

SIMULACIÓN MEDIANTE LENGUAJE VHDL DE UNA UNIDAD DE CONTROL MICROPROGRAMADA CON FINES DIDÁCTICOS.

R. Rico López, J. A. de Frutos Redondo, J. M^a Clemente Párraga y D. Meziat Luna.
Departamento de Automática. Universidad de Alcalá.

En el presente trabajo se describe una aplicación desarrollada en lenguaje VHDL con el fin de realizar la simulación de la arquitectura de un microprocesador de 32 bits cuya unidad de control es microprogramada. El fin de esta aplicación es didáctico.

Hemos querido aplicar toda la potencia del VHDL a la didáctica de la Arquitectura de Computadores ya que nos permite, en el nivel más bajo, abstraer conocimientos obviando toda la problemática del conexionado de circuitos físicos y, en el nivel más avanzado, estudiar la implementación de algoritmos, técnicas arquitectónicas y sus rendimientos de una forma gráfica que sería muy difícil llevar a cabo con dispositivos comerciales.

El diseño de la aplicación cubre todos los aspectos propios de la Arquitectura de Computadores excepto el nivel tecnológico de implementación en silicio. Se ha construido la ruta de datos, se ha diseñado el juego de instrucciones y su formato, se ha determinado el tipo de secuenciamiento del microprograma y el formato de las microinstrucciones y se han corrido programas sobre esta máquina evaluando su correcto funcionamiento. Este trabajo no solo no agota las posibilidades didácticas de esta línea sino que permite explorar nuevos caminos.

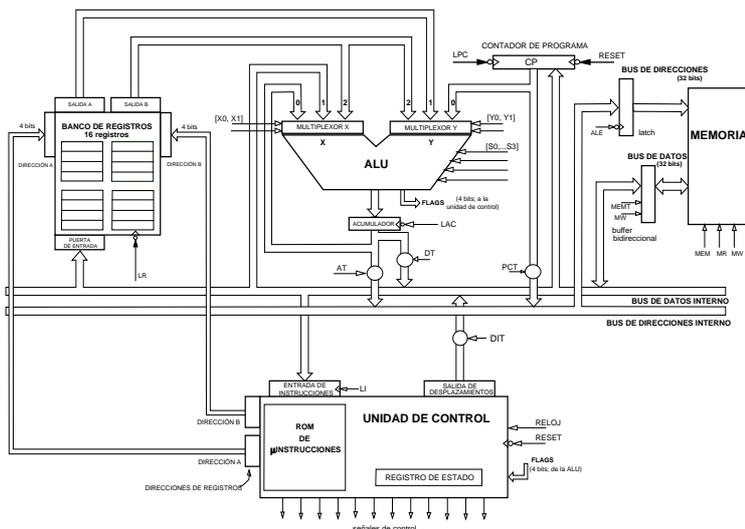


Figura 1. Arquitectura del procesador simulado.

2. El juego de instrucciones

El diseño del juego de instrucciones de esta máquina está sujeto a muy pocas restricciones por tratarse de una arquitectura simulada. Solo hemos de ajustarnos a un formato tratable por el hardware y a las especificaciones a experimentar. Los modos de direccionamiento permitidos son el directo a registro, el relativo a registro más desplazamiento y el inmediato[1]. Cada vez que se realiza una fase de búsqueda se leen los 4 bytes del formato aun cuando alguno pueda no ser utilizado.

3. La unidad de control

En la figura 2 se ilustra la unidad de control microprogramada con secuenciamiento implícito[2]. La ROM de microinstrucciones en VHDL es un

1. Arquitectura de la ruta de datos

Dada la limitación de espacio no nos vamos a detener en la descripción de la aplicación pero si queremos mostrar un modelo terminado y probado con éxito y que abre perspectivas nuevas. La figura 1 ilustra la arquitectura del computador implementado en lenguaje VHDL[3]. Esta máquina de pizarra se ajusta a la arquitectura de Von Neumann[2], con la simplificación de que no tiene sistema de E/S.

Esta formado por una unidad aritmético-lógica, un banco de 16 registros, un interfaz con memoria, una unidad de direccionamiento y la unidad de control. Los buses y los registros de la máquina son de 32 bits.

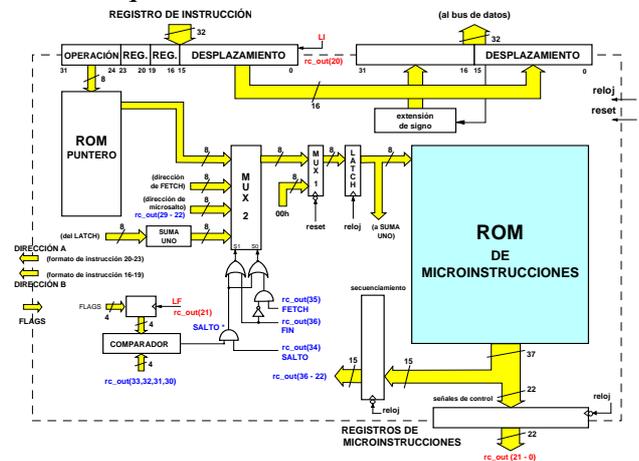


Figura 2. Unidad de control.

fichero ASCII[4,5]. De esta manera, cambiar la forma de ejecutar una instrucción es tan fácil como grabar una ROM (reescribir fichero ASCII) y añadir instrucciones nuevas supone añadir microprogramas en el fichero.

3.1 Formato de las microinstrucciones

Las microinstrucciones tienen un formato en el que se pueden distinguir dos partes: el conjunto de señales de control y los bits responsables del secuenciamiento de la propia ROM.

Las posibles microdirecciones son: comienzo del microprograma de la fase de búsqueda, la primera de un microprograma de ejecución de una instrucción, siguiente en la secuencia normal y salto debido a una microbifurcación. Los bits de secuenciamiento son FIN (si es '1' se continua en la secuencia y si es '0' se termina el microprograma), FETCH ('1' corre el microprograma de la fase de búsqueda) y SALTO (este bit funciona en combinación con un comparador del registro de estado que evalúa una condición dada por otros 4 bits, uno por cada *flag*: C, N, Z y OV).

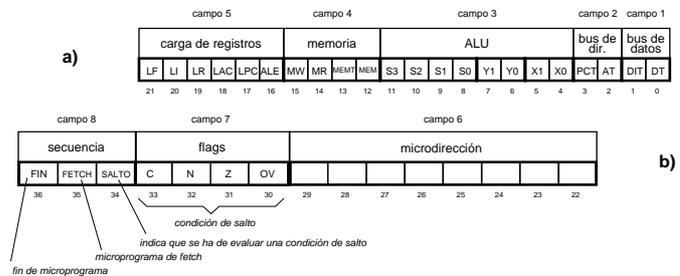


Figura 3. Formato de microinstrucción.

Tomaremos como ejemplo una instrucción sencilla: LD r,n que carga el dato inmediato n en el registro r y cuyo cornograma vemos en la figura 4. Si escribimos el microprograma siguiendo el formato de la figura 3 obtendríamos:

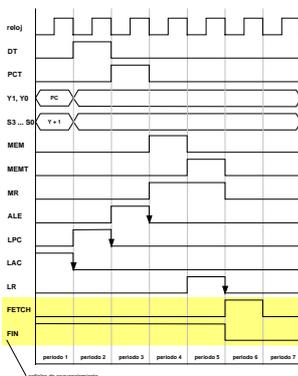


Figura 4. LD r,n.

```

1 0000 0000 0000 0000 0100 0000 1010 0000 0000
1 0000 0000 0000 0000 0010 0000 0000 0000 0001
1 0000 0000 0000 0000 0001 0000 0000 0000 1000
1 0000 0000 0000 0000 0000 0000 0101 0000 0000 0000
1 0000 0000 0000 0000 1000 0110 0000 0000 0000
0 1000 0000 0000 0000 0000 0000 0000 0000 0000
    
```

4. Conclusiones

Sobre esta arquitectura vamos a poder implementar juegos de instrucciones con diferentes formatos y modos de ejecución para evaluar su rendimiento. Además, dado que la unidad de control es microprogramada, se permitirá cambiar el microcódigo de la misma modificando el funcionamiento de la máquina o añadiendo nuevas operaciones con gran facilidad.

Se abren perspectivas nuevas tales como el desarrollo de ensambladores y compiladores, optimizadores de código ejecutable, etc. En todos ellos se trabaja con un código fuente (ensamblador o lenguaje de alto nivel) que es traducido a un fichero ASCII que simula el contenido de la memoria. En la actualidad estamos desarrollando un ensamblador parametrizable de modo que seamos capaces de implementar juegos de nemónicos y diferentes formatos de instrucciones.

También es posible la medida del rendimiento de la arquitectura y la evaluación de algoritmos, técnicas de ordenación de código, etc. mediante la inclusión en la descripción del modelo de variables que registren eventos y generen una traza de la ejecución de un programa.

5. Referencias

1. J. L. Hennessy y D. A. Patterson, *Arquitectura de computadores. Un enfoque cuantitativo*. McGrawHill, 1993.
2. J. A. de Frutos y R. Rico, *Arquitectura de computadores*. Servicio de Publicaciones UAH, 1995.
3. *IEEE Standard VHDL Language Reference Manual*. Published by the IEEE, 1994.
4. R. Lipsett, C. Schaefer y C. Ussery, *VHDL: Hardware Description and Design*. Kluwer Academic Publishers, 1989.
5. P. J. Ashenden, *The VHDL Cookbook*. University of Adelaide, South Australia; FTP desde ftp.cs.adelaide.edu.au (129.127.8.8) pub/VHDL-Cookbook.