

**INFORME**  
**LABORATORIO No.2**  
**Implementación de sistemas secuenciales en VHDL**

Nelson Antonio Becerra Carrillo  
[nelsonabc25@hotmail.com](mailto:nelsonabc25@hotmail.com)

Jaime Alberto López Rincón  
[jaimealopezr@yahoo.com](mailto:jaimealopezr@yahoo.com)

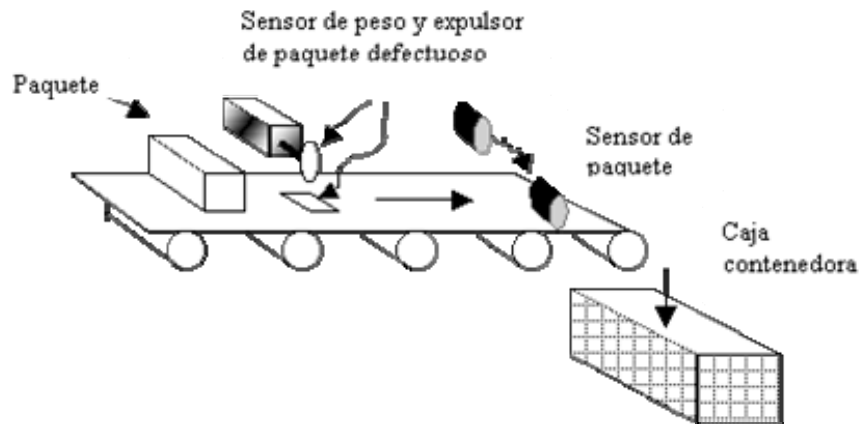
Universidad del Quindío  
Programa de Ingeniería Electrónica  
Facultad de Ingenierías  
Armenia, Colombia  
Octubre de 2004

## **OBJETIVOS**

- Apropiarse de las técnicas de programación en lenguajes de descripción en hardware para la definición de circuitos lógicos secuenciales.
- Implementar una máquina secuencial en VHDL que se comporte como un control de una máquina empacadora de cajas.
- Familiarizarse con el manejo de las herramientas de desarrollo de CPLD/FPGAs de Xilinx.

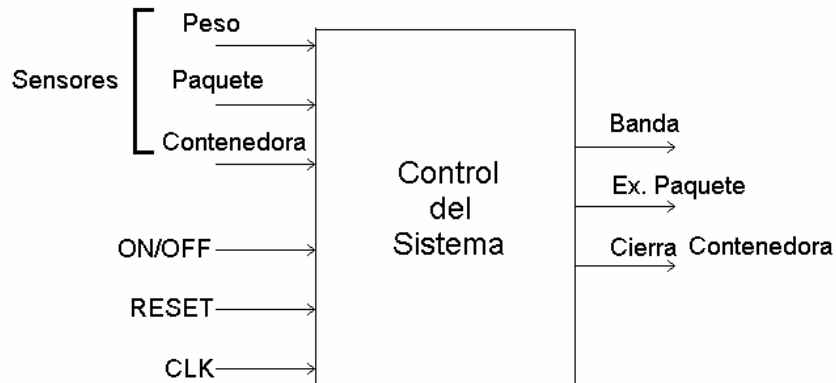
## DESCRIPCION DEL PROBLEMA

**Diseño de una máquina empacadora.** Se desea diseñar una máquina empacadora que dispone de una banda transportadora por la cual pasan los paquetes con el producto hacia una caja de empaque más grande. Los paquetes son sensados con el fin de que cuando hayan pasado 4, cierre la caja contenedora. Además, si el paquete no tiene el peso en gramos especificado, el sistema debe desecharlo. Debe incluir en la máquina un suiche, con el cual el operario le indica al sistema que se detenga momentáneamente, con el fin de revisar posibles errores en la línea de producción. Así mismo, si no se encuentra ninguna caja contenedora presente, la banda transportadora se debe detener, o si aún no se han depositado los 4 paquetes y se retira la caja destino, la cuenta de paquetes debe retornar a cero. Sugerencia: No todas las entradas a la máquina deben considerarse cuando proponga el diagrama de estados, determine inicialmente cuales son aquellas entradas relevantes.



## DESCRIPCIÓN DE LA SOLUCION

Para llevar a cabo la implementación de la máquina empacadora proponemos una entidad como la que se muestra a continuación:



Los sensores ( Peso , Paquete , Contenedora ) junto con ON/OFF, RESET Y CLK son las entradas al dispositivo, mientras que Banda, Ex. Paquete (Expulsor de paquete) y Cierra Contenedora son las salidas.

El Sensor de Peso (S\_peso) se encarga de indicar si el peso del paquete con el producto es el correcto, entonces será 1 cuando el paquete cumple con el peso requerido y 0 cuando no lo cumpla. Cuando el paquete no cumple con esta especificación ( S\_Peso = 0 ) la salida Ex\_Paquete será 1, indicando que hay que expulsar el paquete que está incorrecto.

El Sensor de Paquete(S\_Paq) es el encargado de indicar que está pasando un paquete por la banda y se dirige hay la caja contenedora. Un flanco positivo provocado por este sensor mostrará que el paquete está pasando, y un flanco negativo significará que el paquete paso completamente. Al pasar cuatro paquetes, la salida Cierra\_Caja se pondrá en 1, señalando que se debe cerrar la caja contenedora y colocar una nueva.

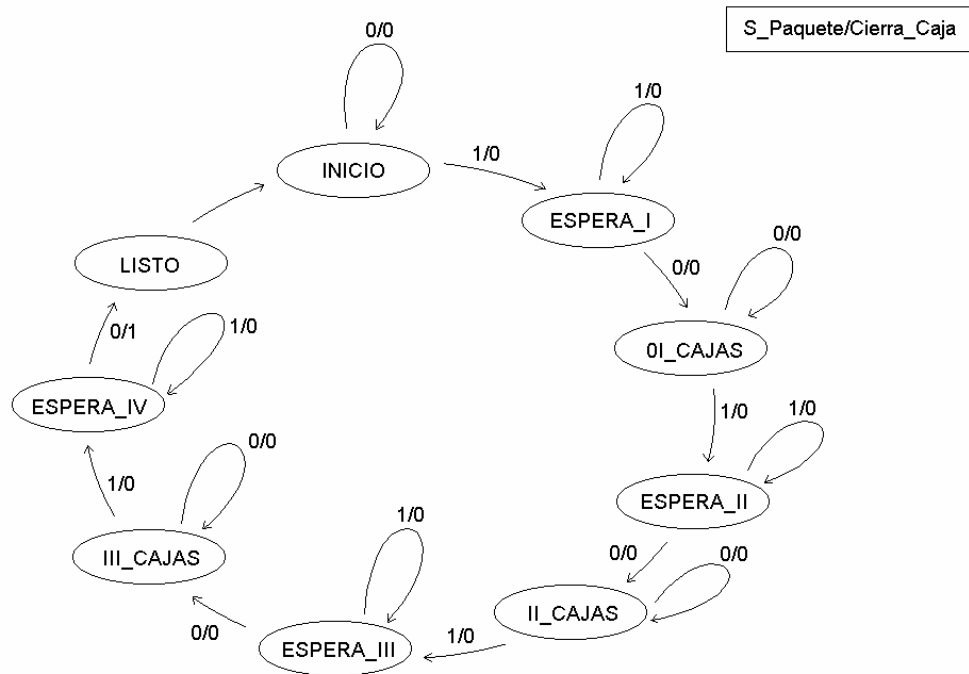
El Sensor de Caja Contenedora (S\_Contenedora) cuando está en 1 indica que se encuentra una caja contenedora disponible para empacar los cuatro paquetes con el producto, mientras que cuando está en 0, muestra que no hay caja contenedora obligando al sistema a parar la banda transportadora ( Ctrl\_Banda = 0 ) y a reiniciar el conteo de los paquetes que se están empacando.

La entrada ON/OFF prende (ON/OFF = 1 ) o apaga (ON/OFF = 0) el sistema, obligando a la banda transportadora a continuar o detenerse, respectivamente.

La entrada RESET se encarga de llevar a cero el conteo de las cajas que están pasando por la banda transportadora, mientras que CLK es la entrada del reloj que se encarga de sincronizar todo el sistema.

Todas las descripciones sobre las entradas y salidas hechas anteriormente son las que se tienen en cuenta a la hora de desarrollar el código para el sistema controlador. La implementación en VHDL estará conformada básicamente por dos descripciones funcionales (process); el primero encargado de calcular el estado siguiente de la máquina y el segundo encargado de establecer el estado actual de la máquina.

Para implementar la máquina secuencial en VHDL proponemos el siguiente diagrama de estados:



Del diagrama de estados se observa que el cambio de estado sólo depende del sensor S\_Paq, ya que este es el que hace que se incremente el conteo de los paquetes que están pasando por la banda transportadora. También cabe destacar que existen estados de ESPERA\_X y estados X\_CAJAS. Los estados de ESPERA\_X se dan en los flancos positivos del sensor S\_paq, ya que es allí cuando comienza a pasar un paquete con el producto rumbo a la caja contenedora. Estos estados permiten esperar el flanco negativo del sensor S\_paq para luego pasar a un estado X\_CAJAS, donde se incrementa la cuenta de los paquetes que están pasando. Al final existe el estado LISTO donde la salida Cierra\_Caja se pone a 1 indicando que ya han pasado los cuatro paquetes y la caja contenedora debe cerrarse. Este último estado se inserta para permitir un pulso en la salida Cierra\_Caja. Cabe notar que el sistema pasa de LISTO a INICIO sin importar la señal que haya en los sensores, por ello no se puso condición alguna entre dichos estados.

## CONCLUSIONES Y OBSERVACIONES

### OBSERVACIONES

- Durante el desarrollo de la máquina secuencial tuvimos muchas veces problemas con la simulación del código realizado. Muchas veces la máquina no cambiaba de estados y en otras ocasiones los pulsos del RESET no funcionaban bien. Al final nos dimos cuenta que cuando sintetizábamos no utilizábamos la familia adecuada y el dispositivo que se pretendía programar.
- Debido a la forma que se implemento el programa para resolver el problema, el pulso final que indicaba que se tenía que cerrar la caja contenedora tenía una duración de 1 ciclo de reloj. Dicha configuración no es la mejor ya que la velocidad del reloj puede llegar a ser tan rápida que dicho pulso sería insignificante. La mejor solución hubiera haber sido el dejar la salida en 1 hasta que comenzaran a pasar los paquetes nuevamente.

### CONCLUSIONES

- La programación de dispositivos electrónicos en VHDL permite realizar implementaciones fácilmente como lo son las máquinas secuenciales con un el uso de un solo integrado; dicha labor es mucho más complicada de ser realizada con compuertas lógicas y registros de datos.
- La implementación de los programas en VHDL utilizando la descripción funcional permite poner en práctica máquinas secuenciales sincrónicas que tengan diferentes estados. Esto se debe a que estas requiere de la dependencia de una señal de reloj que cambia por flancos de subida o bajada. La única forma de especificar este tipo de transiciones es a través de `process`, pues en ellos se especifica una lista de sensibilidad que indica que el proceso se ejecuta ante algún cambio en una señal.
- Cuando se codifica una máquina secuencial en VHDL se establecen dos `process`, uno de ellos se encarga de evaluar la lógica de salida y establecer el estado siguiente, y otro se dedica al cambio del estado actual de la máquina cuando se produce un flanco de reloj o se activa la línea de RESET.

## ANEXOS

El código desarrollado para la implementación de la máquina secuencial aparece a continuación:

```
library IEEE;
use IEEE.std_logic_1164.all;

entity empacar is
  port (
    ON_OFF: in STD_LOGIC;
    S_paq: in STD_LOGIC;
    Ctrl_Banda: out STD_LOGIC;
    Cierra_caja: out STD_LOGIC;
    S_contenedora: in STD_LOGIC;
    S_peso: in STD_LOGIC;
    Ex_paq: out STD_LOGIC;
    CLK: in STD_LOGIC;
    RESET: in STD_LOGIC
  );
end empacar;

architecture empacar_arch of empacar is
  type estado is (INICIO, ESPERA_I, OI_cajas, ESPERA_II, II_cajas,
                 ESPERA_III, III_cajas, ESPERA_IV, LISTO);
  signal actual, sgte: estado;
begin
  Ex_paq <= '0' when ( S_peso = '1' ) else '1';

  process( actual , S_paq )
  begin
    case actual is
      when INICIO =>
        if( S_paq='0' ) then
          sgte <= INICIO ;
        else
          sgte <= ESPERA_I ;
        end if;
        Cierra_caja <= '0' ;

      when ESPERA_I =>
        if( S_paq='1' ) then
          sgte <= ESPERA_I ;
        else
          sgte <= OI_cajas ;
        end if;

      when OI_cajas =>
        if( S_paq='0' ) then
          sgte <= OI_cajas ;
        else
```

```

        sgte <= ESPERA_II ;
    end if;

when ESPERA_II =>
    if( S_paq='1' ) then
        sgte <= ESPERA_II ;
    else
        sgte <= II_cajas ;
    end if;

when II_cajas =>
    if( S_paq='0' ) then
        sgte <= II_cajas ;
    else
        sgte <= ESPERA_III ;
    end if;

when ESPERA_III =>
    if( S_paq='1' ) then
        sgte <= ESPERA_III ;
    else
        sgte <= III_cajas ;
    end if;

when III_cajas =>
    if( S_paq='0' ) then
        sgte <= III_cajas ;
    else
        sgte <= ESPERA_IV ;
    end if;

when ESPERA_IV =>
    if( S_paq='1' ) then
        sgte <= ESPERA_IV ;
    else
        sgte <= LISTO ;
        Cierra_caja <= '1' ;
    end if;

when LISTO =>
    sgte <= INICIO ;

end case;
end process;

process( RESET , CLK )
begin
    if( RESET='1' ) then
        actual <= INICIO ;
    elsif ( CLK'event and CLK='1' ) then
        if( ON_OFF = '1' ) then
            if( S_contenedora = '1' ) then
                actual <= sgte ;
                Ctrl_banda <= '1';
            else
                actual <= INICIO;
                Ctrl_banda <= '0';
            end if;
        end if;
    end if;
end process;

```



```

        else
            Ctrl_banda <= '0';
        end if;
    end if;
end process;
end empacar_arch;

```

En la entidad se declaran las entradas y salidas necesarias para el funcionamiento del sistema. Dichas señales ya fueron descritas en la solución del problema.

La arquitectura de la entidad comienza declarando un tipo de dato llamado estado, el cual tiene definido los diferentes estados por los cuales pasará la máquina secuencial. También se declaran las señales de tipos estado actual y sgte, las cuales serán el estado actual y el estado siguiente de la máquina, respectivamente.

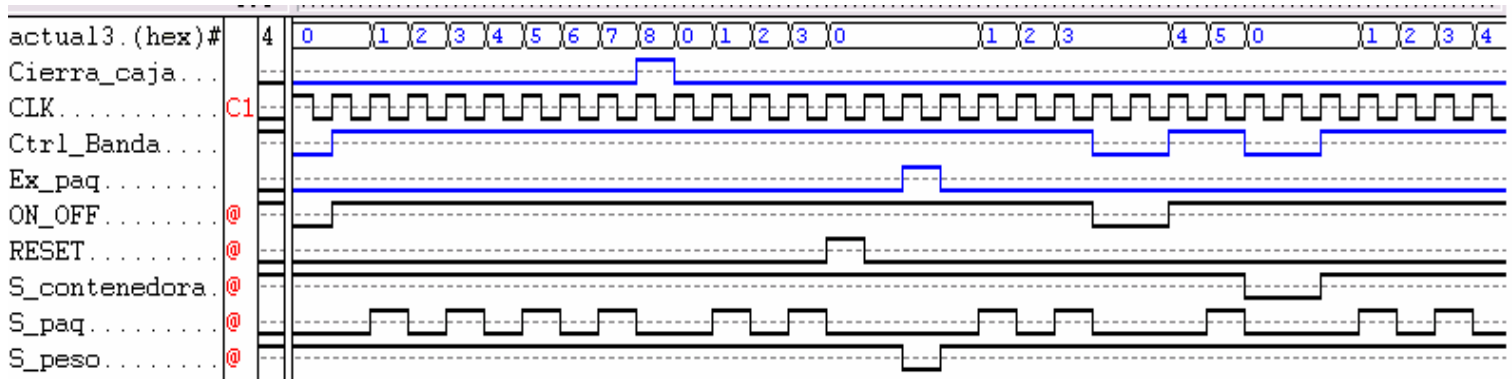
Inicialmente se implementa un parte lógica que se encarga de determina el valor de Ex\_paq comparando la entrada S\_peso, la cual le indica si el peso del paquete es o no correcto. Cuando el peso es incorrecto ( S\_peso <= '0' ) Ex\_paq se pone a 1 expulsando el paquete que es defectuoso.

Después de esta parte lógica aparece el primer process (que depende de actual y S\_paq) y está el encargado de determinar el estado siguiente de la máquina secuencial. En el interior de este se realiza inicialmente un decisión múltiple con actual (estado actual) con el fin de determinar cuál sería el estado siguiente. Nótese que en cada caso when se verifica el valor de S\_paq con el propósito de establecer si ha pasado o no un paquete. En el estado INICIO (estado inicial) y los estados X\_CAJAS, se espera a que S\_paq se igual a 1 (flancos de subida) para permitir el cambio de estado, mientras que en los estados ESPERA\_X, se espera a que S\_paq se igual a 0 (flancos de bajada) para cambiar de estado. Esta implementación asegura que sólo se cuente el paquete cuando este haya pasado en su totalidad. Al final, cuando han pasado las cuatro cajas, la máquina pasa al estado LISTO, el cual ha sido insertado para proporcionar un pulso (que dura un ciclo) en la salida Cierra\_caja.

El segundo process depende de RESET y de la señal de reloj CLK. Inicialmente si RESET está en 1, la máquina vuelve al estado inicial reiniciando el conteo. Si hubo un cambio en la señal de reloj se verifica si el sistema está o no prendido por medio de la instrucción if( ON\_OFF = '1' ); si está prendido comprueba que esté presente una caja contenedora al ejecutar if( S\_contenedora = '1' ), si esta última comparación es verdadera actual pasa al estado siguiente ( actual <= sgte ) y Ctrl\_banda pasa a ser 1, indicando que la banda está activa. Cuando no se encuentra una caja contenedora presente ( if(S\_contenedora = '1') es falso), el sistema vuelve al inicio ( actual <= INICIO ) y se desactiva la banda transportadora

( `Ctrl_banda <= '0'` ). La desactivación de la banda transportadora también ocurre cuando la expresión `if( ON_OFF = '1' )` es falsa.

El código descrito anteriormente al ser sintetizado y simulado dio el siguiente resultado:



En los primeros diez ciclos de reloj se simula el paso de cuatro paquetes por `S_paq`, se observa en la parte superior de la simulación que la máquina secuencial cambia de estado por cada flanco de subida o de bajada de `S_paq`, haciendo que en el último estado se genere un pulso (que dura un ciclo de reloj) en la salida `Cierra_caja`.

En los siguientes cinco ciclos de reloj se simula el paso de dos paquetes más observando que la máquina llega hasta el estado 3 ( `ESPERA_II` ), luego se inserta un pulso en el `RESET` y la máquina vuelve a 0 (estado inicial).

Después se prueba la entrada `S_peso` llevándola a 0 (indicando un peso incorrecto) e inmediatamente la salida `Ex_paq` se pone en 1(indicando que se debe expulsar el paquete).

En los siguientes pulsos de reloj se prueba la entrada `ON_OFF` llevándola a 0, simulando que se apaga la máquina; al suceder esto, la salida `Ctrl_Banda` también se pone en 0, indicando que la banda se debe detener. Obsérvese que a pesar de detener la máquina el conteo de cajas no se pierde. Luego se activa nuevamente el sistema (`ON_OFF` pasa alto) y se pone la entrada `S_contenedora` a 0 (simulando la ausencia de una caja contenedora); cuando esto pasa, la banda se detiene (`Ctrl_Banda` pasa a bajo) y el conteo de paquetes vuelve a empezar. Después de esto el sistema sigue funcionando normalmente.

Después de programar el dispositivo XC9572XL-5-PC44, obtuvimos una distribución de pines que aparece a continuación:

C i e r r a - c											
O N - O C T T T a T T T V T F L I I I j I I I C I F K E E E a E E E C E											
-----											
	/6	5	4	3	2	1	44	43	42	41	40 \
TIE	7									39	RESET
TIE	8									38	TIE
TIE	9									37	S_peso
GND	10									36	TIE
Ctrl_Banda	11				XC9572XL-5-PC44					35	Ex_paq
TIE	12									34	S_paq
S_contenedora	13									33	TIE
TIE	14									32	VCC
TDI	15									31	GND
TMS	16									30	TDO
TCK	17									29	TIE
	\ 18	19	20	21	22	23	24	25	26	27	28 /
-----											
	T	T	T	V	T	G	T	T	T	T	T
	I	I	I	C	I	N	I	I	I	I	I
	E	E	E	C	E	D	E	E	E	E	E

Esta fue la distribución de pines utilizada para probar el funcionamiento de la máquina en el montaje realizado.