# Diseño de Sistemas Electrónicos Digitales

**Sistema de grabación, tratamiento y reproducci´on de audio**

Versi´on 4.2

Pablo Ituero y Fernando Garc´ıa Redondo

Nombre y apellidos de los miembros del grupo:

Carlos Gómez Rodríguez

Alejandro Ramos



*Departamento de Ingenier´ıa Electr´onica*

**Tarea 1.1:**

Disen˜a un circuito que sea capaz de generar las sen˜ales especificadas. Dibuja el es- quem´atico correspondiente a tu disen˜o en el espacio inferior.

**Tarea 1.2:**

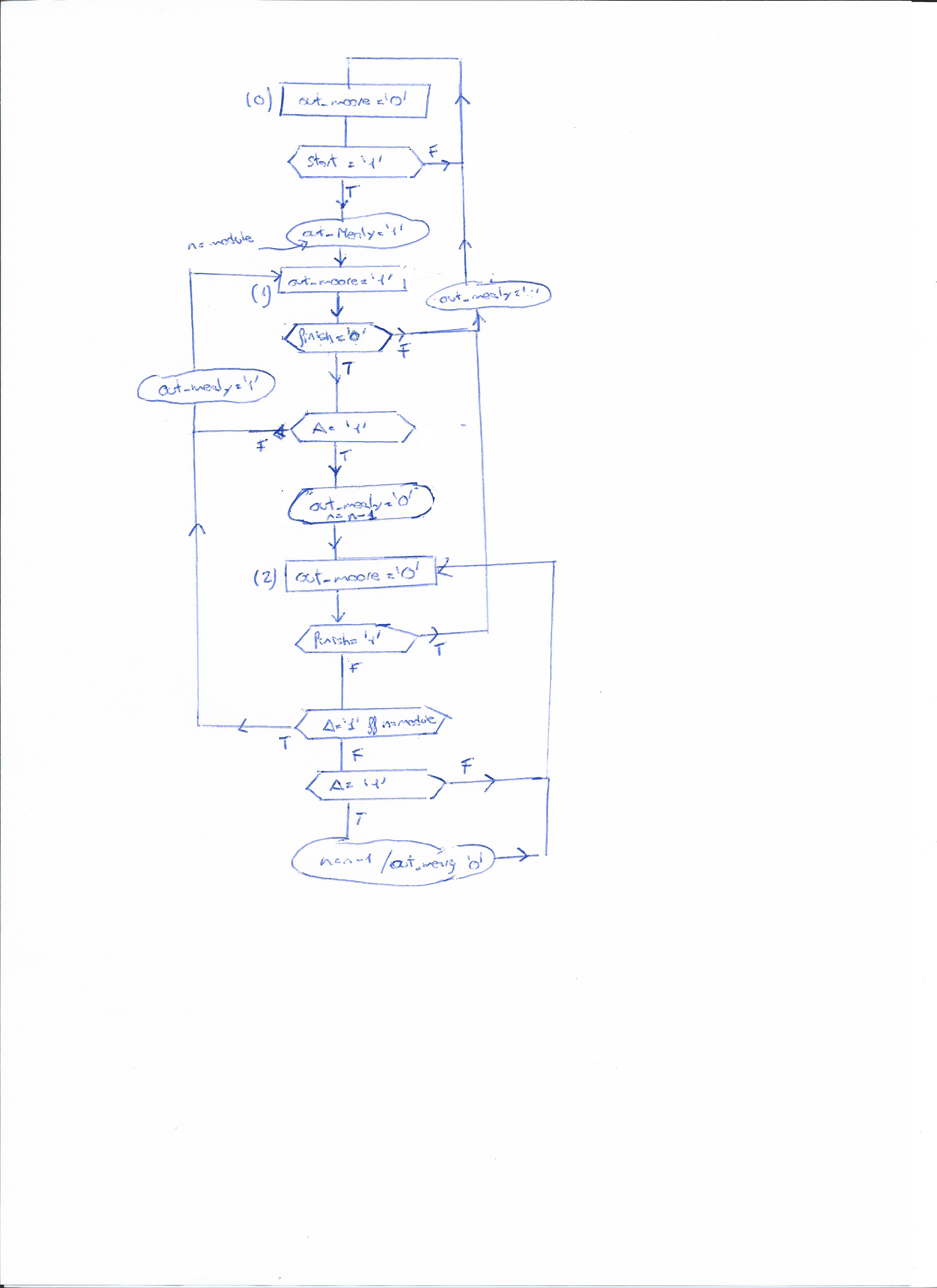
Implementa tu disen˜o en VHDL.

**Tarea 1.3:**

Genera un testbench que verifique la funcionalidad de tu disen˜o.

**Tarea 1.4:**

Realiza el diagrama ASMD que describa la misma funcionalidad que el pseudo-c´odigo anterior.



**Tarea 1.5:**

Implementa el diagrama anterior en un fichero VHDL.

**Tarea 1.6:**

Realiza un testbench que tenga la sen˜al micro data fija a ’1’ para comprobar que las transiciones de estados, la selecci´on de los datos de salida y la activaci´on de sam- ple out ready se produce correctamente.

**Tarea 1.7:**

Realiza un testbench que introduzca una sen˜al pseudo-aleatoria en micro data y com- prueba que la digitalizaci´on se realiza correctamente.

**Tarea 1.9:**

Implementa el esquem´atico anterior en un fichero VHDL.

**Tarea 1.8:**

Modifica el disen˜o propuesto en el libro del Dr. Chu para que sea compatible con las especificaciones de nuestro sistema. Es decir, que el contador cuente de 0 a 299 e incluya reset y enable y que el circuito produzca la salida sample request en el momento apropiado y con la duraci´on apropiada. Dibuja el esquem´atico de tu disen˜o en la parte inferior.

**Tarea 1.9:**

Implementa el esquem´atico anterior en un fichero VHDL.

**Tarea 1.10:**

Realiza un testbench que verifique el funcionamiento de la interfaz de la salida de audio.

**Tarea 1.11:**

Dibuja el esquem´atico a nivel bloque de la interfaz de audio completa. Ten en cuenta que record enable y play enable especifican cu´ando est´an activas las interfaces del micr´ofono y de la salida de audio, respectivamente. Del mismo modo, tienes que asignar un ’1’ tanto a micro LR, como a jack sd.

**Tarea 1.12:**

Implementa en VHDL la interfaz de audio completa. Para ello utiliza un estilo de c´odigo estructural que instancie cada uno de los tres bloques desarrollados anteriormente.

**Tarea 1.13:**

Disen˜a un testbench que verifique la funcionalidad de la interfaz de audio completa.

**Tarea 1.14:**

Implementa en VHDL el controlador. Utiliza estilo estructural. Necesitas emplear el asistente arquitectural de Vivado para generar el reloj de 12 MHz a partir del rejol de 100 MHz.

**Tarea 1.14:**

Implementa en VHDL el controlador. Utiliza estilo estructural. Necesitas emplear el asistente arquitectural de Vivado para generar el reloj de 12 MHz a partir del rejol de 100 MHz.

**Tarea 1.15:**

Disen˜a un testbench que verifique la funcionalidad del controlador. Puedes utilizar los est´ımulos empleados en testbenches anteriores.

**Tarea 1.16:**

Escribe el fichero de restricciones .xdc. Sintetiza y realiza la implementaci´on f´ısica del disen˜o. Genera el fichero .bit y programa la FPGA. Comprueba el funcionamiento correcto conectando unos auriculares a la salida de audio de la placa. Si la salida de audio es muy ruidosa, puedes jugar con el valor de micro LR y ponerlo a ’0’ en lugar de a ’1’.

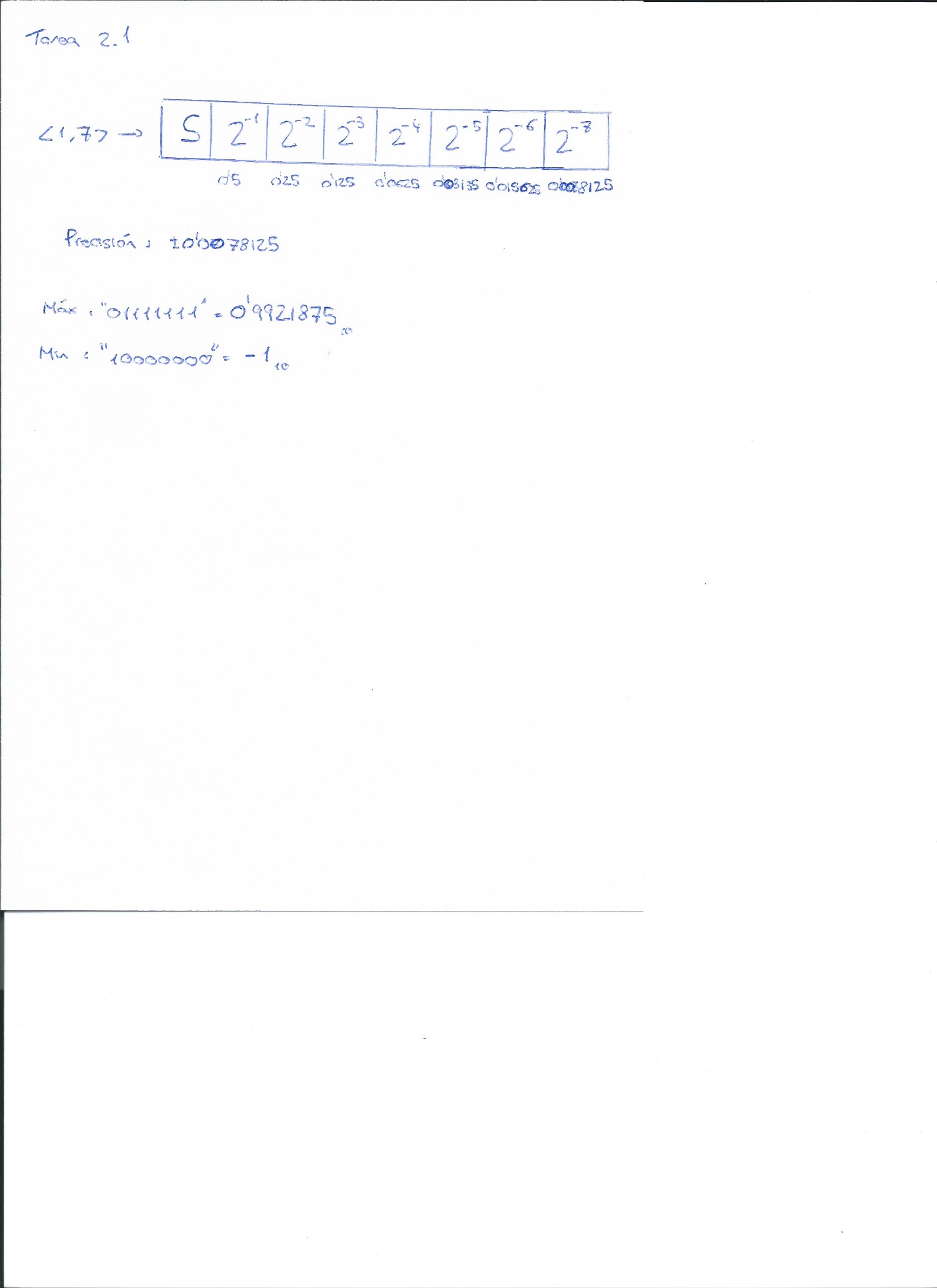
Guarda en un fichero de texto todos los mensajes de warning que hayan aparecido en el proceso anterior.

Avisa al profesor para comprobar el funcionamiento.

**Tarea 2.1:**

¿Cu´al es el rango de un nu´mero de 8 bits en punto fijo *<* 1*,*7 *>* en complemento a dos?

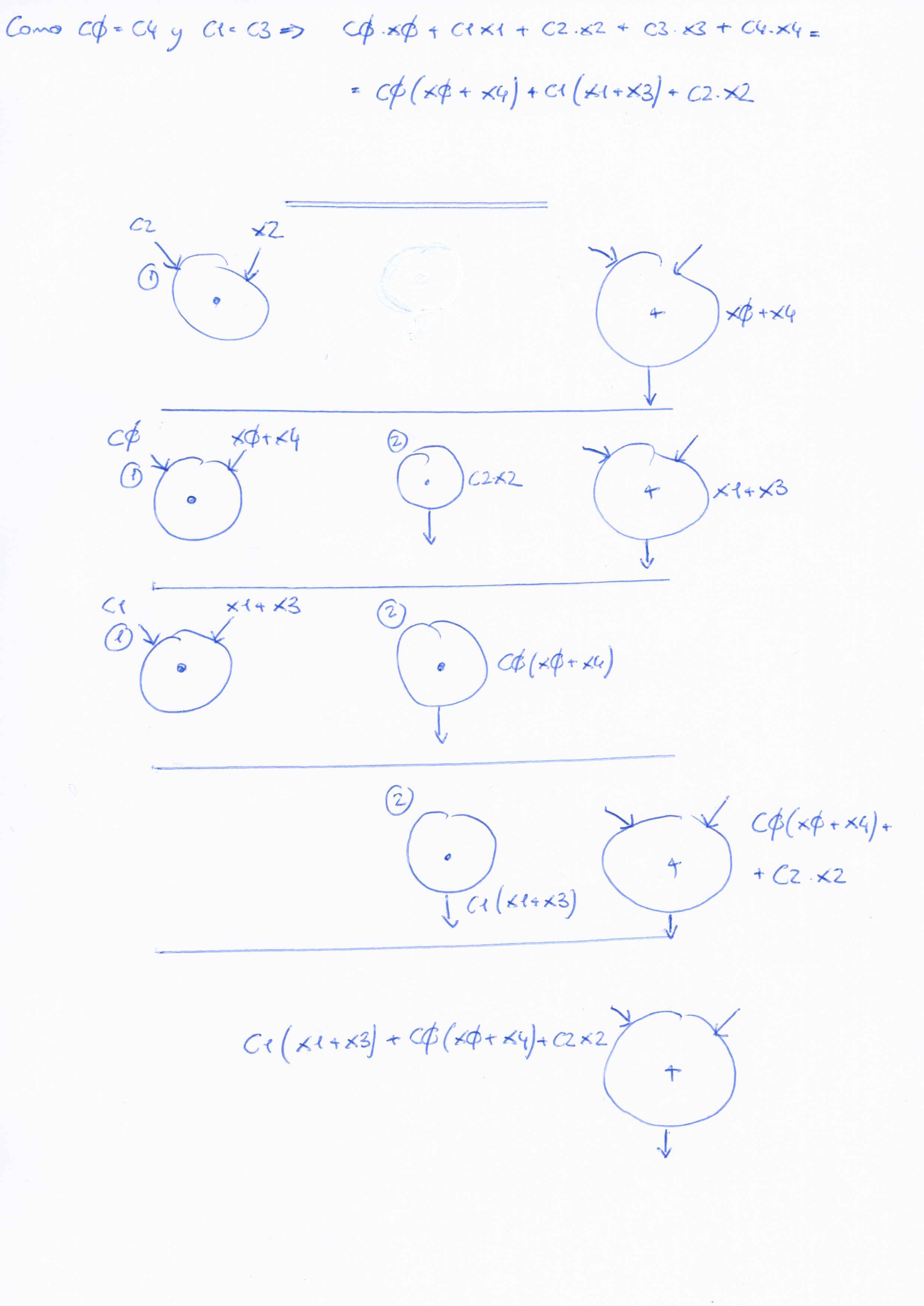
¿Cu´al es su precisi´on?



**Tarea 2.2:**

Realiza sobre papel la implementaci´on de un filtro FIR de 5 etapas con la siguiente asignaci´on: Dos medios multiplicadores y un sumador. Realiza todos los pasos descritos en la secci´on [6.4](#_bookmark21).

**Planificacíon temporal:**



**Vinculaci´on, an´alisis de tiempo de vida de variables y asignaci´on de registros:**

**An´alisis de conexiones para la extracci´on de multiplexores:**

**Implementaci´on:**

**Cronograma:**

**Tarea 2.3:**

Realiza un an´alisis de cuantificaci´on de las sen˜ales, de manera que especifiques cu´antos bits vas a emplear en cada sen˜al y en qu´e posici´on va a estar el punto decimal. Emplea la notaci´on y las metodolog´ıas presentadas en clase.

**Tarea 2.4:**

Escribe modelos VHDL para todos los componentes que se emplean en tu ruta de da- tos. Los operadores aritm´eticos (suma y multiplicaci´on) no necesitan describirse en un estilo estructural. Para conseguir el medio multiplicador, an˜ade un registro extra a la salida de un multiplicador completo, de esta forma se comportar´a como un multiplica- dor perfectamente segmentado. La herramienta de s´ıntesis se encargar´a de mover este registro al interior del multiplicador de manera ´optima. Comprueba el funcionamiento correcto de tus componentes creando los correspondientes testbenches.