

Trabajo práctico: Filtrado digital IIR

5) Filtros IIR de 2do orden tipo Direct I y Direct II con floating point en C

MATLAB permite ejecutar funciones desarrolladas en lenguaje C usando una función *wrapper* cuya finalidad es la de actuar como interfaz entre MATLAB y C. Este wrapper debe adaptar las variables de entrada y salida entre ambos lenguajes.

a) Abra la función wrapper `iir_matlab.c` y analice las diferentes secciones. Al final de la misma se verá que se invoca a la función `iir_filter_I_float()`.

b) Las funciones en C que implementan diferentes filtros IIR de 2do orden se encuentran en el archivo `iir_filters.c`, donde hay funciones para filtros IIR tipo Direct I y Direct II, en punto flotante y punto fijo. Ejecute en consola el comando:

```
>> mex iir_matlab_wrapper.c iir_filters.c
```

```
mex -c iir_matlab_wrapper.c iir_filters.c
```

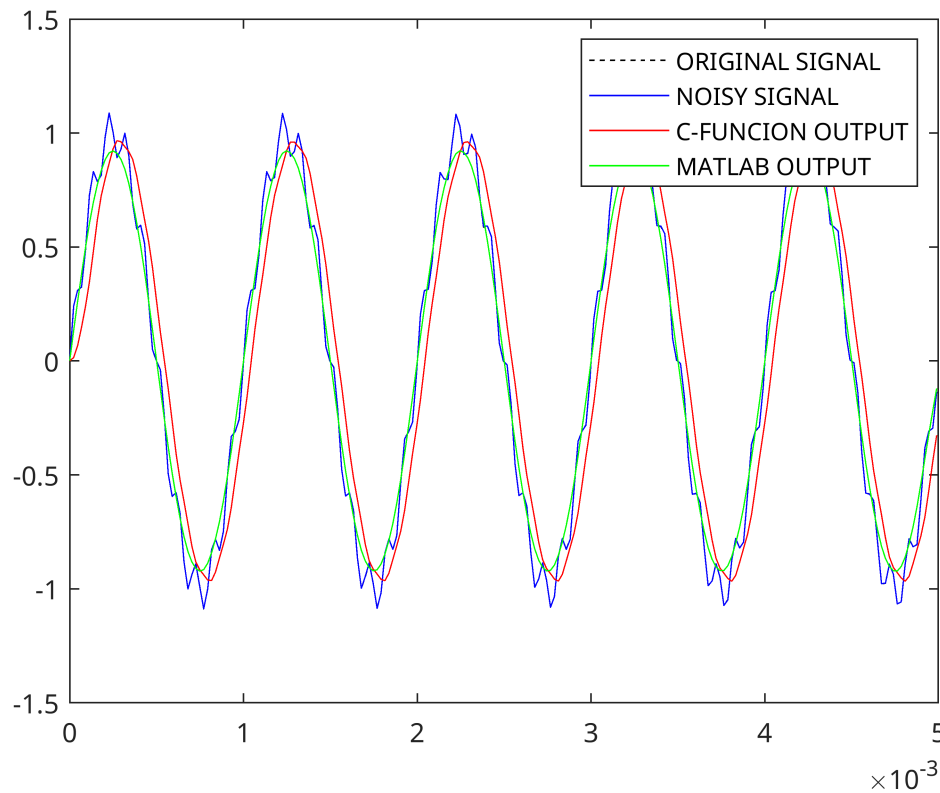
```
Building with 'gcc'.  
MEX completed successfully.
```

La función `mex` es la encargada de compilar los archivos en C bajo MATLAB.

c) Abra en el editor de MATLAB el script `iir_matlab_to_C.m` y ejecútelo. Este se encarga de invocar la función `iir_matlab_wrapper` y de analizar su salida. Concéntrese en comparar las señales de salida para las funciones implementadas en MATLAB en C.

```
iir_matlab_to_C;
```

```
Warning: The first two inputs are interpreted as an SOS matrix and G vector. Use column vectors if  
you meant to specify a transfer function.
```



d) Comente la línea de la función `iir_filter_I_float()`, descomente la línea de la función `iir_filter_II_float()` y vuelva a compilar. Ejecute nuevamente `irr_matlab_to_C.m`. Observe la señal filtrada de salida, ¿detecta alguna diferencia respecto a la salida para el filtrado tipo Direct I?

```
mex -c iir_matlab_wrapper.c iir_filters.c
```

```
Building with 'gcc'.
MEX completed successfully.
```

```
iir_matlab_to_C;
```

Warning: The first two inputs are interpreted as an SOS matrix and G vector. Use column vectors if you meant to specify a transfer function.

