

Control condicional y tablas de datos en RISCv

Departamento de Tecnología Electrónica
Universidad de Sevilla

1 Material

- Ordenador con RARS y CoolTerm instalados.
- Placa Basys3 con HispaRISC implementado sobre una FPGA Basys3 que incluye la ISA completa RV32I.
- Carpeta lab_kit con los archivos necesarios para la práctica.

2 Objetivos

- Aplicar el proceso presentado en la práctica anterior para la simulación y programación del procesador RISCv.
- Realizar programas que:
 - inicializan variables y tablas de datos en memoria
 - evalúan condiciones y en función de estas ejecutan diferentes partes del código.
 - usan bucles para repetir un bloque de código un número determinado de veces o mientras se cumpla una condición.
 - manejan direcciones de memoria y punteros
- Realizar un programa simple mediante subrutina

3 Estudio previo

Revisar y estudiar el conjunto de instrucciones ISA RISCv RV32I.

Repasar la práctica anterior donde se explica cómo trabajar con RARS y cómo programar el micro HispaRISC HR1BS3.

Leer con detalle el apartado 4.1, entender la solución propuesta y simular el programa en RARS.

4 Trabajo en laboratorio.

El trabajo en el laboratorio consistirá en la escritura, ensamblado y simulación de varios programas así como su ejecución sobre el procesador RISCv HispaRISC.

En primer lugar se trabajará con dos programas ya disponibles en lab_kit que habrá que modificar y a continuación con programas que deberán ser desarrollados en su totalidad por el alumno.

4.1 Determinación de la pertenencia a un intervalo para un valor de entrada

Cargue el programa compara.S incluido en el lab_kit. Se trata de un programa que en función de un número de entrada X, muestra un valor en pantalla, hay tres casos:

- $X < 4$: muestra 100
- $4 \leq X < 8$: muestra 140
- $8 < X$: muestra 180

Estructura de Computadores

- a) Analice y comprenda el programa. Compruebe su funcionamiento tanto a nivel de simulación en RARS como de ejecución en HispaRISC.
- b) Modifique el código anterior y obtenga un nuevo programa: compara2.S de forma que, en lugar de mostrar valores, ilumine los LEDs según los siguientes criterios. También haga que el número introducido se muestre en el display.
 - si $X < 4$: LED = 0000 0000 0000 1111
 - si $4 \leq X < 8$: LED = 0000 0000 1111 0000
 - si $8 \leq X < 12$: LED = 0000 1111 0000 0000
 - si $12 \leq X$: LED = 1111 0000 0000 0000

Recuerde (práctica 5) que para iluminar los LEDs basta escribir la palabra correspondiente en la dirección de memoria 0x40.

4.2 Búsqueda del dato de valor máximo de una tabla inicializada en memoria

Realice un programa que busque el máximo de una lista de números y lo muestre por pantalla. Debe inicializar en memoria una tabla de tamaño 10 con los siguientes datos:

14, 13, 28, 19, 16, 12, 12, 18, 2, 11

Cree un archivo nuevo (maximo.S) en RARS e introduzca su código ensamblador. Incorpore algún comentario para explicar el programa.

Compruebe su funcionamiento tanto a nivel de simulación en RARS como de ejecución en HispaRISC.

4.3 Búsqueda del dato de valor máximo o mínimo de una tabla inicializada en memoria

Obtenga un nuevo programa, minimax.S, de modo que busque el mínimo o el máximo de la siguiente lista de 20 números en función del valor del switch0. Si switch0 = 0 buscará el mínimo y si switch0 = 1 buscará el máximo. El resultado ha de mostrarse por pantalla.

17, 1, 23, 19, 18, 33, 15, 24, 4, 22, 28, 20, 7, 21, 5, 10, 14, 3, 7, 6

Compruebe su funcionamiento tanto a nivel de simulación en RARS como de ejecución en HispaRISC.

Recuerde (práctica 5) que para leer los switches basta leer la dirección de memoria 0x20.

4.4 Realización de un programa con subrutina

Realice un programa (parpadeo.S) que haga encenderse y apagarse continuamente un LED de la placa a una frecuencia del orden de Hz (apreciable por el ojo humano). Para ello, utilice una subrutina de espera consistente en decrementar un contador que habrá inicializado previamente a un valor alto (por ejemplo 2500000).

El procedimiento es el siguiente: encender diodo - saltar a subrutina de espera-apagar diodo - saltar a subrutina de espera - encender diodo -