz, mentre i valori delle resistenze e del condensatore devono essere espressi rispettivamente in Ohm e Farad. Potete anche determinare il *duty cycle* della forma d'onda di uscita (ovvero la proporzione fra il tempo che l'uscita vale 1 e il tempo in cui vale 0 durante un periodo), utilizzando la formula:

$$D = R16 / (R15 + 2 \cdot R16)$$

dove D è espresso in percentuale (un valore di 0.6 vuol dire che per il 60% del periodo di oscillazione l'onda quadra è a livello alto, per il restante 40% a livello basso). Per progettare un oscillatore utilizzando questo componente, si deve innanzitutto fissare un valore a piacere di C (perché ragionevole), decidere quale dovrà essere D e, a questo punto, risolvere il sistema formato dalle due equazioni ricavando i corrispondenti valori di $R15$ e $R16$. Chiaramente, la scelta del parametro da fissare non è vincolante. Si può per esempio fissare a priori $R15$ e $R16$ e calcolarsi di conseguenza i valori di C e D , eccetera. L'unica cosa di cui dovete tenere conto, è che la massima frequenza generabile si aggira, a seconda delle versioni del componente, intorno al MHz. Per frequenze superiori occorre ricorrere ad altri tipi di schema.

siasi valore di tensione negativa che si trovasse applicata sugli ingressi analogici (perché essendo il riferimento negativo posto a massa, la minima tensione che può riconoscere non può essere più bassa di questo valore). Come facciamo allora a campionare un segnale audio che, a priori, può essere costituito da una forma d'onda che assume valori sia positivi che negativi? Il metodo è quello di sommare un "offset" alla tensione d'ingresso, in modo da renderla sempre e comunque positiva. In effetti, il partitore formato da R3-R4 (oppure R10-R11) esegue proprio questa operazione: somma al segnale proveniente dall'esterno una tensione costante di 2,5 volt (con alimentazione Vcc=5v, perché il partitore R3-R4 è formato da due resistenze identiche e perciò, come dice la legge di Ohm, la tensione ai capi del partitore si divide esattamente a metà al centro dello stesso), permettendo al segnale di assumere solamente valori positivi.

Ai fini del segnale, va ricordato che la forma d'onda non è minimamente modificata da questa operazione, è semplicemente traslata in positivo di una costante. Risulta perciò evidente che, quando il convertitore ci fornisce in uscita il valore 127 (metà della scala di valori 0-255), vuol dire che la tensione d'ingresso (per esempio su inL) vale istantaneamente (siamo in alternata) 0V, perché il nuovo "zero" è stato traslato verso l'alto di 2,5 volt. Una tensione di 2,5 V verrà ad assumere allora il valore 255, e così via. Ricordatevi che questo non vuole dire che la massima tensione applicabile in ingresso non può superare i 2,5 V. Come già detto, è stato inserito nel circuito uno stadio di amplificazione/attenuazione (IC3-2 e IC3-3) con lo scopo di scalare opportunamente i valori da leggere (non superate comunque i 5 Volt per l'ingresso).

Prestazioni

Le prestazioni del circuito sono interessanti, anche se dipendono chiaramente dalla potenza della macchina che avete a disposizione. Sull'Amiga 4000/040 "overclockato" a 33 MHz si riescono a raggiungere frequenze di campionamento di 35-38 KHz (in mono) con un'ottima fedeltà di risposta. Chiaramente, campionando segnali stereofonici le prestazioni diminuiscono di conseguenza, ma il risultato è comunque molto soddisfacente.

Vorrei ancora farvi notare che l'uso dello schema non è certo limitato all'acquisizione di segnali audio: pensate per esempio alla realizzazione di un programma che simuli un oscilloscopio digitale...

Le modifiche per utilizzare più di due canali, infine, sono veramente molto semplici: basta realizzare un circuito di filtraggio/amplificazione/attenuazione come quello che avete nello schema per ognuno dei canali da utilizzare, e

quindi collegare le uscite di questi ultimi con gli ingressi del convertitore A/D: la selezione del canale di acquisizione sarà resa possibile, al solito, pilotando opportunamente i pin SEL e POUT.

Conclusioni

Vi consiglio di prestare molta attenzione durante il montaggio della scheda a non creare dei cortocircuiti che potrebbero danneggiare il vostro computer. Controllate sempre, una volta cablato lo schema e prima di collegarlo all'elaboratore, l'esattezza di tutti i collegamenti utilizzando un semplice tester. Il circuito non necessita di tarature e dovrebbe, una volta collegato, funzionare subito. Per provarlo, potete utilizzare anche il listato in C fornito come esempio: non applicando nessun segnale d'ingresso il programma dovrebbe leggere il valore di 127 (centro scala). Ricordatevi che, essendo il fondo scala di 5V, l'escursione di segnale utile è di 2,5 V positivi e 2,5 V negativi. Una tensione positiva di 2,5 V verrà convertita nel valore 255, mentre una tensione negativa di 2,5 V sarà chiaramente riportata con il valore

0. Lo step minimo di tensione rilevabile è di 5V/256 = circa 19,6 mV, e il valore della tensione letta si può calcolare con la formula $V_{camp} = ((\text{Valore in uscita di ADC0808} * 19,6) - \text{offset})$ dove l'offset vale, nel nostro caso, 2,5 V. Attenzione! Applicando una tensione continua agli ingressi continuerete a leggere il valore costante uguale a 127 (corrispondente a una lettura di 0 volt), perché ovviamente il condensatore C2 (o C4) blocca qualsiasi componente continua. Per acquisire questo tipo di segnali, occorre operare alcune lievi modifiche al circuito.

Il costo del marchingegno è veramente contenuto e non dovrebbe superare le 35 mila lire. I componenti sono tutti di facilissima reperibilità e di uso comune in ambito elettronico. Lo schema proposto è stato disegnato utilizzando i componenti secondo configurazioni assolutamente standard e quindi di non è minimamente critico o problematico nel funzionamento.

Per qualsiasi dubbio, commento o domanda, potete scrivere in redazione oppure potete contattare l'autore sulla BBS DLE (Tel. 019/8386452 - 8387038) inviando un mail a "CAT". Buon montaggio.

CG

LISTATO D'ESEMPIO IN LINGUAGGIO C

```
*****
* campionatore.c : acquisisce un segnale di tipo      *
* analogico e lo converte in un valore digitale      *
* stampando a video il risultato                  *
*****
```

```
#include <stdio.h>
#include <stdlib.h>
#include <hardware/cia.h>

extern struct CIA ciaa,ciab; /* Struttura di controllo per le due CIA di Amiga */
                                /* definita in hardware/cia.h */

unsigned char *ParPort = &ciaa.ciaprb;      /* Porta parallela D0-D7 */
unsigned char *ParDDR = &ciaa.ciaddrb;     /* Registro di controllo D0-D7 */
unsigned char *CtrlPort = &ciab.ciapra;    /* Segnali di controllo */
unsigned char *CtrlDDR = &ciab.ciaddra;   /* Registro di controllo */

void main()
{
    unsigned char c;

    *ParDDR=0x00;           /* Tutti i pin D0-D7 sono input */
    *CtrlDDR=0x07;          /* BUSY, SEL e POUT sono output */

    *CtrlPort &= 0xf8;        /* Azzero gli output          */
    *CtrlPort |= 0x01;        /* BUSY=1, SEL=0, POUT=0    */

    while(1)                /* Ciclo infinito */

        *CtrlPort |= 0x04;    /* Seleziona canale L (SEL = 1 e POUT = 0) , BUSY = 1*/
        *CtrlPort &= 0xfe;    /* Canale L (SEL = 1), START conversion (BUSY = 0) */

        c = *ParPort;         /* Leggo i bit D0-D7 */

        printf(" Valore canale L = %d \n",c); /* Visualizzo il risultato */

        *CtrlPort &= 0xf8;
        *CtrlPort |= 0x01;    /* Situazione iniziale */

        *CtrlPort |= 0x02;    /* Seleziona canale R (SEL = 0 e POUT=1) , BUSY = 1*/
        *CtrlPort &= 0xfe;    /* Canale R (SEL = 1 e POUT=0), START conversion (BUSY = 0) */

        c = *ParPort;         /* Leggo i bit D0-D7 */

        printf(" Valore canale R = %d \n",c); /* Visualizzo il risultato */

        *CtrlPort &= 0xf8;
        *CtrlPort |= 0x01;
}
```