



UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO

MICROPROCESADORES

GUIA DE LABORATORIO

ECP 1 de 4

I. TEMA - DISEÑO DE DISPOSITIVOS COMBINACIONALES

II. OBJETIVO DE LA PRACTICA

Al finalizar la presente práctica, el alumno:

1. Implementa dispositivos combinacionales utilizando el lenguaje VHDL y las herramientas GHDL y GTKWAVE.

III. MARCO TEORICO

Se sugiere la lectura previa del texto de diseño digital de la bibliografía o cualquier otro similar, en la parte concerniente al diseño de circuitos combinacionales.

IV. MATERIALES NECESARIOS

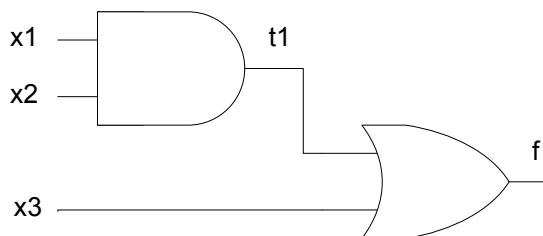
1. Sistema operativo Linux SCLive, el cual incluye el compilador GHDL y GTKWAVE

V. PRE REQUISITOS

1. Manejo del compilador GHDL
2. Conocimiento del lenguaje VHDL
3. Conocimientos previos de lógica y diseño combinacional

VI. DISEÑO DE DISPOSITIVOS COMBINACIONALES

En la presente práctica se implementará el siguiente circuito.



En donde:

$$t1 = x1 \text{ AND } x2$$
$$f = t1 \text{ OR } x3$$



UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO

MICROPROCESADORES

GUIA DE LABORATORIO

ECP 2 de 4

1. Crear una carpeta para los archivos del proyecto. En el ejemplo se crea la carpeta **pCombinacional**

2. Crear el archivo **cl_ent.vhd** con el siguiente código:

```
ENTITY cl_ent IS
    PORT(x1,x2,x3: IN BIT; f:OUT BIT);
END cl_ent;
```

3. Crear el archivo **cl_arch.vhd** con el siguiente código:

```
ARCHITECTURE cl_arch OF cl_ent IS
    SIGNAL t1: BIT;
BEGIN
    t1<= x1 AND x2;
    f <= t1 OR x3;
END cl_arch;
```

4. Crear el archivo **cl_test.vhd** con el siguiente código:

```
ENTITY testboard_ent IS
END testboard_ent;

ARCHITECTURE testboard_arch OF testboard_ent IS

    COMPONENT cl
        PORT(x1, x2, x3: IN BIT; f: OUT BIT);
    END COMPONENT;

    FOR cl1: cl USE ENTITY WORK.cl_ent(cl_arch);

    SIGNAL x1, x2, x3, f: BIT;

BEGIN
    cl1:cl PORT MAP(x1=>x1, x2=>x2, x3=>x3, f=>f);
    x1<='0','1' AFTER 10 NS, '0' AFTER 20 NS,'1' AFTER 25 NS;
    x2<='0','1' AFTER 15 NS, '0' AFTER 25 NS,'1' AFTER 35 NS;
    x3<='0','1' AFTER 5 NS, '0' AFTER 10 NS, '1' AFTER 15 NS, '0'
        AFTER 30 NS;
END testboard_arch;
```

5. Compilación de los archivos vhd

Los archivos antes creados se compilaran utilizando la siguiente secuencia de órdenes:

```
ghdl -s cl_ent.vhd
ghdl -a cl_ent.vhd

ghdl -s cl_arch.vhd
ghdl -a cl_arch.vhd

ghdl -s cl_test.vhd
```



UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO

MICROPROCESADORES

GUIA DE LABORATORIO

ECP 3 de 4

```
ghdl -a cl_arch.vhd
```

6. Ejecutar la simulación

En este punto se debe seleccionar la entidad que se utilizara para la simulación. Para el ejemplo, en el archivo `cl_test.vhd` se ha creado la entidad `testboard_ent`.

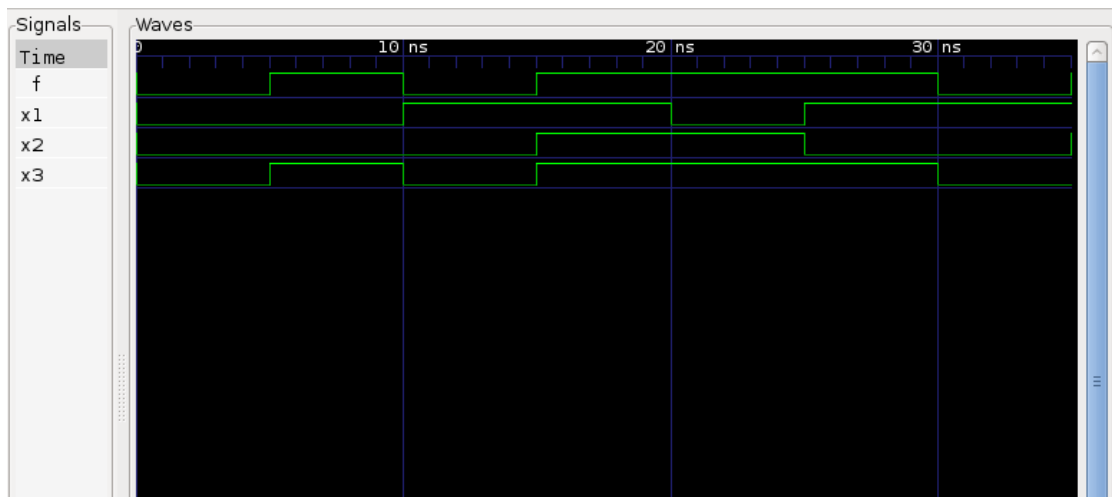
Los comandos para preparar el archivo con la simulación son:

```
ghdl -e testboard_ent
ghdl -r testboard_ent
ghdl -r testboard_ent --vcd=testboard_ent.vcd
```

Finalmente, para representar gráficamente el resultado de la simulación ejecutamos la orden:

```
gtkwave testboard_ent.vcd &
```

Si el proceso se realizó correctamente, el resultado debe ser similar al que se muestra en el gráfico que se muestra a continuación:



VII. TRABAJO DE LABORATORIO

1. Diseñe y simule el circuito $F = (a \text{ XOR } b) \text{ AND } (c \text{ AND } d)$
2. Diseñe y simule el circuito $F = (\neg a \text{ AND } b) \text{ OR } (c \text{ NAND } d)$
3. Diseñe y simule el circuito $F = (a \text{ OR } \neg b) \text{ AND } (\neg a \text{ XOR } c)$
4. Diseñe y simule un decodificador de 2 entradas.
5. Diseñe y simule un multiplexor de 4 entradas
6. Diseñe y simule un demultiplexor de 4 salidas.
7. Diseñe un codificador que convierta una entrada binaria de 3 bits al código Grey equivalente.



UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO

MICROPROCESADORES

GUIA DE LABORATORIO

ECP 4 de 4

8. Diseñe un codificador que convierta una entrada de 3 bits en código Grey a su equivalente en binario.
9. Diseñe y simule un sumador de 2 bits
10. Diseñe y simule un sumador/restador de 2 bits

VIII. BIBLIOGRAFÍA

REFERENCIAS WEB

1. <http://www.cs.tau.ac.il/~orish/structure/similidoc/similiumanual.htm>
2. <http://www.redeya.com/electronica/tutoriales/PDF/vhdl.pdf>
3. <http://es.wikibooks.org/wiki/VHDL>
4. http://www.ite.tul.cz/data/vhdl_ref.pdf
5. <http://users.ece.gatech.edu/~sudha/book/starters-guide/>
6. <http://mikro.e-technik.uni-ulm.de/vhdl/anl-engl.vhd/html/vhdl-all-e.html>
7. <http://www.eng.auburn.edu/departament/ee/mgc/vhdl.html>
8. <http://www.altera.com/support/examples/vhdl/vhdl.html>
9. http://esd.cs.ucr.edu/labs/tutorial/VHDL_Page.html
10. http://www.seas.upenn.edu/~ese201/vhdl/vhdl_primer.html
11. <http://www.vhdl-online.de/tutorial/>
12. <http://www.comms.scitech.susx.ac.uk/fft/vhdl/ESA-ModelGuide.pdf>
13. <http://equipe.nce.ufrj.br/gabriel/vhdlfpga.html>
14. <http://hdlplanet.tripod.com/vhdl/vhdl.html>
15. http://www.e-vhdl.com/vhdl/e-VHDL_vhdl.html
16. http://en.wikipedia.org/wiki/Gray_code

LIBROS

1. Ashenden, Peter "The VHDL Cookbook" 1ed. (<http://www.licm.sciences.univ-metz.fr/IMG/pdf/VHDL-Cookbook.pdf>)
2. Hayes. "Introducción al diseño lógico digital". 1Ed. Addison Wesley.