CAPÍTULO 5

Interfaz *simconverter*

Índice:

INTERFAZ *SIMCONVERTER*

1. Introducción.....31
2. Scripts del programa.....31
3. Visualización de la interfaz.....32

1. Introducción

En el presente capítulo se describirá brevemente la estructura del programa implementado para la interfaz *simconverter* (funciones que se llaman, etc.). En el anexo 3 se ha reflejado el código de las partes más importes de este programa donde se han realizado modificaciones al mismo para mejorarlo, comunicarlo con la interfaz library y darle mayor funcionalidad.

El nombre que se le ha dado al simulador de convertidores es *simconverter* (ésta será la orden que tendremos que teclear desde la ventana de comandos de Matlab para arrancarlo, ya que el script principal es el archivo simconverter.m).

2. Scripts del programa

Este script principal simconverter contiene las llamadas a los dos scripts que ya contendrán el código del programa. A grandes rasgos, estos scripts son:

- **Initial**: Se encarga de la declaración de las variables globales del programa. Dichas variables globales se han usado para poder disponer de los valores de los parámetros de interés del programa en el WorkSpace externo, de forma que desde el exterior del programa el usuario pueda ver el valor de estas variables, y editarlas desde la ventana de comandos. Este script se encuentra en el archivo initial.m.

- **Guiproyecto**: Este script contiene todo el código que controla el funcionamiento de la GUI. Dicho archivo, denominado guiproyecto.m, inicializa la GUI, contiene todas las callbacks (comandos que se ejecutan cuando un usuario hace clic sobre un componente de la GUI) de la GUI, y realiza todas las llamadas a funciones y scripts auxiliares necesarias para la implementación deseada en cada caso (entraremos en detalle en las funciones auxiliares a las que llama este script a continuación).

Las funciones y scripts auxiliares a las que llamará el script de la GUI son las siguientes:

- **INLDNL_matlab**: Script que calcula INL, DNL y los códigos perdidos.

- **CalcINLDNL**: Función que recibe como parámetros el vector de salida del convertidor y el manejador de la GUI, y ejecuta el script INLDNL_matlab para calcular INL, DNL y los códigos perdidos.

- **CalcSNDR**: Función que recibe como parámetros el vector de salida del convertidor, el ancho de banda, la banda mínima de señal, el número de

muestras que consideramos como banda de señal (del centro a un extremo), el manejador de la GUI y el manejador del objeto figura. Calculará SNDR, SNR y ENOB.

- **SimulacionNoIdeal**: Función que gestiona los análisis simples, es decir, realiza una única simulación para unos valores concretos de los parámetros. Realiza sus cálculos usando las funciones CalcINLDNL y CalcSNDR.

- **Funciones de barrido**: Funciones que gestionan los barridos simples y dobles, tenemos una función por cada barrido simple o doble que se haya implementado. La característica principal de las funciones de barrido es que su nombre siempre comienza por la palabra "Estudio". Usará para realizar sus cálculos las funciones CalcINLDNL y CalcSNDR.

2. Visualización de la interfaz

La interfaz[1] finalmente desarrollada se muestra a continuación:

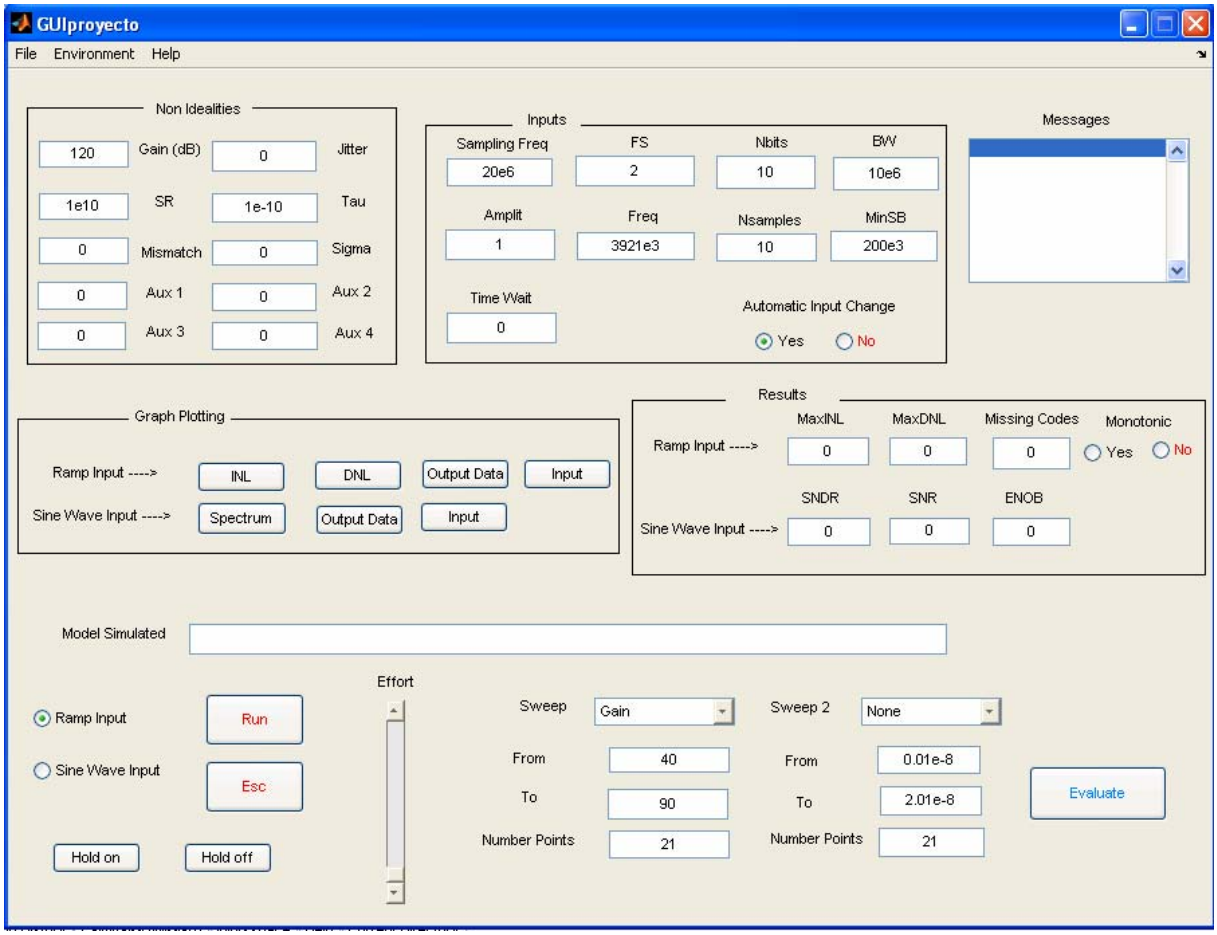

Fig. 5.1. Interfaz simconverter versión 2.o

[1] La descripción de su utilización se realiza en el manual de usuario que se encuentra en el Anexo1.