

Diseño de un conversor A/D

Juan N. Zambrano C.¹

Resumen

En el presente artículo se hace primero una introducción a los convertidores analógico-digitales, luego se resalta la importancia del proceso de conversión en el mundo actual y la necesidad de hacerlo y más adelante se presentan las definiciones más importantes relacionadas con el tema y se muestran algunos cálculos sencillos de las características de un convertidor A/D. Luego se muestra el diseño y la codificación utilizada para hacerlo, seguidamente se hace el análisis del convertidor diseñado y se termina el artículo con un análisis de error.

Palabras clave

Cuantización, niveles de cuantización, bits, resolución del conversor, rango dinámico, Simulink, bloques.

Abstract

The present article starts with an introduction to the analog-to-digital converters, highlighting the importance of the conversion process in the current world and the need to do it, then the most important definitions related to the topic are given and some simple calculations of the characteristics of an A/D converter are shown. Afterwards, the design and the codification of one of these are examined, followed by the analysis of the designed converter, and the article finishes with an error analysis.

Keywords

Quantification, quantification levels, bits, converter resolution, dynamic range, Simulink, blocks.

¹ Universidad EAFIT, Medellín, jzambra3@eafit.edu.co, juanchoza@gmail.com.

1. Introducción

En el tratamiento con señales es conveniente, la mayoría de las veces, expresar señales definidas en tiempo continuo (para todo instante), es decir análogas, como señales en tiempo discreto, es decir digitales. Las primeras son llamadas así porque generalmente tienen la misma forma de onda de la variable física que representan; las segundas son cadenas de dígitos binarios que no se parecen a la forma de onda del fenómeno físico y por tanto son llamadas no análogas o digitales. Es importante aclarar que una señal análoga es continua en el dominio (tiempo) pero no necesariamente en el rango, ya que puede tener discontinuidades de tipo no evitable. De la misma forma, un sistema que recibe y genera señales análogas se llama sistema análogo, y un sistema que recibe y genera señales digitales es llamado sistema digital. Sin embargo, un sistema análogo puede ser modelado como un sistema digital por conveniencia de análisis y diseño, como es usual en sistemas de control. Veamos algunas diferencias entre las dos técnicas (Chi-Tsong, 2006):

- Las señales digitales son codificadas en cadenas de dígitos binarios que representan niveles de voltaje. Este tipo de representación es menos susceptible al ruido y a las corrientes parásitas de las fuentes de poder.
- La precisión de los sistemas y transductores análogos es limitada, mientras que en los dispositivos digitales se puede incrementar simplemente aumentando el número de bits, lo que los hace más precisos y más confiables.
- Los sistemas digitales son más flexibles y versátiles que los sistemas análogos que, una vez construidos, no pueden ser modificados sin tener que reemplazar algunos componentes o el sistema completo. Los sistemas digitales se pueden

modificar, la mayoría de las veces, por programación.

- Debido al avance de la tecnología VLSI (very-large-scale integration) los sistemas digitales son cada vez más económicos, aun para sistemas de control pequeños.

En la siguiente sección definiré algunos términos necesarios en el proceso de diseño de un sistema que puede hacer la conversión de una señal análoga a una señal digital, al cual llamaré de ahora en adelante conversor A/D. Luego mostraré gráficamente la forma de una señal que ha sido digitalizada y finalmente haré un análisis del error que se presenta en el proceso.

2. Proceso de conversión

El conversor A/D consta básicamente de tres etapas: el muestreo, la cuantificación y la codificación de la señal de salida para poderla visualizar. Veamos cómo se desarrolla cada una de ellas en nuestro diseño:

2.1 Muestreo

Cuando una señal continua entra al conversor para ser cuantificada, éste no puede tomar todos los valores de ella debido precisamente a su carácter continuo. Por ello debe tomar muestras de la función cada determinado tiempo, llamado periodo de muestreo, y debe ser definido por el usuario de acuerdo a las capacidades de software y hardware que posea. En nuestro caso utilizaremos periodos de muestreo uniforme aunque ello puede introducir ambigüedad en la señal obtenida e impone una restricción esencial: la máxima frecuencia analógica (ver teorema de Nyquist: Soria et al., 2003, p. 4) que podemos recuperar tras un muestreo de la señal a F_m (frecuencia de muestreo) es $F_{\max} = \frac{1}{2}F_m$ donde $F_m = \frac{1}{T}$, de donde:

$$F_{\max} = \frac{1}{2T}$$

2.2 Cuantificación

La cuantificación se define como la conversión de una señal en tiempo discreto con valores continuos en una señal en tiempo discreto con valores discretos (señal digital). El valor de cada muestra se representa mediante un valor seleccionado de un conjunto finito de valores (niveles de cuantificación) (Soria et al., 2003). La cuantificación es un proceso irreversible, no invertible, ya que siempre se produce una pérdida de información. Teóricamente esto se fundamenta en el hecho de que al existir un número discreto y finito de niveles para clasificar un conjunto continuo de valores, a diferentes valores continuos les corresponderá la misma clase, es decir el mismo valor discreto, por lo que no se pueden obtener valores originales a partir de valores cuantificados.

En el proceso de cuantificación se debe tener en cuenta el rango de operación (ΔV) del conversor, también conocido como rango dinámico, el cual se calcula como:

$$\Delta V = V_{\max} - V_{\min}$$

Igualmente se debe tener en cuenta el número de bits del conversor o longitud de palabra del sistema (n) y se puede definir el máximo número posible de niveles de cuantificación como 2^n , representado por los enteros $\{0, 1, 2, \dots, 2^n - 1\}$:

$$N_{\max} = 2^n - 1$$

Luego la resolución del cuantificador será:

$$dv = \frac{\Delta V}{2^n}$$

Lo cual significa que un conversor de 4 bits, con un rango de 0 a 10 voltios, tendrá 8 niveles de cuantificación, es decir, a los infinitos valores continuos en $[0,10]$ sólo se les podrá asignar²² un número entero entre 0 y 7. Y dicho conversor tendrá una resolución de:

$$dv = \frac{10}{16} = 0.625 \text{ voltios}$$

Si consideramos que en nuestro caso la aproximación se hará por truncamiento, entonces el error máximo posible que se puede cometer se llama error por truncamiento y será: $e_{tru} < 0.625 \text{ voltios}$

El conversor sólo reconocerá valores de entrada que no excedan el rango dinámico del mismo. Cuando esto suceda la señal será recortada y producirá graves distorsiones, las cuales se conocen como ruido de sobrecarga. La calidad de la salida del conversor, que es la relación señal-ruido de cuantificación (SQNR, por sus siglas en inglés) (Soria et al., 2003), define la relación entre la potencia de la señal y la del ruido:

$$SQNR = \frac{P_{\text{señal}}}{P_{\text{ruido}}} = \frac{3}{2} 2^{2n}$$

$$P_{\text{señal}} = \frac{A^2}{2}, \quad P_{\text{ruido}} = \frac{\frac{A^2}{3}}{2^{2n}} \text{ para sinusoides}$$

La calidad (dB):

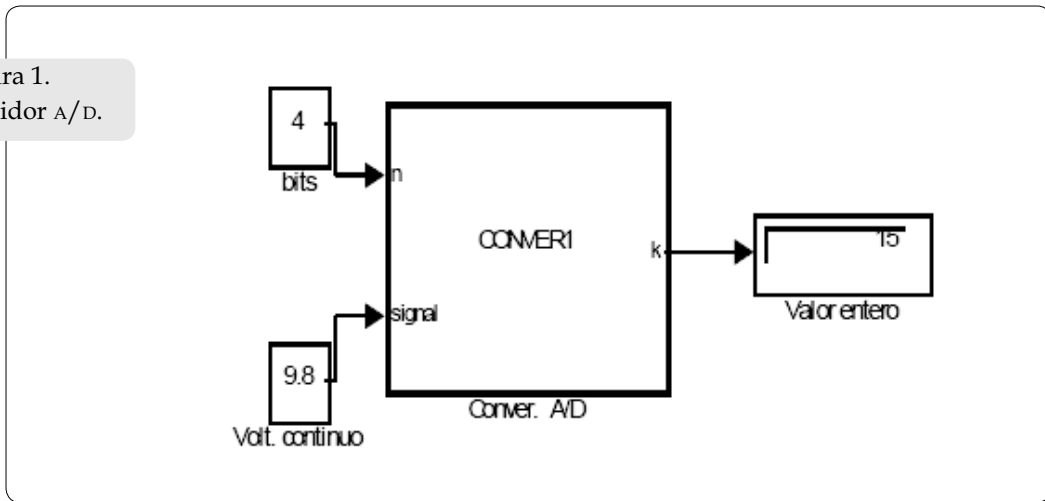
$$SQNR(\text{dB}) = 10 \log_{10}(SQNR) \text{ para sinusoides}$$

$$SQNR(\text{dB}) = 1,76 + 6,02n$$

La regla de los 6 dB: de la expresión anterior se deduce que cada bit adicional del

2 Si la asignación del entero se hace por redondeo, se le asigna a cada muestra el entero más cercano.

Figura 1.
Convertidor A/D.



conversor proporciona un aumento de unos 6 en la SQNR..

2.3 Codificación

Para poder dibujar la señal digital a la salida del conversor es necesario expresar estos valores enteros en valores que recorran la misma escala de medida de la función original. Para ello multiplicamos el entero obtenido por la resolución del conversor, es decir:

$$V_{discreto} = dv N_i \text{ con } i = 0, \dots, 2^n - 1$$

En caso de que la función de entrada tome valores negativos, es necesario hacer una traslación hacia abajo de la función de salida ya que todos los enteros utilizados para reconstruirla son positivos.

3. Diseño del convertidor

3.1 Una función cuantificadora

La primera parte del diseño se escribió utilizando Simulink, una función de dos entradas y una salida que acepta el número de bits del conversor y el voltaje a ser discretizado y entrega en la salida el entero

correspondiente a dicho voltaje. Se utilizó en User-Defined functions la opción Embedded MATLAB Function (Ogata, 1996) para construir el bloque (figura 1).

Dentro del cual se escribieron las instrucciones

```
function k = CONVER1(n, signal)
    dv=(10+10)/(2^n);
    k=0;
    while (signal> -10+(k+1)*dv)
        k=k+1;
    end
```

que muestran claramente que este bloque acepta valores continuos entre -10 y 10. Pero si quisiéramos dejar estos valores variables para adaptarlos a cualquier rango, modificamos el código así:

```
function k = CONVER1(n,signal,Vmax,Vmin)
    dv=(Vmax-Vmin)/(2^n);
    k=0;
    while (signal > Vmin+(k+1)*dv)
        k=k+1;
    end
```

y el bloque de Simulink tomaría la forma que se muestra en la figura 2.

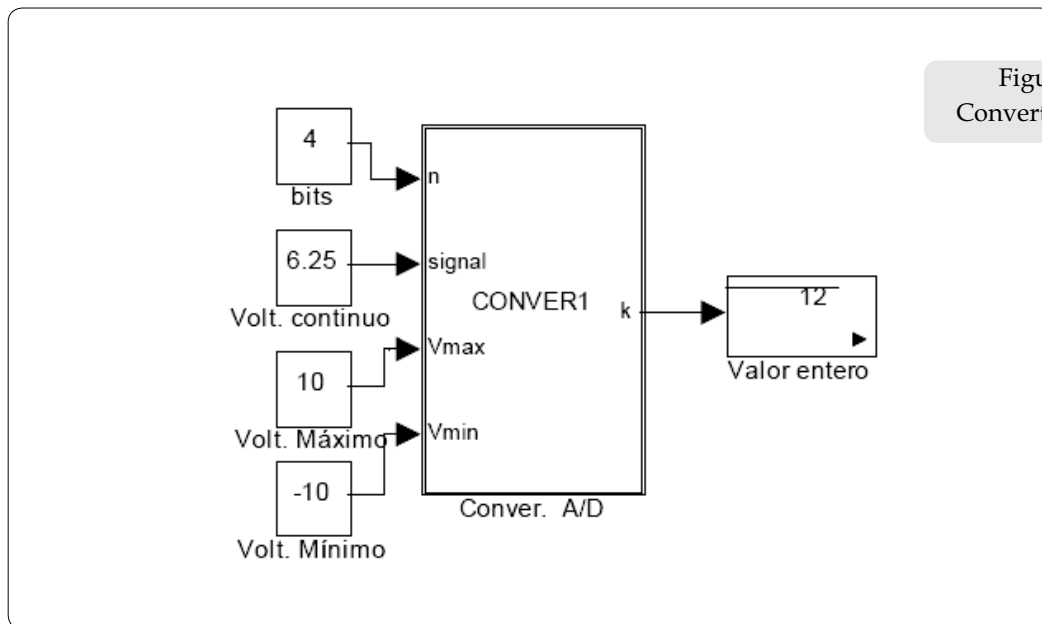


Figura 2.
 Convertidor A/D.

Cabe anotar que las cajas pequeñas de la izquierda actúan como las entradas del convertidor, que pueden ser ajustadas a gusto del usuario, y el display del lado derecho registra la salida del sistema.

3.2 El convertidor

Para el diseño del convertidor analógico-digital se modificó el código anterior de tal forma que se pudieran visualizar las funciones de entrada y salida, lo mismo que el error y la resolución del cuantificador. El código modificado tomó la forma:

```

1. function [y,dv,x]
    =
    CONVERSOR(n,vmax,vmin,signal)
2. dv=(vmax-vmin)/(2^n);
3. k=0;
4. while (signal > vmin+(k+1)*dv)
    k=k+1;
    end
5. y=k*dv+vmin;
6. x=signal-y;
```

Los comentarios dentro del código han sido eliminados para invitar al lector a escribir sus propios programas, ya que éstos se pueden construir de muchas formas. Incluso algunos comandos como `while` ya se encuentran en forma de bloques en Simulink.

El convertidor queda esquematizado por el diagrama de bloque (figura 3).

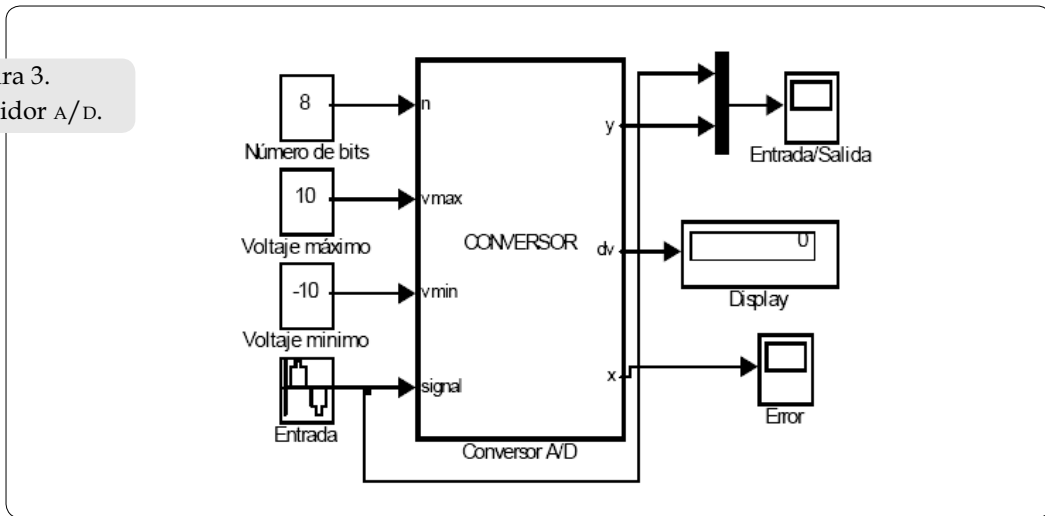
Como se ve el convertidor muestra cuatro entradas las cuales representan:

1. Número de bits del convertidor.
2. Voltaje máximo de operación.
3. Voltaje mínimo de operación.
4. Señal de entrada: cualquier señal continua cuyo rango esté entre V_{max} y V_{min} . Cualquier valor que caiga fuera del rango dinámico será recortado.

Igualmente se aprecian tres dispositivos de salida:

1. Un osciloscopio en el que se visualizan la señal de entrada y la señal de salida.

Figura 3.
Convertidor A/D.



2. Un display en el que se registra la resolución del cuantificador, el cual depende solamente del rango dinámico y del número de bits.
3. Un osciloscopio en el que se visualiza la señal de error.

En la siguiente sección se muestran y analizan los resultados de las pruebas realizadas al convertidor.

4. Análisis del sistema

Veamos lo que sucede en la salida del convertidor de la figura 3 cuando aplicamos un voltaje fuera de rango. Tomaremos:

$$V_{min} = 0$$

$$V_{máx} = 10$$

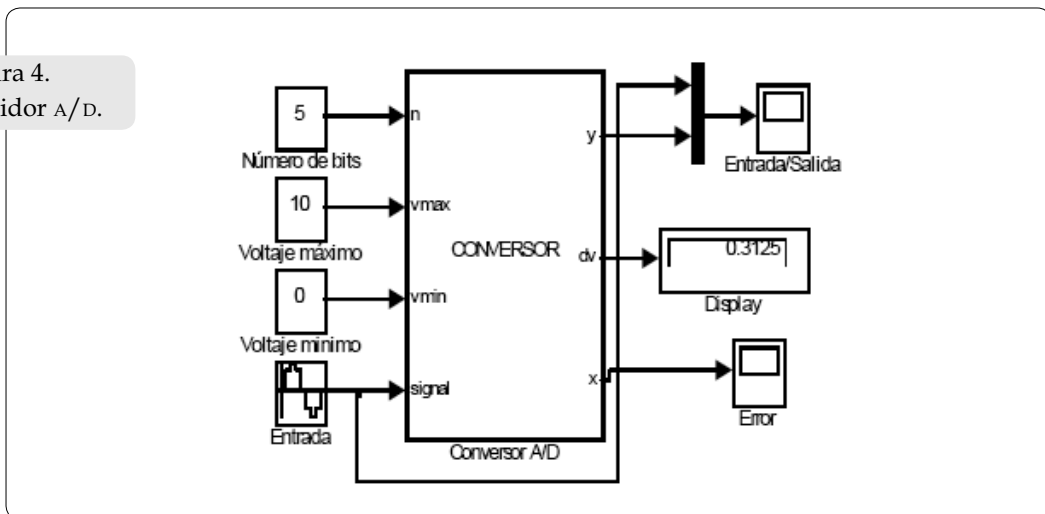
$$Entrada = 10 \sin(x),$$

$$T = 0,005,$$

$$n = 5bits$$

El sistema se ve de la siguiente forma:

Figura 4.
Convertidor A/D.



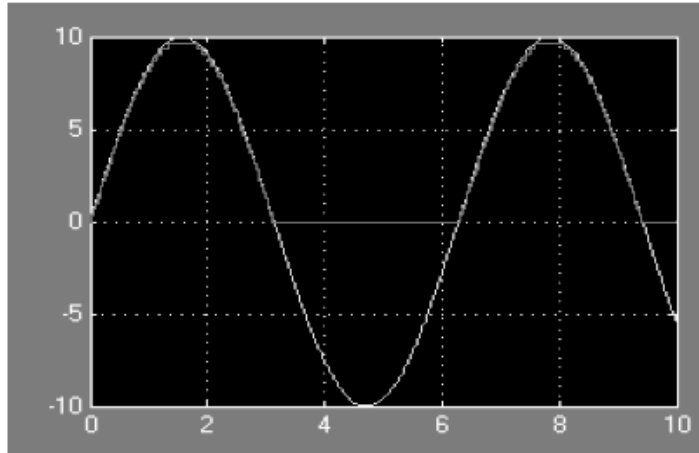


Gráfico 1.
 Entrada y salida
 del conversor.

En donde se observa en el display que la resolución del conversor es

$$dv = 0,3125$$

y las señales de entrada y salida se ven como se muestra en el gráfico 1.

Lo cual prueba que los datos por fuera del rango dinámico del conversor son recortados. Ahora veamos un poco más de cerca las

señales para ver el efecto de la cuantificación (gráfico 2).

Se observa mucho mejor si tomamos menos bits. Probemos con:

$$V_{min} = -10$$

$$V_{max} = 10$$

$$Entrada = 10 \sin(x),$$

$$T = 0,005$$

$$n = 4bits$$

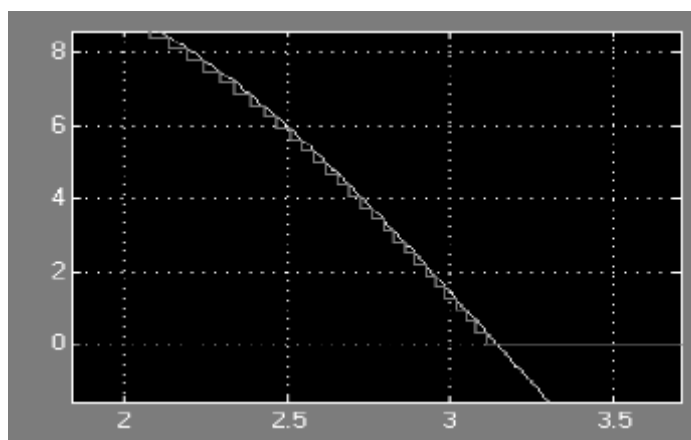
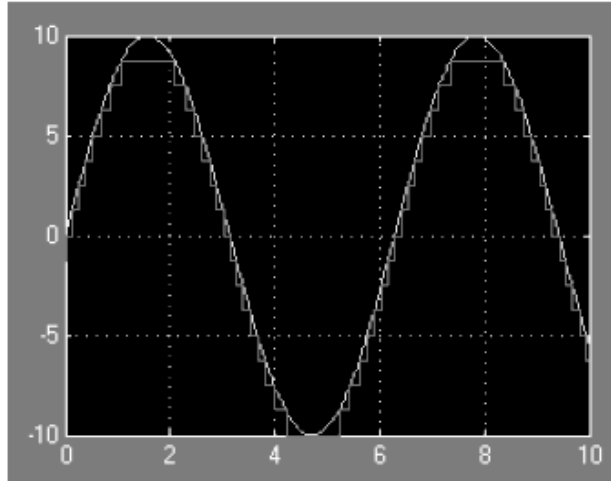


Gráfico 2.
 Entrada-salida
 (5 bits).

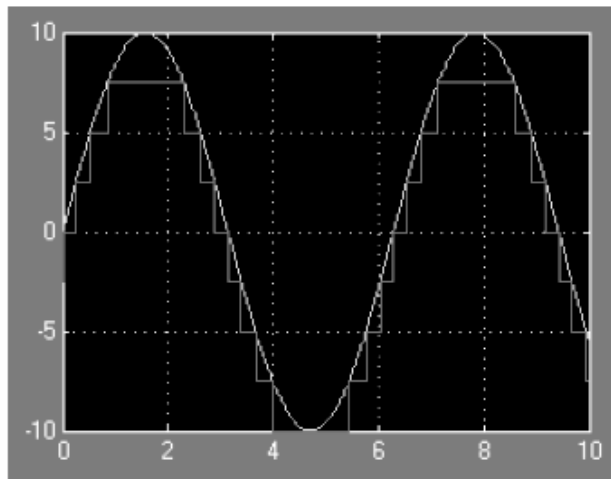
Las señales de entrada y salida son:

Gráfico 3.
Entrada-salida
(4 bits).



Y veamos la diferencia con 3 bits:

Gráfico 4.
Entrada-salida
(3 bits).



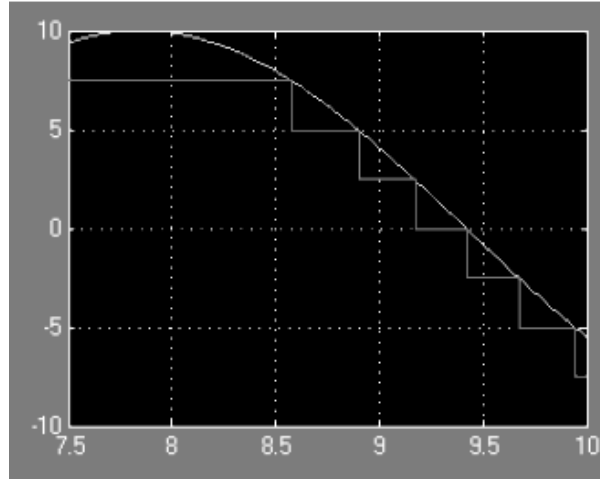


Gráfico 5.
 Entrada-salida
 (3 bits / T=0.0005).

En todos los casos anteriores se ha utilizado un periodo de muestreo $T=0,005$. Si cambiáramos dicho tiempo, ni los niveles de cuantificación ni la resolución del convertidor cambiarían, solamente cambiaría el número de muestras tomadas de la señal de entrada y así podría verse más suave su función. Veamos el gráfico anterior con $T=0.0005$.

5. Error

En la sección 2 se trataron algunas cosas acerca del error, que en nuestro caso se llamó de

truncamiento porque cada valor se aproxima al dígito inferior. Ahora veamos cómo se calcula y visualiza dicho error. Definimos el error como la diferencia entre la señal de entrada y la salida cuantificada:

$$x = \text{signal} - y \text{ de la figura 4.}$$

El error para un convertidor de 4 bits, con $\text{signal} = 10 \sin(x)$, $V_{\min} = -10\text{vol}$ y $V_{\max} = 10\text{vol}$ se observa en el gráfico 6.

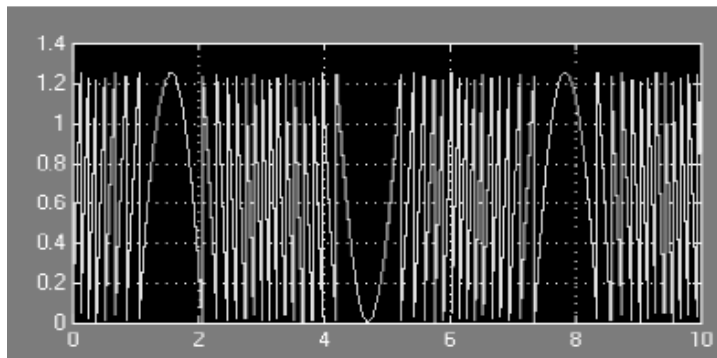
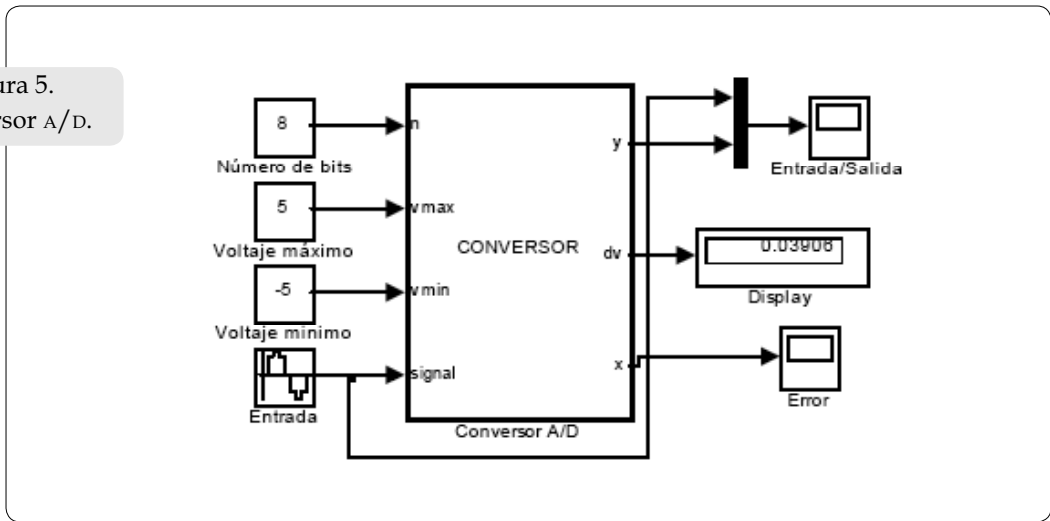


Gráfico 6.
 Error para 4 bits,
 $\Delta V = 20$.

Figura 5.
Convertor A/D.



Como se nota en la figura el error es una función cuyo rango es (0125) donde 1.25 es precisamente el valor que se observa en el display cuando se hace la simulación. Esto es porque:

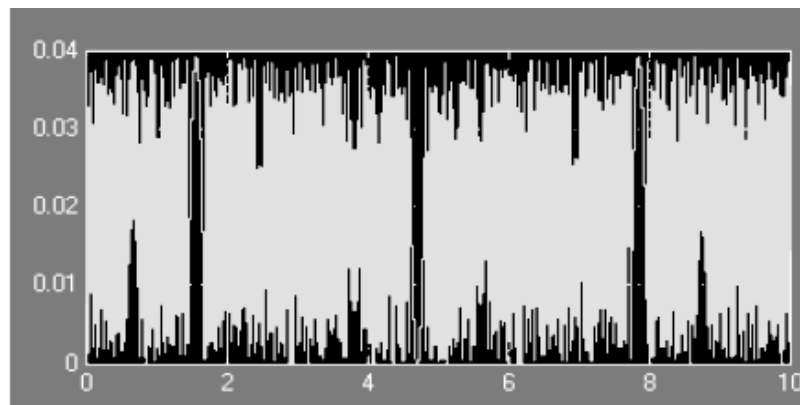
$$dv = \text{resolución del convertor} = \frac{20}{2^4} = 1.25$$

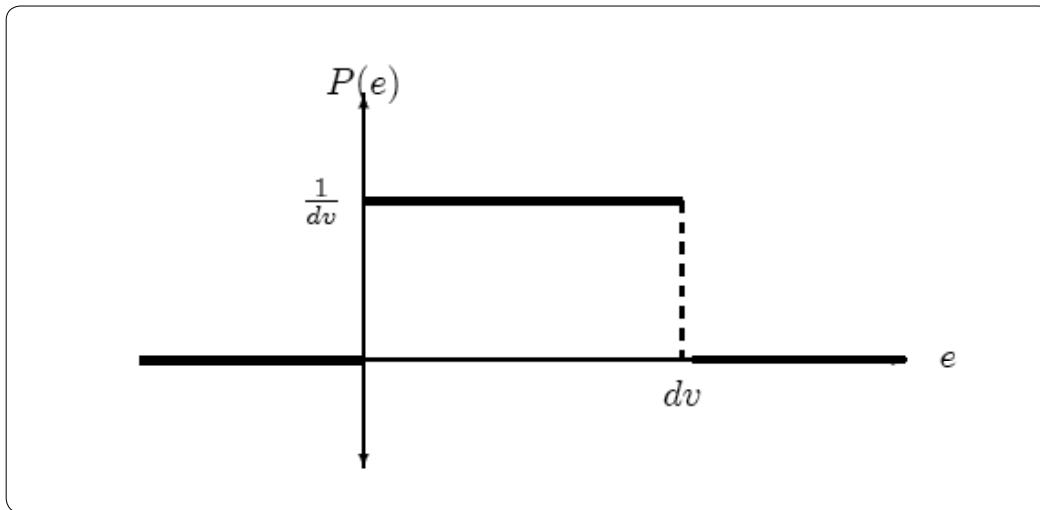
Es precisamente el error máximo que se tiene cuando se hace el proceso de conversión, como

se explicó en la sección 2. Esto nos muestra que al aumentar el número de bits, disminuye la probabilidad de error y viceversa. Veamos una simulación para $signal = 5 \sin(x)$, $V_{min} = -5vol$ y $V_{max} = 5vol$ con un convertor de 8 bits (figura 5).

En la figura 5 se nota que la resolución del convertidor es 0,03906. La función de error se muestra en el gráfico 7.

Gráfico 7. Error para conversión de la figura 5.





Como se observa éste coincide con el valor límite establecido en la figura anterior. Estos resultados muestran que el error se distribuye uniformemente desde 0 hasta dv (gráfico supra):

Cuyo valor esperado es

$$E(e) = \frac{dv}{2}$$

con una varianza

$$\sigma^2 = V(e) = \frac{(dv)^2}{12}$$

6. Conclusiones y sugerencias

1. Se ha mostrado una metodología muy práctica en el diseño de convertidores A/D, como el uso de Simulink a manera de alternativa a los procesos de programación y simulación de sistemas.
2. Se recomienda ahondar en el uso de esta herramienta (Simulink) ya que se puede programar utilizando más el sistema de bloques, que aún no conocemos muy bien, pero que con la práctica y el estudio podremos dominar.

3. En el diseño de convertidores el tiempo de muestreo no afecta la resolución del convertidor; como se logró ver, ésta depende solamente del número de bits.

4. Aquí se analizó la distribución del error utilizando el truncamiento, pero es fácil ver que si se utiliza el redondeo la distribución de probabilidad del error es la misma obtenida acá: con

$$E(e) = 0 \text{ y } V(e) = \frac{(dv)^2}{12}$$

5. En el diseño de convertidores, el número de bits (n) está estrechamente ligado al error; estas dos cantidades varían en forma inversa.

Bibliografía

- Chi-Tsong, C. (2006). *Analog & Digital Control System Design. Transfer-function, State-space & Algebraic Methods*. State University of New York at Stony Brook.
- Ogata, K. (1996). *Sistemas de control en tiempo discreto*. Pearson educación.
- Soria, E. et al. (2003). *Tratamiento digital de señales*. Pearson-Prentice Hall.