

A/D Converter, esercitazione

come funziona un ADC

Il **progetto** realizzato con Scratch permette di entrare nei dettagli di un ADC.

Il convertitore è capace di lavorare per tensioni in ingresso che variano da 0 a 5 V, come succede per il convertitore della scheda Arduino.

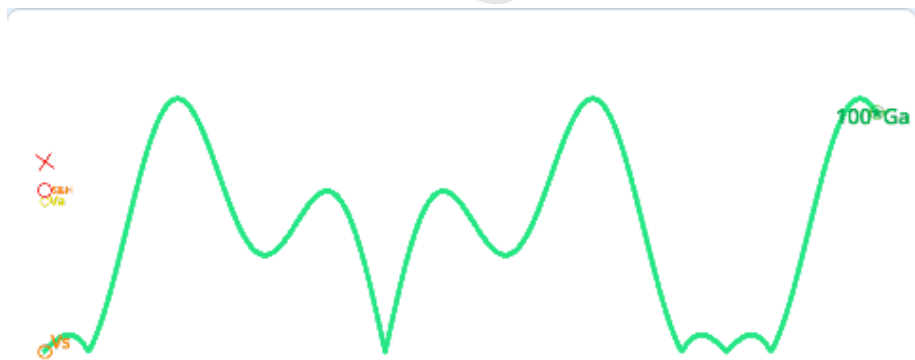
La variabile da codificare

La grandezza da codificare è costruita di default a partire dal valore assoluto di una sinusoide combinata con un quarta armonica:

$$s = abs[\sin(t) - 0,4 * \sin(4t)]$$

Si assume che sia una grandezza ambientale qualunque che la sonda convertirà in tensione nella gamma che va da 0 a poco più di 1V.

Con **[tasto F]** si fa disegnare la grandezza fisica amplificata 100 volte per poterla disegnare sullo stage in modo che occupi buona parte dello spazio a disposizione.



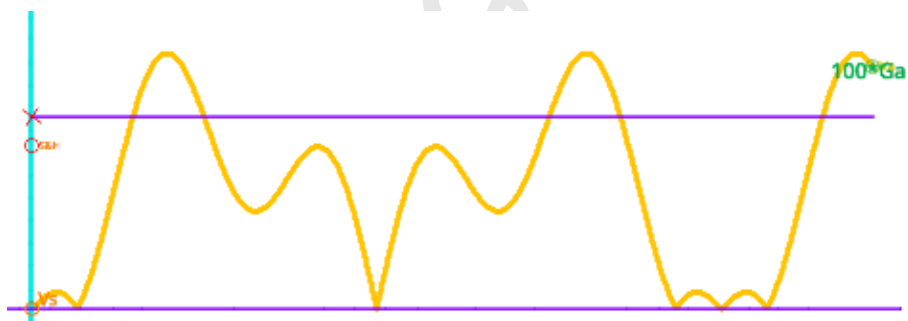
Con **[tasto V]** viene disegnata la tensione in uscita dell'amplificatore che ha un guadagno iniziale pari a 5 corrispondente al 100% delle sue prestazioni:



Si suppone che l'amplificatore sia molto preciso per cui la tensione in uscita riproduce in modo fedele la grandezza fisica da codificare.

Mentre la grandezza in ingresso ha un'ampiezza che è indipendente dalla volontà del misuratore e le prestazioni della sonda sono legate alla tecnologia dei trasduttori, l'amplificatore può essere costruito per assicurare al convertitore A/D di avere al suo ingresso una tensione variabile utile da convertire.

Con **[tasto A]** si disegnano gli assi così da evidenziare i limiti del convertitore A/D:

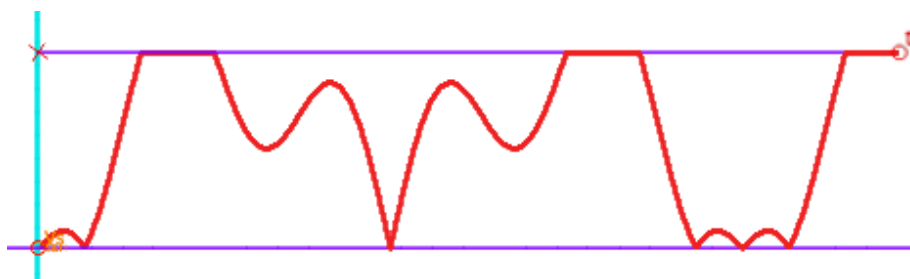


Si vede chiaramente che con un guadagno del 100%, la tensione in ingresso del convertitore supererà i limiti.

Si procede comunque per vedere l'effetto dell'uso dell'amplificatore.

Con **[tasto T]** si può vedere come la tensione verrà tagliata per effetto dei limiti della gamma del convertitore:

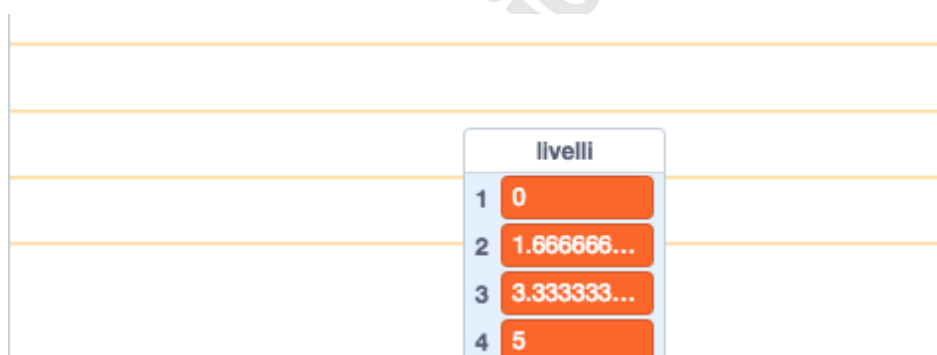
scheda microbit



Si tratta ora di prevedere come si comporterà il convertitore.

Per vedere meglio cosa accade, si opera nel caso di una conversione a 2 bit con un periodo di conversione pari a 10 tic.

Con una conversione a 2 bit sono disponibili 4 valori per la conversione digitale che sono visibili con **[tasto L]**:



La lista scompare premendo di nuovo **[tasto L]**.

Il convertitore dovrà decidere a quale livello deve appartenere la tensione rilevata al suo ingresso potendo assegnare solo uno dei quattro valori elencati.

La tensione in ingresso che ha infiniti valori ne potrà avere in uscita solo quattro:

0

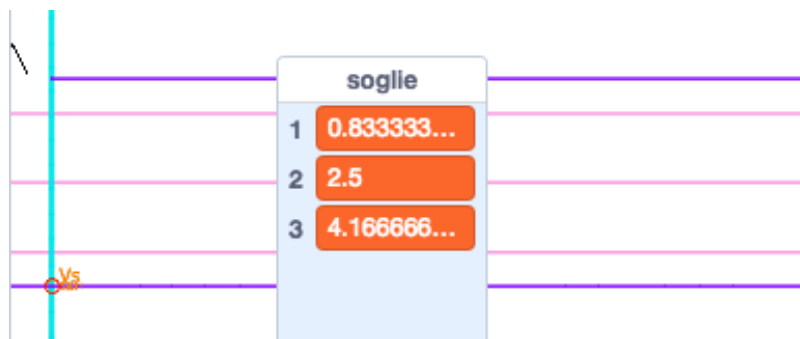
1,666

3,333

5.

La decisione viene presa in base al fatto che la tensione sia sopra o sotto le tre soglie che sono poste nel bel mezzo dell'intervallo che c'è fra i livelli.

Con **[tasto S]** si disegnano le soglie di conversione, i cui valori sono indicati nella lista che compare contemporaneamente. La lista scompare premendo di nuovo il **[tasto S]**:



Le soglie sono 3 e sono poste alle tensioni:

- 0,8333
- 2,5
- 4,1666.

Il convertitore oltre a non poter catturare tutti i valori possibili perché sono in quantità infinita, non può nemmeno farlo per tutti gli istanti di tempo possibili perché anche questi sono in quantità infinita.

Non c'è nessun computer che può fare infiniti calcoli in un tempo limitato e nessuna memoria che può contenere infiniti valori con infinite cifre decimali di precisione.

Il convertitore, inoltre, ha bisogno di tempo per decidere a quale intervallo deve appartenere la tensione in ingresso per cui un orologio (clock) scandisce ogni quanto tempo si fa la conversione.

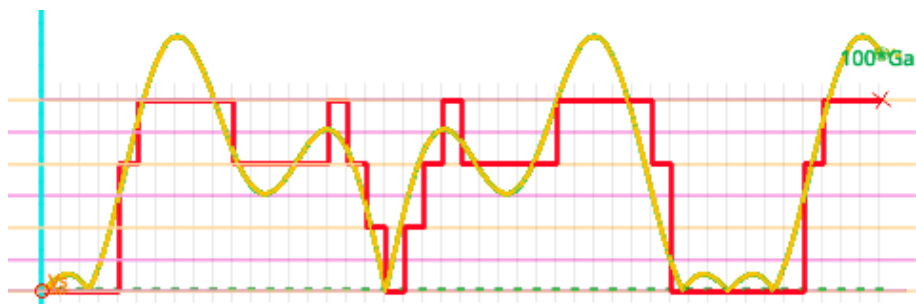
Con **[tasto X]** si disegnano gli istanti che devono essere usati per la conversione:



scheda microbit

La figura mostra una griglia con tutti i parametri che servono per fare la conversione, anche a mano.

Con **[tasto 2]** si avvia al conversione a 2 bit:

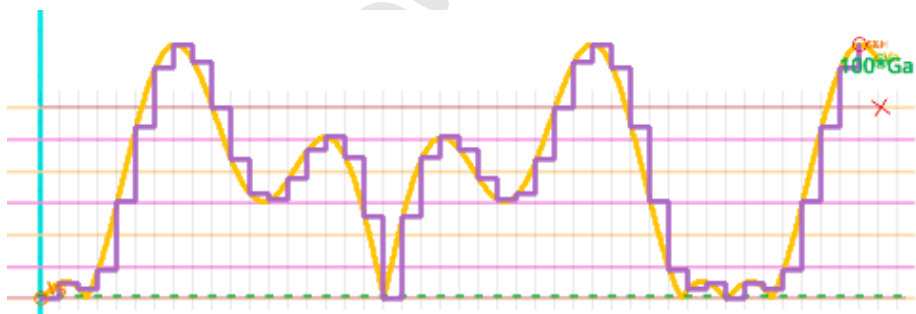


Per vedere tutte gli elementi della figura occorre fare disegnare anche gli assi, i livelli e le soglie.

La figura mostra solo l'esito ma non il meccanismo che lo ha generato.

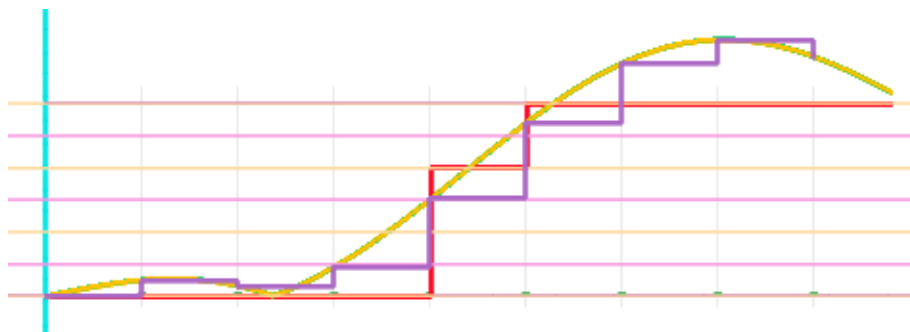
Si può cancellare tutto con on **[tasto E]** e disegnare la tensione V_a con on **[tasto V]**.

Se si preme **[tasto H]** si mostra la tensione che viene trattenuta da uno speciale componente (S&H) che deve catturare la tensione nell'istante stabilito (Sample) e mantenerla (Hold) fino all'istante successivo per consentire al convertitore di fare la conversione:



Per vedere bene cosa accade si ingrandisce la base dei tempi utilizzando $\text{scalaX} = 0.2$ per avere questa figura:

scheda microbit



si vedono:

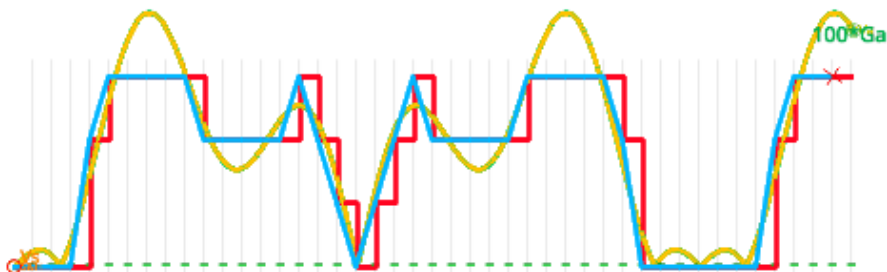
- fino al 4° istante (il primo coincide con l'asse delle y) la tensione (gialla) produce la tensione SH (viola) che è sempre sotto la prima soglia per cui il valore digitale appartiene al primo livello che vale 0;
- al 5° istante viene superata, seppur di poco, la seconda soglia per cui il valore digitale è 2;
- dal 6° istante in poi il valore supera abbondantemente la terza soglia ma il convertitore non può dare più di 3.

La lista conferma i suddetti valori:

	digit
1	0
2	0
3	0
4	0
5	2
6	3
7	3
8	3

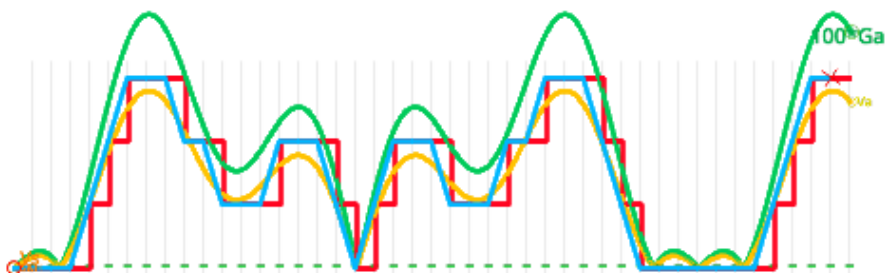
La spezzata viola, che rappresenta le tensioni S&H è uguale alla tensione in giallo negli istanti di campionamento, la spezzata rossa, che è l'uscita del convertitore, deve stare sui livelli prestabiliti.

Con **[tasto I]** si vede la tensione che esce da un convertitore D/A, digitale analogico, che effettua conversioni con modulazione di ampiezza (non è quello di Arduino) con interpolazione lineare:

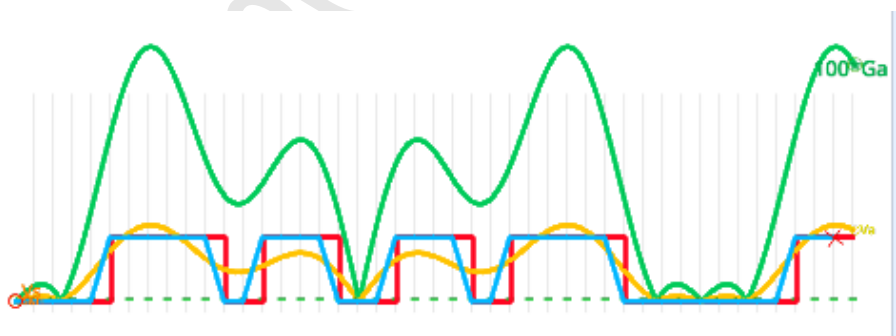


Non è per niente una buona approssimazione.

Con **[tasto G]** si porta il guadagno dell'amplificatore al 70% e con **[tasto R]** si ripete la conversione ottenendo questo risultato:



Se si esagera ad abbassare il guadagno, per esempio riducendolo al 30% si ottiene questo:



Il convertitore si trova a funzionare come se avesse un solo bit a disposizione in quanto fornisce solo due valori in uscita.