

MICROPROCESADOR

MICROCONTROLADORES II

UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS

FACULTAD TECNOLÓGICA

TECNOLOGÍA EN ELECTRÓNICA

SANTA FE DE BOGOTÁ, MARZO DEL 2001

INTRODUCCIÓN

El presente trabajo es con el fin de conocer y profundizar en el área de arquitectura de computadores, además el de conocer e investigar sobre cada una de las funciones y tareas que realiza un microprocesador, tanto a nivel máquina como a nivel del programador.

Lo primero que se establece en este trabajo es el conocimiento de un set de instrucciones totalmente nuevo que sea un tanto amigable para el usuario, además se describe completamente el proceso para reconocer todos y cada uno de los pasos a seguir, y por último lo que se desea es el conocer y mostrar las ventajas y desventajas que se pueden tener en un microprocesador de estas características.

Con esto los problemas y soluciones a seguir para nuestro microprocesador.

OBJETIVOS

- La idea de realizar un microprocesador, nos lleva a pensar en uno que sea competente en el mercado actual, que posea gran velocidad en la ejecución de datos e instrucciones y tenga una buena eficiencia y variedad de funciones.
- Planear un procesador con un set de instrucciones bastante grande con el fin de que el usuario tenga muchas más opciones a la hora de trabajarlo, y sirva para implementar en varias cosas.
- Tener un set de instrucción completo para mayor facilidad a la hora de programar mi procesador.
- Construir un procesador que tenga una zona fuerte o mejor completa, en este caso nuestro fuerte serían las señales analógicas, quienes poseen un mayor trabajo en cuanto a comunicaciones y tal vez modulación.
- Además, de este aspecto también tenga una fácil programación para implementarlo en automatizaciones y control a nivel de empresas, que este en la capacidad de trabajar con sensores, control de motores, control remotos y demás.

DESCRIPCIÓN DEL SET INSTRUCCIONES

- **INSTRUCCIONES DE TIPO – R** : Son operaciones ALU registro a registro:

rd rs1 func rs2. Función codifica la operación del camino de datos. Registros especiales de lectura/escritura y transferencia.

En nuestro procesador encontramos varias funciones de este tipo como son :

SUM R1, R2, R3, : Esta instrucción sirve para la realización de sumas entre dos registros R2 y R3 y el resultado de esta suma será guardada en el registro R1, en parte interna, la operación suma será trabajada por medio de la ALU, registro A, registro B y su respuesta en el registro C.

RES R1, R2, R3, : Esta instrucción sirve para la realización de restas entre dos registros R2 y R3 y el resultado de esta suma será guardada en el registro R1.

AND, OR, XOR, R1, R2, R3, : También utilizan el tipo de instrucción R y cada una de estas realiza su operación lógica por medio de la ALU.

- **INSTRUCCIONES TIPO – I :** Codifica: Carga y almacenamiento de bytes, palabras, medias palabras. Todos los inmediatos (rd rs1 función inmediato)

Instrucciones de salto condicional (rs1 es registro, rd es registro de destino, inmediato es el número hexagesimal que introduce el programador para realizar la operación deseada).

En nuestro procesador encontramos varias funciones de este tipo como son :

SUMN, R1, R2, N : Esta función realiza sumas entre los datos que hay en un registro con el dato o número que el programador necesite realizar la operación. N tiene que ser un número hexagesimal y no puede ser mayor que FF. La operación suma será trabajada por medio de la ALU, registro A, número predeterminado y su respuesta en el registro C.

RESN, ANDN, ORN, XORN, R1, R2, N : Con estas instrucciones podemos realizar cualquier operación anterior con un número cualquiera que debe ser hexagesimal, por ejemplo en la XOR podemos comparar o filtrar palabras o datos que nos estén sobrando y nos causen problemas con el carry o demás.

ROTDN, ROTIN, R1, R2, N: Rote a la derecha N veces y Rote a la izquierda N veces también son instrucciones de tipo – I, ya que utilizan también el inmediato, esta instrucción se utiliza en casos donde necesitemos invertir nuestra palabra, o sea que nuestra media palabra que está a la izquierda resulte en la derecha y la otra media palabra que estaba en la izquierda resulte a la derecha. Gracias al N veces que puedo rotar no solo se puede hacer media palabra sino además un cuarto o bit por bit de palabra. Utiliza este tipo de instrucción–I, y su código de operación es el de rotar.

MOVNR R1, N : La instrucción mencionada mueve el valor introducido por el programador a un registro R1 determinado por el mismo, o sea que en un registro de propósito general le da el valor correspondiente para utilizar como variable en su programa. Utiliza este tipo de instrucción–I, y su código de operación es el de mover.

- **INSTRUCCIONES DE TIPO – D:** Realiza las operaciones en la ALU, con un registro r1 función r2, r1 es el registro de destino, a r2 le realizo la operación y su resultado la guardo en r1.

En nuestro procesador encontramos varias funciones de este tipo como son :

NOT R1, R2 : Esta instrucción realiza la negación total de la palabra, o sea, lo que tenga con valor 1 lógico será 0 lógico y viceversa, entonces me sirve para hallar el complemento de palabras en un ciclo de reloj.

ROTI, ROTD, R2, R3 : Rotar la palabra a la izquierda y rotar la palabra a la derecha son funciones de esta clase de instrucción, por que no necesita de otro registro para poderse operar simplemente realiza la operación

rotar y su nuevo resultado lo deja en un nuevo registro llamado R2, en caso que el programador desee poner la rotación de su palabra en el mismo registro es necesario poner el mismo nombre para R1 y R2.

MOVRR R1, R2 ; MOVRR M, R2 ; MOVRR R1,M : La instrucción Mover el dato del registro R2 R1 y dejar guardado en R1, la instrucción Mover el dato del registro R2 M, y dejar guardado en la memoria, la instrucción Mover el dato de la memoria M R1, y dejar guardado en el registro R1. Son instrucciones de esta clase ya que no necesitan ser operadas por ningún otro registro, solamente necesitan el código de operación que para estas será en el mover o transferir datos, en M se debe especificar la dirección en donde se quiere guardar el dato deseado, no se podrá ejecutar bien la operación si se coloca MOVRR y dos registros de propósito general.

COMP R1, R2 : Esta función también es de este tipo da instrucción por que solo necesita el registro R1 y el registro R2 y operar, pero no necesita un registro de destino, ya que necesita trabajar con la ALU, quien es la que va decir, por medio de una bandera si son iguales o diferentes, por ejemplo cuando necesito comparar un resultado de una SUM R1, R2, con un valor de un registro entonces utilizo esta instrucción. Quien va a llevar los dos valores o datos a la ALU y está me activa una bandera B en caso que sean iguales, el paso a seguir sería la de verificar esta bandera, que se encuentra en el registro BAND en el bit BD.

- **INSTRUCCIONES DE TIPO – J :** Bifurcación y bifurcación y enlace, refieren a los saltos que puede realizar mi programas, como llamados y para realizar un ciclo en determinada operación de mi programa. Se necesita el nombre o condición para producir un salto y el desplazamiento añadido al PC (Contador del Programa).

En nuestro procesador encontramos varias funciones de este tipo como son :

IRA Nombre; LLAM Nombre,: Ir a la etiqueta Nombre, Hacer un Llamado a Nombre con la condición de que debe regresar a este mismo sitio, es decir, que antes de moverse debe guardar la dirección donde esta en la pila. Son instrucciones que no necesitan r1, y simplemente tienen una operación determinada y un lugar a donde ir o desplazamiento en el PC, en verdad lo que hacen estas instrucciones es: Nombre PC.

SIC, SNIC R, Nombre : Salte si R es Igual a Cero y Salte si R No es igual a Cero, son funciones que deben saltar a una etiqueta Nombre, sin condición para que se devuelva simplemente si cumple la comparación con cero añade Nombre PC.

RLLAM : Retorne al Llamado sirve para cuando termine de realizar el llamado provocado por LLAM poderse devolver al sitio que estaba trabajando, lo que hace es sacar el valor guardado en la Pila PC.

CÓDIGO DE rs1 rs2 rd FUNCIÓN

OPERACIÓN

CÓDIGO DE rs1 rd INMEDIATO

OPERACIÓN

CÓDIGO DE rs1 rs2

OPERACIÓN

CÓDIGO DE r1 DESPLAZAMIENTO AÑADIDO AL PC

OPERACIÓN (Si es necesario)