

APOYO DIDÁCTICO DEL LENGUAJE VHDL EN EL ESTUDIO DE UNA UNIDAD DE CONTROL MICROPROGRAMADA

Carlos Federico Hernández Farfán

Tecnológico Nacional de México/Instituto Tecnológico Superior de Irapuato
cahernandez@itesi.edu.mx

Abissay Natanael Hernández Armenta

Tecnológico Nacional de México/Instituto Tecnológico Superior de Irapuato
abbisayhdz@outlook.com

José Humberto Sánchez Galeana

Tecnológico Nacional de México/Instituto Tecnológico Superior de Irapuato
beto_9303@hotmail.com

Resumen

En este trabajo se presenta una experiencia educativa en el campo de Arquitectura de Computadoras, en la cual se empleó el lenguaje VHDL como apoyo didáctico para cubrir los conceptos relacionados a una unidad de control microprogramada. En el estudio de esta unidad de control microprogramada se hizo énfasis en el funcionamiento interno de un procesador, en el cual se realizan transferencias de datos entre componentes en intervalos de tiempo muy pequeños y en el que se generan señales internas que resultan complicadas de verificar sin el equipo especializado de laboratorio. Se pretende que se puedan verificar los conceptos teóricos mediante las simulaciones del funcionamiento interno de los componentes del procesador, especialmente la unidad de control. Mediante el lenguaje VHDL y la técnica de diseño asistido por computadora, se realizó la descripción de los componentes digitales que conforman un pequeño procesador de 16 bits y su unidad de control. Se estudió qué señales son necesarias para el

control del procesador y cómo se generan empleando la técnica de control microprogramado. Finalmente se muestra la percepción de los estudiantes del grado de aceptación del uso del lenguaje VHDL como apoyo didáctico para el estudio de una unidad de control microprogramada

Palabra(s) Clave: Arquitectura de Computadoras, Control microprogramado, Procesador, VHDL.

Abstract

This article shows an educational experience in the area of computer architecture, in which the language VHDL is used as didactic support to cover the concepts related to a microprogrammed control unit. In the study of this unit, emphasis was done on the internal operation of a processor, in which data transfers between components is performed in very small time intervals and in which signals generated are complicated to verify without the specialized laboratory equipment. It is intended to verify the theoretical concepts through simulations of the internal operation of the processor components, especially the control unit. Using the VHDL language and the computer assisted design technique was performed the description of the components that make up a small 16 bit processor and its control unit. Signals for processor control were studied and how they are generated using the microprogrammed control technique. Finally, the students' perception is shown of the degree of acceptance of the use of the VHDL language as didactic support for the study of a microprogrammed control unit.

Keywords: Computer architecture, Microprogrammed control, Processor, VHDL

1. Introducción

En el programa de estudios de la carrera de Ingeniería en Sistemas Computacionales, se imparte la materia de Arquitectura de Computadoras, aunque en el temario se incluye el estudio de la unidad central de procesamiento y la unidad de control, no se especifica el estudio de una unidad de control microprogramada, tal como puede verse en Instituto Tecnológico Superior de Irapuato [ITESI, 2017]. Sin embargo, es de interés el estudio de la unidad de

control microprogramada ya que constituye una alternativa elegante y sistemática a las técnicas de diseño lógico convencional, tal como sugiere Morris [2000].

El tema de unidad de control microprogramado se puede considerar como intermedio en el grado de complejidad para los estudiantes de nivel licenciatura, sin embargo, permite dar continuidad en el estudio de Arquitectura de Computadoras hacia temas más avanzados. Tal como muestra Hernández [2015], en el que se explica el uso del lenguaje VHDL como apoyo en la enseñanza de la materia de arquitectura de computadoras, en este trabajo se extiende su uso para mostrar una experiencia en el estudio de la unidad de control microprogramada.

En la revisión bibliográfica, se encuentran libros de Arquitectura de Computadoras o Lógica Programable que, aunque son afines no incluyen el tema de control microprogramado. Por ejemplo, el libro de Miguel [1997], de Arquitectura de Computadoras, no incluye el tema de control microprogramado. El libro de Maxinez y cols. [2003], incluye la descripción de un procesador y una unidad de control parecida a una unidad de control microprogramada, pero no explica el funcionamiento de las señales de control y la explicación de los conceptos de control es breve. El libro de Larrea [2002], incluye varios ejemplos de sistemas digitales descritos en VHDL como unidades aritméticas y lógicas, pero no incluye el tema de control microprogramado. EL libro de Parhami [2007], incluye un capítulo denominado Síntesis de Unidad de Control e incluye la microprogramación como subtema, se explican de manera extensa los conceptos, sin embargo, la implementación del sistema digital correspondiente tiende a ser más complicada ya que se refiere a un procesador un tanto complejo. En el libro de Morris [2000], se presenta un capítulo denominado Unidad de Control Microprogramada, la cual es explicada basada en un procesador simplificado, por lo que las descripciones en VHDL de estos circuitos digitales pueden realizarse siguiendo una secuencia didáctica acorde a un curso introductorio. En el libro de Romero [2007] se incluye la descripción en VHDL de circuitos digitales desde básicos hasta complejos, aunque en algunas descripciones se emplean diferentes paquetes de la librería estándar de la IEEE. De acuerdo a esta revisión, se emplea como apoyo para el desarrollo de este trabajo los dos últimos libros, ya que

presentan el tema de una manera creciente en grado de complejidad y la descripción en VHDL de los componentes básicos de un procesador y una unidad de control, aunque este trabajo se emplea sólo un paquete de la librería.

En diferentes artículos como el de Morales *et al.* [2012], y el de Romero *et al.* [2006], se muestra el desarrollo de procesador mediante el uso del lenguaje VHDL, sin embargo, éstos procesadores presentan mayores capacidades y por lo tanto un poco más complicados para su estudio. En los artículos de Rosado y cols. [2008] y el de Perales y cols. [2012] se muestran algunas estrategias para el estudio de electrónica y sistemas digitales, en los que se busca acercar los conceptos teóricos a la práctica apoyándose en herramientas tecnológicas para la enseñanza.

De esta manera se optó por utilizar la estructura de un procesador simplificado que permita conocer qué señales se requieren para ejecutar una instrucción de un programa y cómo se generan estas señales empleando la técnica de control microprogramado; apoyándose para su estudio en el lenguaje VHDL y simulaciones del funcionamiento del procesador y la unidad de control, buscando un balance entre la teoría y la práctica.

2. Metodología

En esta sección se muestra cómo se empleó el lenguaje VHDL como apoyo didáctico para el estudio de una unidad de control microprogramada y la percepción del grado de aceptación de los estudiantes del uso de esta herramienta.

Se revisó la información documental y libros de texto para tomar como base el material que presenta el tema de control microprogramado de manera gradual en orden de complejidad, y que presenta el tema de manera simplificada para su estudio, pero lo suficientemente completo para cubrir los conceptos.

Se empleó el lenguaje de descripción de circuitos digitales VHDL debido a su versatilidad para la construcción de sistemas digitales, además de apegarse al estándar, así como su interoperabilidad en diferentes plataformas. Se describieron los sistemas digitales a partir de sus componentes básicos que lo integran

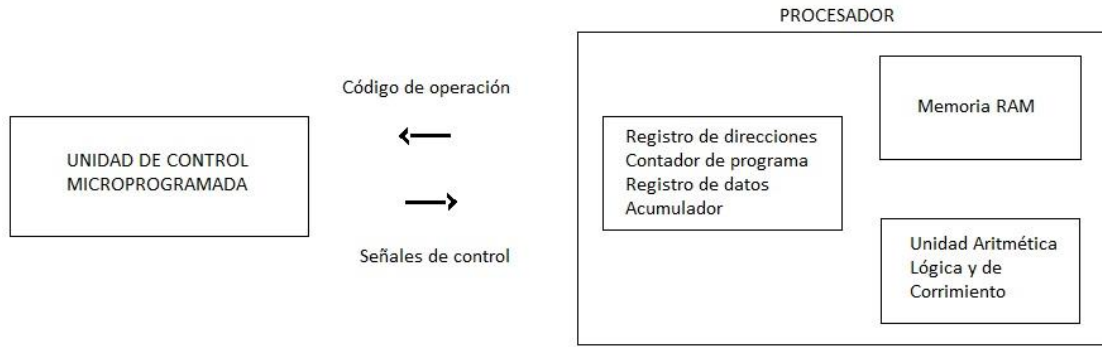
empleando la técnica de descripción jerárquica. De esta manera se empleó una sola librería estándar de la IEEE y un solo paquete de la librería, [Librería IEEE, paquete std logic 1164].

Con el objeto de verificar de forma didáctica los conceptos teóricos relacionados a la unidad de control microprogramada y debido a que son señales internas del procesador y que ocurren en intervalos de tiempo sumamente pequeños; se estudiaron a partir de simulaciones del funcionamiento de los sistemas digitales descritos en VHDL. Mediante las simulaciones de los componentes, es posible determinar qué señales son necesarias para el control del procesador y cómo se generan empleando la técnica de control microprogramado.

Para la descripción de los circuitos, compilación y simulación de los componentes digitales tanto del procesador como de la unidad de control se empleó el software Active-HDL Student Edition de ALDEC® versión 2016.

Dado que el enfoque es didáctico, los componentes que integran tanto el procesador como la unidad de control se describen a partir de componentes digitales básicos y empleando una sola librería estándar del lenguaje y un solo paquete de la librería; IEEE.std_logic_1164. Se describieron en VHDL los componentes individuales de los sistemas digitales; tales como multiplexores, registros y decodificadores, tal como se muestra en el libro Romero [2007]. Los cuales posteriormente se integran para conformar componentes más complejos como la unidad aritmética y lógica, la memoria RAM y su vez integrarlos con otros componentes para formar un pequeño procesador y la unidad de control de acuerdo a libro de Morris [2000].

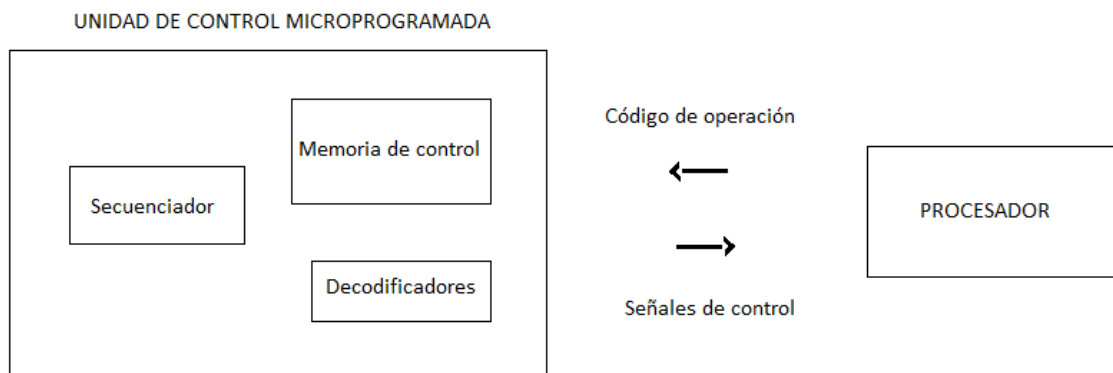
En la figura 1 se muestra un diagrama a bloques simplificado de un procesador basado en Morris [2000]. Este procesador consta de una memoria RAM en la que se almacena el programa principal y los datos del programa. Cuenta con una unidad aritmética y de corrimiento capaz de realizar 14 operaciones. Cuenta con registro de dirección para la memoria RAM, registro contador de programa, registro de datos, registro acumulador y multiplexores para el control del flujo de datos.



Basado en Morris [2000].

Figura 1 Diagrama a bloques del procesador.

En la figura 2 se muestra el diagrama a bloques simplificado de la unidad de control microprogramada basado en Morris [2000]. Consta de una memoria ROM, la cual contiene las diferentes rutinas de microinstrucciones necesarias para la ejecución de las diferentes instrucciones del procesador. Cuenta con un registro de direcciones para la memoria de control y un secuenciador para generar la próxima dirección de la memoria de control. La palabra de control que se obtiene de la memoria ROM, consta de 20 bits, éstos bits se agrupan en diferentes campos de los cuales una parte se emplea para generar la próxima dirección de la memoria de control y el resto se decodifica para generar las señales hacia el procesador. Esta unidad genera las señales necesarias para que el procesador realice las transferencias de datos y operaciones de la ALU con el fin de que se completen las instrucciones del programa.



Basado en Morris [2000].

Figura 2 Diagrama a bloques de la unidad de control basada en Morris (2000)

En figura 3 se muestra la declaración de las entradas y salidas de un procesador de 4 bits. Este procesador está integrado por diferentes componentes como multiplexores, registros, memoria RAM, unidad aritmética lógica y de corrimiento. Los cuales se integran tal como se muestra en la figura 4. Posteriormente se realizan las pruebas del funcionamiento del procesador aplicando las señales de control directamente como entradas de usuario en la simulación. Una vez comprobado mediante la simulación que el procesador es capaz de realizar diferentes microoperaciones, se realiza la descripción y simulación del procesador incrementando su capacidad a 8 y 16 bits. Con esto se pretende que el estudiante comprenda cómo funciona internamente el procesador para que posteriormente estudie cómo se generan las señales necesarias que controlan su funcionamiento.

```
library ieee;
use ieee.std_logic_1164.all;

entity procesador is port(
    load: in std_logic_vector(3 downto 0);
    reloj,reset: in std_logic;
    sar: in std_logic;
    sdr: in std_logic_vector(1 downto 0);
    as: in std_logic_vector(3 downto 0);
    sel1,sel2: in std_logic_vector(1 downto 0);
    we : in std_logic;
    cin,cero: in std_logic;
    cout: out std_logic);
end procesador;
```

Figura 3 Entradas y salidas del procesador

```
begin
    u0:mux214 port map(pc,dr,sar,muxar);
    u1:registro port map(muxar,load(0),reset,reloj,ar);
    u2:registro port map(ar,load(1),reset,reloj,pc);
    u3:mux414 port map(ram,ac,pc,inha,sdr,muxdr);
    u4:registro port map(muxdr,load(2),reset,reloj,dr);
    u5:registro port map(alus,load(3),reset,reloj,ac);
    u6:ram4 port map(dr,ar,we,as,ram,reset);
    u7:ALU port map(dr,ac,sel1,sel2,cin,cero,alus,cout);
end arqproc;
```

Figura 4 Interconexión de componentes del procesador.

En la figura 5 se muestra una prueba del funcionamiento interno del procesador y se verifica las señales que lo controlan. En este caso el contenido del contador de programa (8), se carga en el registro de datos, el cual a su vez se aplica a la

memoria como dato de entrada, y este mismo valor se carga en el registro de direcciones para la memoria. Con el dato de entrada (8) y la dirección (8), se aplica el pulso para almacenar el dato en memoria.

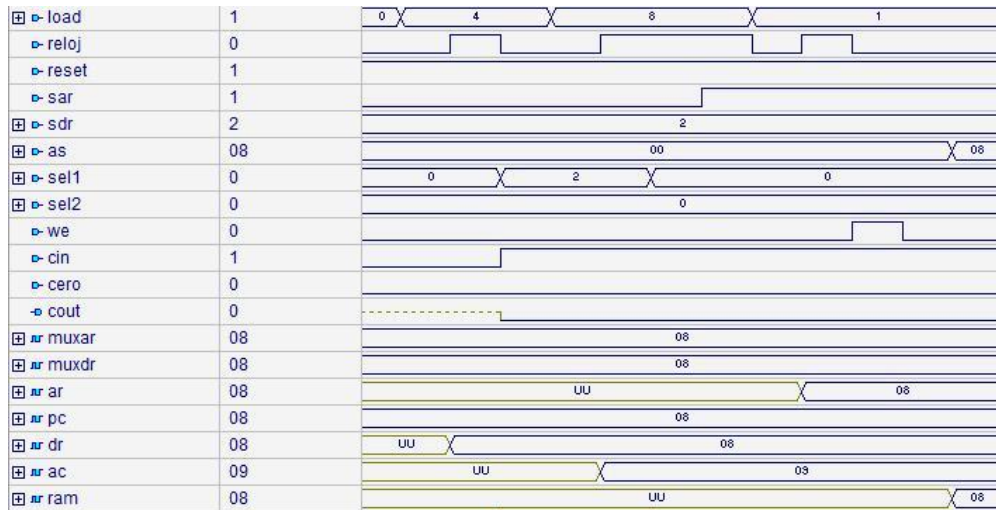


Figura 5 Prueba del procesador de 8 bits.

En la ALU se realiza el incremento del dato (9). Finalmente se aplica la dirección de salida de la memoria para leer el dato almacenado. Una vez que se ha descrito en VHDL y simulado el procesador básico, se realiza la descripción en VHDL e interconexión de los componentes que conforman la unidad de control microprogramada de acuerdo la arquitectura mostrada en Morris [2000].

En esta sección se realizan las simulaciones correspondientes a mostrar y verificar el funcionamiento interno de la unidad de control y los conceptos relacionados con una unidad de control microprogramada tales como memoria de control, microprograma, instrucción, microinstrucción, código de operación, palabra de control, microoperaciones, ciclo fetch, ciclo de instrucción, direccionamiento, etc. Se verificó que la unidad de control al recibir un código de operación del procesador sea capaz de generar la dirección de lectura de la memoria de control, y generar las señales para la ejecución de las microoperaciones del procesador a fin de que se ejecute una instrucción. Para realizar la descripción en VHDL de la unidad de control microprogramada, primero se realiza la descripción de los

componentes individuales y una vez comprobado su funcionamiento, se interconectan empleando la técnica jerárquica.

En la figura 6 se muestran las señales de entrada y de salida de la unidad de control microprogramada. De acuerdo a Morris [2000], en la memoria ROM de la unidad de control, se almacenan microinstrucciones que permiten generar las señales necesarias hacia el procesador y que determinan una microoperación; así como obtener la dirección de la próxima microinstrucción.

```
library ieee;
use ieee.std_logic_1164.all;
entity UnidadDControl is port(
    codop: in std_logic_vector(3 downto 0);
    reset, reloj, load: in std_logic;
    cin, U, I, S, Z : in std_logic;
    yincrc : in std_logic_vector (6 downto 0);
    cout : out std_logic);
end UnidadDControl;
```

Figura 6 Entradas y salidas de la unidad de control.

En la figura 7 se muestra el código VHDL de las señales de entrada de la memoria ROM. Como puede observarse, la memoria ROM, recibe una señal "a" que se emplea como dirección de lectura de la memoria. La señal de salida de la memoria ROM es de 20 bits, cuyos campos son: campo de dirección para la memoria de control, campo de condición y campo de tipo de transferencia. Los campos se emplean para determinar una microoperación en particular, la cual es decodificada y mediante la cual se genera la señal requerida por el procesador para realizar la microoperación.

```
library ieee;
use ieee.std_logic_1164.all;
entity rom128x20 is port (
    a : in std_logic_vector (6 downto 0);
    ad : out std_logic_vector (6 downto 0);
    br, cd : out std_logic_vector (1 downto 0);
    f1, f2, f3 : out std_logic_vector (2 downto 0));
end rom128x20;
```

Figura 7 Entradas y salidas de la memoria de control.

La descripción en VHDL de la memoria ROM y su contenido se realiza por medio de la estructura de control "Case", tal como se muestra en la figura 8. En la cual por cada dirección "a", se muestra a la salida de la memoria su contenido "d" en 20 bits. Para verificar los conceptos relacionados a la memoria de control, se muestran a continuación algunos ejemplos de simulaciones.

```
architecture arqrom of rom128x20 is
signal d: std_logic_vector (19 downto 0);
begin process (a,d)
begin
case a is
when "0000000" => d <= "000000000001011000011";
when "0000001" => d <= "00010000000000000010";
when "0000010" => d <= "00100000000001000000";
```

Figura 8 Descripción de la memoria de control.

De acuerdo a Morris [2000], la unidad de control recibe del procesador un código de operación relacionado a la instrucción que se debe ejecutar. Para cada instrucción, se cuenta en la memoria de control una rutina con palabras de control (microinstrucciones) que se leen de esta memoria, se decodifican y se generan las señales hacia el procesador para que se realicen las transferencias entre registros del procesador y operaciones en la ALU (microoperaciones). Por ejemplo, para una instrucción, se tiene un código de operación del cual se obtiene una dirección para leer la memoria de control, y ésta dirección hace referencia a una microinstrucción de la rutina de la instrucción, puede provocarse un salto a una rutina recurrente para todas las instrucciones, la rutina de direccionamiento indirecto. Posteriormente se ejecutan las microoperaciones de la instrucción como tomar el dato de la memoria y cargarlo en el registro de datos, así como obtener la dirección de la próxima microinstrucción, figura 8.

Al decodificarse los campos de la palabra de control se generan señales que pueden indicar, por ejemplo, qué transferencia se requiere entre los registros del procesador para completar esta instrucción y determinan cómo se va a obtener la siguiente dirección para la memoria de control.

En la figura 9 se observa la lectura de la memoria ROM empleando la dirección 02hx. En esta localidad se encuentra a microinstrucción de sumar el contenido del

acumulador y el registro de datos. Esta dirección se determinó mediante la palabra de control anterior. En la figura 10 se observa la lectura de la memoria ROM empleando la dirección 40hx. Esta dirección se determinó mediante la palabra de control anterior. Esta dirección también es incrementada. Al ocurrir el séptimo pulso de reloj, se obtiene la dirección para leer la memoria. La palabra de control provoca que se active la señal de transferencias del procesador y cómo obtener la próxima dirección para la memoria de control.

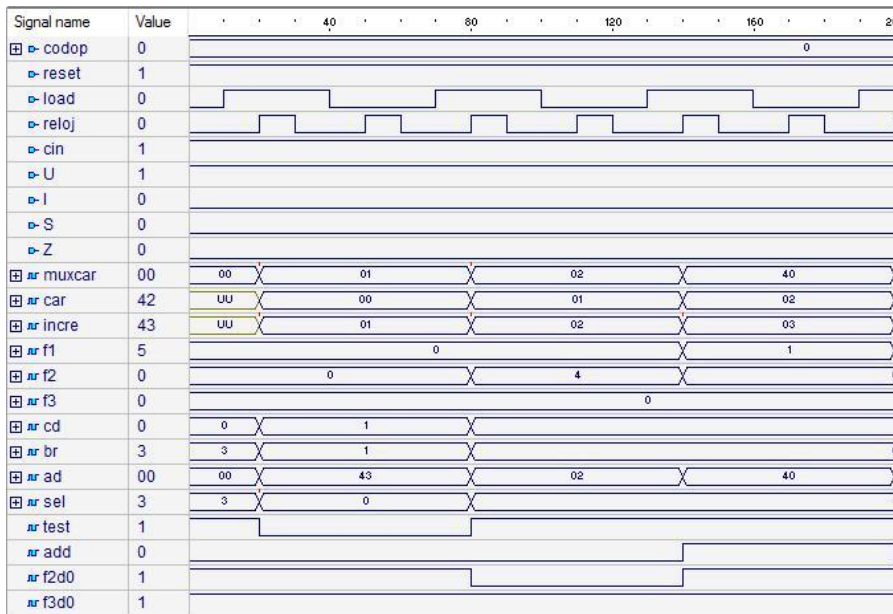


Figura 9 Rutina ADD.

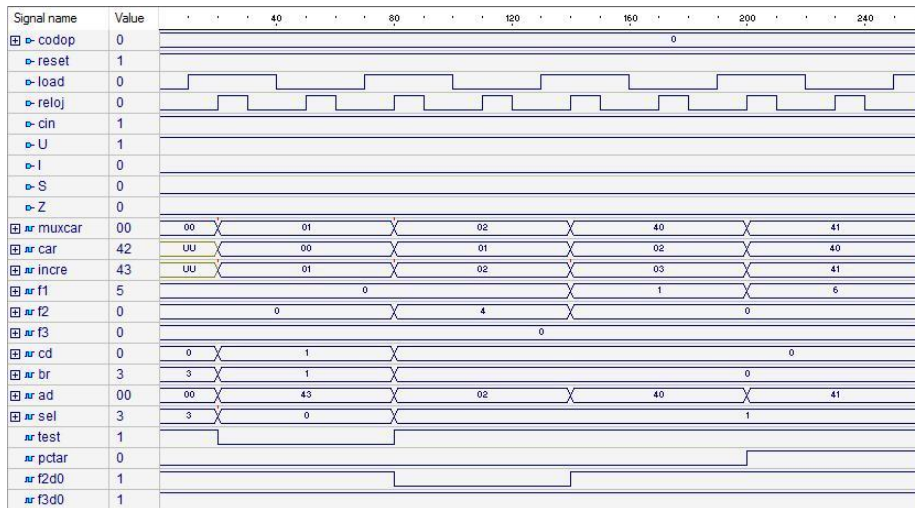


Figura10 Rutina FETCH.

Para el desarrollo de este proyecto se trabajó con grupo piloto de 32 estudiantes del Instituto Tecnológico Superior de Irapuato, de la carrera de Ingeniería en Sistemas Computacionales en la materia de Arquitectura de Computadoras del periodo agosto diciembre del 2016 y se les realizó una encuesta para conocer su percepción del uso del lenguaje VHDL como herramienta didáctica.

3. Resultados

En este trabajo se mostró el uso del lenguaje VHDL como apoyo didáctico para el estudio de una unidad de control microprogramada en el campo de Arquitectura de Computadoras. El énfasis en este trabajo consistió en verificar mediante simulaciones, conceptos teóricos tales como código de operación, memoria de control, microoperaciones, rutina de instrucción, secuenciamiento, ciclo FETCH; y principalmente qué señales internas son requeridas para que se ejecute una instrucción y cómo se generan estas señales. En la tabla 1 se muestran los grupos impartidos de la materia de Arquitectura de Computadoras en el año 2016 y actual. Para el desarrollo de este proyecto se trabajó con el grupo vespertino del periodo agosto - diciembre del 2016, que representa el 51.6% del total de los estudiantes del periodo. La materia se imparte en el quinto semestre, participaron 25 hombres y 7 mujeres, 3 estudiantes se encontraban en repetición de curso.

Tabla 1 Grupos de Arquitectura de Computadoras.

	2016	2017
Grupo matutino	30 alumnos	29 alumnos
Grupo vespertino	32 alumnos	27 alumnos

En la figura 12 se muestra la encuesta que se aplicó a los estudiantes con el objeto de conocer su percepción del uso del lenguaje VHDL como apoyo didáctico en el estudio de una unidad de control microprogramada.

En la figura 13 se muestra que prácticamente todos los alumnos del grupo piloto, consideran que el uso del lenguaje VHDL contribuye en su aprendizaje y se puede verificar el funcionamiento interno del procesador y la unidad de control microprogramada.

- 1.- ¿La primera vez que escuchaste sobre el concepto del procesador se te hizo difícil entenderlo?
- 2.- ¿La primera vez que escuchaste sobre el concepto de la unidad de control microprogramada se te hizo difícil entenderlo?
- 3.- ¿Alguna vez habías visto el funcionamiento interno de un procesador?
- 4.- ¿Alguna vez habías visto el funcionamiento interno de una unidad de control microprogramada?
- 5.- ¿Conoces alguna manera de poder visualizar el funcionamiento de un procesador y una unidad de Control microprogramada?
- 6.- ¿Considera que el uso del lenguaje VHDL ha contribuido satisfactoriamente en el aprendizaje de los conceptos de la materia de Arquitectura de computadoras?
- 7.- ¿Es más fácil comprender los conceptos de la materia Arquitectura de computadoras con el uso de simulaciones en VHDL?
- 8.- ¿Qué tan difícil le ha resultado describir y simular componentes en VHDL?
- 9.- ¿Considera que mediante el uso del lenguaje VHDL es posible cubrir los temas de Arquitectura de computadoras en forma fluida y gradual en su grado de complejidad?
- 10.- ¿Considera que mediante el uso del lenguaje VHDL es posible ver el funcionamiento interno de un procesador y una unidad de control Microprogramada?

Figura 12 Encuesta para alumnos

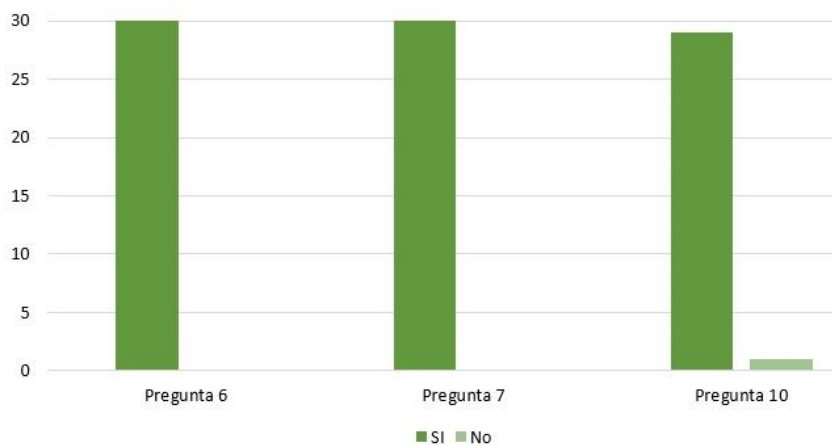


Figura 13 Uso de VHDL como apoyo didáctico.

En la figura 14 se muestra que el 77 % consideró que se pueden cubrir los conceptos teóricos de manera fluida y gradual en el grado de complejidad de los sistemas digitales. En la figura 15 se muestra que el 33% considera que es un lenguaje difícil de aprender y sólo el 10% lo considera fácil.

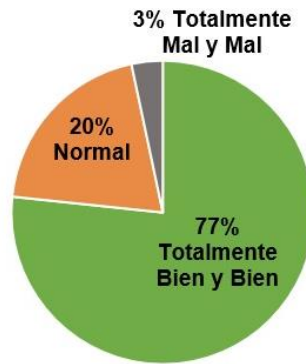


Figura 14 Uso de VHDL en forma gradual en complejidad

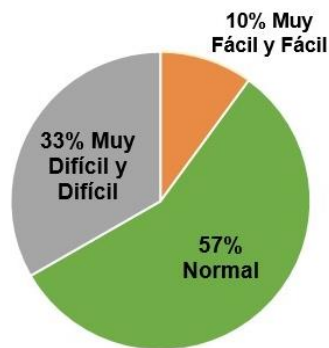


Figura 15 Dificultad del uso de VHDL

Finalmente, como resultado de esta experiencia se realizó una Guía de Instrucción para el docente con la planeación de las actividades a realizar en el periodo de un curso de un semestre a fin de cubrir los conceptos básicos hasta la unidad de control microprogramada.

4. Discusión

De acuerdo con lo mostrado en la metodología del trabajo, se realizó la descripción en VHDL y simulación de un procesador básico y su unidad de control. Debido al enfoque didáctico ambos componentes se probaron por separado, con el objetivo de identificar qué señales requiere el procesador para realizar transferencias y operaciones, y cómo se generan estas señales por medio de la unidad de control.

En las simulaciones mostradas en este trabajo sólo incluyen una parte representativa del funcionamiento del procesador y la unidad de control; tales

como la escritura y lectura de la memoria del procesador, operación aritmética en la ALU, generación de las señales de control para la ejecución de una instrucción en particular o el inicio del ciclo FETCH. Para la revisión de otro concepto en particular del funcionamiento de los componentes se debe preparar una simulación adicional.

Aunque el lenguaje VHDL y la simulación de los sistemas digitales pueden ser usados como una herramienta didáctica, se requiere de un esfuerzo adicional por parte de los estudiantes para aprender primero este lenguaje y posteriormente estudiar los sistemas digitales.

Como conclusiones se tiene que mediante el uso del lenguaje VHDL y las simulaciones de los sistemas digitales se verificaron conceptos relacionados al funcionamiento interno de un procesador y al funcionamiento de una unidad de control. Es decir, se verificó qué señales son requeridas para el funcionamiento del procesador y cómo se generan dichas señales empleando la técnica del control microprogramado.

Se describieron los sistemas digitales llevando una secuencia didáctica en la que se parte de los componentes digitales básicos, los cuales posteriormente se integran para conformar componentes digitales mayores hasta conformar un procesador de 16 bits y una unidad de control microprogramada.

Aunque el lenguaje VHDL y las simulaciones constituyen una buena herramienta para el estudio de los sistemas digitales, requiere un esfuerzo adicional el aprendizaje del lenguaje; sin embargo, abre nuevas posibilidades para el estudio de temas más avanzados o implementaciones en sistemas de desarrollo basados en dispositivos programables FPGA (Field Programmable Gate Array).

5. Bibliografía y Referencias

- [1] Hernández, C.F., Navarro D.A., Hernández E.E., Vargas J.P. (2015, noviembre). Uso del lenguaje VHDL como apoyo a la enseñanza de arquitectura de computadoras. *Pistas Educativas*. 112, 1842-1861.
- [2] Ingeniería en Sistemas Computacionales, Instituto Tecnológico Superior de Irapuato [ITESI] (2017): goo.gl/gS8WCg.

- [3] Maxinez, D. G., Alcalá J. (2003). VHDL El arte de programar sistemas digitales. México: CECSA.
- [4] Miguel, de M. A. (1997). Arquitectura de Computadoras. México: Alfaomega.
- [5] Morales, L., Osornio, R.A., Romero, R. de J. (2012). FPGA embedded single-cycle microprocessor and tools. International Conference on Reconfigurable Computing and FPGAs, doi: 10.1109/ReConFig 2012.6416749
- [6] Morris, M. (2000). Arquitectura de Computadoras (3ª ed.). México: Prentice Hall.
- [7] Parhami, B. (2007). Arquitectura de Computadoras. México: Mc Graw Hill.
- [8] Perales, M. A., Barrero, F. J., Toral, S. L., Durán, M. J. (2012, noviembre). Experiencia PBL en una asignatura básica de electrónica. IEEE-RITA, 7 (4), 223-230.
- [9] Romero, R. de J., Ordaz, A., Vite, J.A., García, A. (2006). 8-bit CISC Microprocessor Core for Teaching Applications in the Digital Systems Laboratory. IEEE International Conference on Reconfigurable Computing and FPGAs, doi: 10.1109/RECONF 2006.307782
- [10] Romero, R. de J. (2007). Electrónica Digital y Lógica Programable. México: Universidad de Guanajuato.
- [11] Rosado, A., Bataller, M., Guerrero, J. F. (2008, noviembre). Aprendizaje por proyectos: Una aproximación docente al diseño digital. IEEE-RITA, 3 (2), 87-95.