Práctica 5:
Computador simple 3 (CS3)

Objetivos

El Departamento de Tecnología Electrónica de la Universidad de Sevilla ha desarrollado una implementación del CS3 para las placas Basys2de Digilent, que usa una FPGA de Xilinx, así como una herramienta para el PC que permite la carga, ensamblado y depuración de los programas escritos para dicho computador. La herramienta visualiza la unidad de datos y muestra la ejecución de las instrucciones paso a paso e incluso microinstrucción a microinstrucción, ayudando a una mejor comprensión de los conceptos estudiados en teoría.

En esta práctica se cubren los tres objetivos siguientes:

- Programar el CS3 en ensamblador.
- Comprobar el funcionamiento de las unidades de datos y control del CS3.
- Ver la ejecución del programa en una implementación real del computador.

Conocimientos previos

En esta práctica de laboratorio el alumno debe poseer los conocimientos teóricos de funcionamiento y programación del CS3.

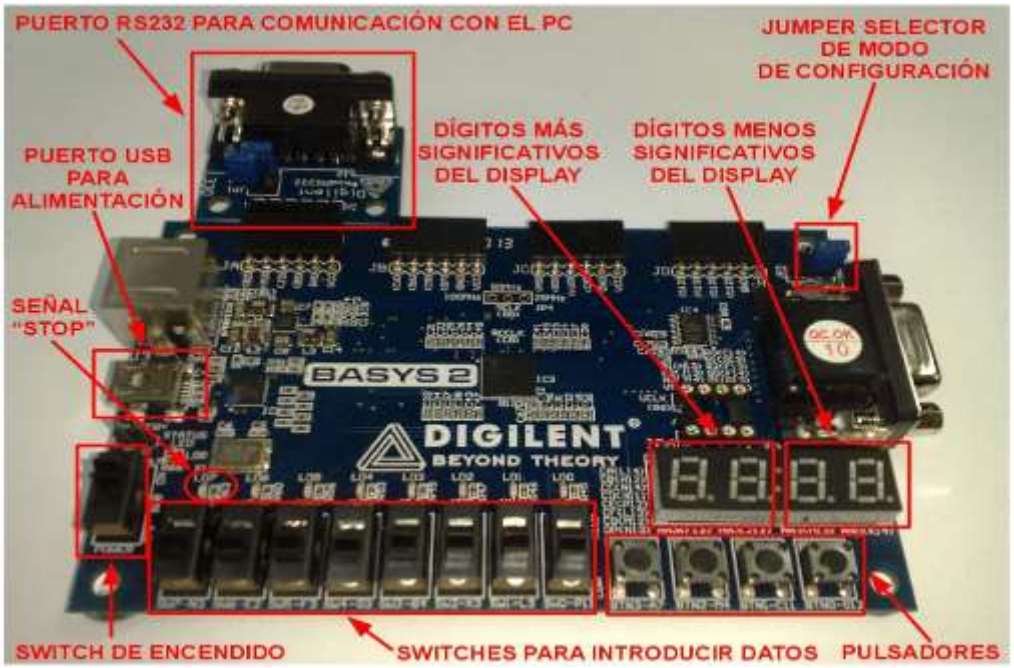
Componentes a utilizar

- Placa de Digilent Basys2.
- Software de programación de la placa.
- Software de interacción con el CS3.

Parte 1: Información básica

1. Implementación de CS3

Se ha implementado[1] un sistema basado en el CS3 usándola placa de Digilent Basys2 que se muestra en la figura:



El sistema implementado incluye los siguientes componentes:

- Procesador.
- Memoria de código.
- Memoria de datos.
- Dispositivos de entrada-salida mapeados en memoria.
- Unidad de depuración.

Los dispositivos de entrada-salida se han mapeado en el espacio de direccionamiento de la memoria de datos tal y como muestra la siguiente tabla:

Dirección	Sentido	Dispositivo	Descripción
$80	Entrada	Interruptores	Cada uno de los 8 bits de esta posición de memoria indica si el interruptor correspondiente está activo.
$81	Entrada	Pulsadores	Los 4 LSB de esta posición de memoria indican si los botones correspondientes están pulsados. Los 4 MSB valen 0.
$82-83	Salida	Display	La posición $83 controla los dos dígitos más significativos y la$82 los dos menos significativos. Se muestran en hexadecimal.

1. La implementación ha sido realizada por Jonathan Ruiz Páez como proyecto fin de carrera de Ingeniería Informática, dirigido por el profesor David Guerrero Martos.

La unidad de depuración es un componente que se comunica con el PC a través del puerto RS232 y le permite leer y escribir la memoria de código y datos. Además, hace posible depurar los programas descargados mediante:

- Ejecución de ciclos de reloj individuales.
- Ejecución de instrucciones individuales.
- Inspección del contenido de la memoria y los registros.
- Inspección del estado de las líneas de control.
- Habilitación/inhibición del reloj del procesador.
- La interacción con la unidad de depuración se realiza a través del software descrito a continuación.

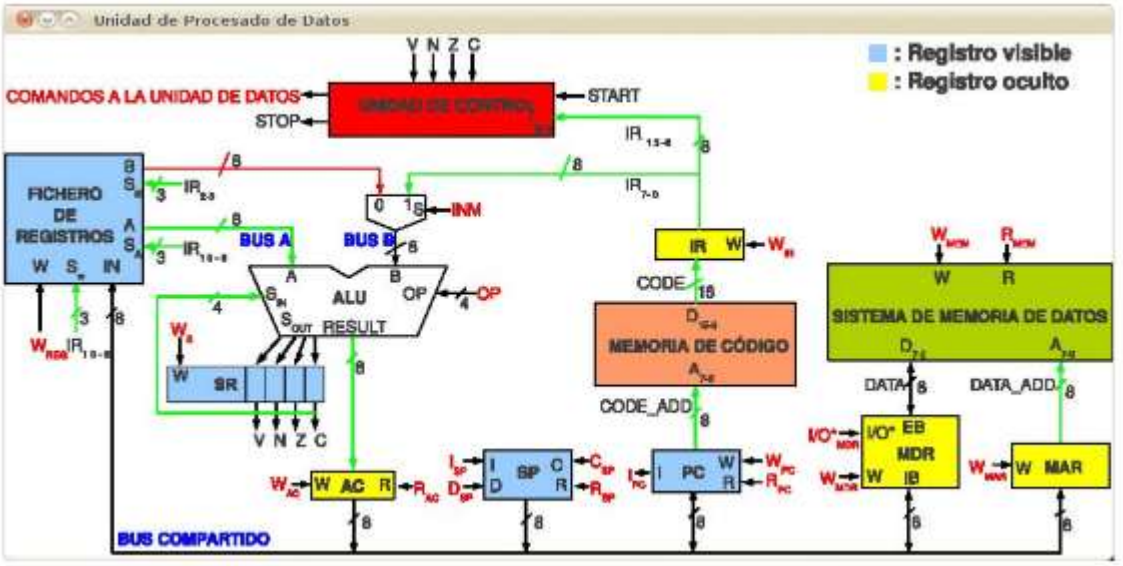
2. Sofware de apoyo en el PC.

Para interactuar con la unidad de depuración usaremos el software instalado en los PC del laboratorio. Dicho software está disponible en la página web de la asignatura. Para lanzarlo simplemente pulsamos sobre el icono correspondiente en el escritorio. Al hacerlo nos aparecerá la siguiente ventana:

La parte superior de la ventana contiene una serie de controles para interactuar con la unidad de depuración del CS3 que se describen a continuación:

- Campo para la ruta del puerto serie. Aparece relleno por defecto con la ruta correspondiente al puerto serie de los ordenadores del laboratorio, por lo que en principio no debe cambiarse.
- Campo de fichero de código binario (BIN): este campo debe contener la ruta del fichero binario cuyo contenido se escribirá en la memoria de código cuando se inicialice el procesador. Al rellenarse se mostrará en la parte superior derecha de la ventana el contenido de la memoria desensamblado. Normalmente no tendremos que preocuparnos de rellenar este campo ya que usaremos un fichero en formato ensamblador para especificar el programa que deberá escribirse.
- Campo de fichero de código ensamblador (ASM): este campo debe contener la ruta de un fichero de código ensamblador correspondiente al programa que será escrito en la memoria de código. Al rellenar este campo la herramienta ensamblará el fichero informando de los posibles errores y escribirá la ruta del fichero binario resultante en el campo comentado anteriormente. Rellenar este campo no es obligatorio, pero es la forma habitual de trabajar con la herramienta.

- Campo de fichero de datos: si deseamos inicializar la memoria de datos con el contenido de un fichero binario, podemos indicar su ruta en este campo.
- Botón de establecimiento de conexión e inicialización: al pulsarlo el software establece una conexión a través del puerto RS232con la unidad de depuración, ordena inicializar el CS3 y escribe las memorias de código y datos con el contenido de los ficheros especificados. Únicamente tras esto podremos interactuar con la unidad de depuración.
- Botón "Mostrar Unidad De Datos": al pulsar este botón aparecerá la ventana siguiente:



Durante las ejecuciones ciclo a ciclo se resaltarán en esta ventana las señales que se vayan activando.

- Botón "Habilitar Reloj"/"Deshabilitar Reloj": Si el reloj delCS3 está detenido, este control permite ponerlo en marcha para que se ejecute el programa de forma autónoma hasta encontrar una instrucción de STOP. Si el reloj está ya en marcha permite detenerlo para inspeccionar el estado del procesador y la memoria.
- Botón "Activar Start": Le indica a la unidad de depuración que genere un pulso alto en la señal de inicio del CS3.
- Botón "Ejecutar Un Ciclo": Le indica a la unidad de depuración que habilite el reloj durante un solo ciclo.
- Botón "Ejecutar Una Instrucción": Le indica a la unidad de depuración que habilite el reloj hasta que se ejecute la instrucción en curso o hasta que se llegue al estado de espera.

Cuando el reloj del CS3 está detenido se muestra el estado del procesador incluyendo:

- Contenido de la memoria de datos.
- Contenido de los registros visibles.
- Contenido de los registros ocultos.
- Estado de las líneas de control.
- Valor de los buses de entrada y salida de la ALU y del bus común.

Además, la instrucción en curso aparecerá resaltada. El contenido de la memoria de datos puede ser editado manualmente. También es posible guardar el contenido de dicha memoria en un fichero binario pulsando sobre el botón "Guardar Memoria Datos".

Parte 2: Estudio teórico

1. Escriba la subrutina MUL8 en ensamblador delCS3 que reciba en R5 y R6 dos números sin signo y devuelva el producto de ambos en R7.
2. Escriba un programa que use la subrutina anterior para calcular el producto de 22 por 10 y almacene el resultado en la posición de memoria 4.
3. Escriba una tabla que indique qué señales se activan y qué transferencias entre registros se realizan en cada ciclo de reloj durante la ejecución de las dos primeras instrucciones del programa del apartado anterior.
4. Usando la subrutina anterior, escriba un programa que implemente el siguiente algoritmo:

```
HACER
        MIENTRAS BOTON NO PULSADO
                NO HACER NADA
        FIN MIENTRAS
        R5 = NUMERO CODIFICADO EN INTERRUPTORES
        MIENTRAS BOTON3 NO PULSADO
                NO HACER NADA
        FIN MIENTRAS
        R6 = NUMERO CODIFICADO EN INTERRUPTORES
        R7 = R5 *R6
        MOSTRAR R7 EN EL DISPLAY
POR SIEMPRE
```

Compruebe que la sintaxis de los programas escritos es correcta usando el software disponible en la página de la asignatura.

NOTA IMPORTANTE: Es imprescindible traer a la sesión de laboratorio los ficheros de los programas ya escritos en soporte electrónico y con la sintaxis ya chequeada.

Parte 3: Estudio experimental

1. Asegúrese de que el puente de selección del modo de configuración de la placa Basys2 (JP3) conecta el pin central con el etiquetado con "ROM".
2. Conecte la placa al puerto RS232 del PC.
3. Alimente la placa de prototipado a través del puerto USB del PC.
4. Cargue en la memoria de código el programa que realiza la multiplicación de 22x10.
5. Ejecute ciclo a ciclo las dos primeras instrucciones comprobando que en cada ciclo se activan las señales de control y se realizan las transferencias entre registros que describió en el estudio teórico.
6. Ejecute instrucción a instrucción las dos líneas siguientes y compruebe que tienen el efecto esperado.
7. Habilite el reloj para ejecutarlas instrucciones restantes y vuelva a deshabilitarlo para comprobar el resultado final.
8. Cargue en la memoria de código el programa que implementa el algoritmo del apartado 4 del estudio teórico y haga que se ejecute de forma autónoma habilitando el reloj del CS3. Compruebe que el programa funciona correctamente multiplicando distintos valores.

NOTA: Si el software indica que no puede realizarse la conexión a pesar de haber conectado correctamente la extensión RS232, es posible que la placa no esté aún configurada para implementar el CS3. En tal caso pulse sobre el icono "configurar BASYS2 para CS3"y, cuando la grabación termine, apague y vuelva a encender la placa.