



**Dipartimento di Ingegneria
Industriale e dell'Informazione**

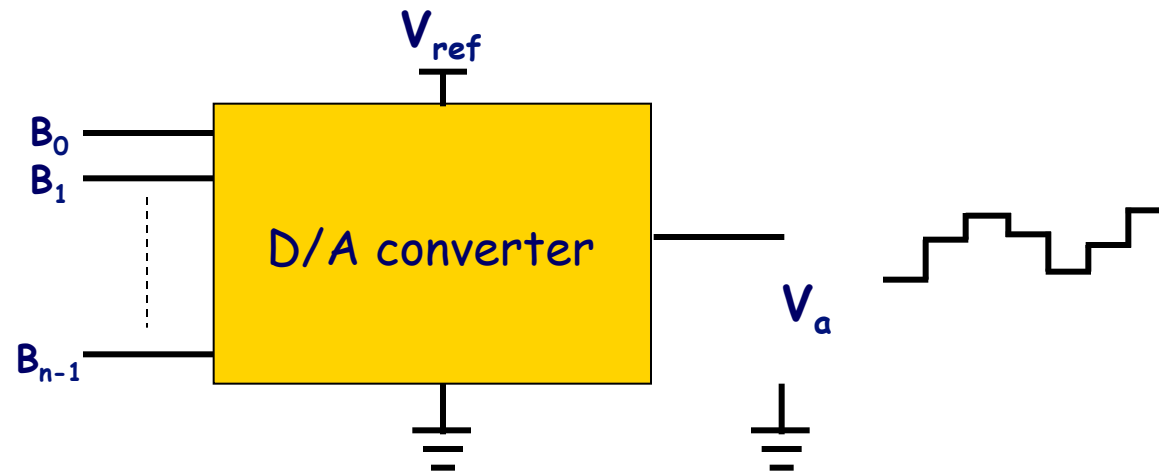


Realizzazione e programmazione di un convertitore digitale/analogico



Convertitore digitale/analogico

Un convertitore digitale/analogico (digital to analog converter) accetta in ingresso una parola digitale di n bit e la converte in una grandezza (tensione) analogica



In generale, il dato analogico all'uscita del convertitore D/A viene inviato ad un circuito di sample and hold

La forma d'onda a "gradinata" viene poi inviata ad un filtro passabasso, che completa la ricostruzione del segnale analogico



Convertitore digitale/analogico

- Il dato analogico all'uscita del convertitore, v_a , si ottiene in genere attribuendo a ciascuno dei bit che compongono la parola digitale all'ingresso un opportuno peso, dipendente dalla sua posizione

$$\begin{aligned} V_a &= V_{\text{ref}} \cdot \left(\frac{B_{n-1}}{2^1} + \frac{B_{n-2}}{2^2} + \dots + \frac{B_0}{2^n} \right) = \\ &= \frac{V_{\text{ref}}}{2^n} \cdot (B_{n-1} 2^{n-1} + B_{n-2} 2^{n-2} + \dots + B_0 2^0), \quad B_i = 0, 1 \end{aligned}$$

- Si osservi che il valore massimo di V_a , $V_{a,\text{max}}$, non è V_{ref} ma

$$V_{a,\text{max}} = \frac{V_{\text{ref}}}{2^n} \cdot (2^{n-1} + 2^{n-2} + \dots + 2^0) = V_{\text{ref}} \frac{2^n - 1}{2^n} = V_{\text{ref}} - \frac{V_{\text{ref}}}{2^n}$$




dove $V_{\text{ref}}/2^n$ è il valore analogico corrispondente al bit meno significativo (least significant bit, LSB)



Obiettivi dell'esperienza





Realizzazione di un sistema che consenta

-  la programmazione di un DAC a 8 bit attraverso 8 delle 24 porte digitali disponibili sulla piattaforma SC2075
-  il calcolo del valore teorico atteso all'uscita del convertitore sulla base della parola digitale impostata
-  l'acquisizione del dato analogico all'uscita del convertitore

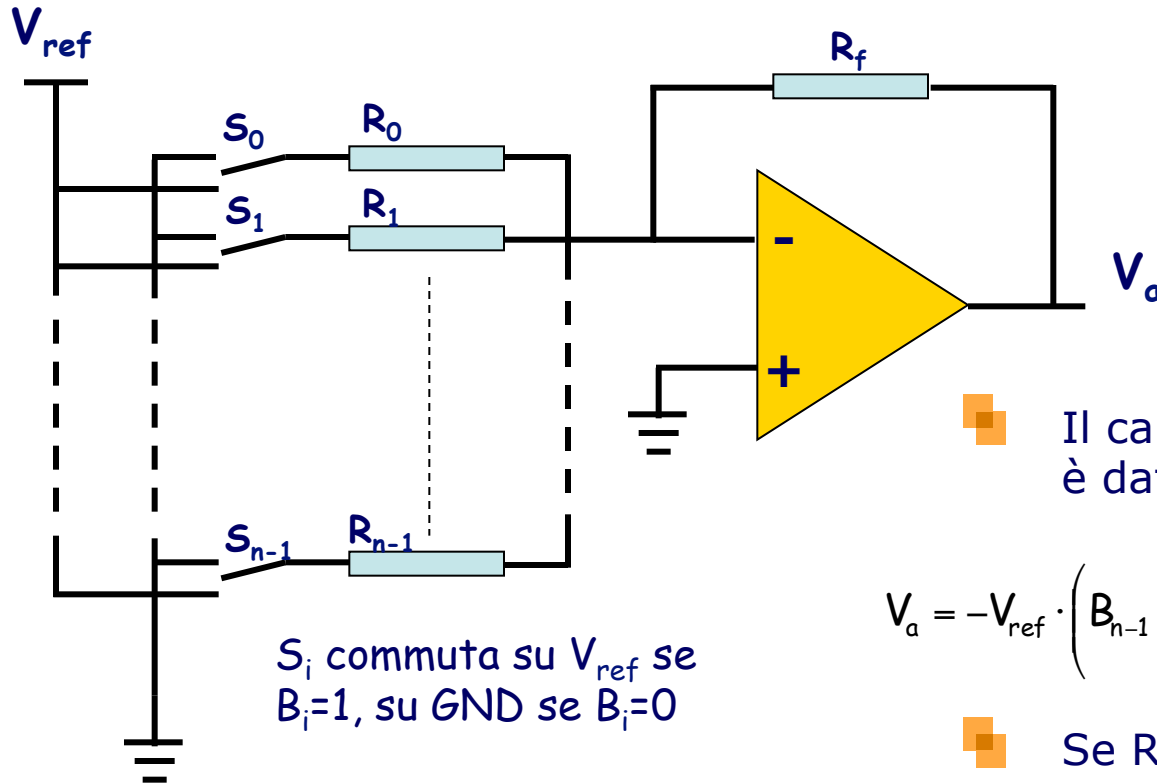


Il sistema dovrà includere

-  un DAC a resistori pesati, realizzato sulla breadboard
-  uno strumento virtuale realizzato in ambiente di programmazione LabView con funzione di interfaccia tra il sistema e l'utente



DAC a resistori pesati



Il campione analogico in uscita è dato da

$$V_a = -V_{ref} \cdot \left(B_{n-1} \cdot \frac{R_f}{R_{n-1}} + B_{n-2} \cdot \frac{R_f}{R_{n-2}} + \dots + B_0 \cdot \frac{R_f}{R_0} \right), \quad B_i = 0, 1$$

Se R_0 è la resistenza relativa al LSB, posto $R_i = R_0 / 2^i$, risulta

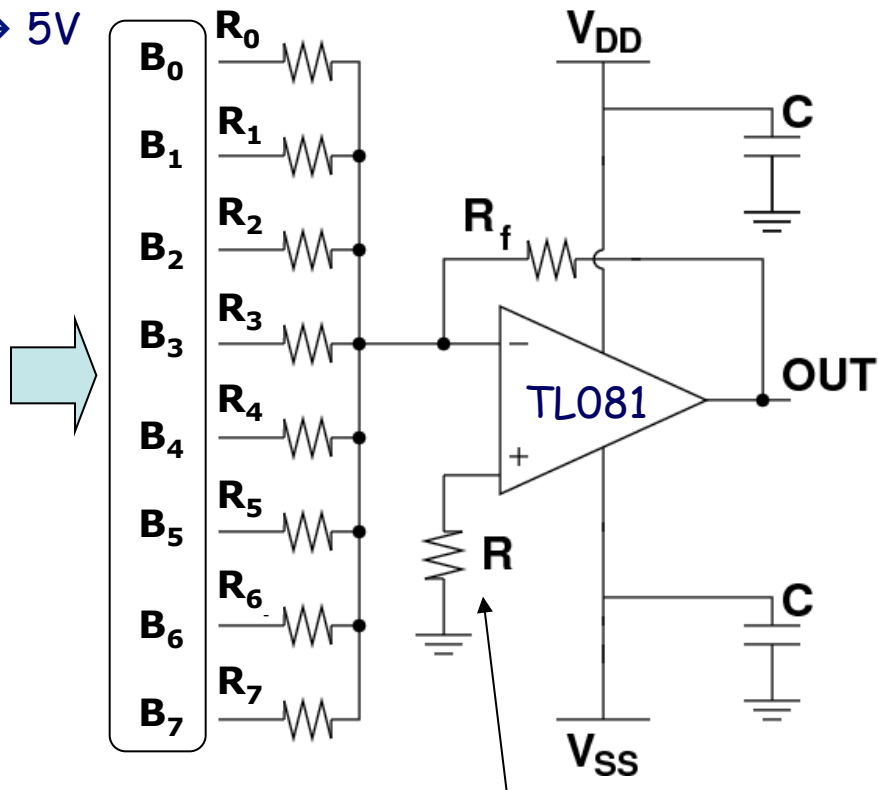
$$V_a = -\frac{R_f}{R_0} \cdot V_{ref} \cdot (B_{n-1} \cdot 2^{n-1} + B_{n-2} \cdot 2^{n-2} + \dots + B_0 \cdot 2^0), \quad V_{a,max} = -\frac{R_f}{R_0} \cdot V_{ref} (2^n - 1)$$

Realizzazione del DAC a resistori pesati

$B_i=0 \rightarrow 0V$

$B_i=1 \rightarrow 5V$

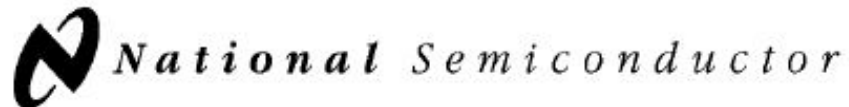
Dalla porta digitale della
scheda SC2075



resistenza di
compensazione delle
correnti di polarizzazione

$R_0 = 511k\Omega$
 $R_1 = 243k\Omega$
 $R_2 = 130k\Omega$
 $R_3 = 61.9k\Omega$
 $R_4 = 33.2k\Omega$
 $R_5 = 16.2k\Omega$
 $R_6 = 8.2 k\Omega$
 $R_7 = 3.92k\Omega$
 $R_f = 2 k\Omega$
 $R = 2 k\Omega$
 $C = 100 nF$
 $V_{DD} = 15 V$
 $V_{SS} = -15 V$

TL081 JFET input OpAmp



December 1995

TL081 Wide Bandwidth JFET Input Operational Amplifier

General Description

The TL081 is a low cost high speed JFET input operational amplifier with an internally trimmed input offset voltage (BI-FET II™ technology). The device requires a low supply current and yet maintains a large gain bandwidth product and a fast slew rate. In addition, well matched high voltage JFET input devices provide very low input bias and offset currents. The TL081 is pin compatible with the standard LM741 and uses the same offset voltage adjustment circuitry. This feature allows designers to immediately upgrade the overall performance of existing LM741 designs.

The TL081 may be used in applications such as high speed integrators, fast D/A converters, sample-and-hold circuits and many other circuits requiring low input offset voltage, low input bias current, high input impedance, high slew rate and wide bandwidth. The devices has low noise and offset voltage drift, but for applications where these requirements

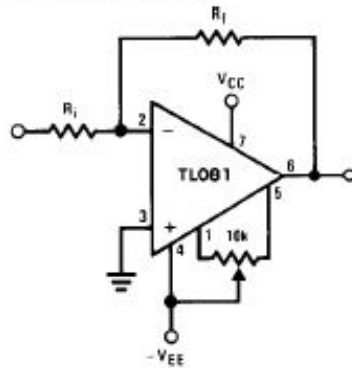
are critical, the LF356 is recommended. If maximum supply current is important, however, the TL081C is the better choice.

Features

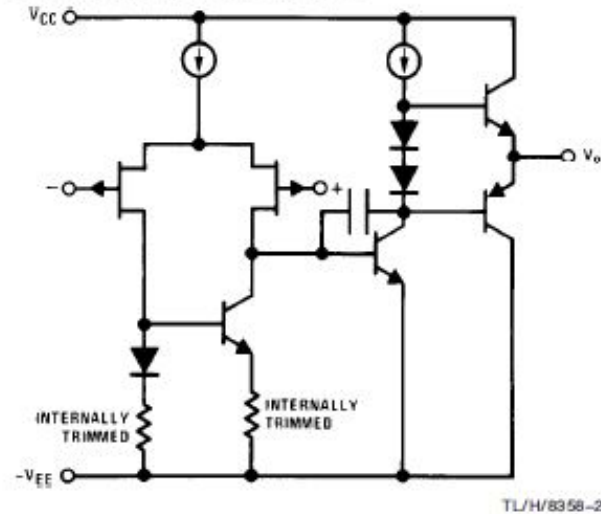
■ Internally trimmed offset voltage	15 mV
■ Low input bias current	50 pA
■ Low input noise voltage	25 nV/ $\sqrt{\text{Hz}}$
■ Low input noise current	0.01 pA/ $\sqrt{\text{Hz}}$
■ Wide gain bandwidth	4 MHz
■ High slew rate	13 V/ μs
■ Low supply current	1.8 mA
■ High input impedance	$10^{12}\Omega$
■ Low total harmonic distortion $A_V = 10$, $R_L = 10\text{k}$, $V_O = 20\text{ Vp-p}$, $\text{BW} = 20\text{ Hz} - 20\text{ kHz}$	<0.02%
■ Low 1/f noise corner	50 Hz
■ Fast settling time to 0.01 %	2 μs

TL081 JFET input OpAmp

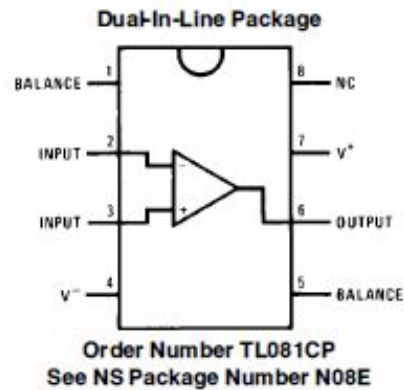
Typical Connection



Simplified Schematic



Connection Diagram



Pannello frontale

Controllo per la programmazione della parola digitale (cluster di controlli booleani)

Sezione di acquisizione (del dato analogico)

The front panel is divided into three main sections:

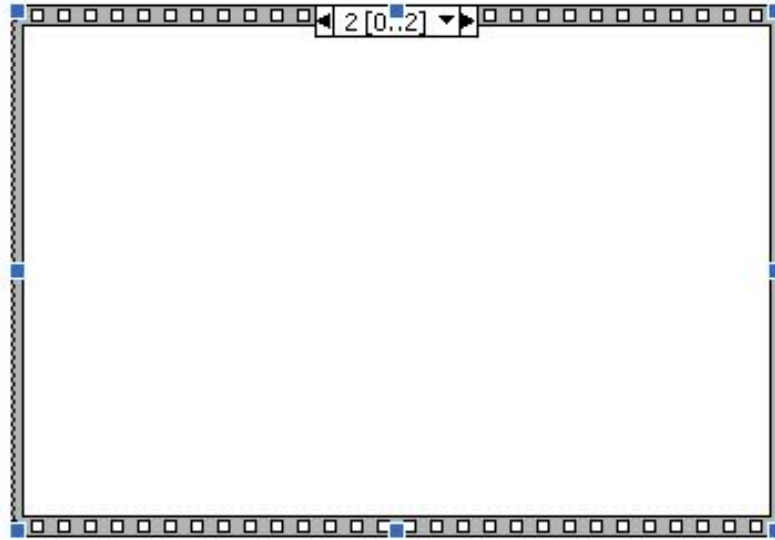
- Digital Control Section (Left):** Includes a 'Digital Device' selector set to 1, a 'Stop' button, an 'Error' status box with Boolean, Numeric, and String indicators, a 'DAQ Digital Channel Name' dropdown set to 0, and a 'Switch' panel with 8 toggle switches labeled 7 to 0.
- Analog Acquisition Section (Right):** Includes an 'Analog device' selector set to 1, a 'DAQ Analog Channel Name' dropdown set to 1, 'Number of measures' set to 100, 'Max. value' set to 0.00, 'Min. value' set to -5.00, and a 'Measured DAC Output' display showing 0.000.
- Digital Value Section (Bottom):** Displays eight resistors (R0 to R7) with values: R0=512.00, R1=256.00, R2=128.00, R3=64.00, R4=32.00, R5=16.00, R6=8.00, R7=4.00. It also includes a 'V Ref' set to 5.00 and a 'Calculated DAC Output' display showing 0.000.

Sezione di controllo (della parola digitale)

Valore teorico di V_a

Struttura sequence

Dal punto di vista dello schema a blocchi, lo strumento virtuale LabView può essere realizzato mediante una struttura *sequence* con tre *frame* (per aggiungere un frame fare click con il pulsante destro del mouse sulla cornice e selezionare “Add Frame After”). Questa struttura consente l’esecuzione di una serie di istruzioni secondo una sequenza definita dal programmatore (vengono eseguite prima le istruzioni contenute nel *frame* 0, poi quelle contenute nel *frame* 1, etc.)



La proposta di utilizzare questa struttura ha il solo scopo di rendere possibile una distinzione temporale tra le tre fasi dell’attività svolta dal programma:

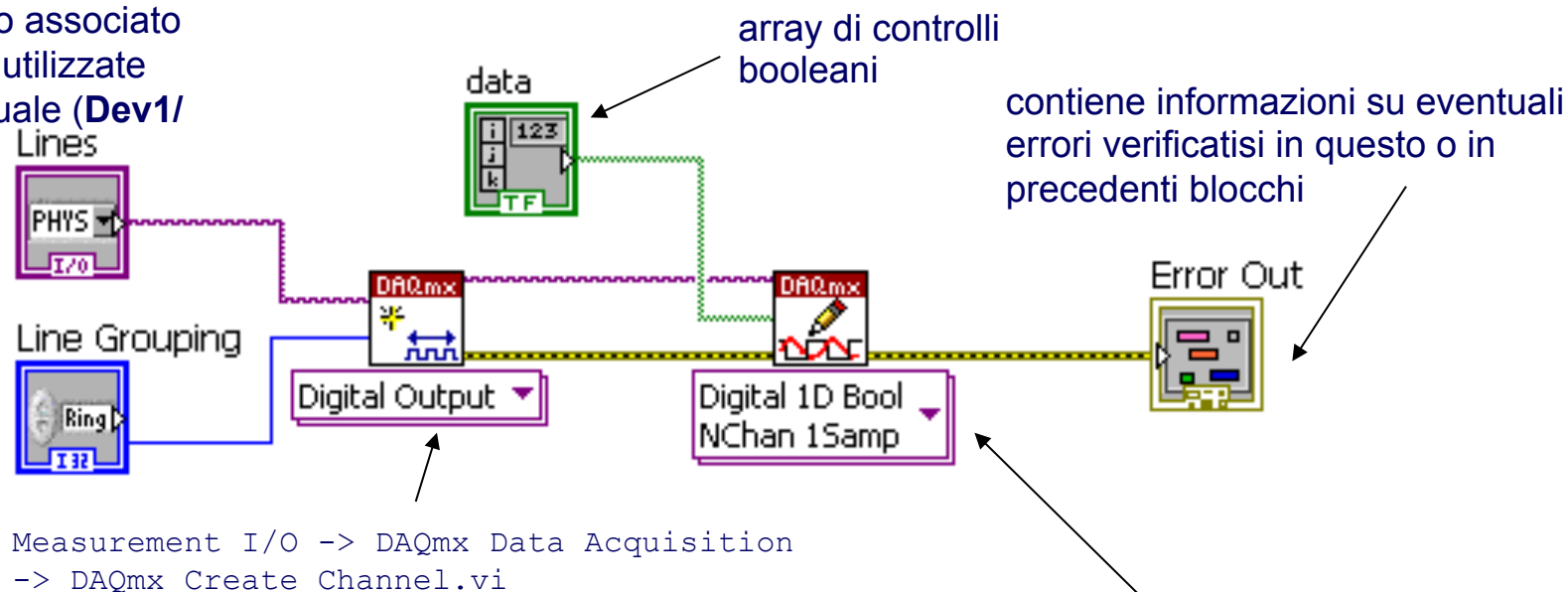
- **programmazione del DAC** (frame #0)
- **calcolo del valore teorico del dato analogico all’uscita del DAC** (frame #1)
- **acquisizione del dato analogico reale** (frame #2)

Programmazione del DAC (frame #0)

- DAQmx Create Channel.vi provvede alla configurazione del canale digitale della scheda di acquisizione
- DAQmx Write.vi configura le uscite digitali della scheda individuate dal blocco precedente con i dati forniti dall'opportuno blocco di controllo

specifica il nome delle linee digitali o il numero identificativo associato alle porte che vengono utilizzate per creare il canale virtuale (**Dev1/port0/line0:7**)


modalità di raggruppamento delle linee digitali in uno o più canali virtuali (**one channel for each line**)



Measurement I/O -> DAQmx Data Acquisition -> DAQmx Write.vi



Calcolo del valore teorico di V_a (frame #1)

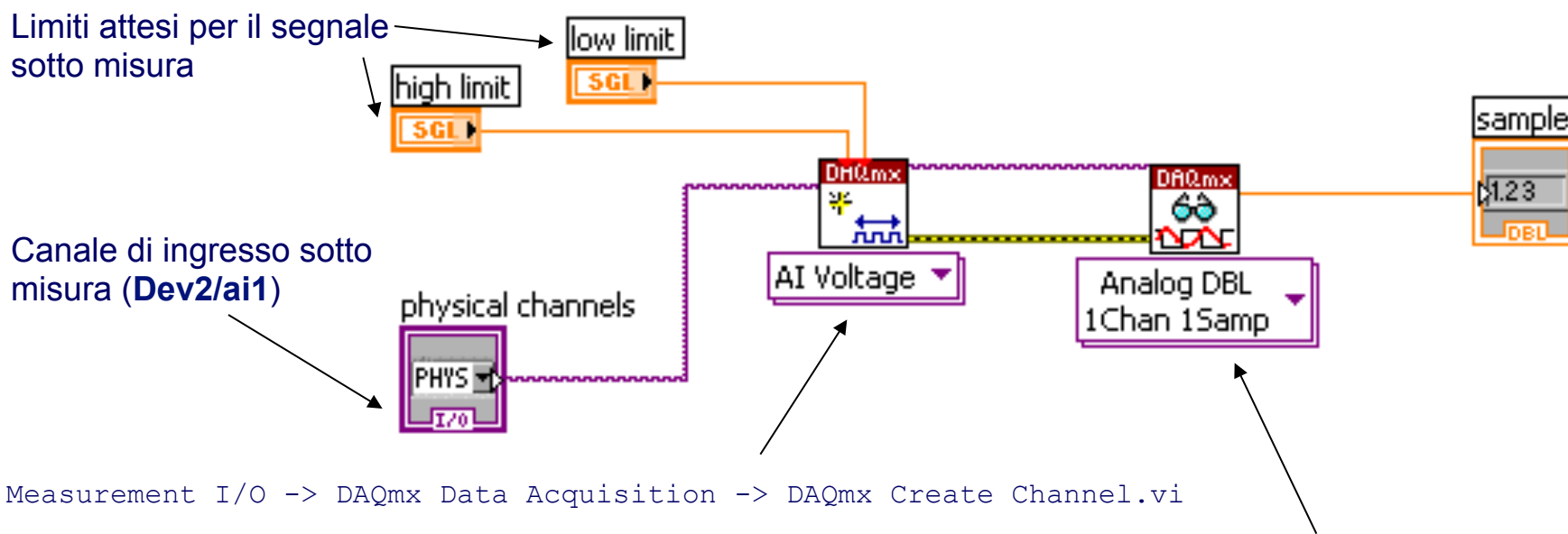
 Il frame #1 dovrà provvedere al calcolo del valore teorico dell'uscita del convertitore D/A sulla base della parola binaria impostata. Per far questo sarà necessario applicare la relazione già vista:

$$V_a = -V_{ref} \cdot \left(B_7 \cdot \frac{R_f}{R_7} + B_6 \cdot \frac{R_f}{R_6} + \dots + B_0 \cdot \frac{R_f}{R_0} \right), \quad B_i = 0, 1$$



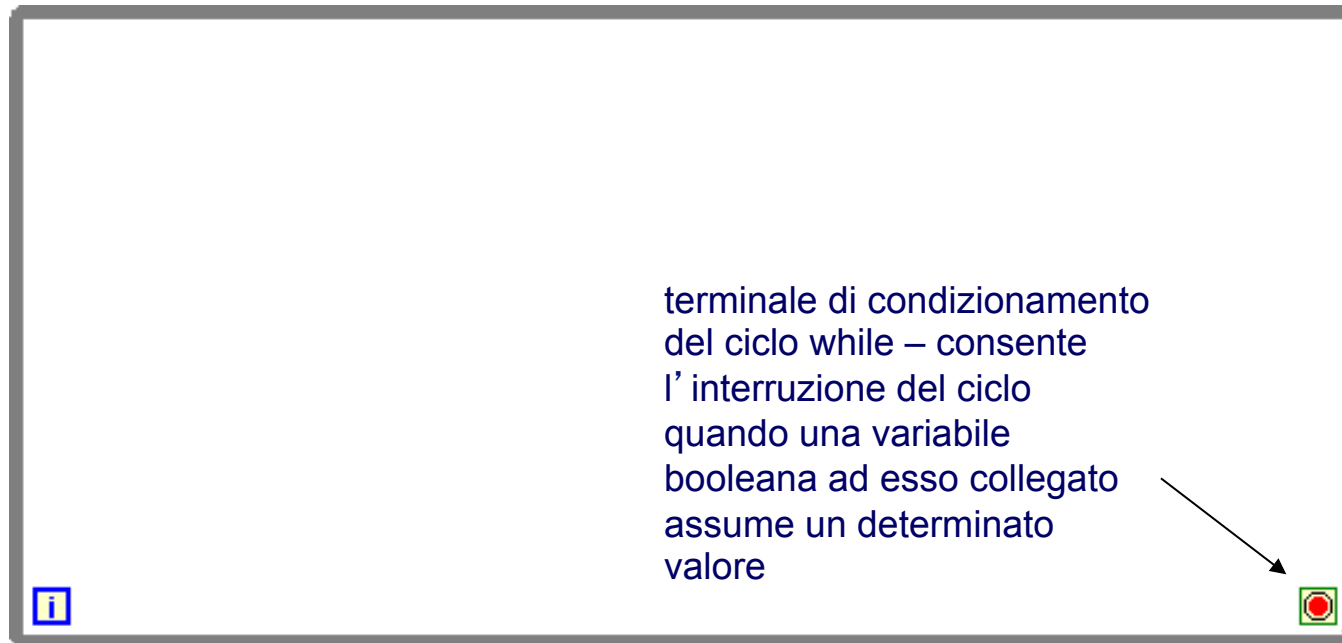
Acquisizione del dato analogico reale (frame #2)

- DAQmx Create Channel.vi fornisce alla scheda di acquisizione informazioni riguardanti il tipo e il range dei segnali che deve ricevere in ingresso e il canale da cui deve leggere questi dati
- DAQmx Read.vi campiona il segnale proveniente dal canale specificato e riporta il valore misurato



Ciclo while

Un ciclo while (si trova nel menu `Structures` della Functions palette) può essere utilizzato per rendere possibile la conversione in tempo reale della parola digitale (si trova nel menu `Structures` della Functions palette). Il programma deve consentire l'interruzione dell'acquisizione mediante un interruttore booleano (pulsante "stop")

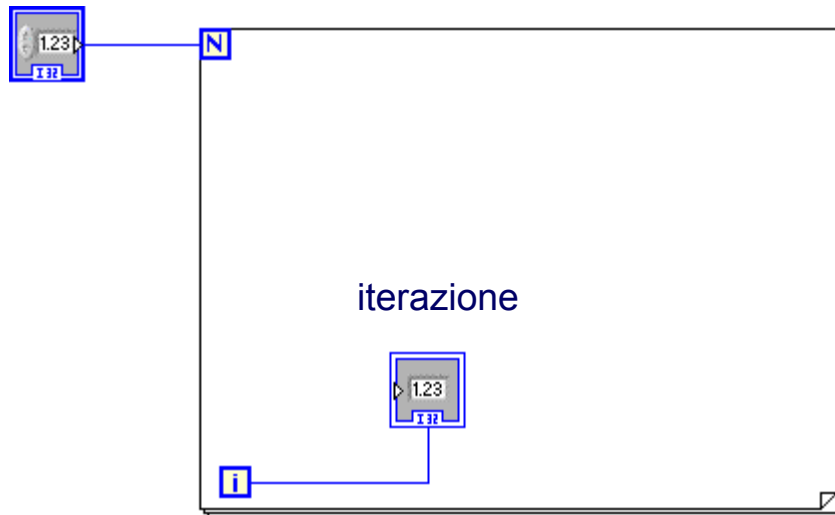




Ciclo for

- Utile per migliorare l'accuratezza della misura (frame #2) poiché consente di ridurre l'effetto dei disturbi a media nulla sovrapposti alla tensione da misurare

numero di cicli



- Invece di rappresentare ogni singolo campione acquisito si rappresenta la media di **n** campioni; di conseguenza la velocità con cui i dati misurati vengono riportati sul grafico si riduce di un fattore **n**





Suggerimenti e possibili miglioramenti

- Può risultare utile ed interessante confrontare il dato teorico con il dato acquisito tramite lo strumento virtuale ed il dato acquisito mediante voltmetro digitale. Sarebbe opportuno fornire una spiegazione delle (inevitabili) differenze tra i tre gruppi di dati
- Modificare lo strumento virtuale in modo tale che sia in grado di riportare, in forma di tabella numerica in un file ed in forma di grafico sul pannello frontale, la caratteristica ingresso-uscita teorica e misurata del DAC
- Modificare lo strumento virtuale in modo tale che sia in grado di fornire, in forma grafica, la non linearità differenziale (DNL) e la non linearità integrale (INL) del DAC



Rappresentazione grafica della caratteristica I/O e di DNL e INL

- XY graph (Modern-> Graph ->XY Graph o Classic ->Classic Graph ->XY Graph)

- Richiede l'uso di una funzione Bundle (Programming-> Cluster, Class & Variant-> Bundle) per raggruppare in un unico array le variabili indipendente (X) e dipendente (Y)

ingressi in forma di vettore (pari dimensioni)

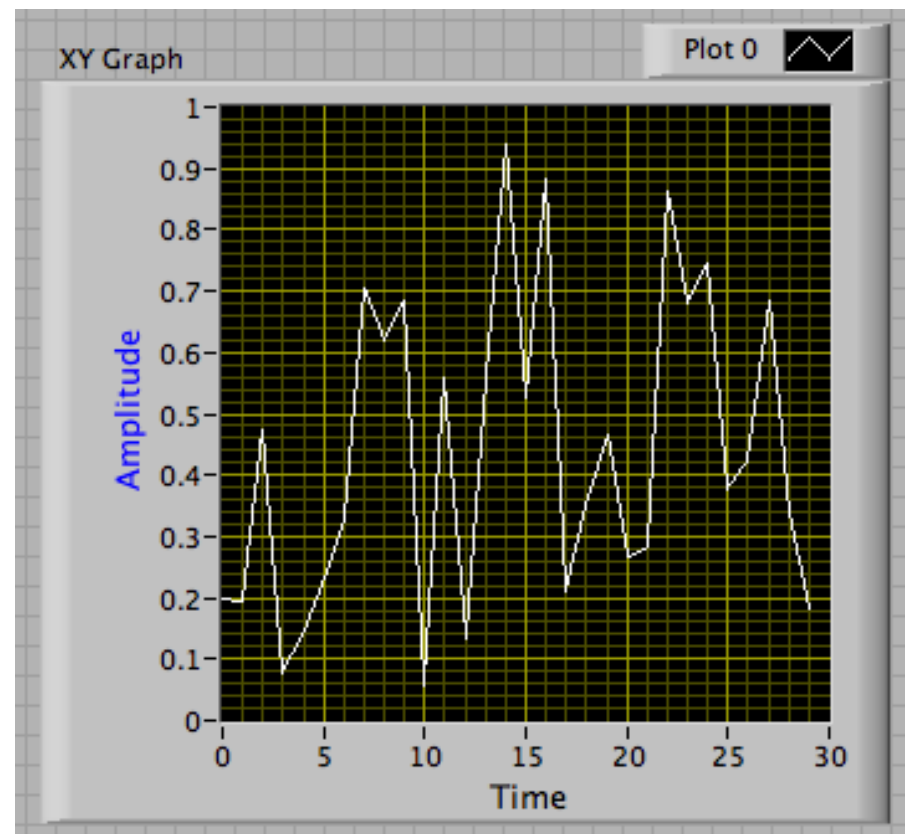
Single Plot XY Graph:

x array

y array

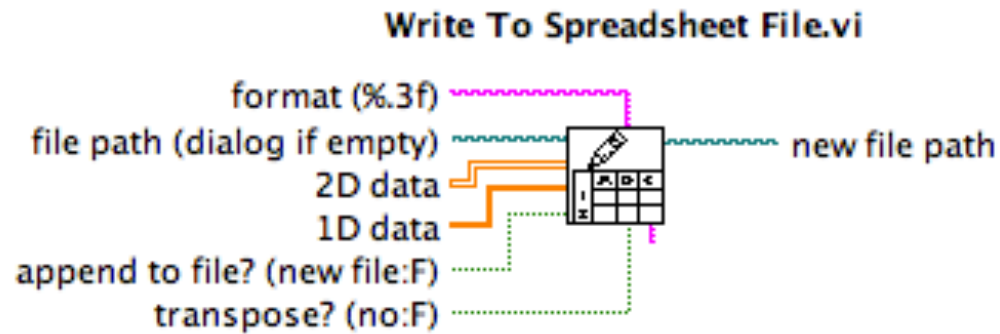


XY Graph
(Single Plot)



Suggerimento: per la generazione dei vettori si potrebbero utilizzare i "tunnel" di una struttura for in modalità "indexed"

Scrittura su file

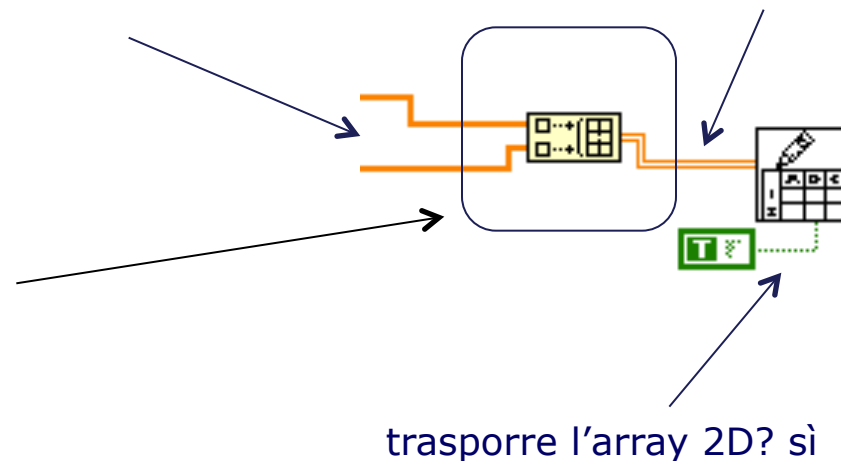


Programming-> File I/O->
Write To Spreadsheet File.vi

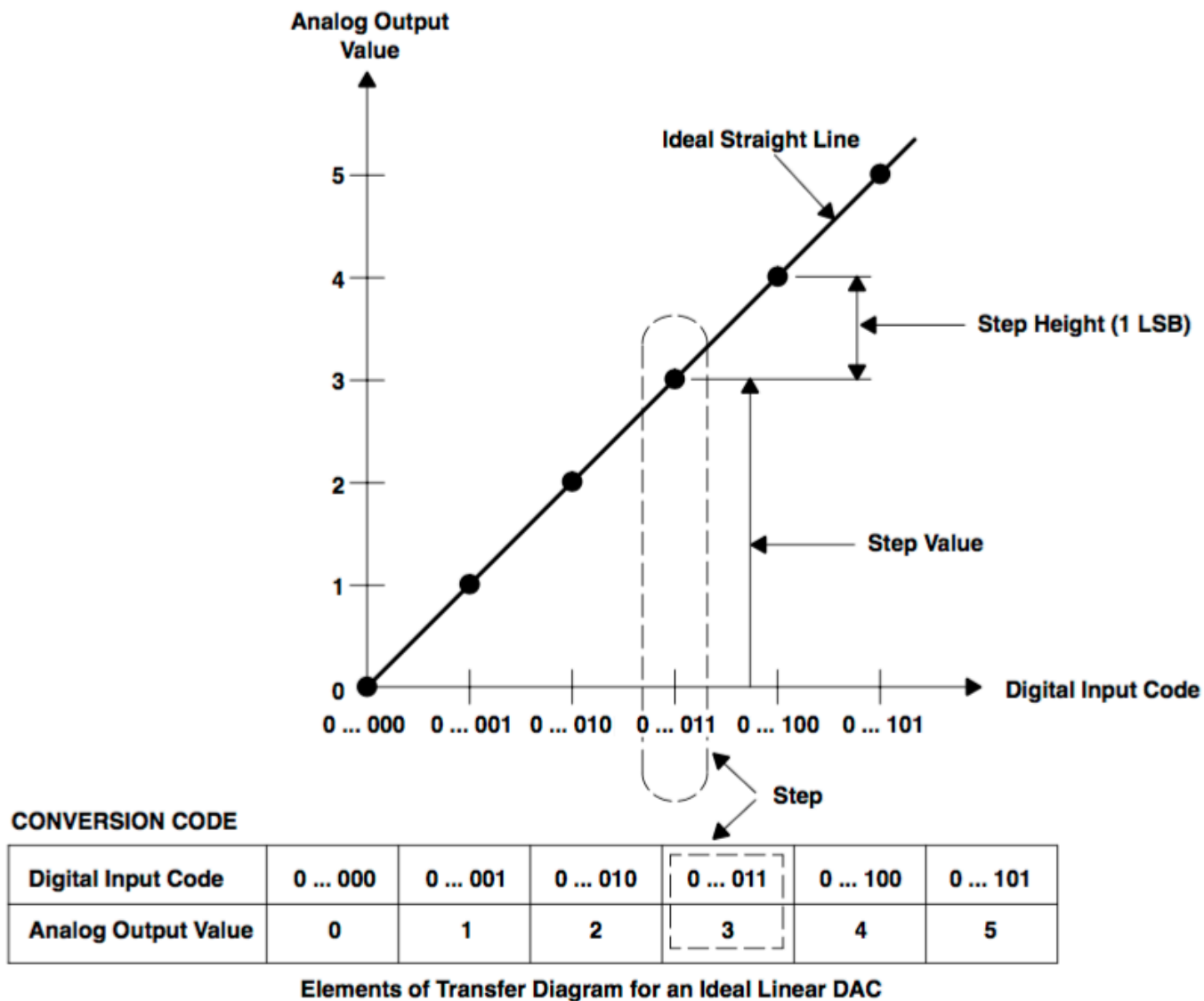
Programming-> Array->
Build Array

2 array 1D in ingresso

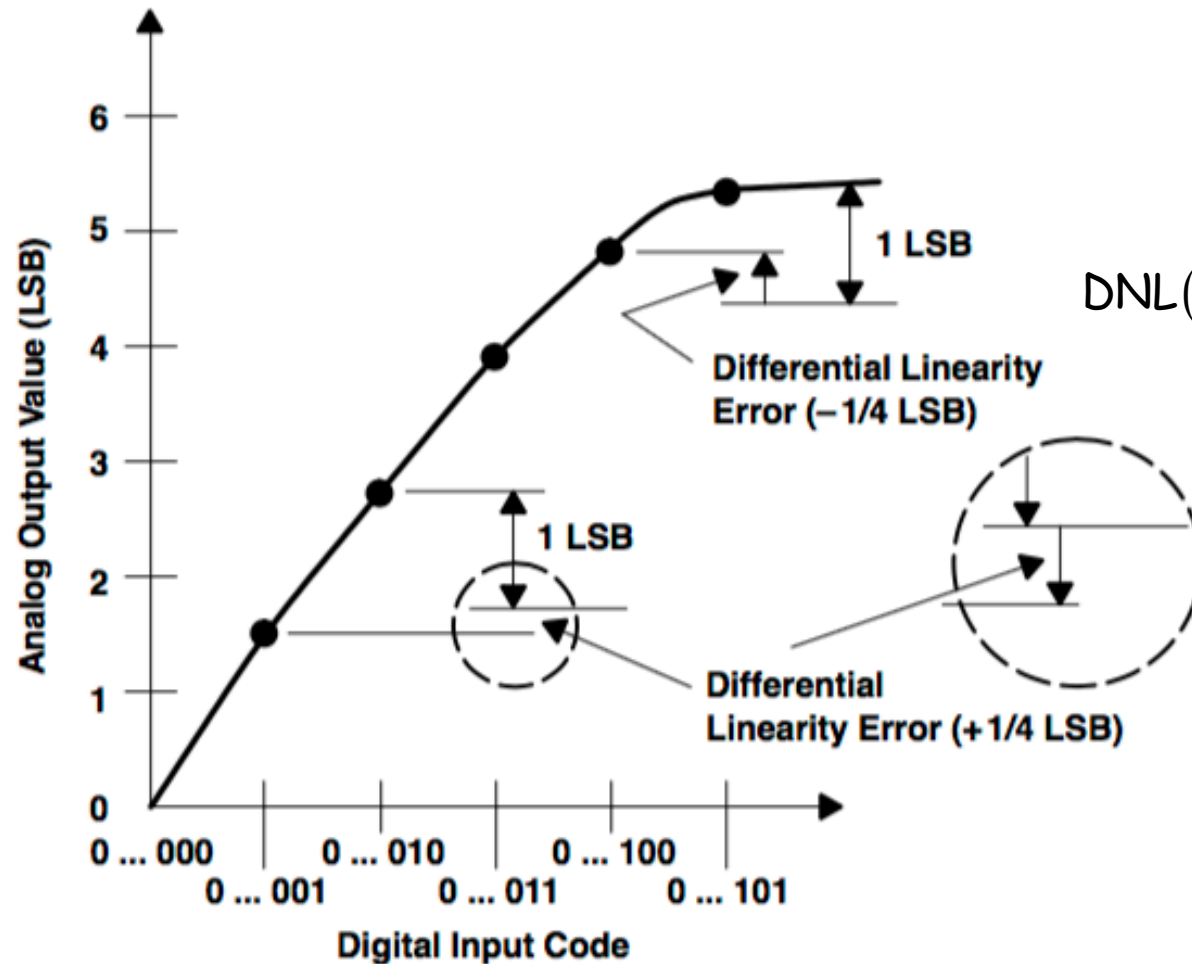
1 array 2D in uscita
(tabella di dati a 2
colonne)



Caratteristica ingresso-uscita



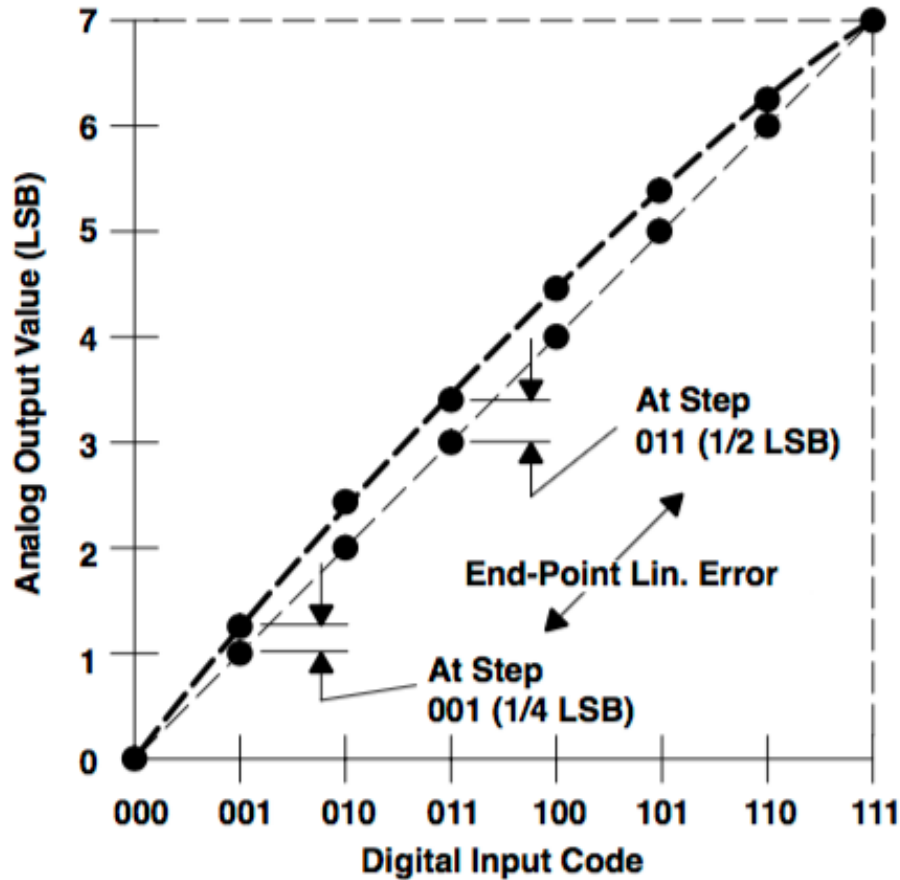
Non-linearità differenziale



$$DNL(k) = \frac{V_{out}(k+1) - V_{out}(k)}{LSB} - 1, k \in [1, 2^n - 1]$$

$$LSB = \frac{V_{REF}}{2^n}$$

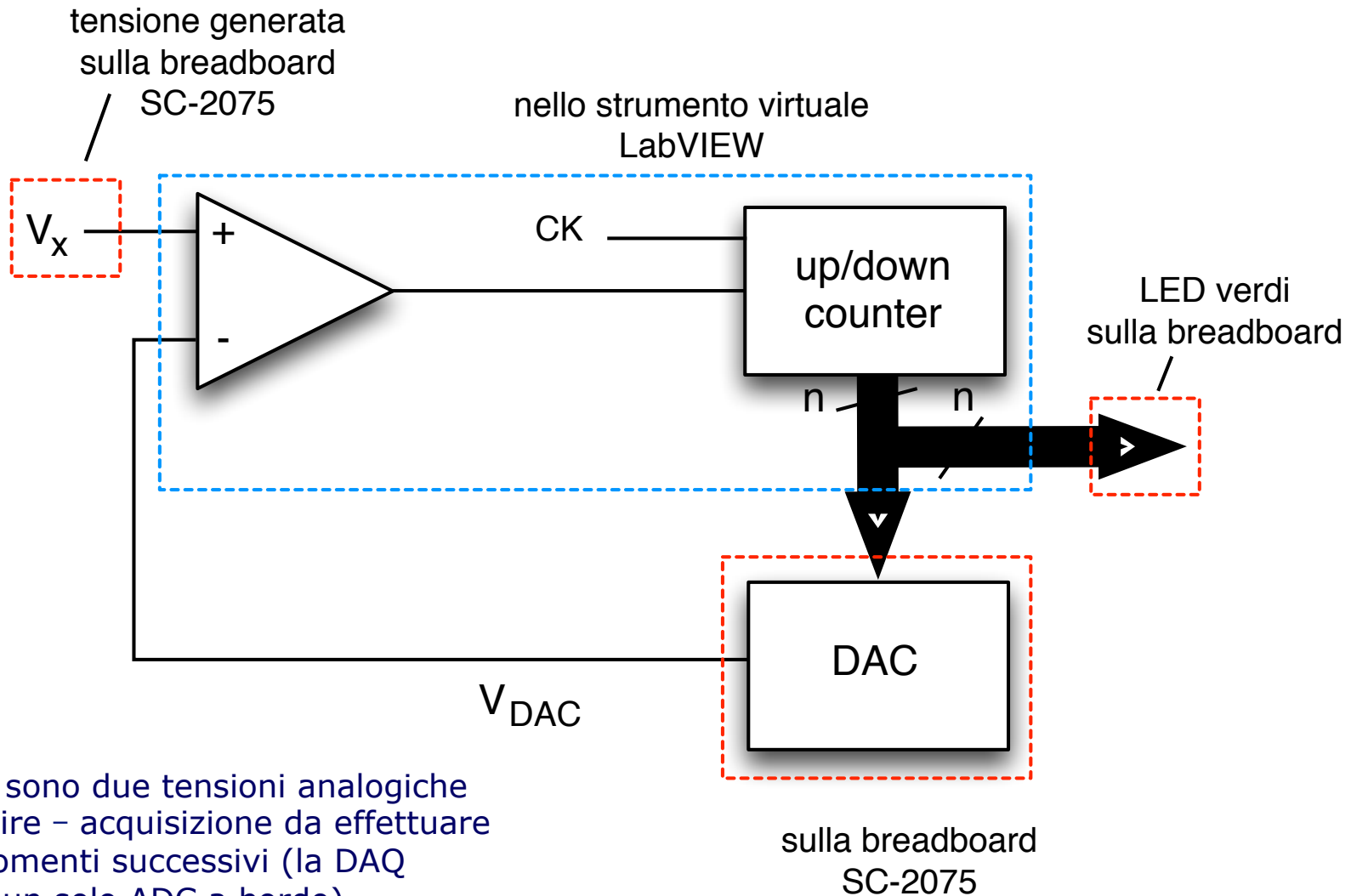
Non-linearità integrale



$$INL(k) = \frac{V_{out}(k) - k \cdot LSB}{LSB}, k \in [0, 2^n - 1] = \frac{V_{out}(k)}{LSB} - k$$

$$LSB = \frac{V_{out}(2^n - 1) - V_{out}(0)}{2^n - 1}$$

ADC ad inseguimento



V_x e V_{DAC} sono due tensioni analogiche da acquisire – acquisizione da effettuare in due momenti successivi (la DAQ board ha un solo ADC a bordo)

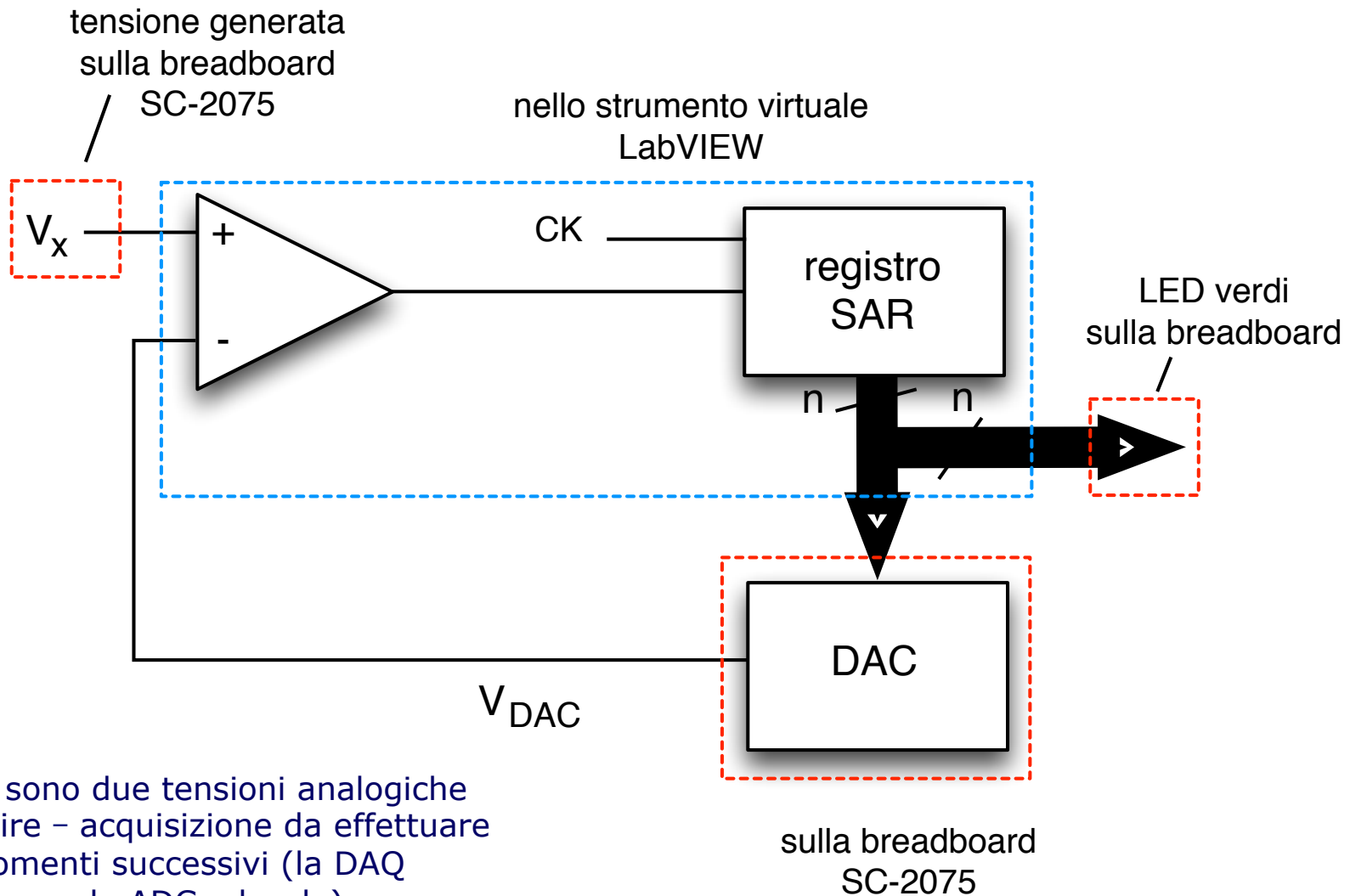


Struttura del VI LabVIEW

- Porre a 0 tutti i bit di ingresso del DAC. In una struttura Sequence all'interno di un ciclo While
 - acquisire V_x
 - acquisire V_{DAC}
 - confrontare V_x e V_{DAC} ; se $V_x > V_{DAC}$ incrementare di un'unità la parola all'ingresso del DAC, altrimenti decrementarla
- Alla fine della conversione, i bit di ingresso del DAC sono i bit di uscita dell'ADC.
- L'ADC funziona anche quando si fa variare (lentamente) la tensione V_x al suo ingresso




ADC ad approssimazioni successive (SAR)



V_x e V_{DAC} sono due tensioni analogiche da acquisire – acquisizione da effettuare in due momenti successivi (la DAQ board ha un solo ADC a bordo)



Struttura del VI LabVIEW

- 
- Acquisire V_X . In una struttura Sequence all'interno di un ciclo For (con un numero di cicli pari al numero di bit)
- porre a 1 il bit più significativo all'ingresso del DAC e a 0 tutti gli altri
 - acquisire V_{DAC}
 - confrontare V_X e V_{DAC} ; se $V_X > V_{DAC}$ lasciare a 1 il bit più significativo, altrimenti azzerarlo
 - porre a 1 il secondo (a partire dal MSB) bit
 - acquisire V_{DAC}
 - confrontare V_X e V_{DAC} ; se $V_X > V_{DAC}$ lasciare a 1 il secondo bit, altrimenti azzerarlo
 - ripetere la medesima operazione sui rimanenti bit.

