

Proyecto final

Valor 30%

Fecha de entrega: 21 de agosto, 2017

Introducción

Las especificaciones y planeamiento de la arquitectura del Sistema son importantes para iniciar un proyecto encaminado al éxito. En este proyecto se realizará un pequeño ejercicio en cuanto a estos dos elementos del diseño digital. Además se evaluará la parte práctica del curso.

Escriba un reporte donde presente los resultados de cada una de las partes de este proyecto. El documento final puede ser a una sola columna. En este caso el formato es libre, con una portada, introducción, desarrollo de los puntos de cada parte y en el caso de la parte práctica resultados y conclusiones.

Parte 1 – Especificaciones, arquitectura, diseño y prototipado

***Nota: esta parte es solo teórica, no es necesario implementar un sistema.**

Usted labora como ingeniero dentro de un grupo de desarrollo de hardware. Su grupo de trabajo se encuentra desarrollando un nuevo micro-controlador DSP. Como base usted conoce lo siguiente:

1. Su microcontrolador debe ser capaz de correr a frecuencias de 50MHz.
2. La arquitectura del núcleo de procesamiento va a ser RISC. Este bloque va a ser reciclado de otro diseño por lo que sería un IP core de su empresa.
3. Como periféricos debe poseer lo siguiente:
 - a. Controlador de reloj. Toma como entrada un reloj externo, y lo distribuye dentro del IC: También es capaz de generar a partir del reloj principal dos relojes individuales que pueden ser utilizados por todos los demás periféricos. Estos relojes pueden los siguientes múltiplos del reloj maestro: 2, 1, 1/2, 1/4, 1/8.
De esta forma un periférico puede usar el reloj maestro o cualquiera de estos dos relojes.
 - b. Dos contadores-timer hacia abajo de 16 bits. Estos producen una interrupción al terminar la cuenta. Pueden ser inicializados.
 - c. Un watchdog timer. Este debe tener un periodo máximo de 30 segundos a máxima frecuencia.
 - d. Dos periféricos para calcular transformadas de Fourier. Los datos de entrada pueden ser en 8 bits, y el largo máximo es de 512 puntos.

- e. Un ADC de 8 bits, puede trabajar a 10Msamples/s.
- f. Un DAC de 8 bits, puede trabajar a 2Msamples/s.
- g. Dos Interfaces SPI e I2C.
- h. 16Megabits de memoria en chip de un solo acceso. Bus de datos multiplexado.
- i. Controlador de potencia. El sistema puede entrar en 3 modos de operación:
 - i. Modo activo: El núcleo recibe energía y reloj, los demás periféricos pueden apagarse a discreción).
 - ii. P1: El núcleo recibe energía y reloj, los periféricos están todos apagados, a excepción de un timer que puede usarse para despertar el sistema
 - iii. P2: Todos los sistemas están apagados, el núcleo no recibe reloj. Un pin externo puede despertar el sistema.
- j. El sistema posee 4 bancos de 8 pines I/O cada uno. Estos pines pueden ser configurados como entrada o salida.

Según la base anterior:

1. Defina que bloques pueden prototiparse por completo, cuales parcialmente y cuales no vale la pena prototipar (o que no es posible).
2. Busque dos opciones de FPGA/Tarjeta de desarrollo que puedan permitir prototipar su sistema. Justifique la decisión basándose en los recursos necesarios, para esto primero defina un estimado de dichos recursos. (puede usar un sistema qsys o Ip Cores como base para la estimación). Recuerde dejar un margen de diseño (indique de cuanto es)
3. Defina un documento de “arquitectura y diseño” de su sistema prototipo. En este debe especificar detalles sobre cómo se implementaría el prototipo, a nivel top (diagrama de bloques) y cada bloque. (I.e. El núcleo se va a implementar usando un NIOS II porque es RISC y puede servir para probar el software...). Si va a utilizar IP Cores indique la fuente y porque se escogerá un core específico.
4. Detalle un plan de pruebas para el sistema prototipo.

Parte 2 – Ejercicio Práctico

En los sistemas empuetrados el uso de aceleradores de hardware permite optimizar el sistema para realizar ciertas operaciones de la manera más rápida. En el caso de su sistema usted requiere de un módulo que permita realizar la multiplicación de matrices 2x2 y retorne el determinante del resultado.

- Cada celda en las matrices contiene datos de 7 bits. Se asume que estos datos representan valores enteros en complemento a 2 (un bit de signo y 6 de magnitud).
- Genere un bloque de hardware que permita la multiplicación de matrices 2x2 y calcule el determinante del resultado en hardware donde se haga uso extensivo de cálculos en paralelo. Recuerde tomar en cuenta la ruta crítica de su diseño.
- Genere un sistema empuetrado NIOS (no usar multiplicador de hardware) que contenga un timer de 32 bits.
- Genere una interfaz entre un sistema empuetrado NIOS y su módulo de hardware.

- Escriba una aplicación de prueba que realice multiplicaciones de matrices de 2x2 y el cálculo del determinante de cada matriz resultado usando solamente el procesador (i.e. las operaciones se realizan por medio de código C).
- A modo de prueba ejecute su rutina **16** veces consecutivas y mida el tiempo de procesamiento total utilizando el timer.
- Re-escriba una nueva aplicación para que utilice el módulo de hardware para realizar las mismas operaciones (cargar datos al módulo y leer el resultado de vuelta). Mida el tiempo de procesamiento usando el timer.
- De igual forma ejecute la rutina **16** veces consecutivas y mida el tiempo de procesamiento total utilizando el timer.
- Incluya en su reporte el proceso de diseño y sus resultados.
- Junto con este documento se incluye una rutina de generación de valores para las matrices. Las matrices están implementadas como un set de datos que puede ser leído como un arreglo signed char dos por dos ejemplo (*a).cd[0][1]. En este caso interprete los datos como una matriz en el siguiente orden:

Matriz X: (Primer índice es fila y segundo índice es columna)

cd[0][0]	cd[0][1]
cd[1][0]	cd[1][1]

Recomendaciones:

- Para enviar los datos a su bloque tome ventaja de que los datos son de 7 bits, por lo que cada matriz puede ser enviada al bloque de hardware como un paquete de 28 bits en una sola transacción.
- Separe sus operaciones en el bloque RTL en dos grandes segmentos:
 - Cálculo de multiplicación de matriz
 - Cálculo de determinante
 - Separe ambos segmentos por medio de registros para romper con rutas largas
- Verifique si el CPU va a necesitar ciclos de espera para leer el dato respuesta. Si necesita ciclos extra determine cuantos y use este valor cuando genere su instancia en qsys.
- Haga uso de operaciones de alto nivel como "*" para realizar la multiplicación, de esta manera el sistema va a utilizar los recursos aritméticos disponibles.
- Haga un cálculo de cuantos bits va a ocupar para datos en cada etapa. El bus de datos que contiene el dato del determinante va a ser más largo que el bus de datos de cada celda en las matrices.... Recuerde cuantos bits se generan en una multiplicación o en una suma/resta binaria...