# Módulo IP de Filtro FIR con Interfaz de comunicación AXI Lite

Para el proyecto de la asignatura de Design of Embedded Systems, se ha realizado un módulo IP consistente en un Filtro Fir de 4 coeficientes.

Para ello se ha descargado un código VHDL de la siguiente página web: [Código VHDL Filtro FIR](https://surf-vhdl.com/how-to-implement-fir-filter-in-vhdl/).

Se ha modificado el código para dar una señal de Valid cuando el Dato de Salida esté disponible para lectura.

La entidad generada para el código del Filtro FIR es la siguiente:

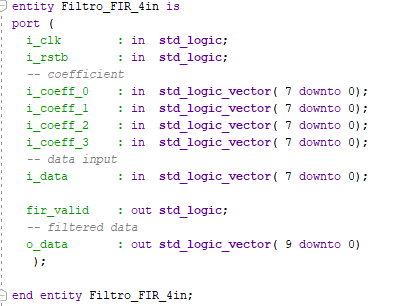


Ilustración Entidad Filtro FIR

Para la comunicación con el procesador y la escritura de los valores de los coeficientes y dato de entrada y lectura del dato de salida, se ha usado comunicación a través de la interfaz AXI Lite. La instanciación y conexión con el IP del AXI ha sido la siguiente:

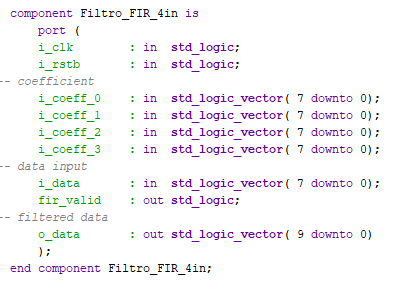


Ilustración Declaración Módulo FIR

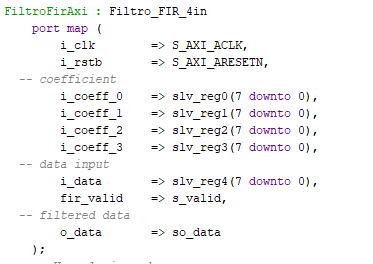


Ilustración Instanciación y Conexionado

A pesar de que en los registros quedaban muchos bits “libres”, se eligió separar cada coeficiente y dato de entrada en registros diferentes para el mejor manejo en el SDK a la hora de escribir en dichos registros.

Para el dato de salida, se ha asignado a una señal su valor. Dicho valor se escribe en el registro 5 (***slv\_reg5***) usando solo los 10 primeros bits. Dicha escritura en el registro 5 (para posteriormente leerlo en el proceso de lectura) se hace al final del proceso de escritura de los registros, de manera que, si por error se escribe en el registro 5, el valor que finalmente quedará guardado será el que queremos, el dato de salida.

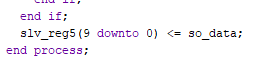
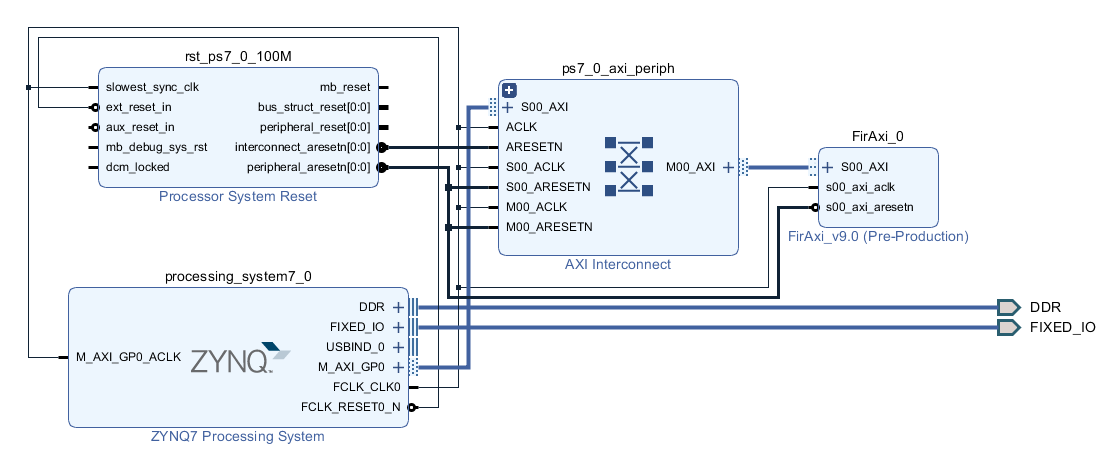


Ilustración Escritura Dato de Salida

Realizado todo el conexionado de señales, se finaliza el módulo AXI de nuestro IP, por lo que se pasa al diseño de Bloques. El diseño de bloques generado es el siguiente:



El siguiente paso, fue la realización del Bare metal en SDK, donde se generó un código sencillo para la comprobación del correcto funcionamiento de nuestro módulo.

Dicha comprobación se ha realizado mediante una señal seno como dato de entrada, generada en un bucle for para cada punto de la señal y la escritura de los coeficientes.

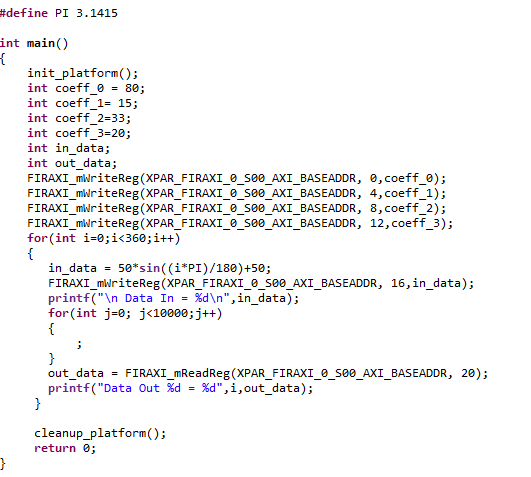


Ilustración Código SDK

Puesto que en la FPGA no se podían sacar valores negativos, se sumó un valor entero de 50 (el mismo valor de la amplitud de nuestra señal) para llevar toda la señal a la parte positiva.