

Escuela Politécnica Superior

Asignatura: Tecnología de Computadores

PROYECTO FINAL:

Memoria

NOMBRE Y APELLIDOS DE AMBOS ESTUDIANTES:

- Ignacio Murube Crego
- Jerónimo Boza García



UNIVERSIDAD
NEBRIJA

Índice/Tabla de contenidos

Tabla de contenido

Índice/Tabla de contenidos	2
1. Introducción Proyecto.....	3
2. Procedimientos	4
3. Explicación códigos VDHL.....	5
4. Simulación	6
5. Análisis	7
6. Conclusión	8

1. Introducción al Proyecto

En este documento se va a comentar, describir y analizar el proyecto llevado a cabo en su totalidad, dicho proyecto constituido por las siguientes partes:

1. Introducción al Proyecto
2. Procedimiento realizados y planificados para su elaboración.
3. Explicación del código VDHL.
4. Simulación del proyecto.
5. Análisis de los resultados obtenidos.
6. Conclusiones sobre el proyecto.

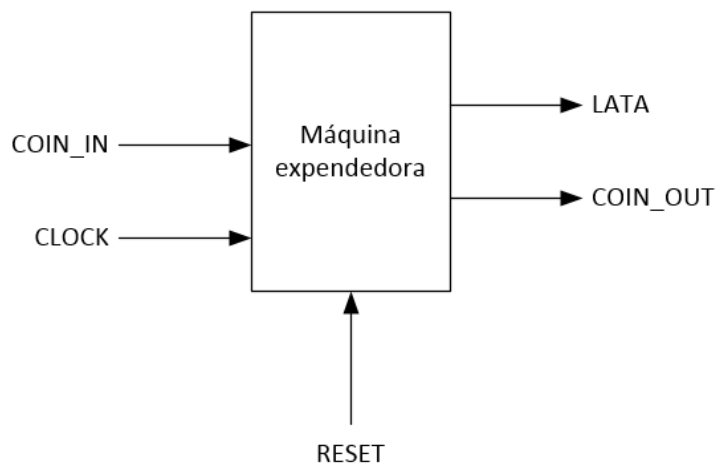
Primero empezaremos planteado el enunciado:

“El trabajo consiste en diseñar el sistema de control de una máquina expendedora de bebidas que vende cada lata de refrescos a 2€. Se deben cumplir las siguientes características:

- La máquina puede recibir monedas de 1€ y de 2€ y billetes de 5€.
- La máquina devuelve cambio si se introduce más dinero del que cuesta la lata.
- La máquina siempre tiene latas disponibles, por lo que, para simplificar, no se considera la posibilidad de quedarse sin stock.

Para diseñar la máquina expendedora se recomienda utilizar una máquina de estados síncrona. Asimismo, se recomienda que los alumnos realicen su diseño paso a paso empezando por un análisis de posibilidades para entender las posibles combinaciones de monedas/billetes que se pueden introducir en la máquina expendedora y su comportamiento esperado en cada caso.

Para mayor claridad y homogeneidad en los trabajos entregados por los alumnos, se recomienda ceñirse al diagrama presentado a continuación y los nombres escogidos para las señales.”



1. Diagrama

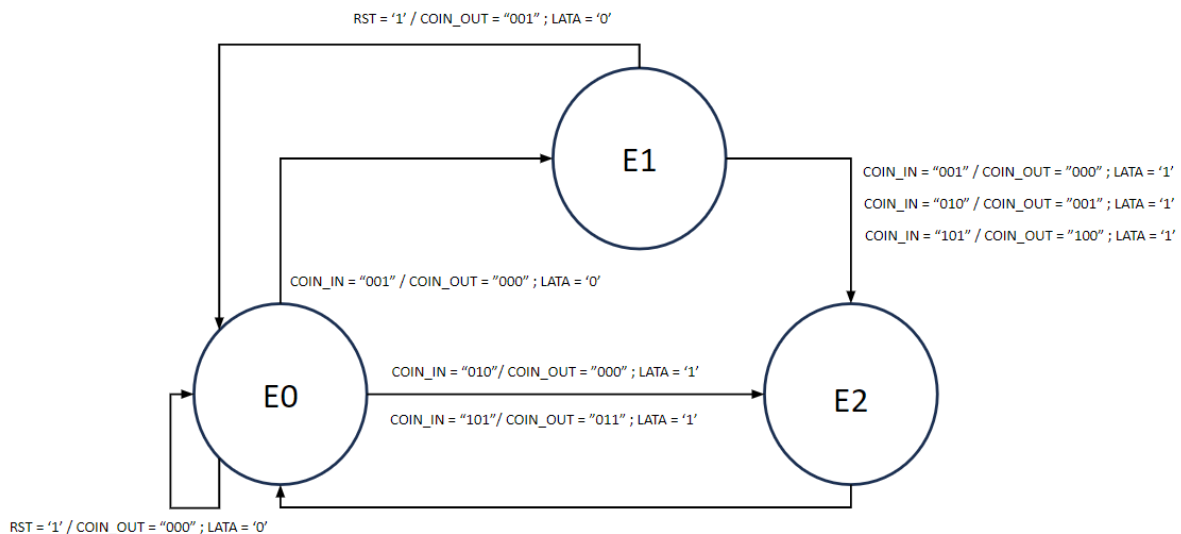
2. Procedimientos

Primer paso fue decidir el tipo de máquina de estado íbamos a realizar: una máquina de Moore o una máquina de Mealy.

Máquina de Moore → Las salidas dependen solamente del estado actual y no de las entradas.

Máquina de Mealy → Las salidas si dependen tanto de su estado actual como de la entrada.

Una vez planteado el ejercicio propuesto decidimos usar **una máquina de estados Mealy**.



II. Esquema Máquina de Mealy

En el esquema podemos apreciar que el estado 'E0' es el estado inicial, cuando la máquina espera la entrada de dinero derivando en varias posibilidades:

- 1) Entrada moneda de 1€ y paso a estado 'E1'.
- 2) Entrada moneda de 2€ y paso a estado 'E2'.
- 3) Entrada billete de 5€ y paso a estado 'E2'.
- 4) Resteo máquina en el caso de esperar entrada de dinero y no tener ninguna, vuelta a estado 'E0'.

Como segundo estado tendríamos a 'E1' donde:

- 1) Entrada moneda de 1€ y paso a estado 'E2'.
- 2) Entrada moneda de 2€ y paso a estado 'E2'.
- 3) Entrada moneda de 5€ y paso a estado 'E2'.
- 4) Reseteo de máquina al no recibir más dinero y paso a estado 'E0'.

Y por último tendremos el estado final que corresponde con 'E2'.

Una vez hemos definido el esquema y el proceso de paso entre estados, pasamos a realizar el código VHDL donde definiremos dichos estados y su funcionamiento para manejar las entradas posibles.

3. Explicación códigos VHDL

Para la estructuración del proyecto el diseño se ha dividido en 2 bloques:

CanMachine.vhd: es el bloque destinado al diseño de la máquina de estados descrita anteriormente que gestiona toda la lógica interna de nuestra máquina expendedora y para la tarea de ampliación mantendrá un control del inventario restante mediante la señal 'empty' generada por el segundo bloque.

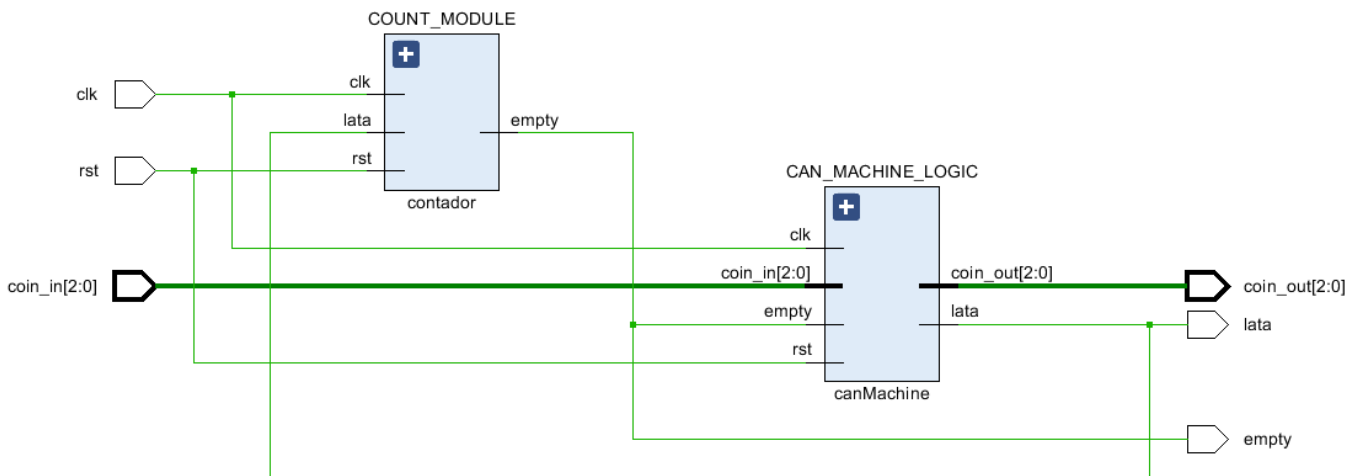
Contador.vhd: es el bloque que se encargará de mantener la cuenta de latas restantes en el inventario de la máquina expendedora y activará la señal 'empty' dirigida hacia el bloque de lógico de nuestro diseño.

CanMachineUnit.vhd: diseño final de bloques interconectados.

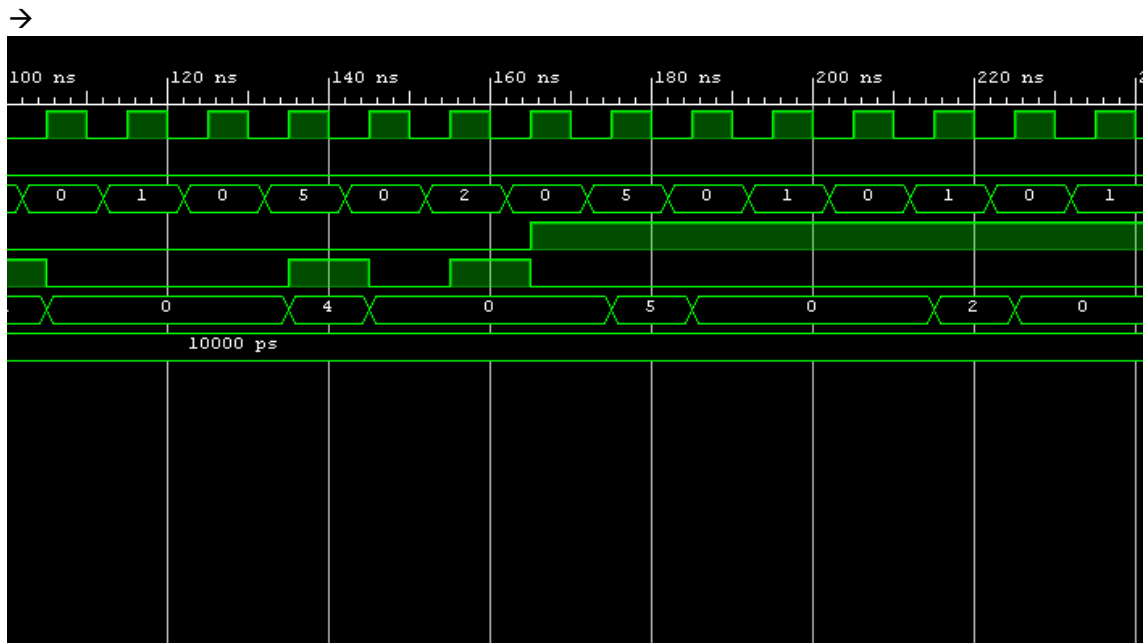
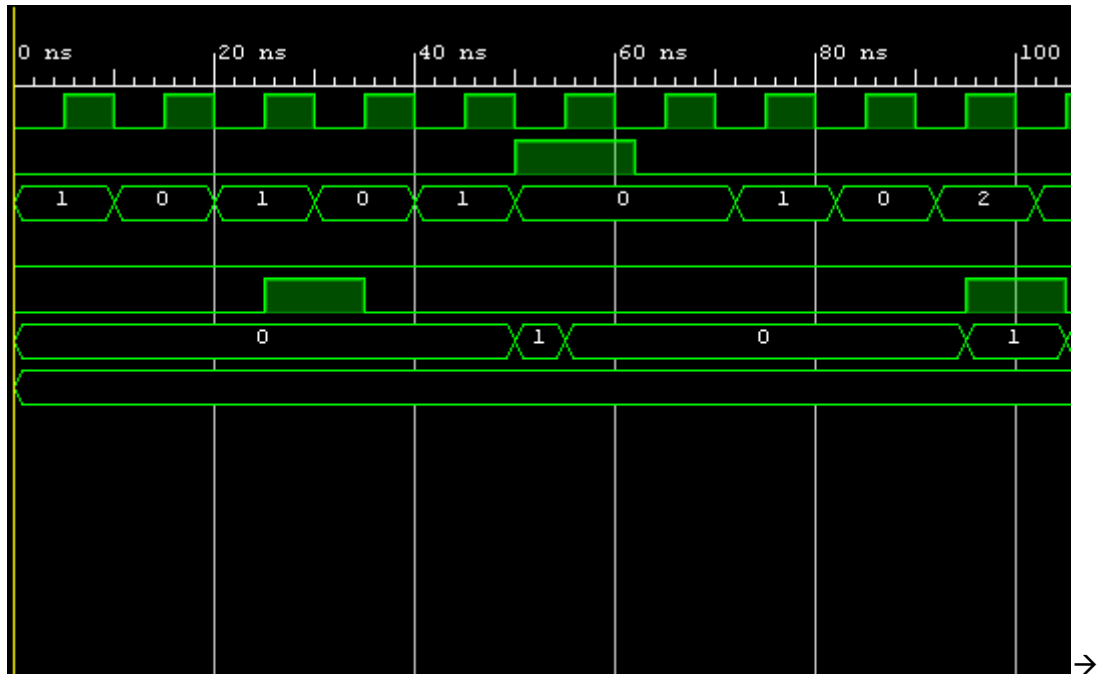
Para la tarea de ampliación se ha establecido según el enunciado el número de latas en inventario por defecto igual a 3 latas y dicha cantidad se restaura cada vez que se activa la señal de entrada 'RST'.

Se ha dejado comentado el código alternativo para el caso donde se haya acabado el inventario y se quiera expender una lata.

Estos cambios consistirían en que en el estado E2, donde se procesa la entrega de la lata y el cambio, si 'empty' está en estado bajo (indicando que no hay inventario), se establece que la señal de salida 'lata' es '0' y se devuelve el dinero ingresado almacenado en la señal interna 'money_in' mediante 'coin_out'. Además, después de esto, la máquina vuelve al estado inicial E0.

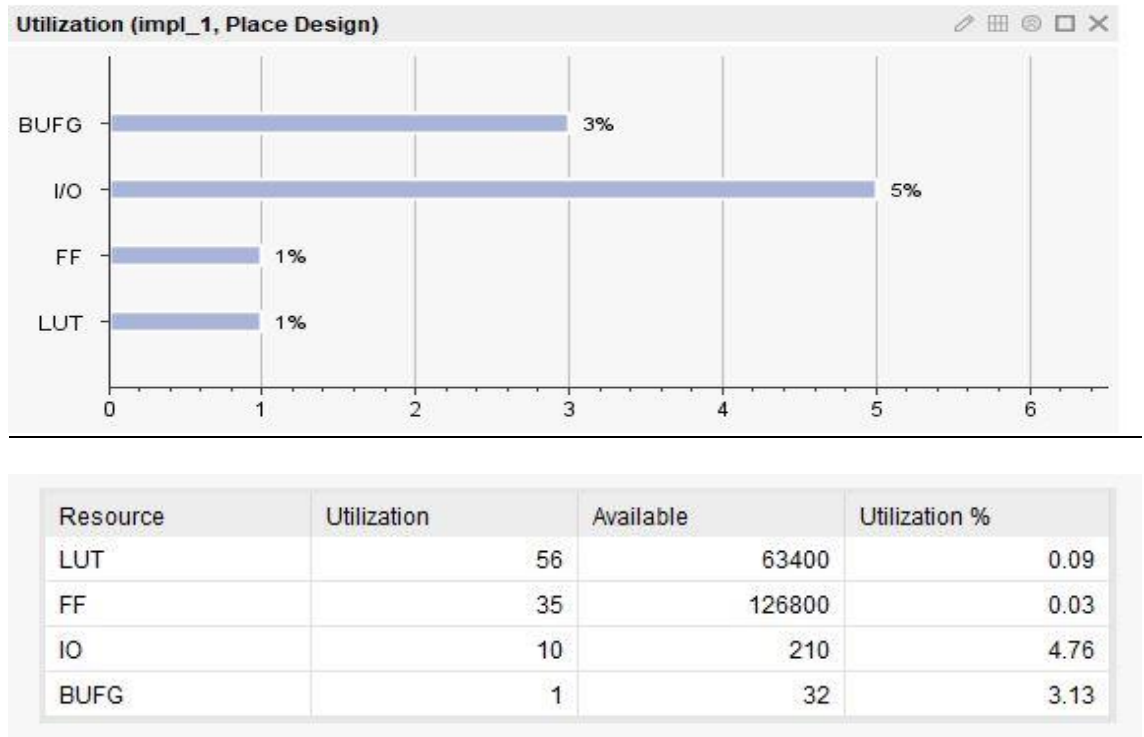


4. Simulación



5. Análisis

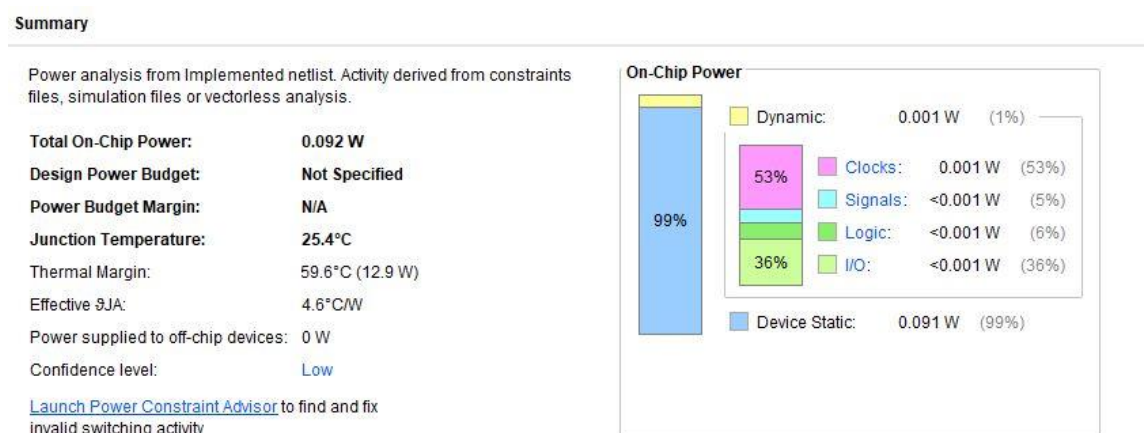
A) Análisis de área



B) Análisis de tiempo

Design Timing Summary					
Setup		Hold		Pulse Width	
Worst Negative Slack (WNS): 5.393 ns		Worst Hold Slack (WHS): 0.287 ns		Worst Pulse Width Slack (WPWS): 4.500 ns	
Total Negative Slack (TNS): 0.000 ns		Total Hold Slack (THS): 0.000 ns		Total Pulse Width Negative Slack (TPWS): 0.000 ns	
Number of Failing Endpoints: 0		Number of Failing Endpoints: 0		Number of Failing Endpoints: 0	
Total Number of Endpoints: 67		Total Number of Endpoints: 67		Total Number of Endpoints: 36	
All user specified timing constraints are met.					

C) Análisis de consumo



6. Conclusión

Área: El diseño utiliza un número moderado de LUTs y FFs, lo que indica una eficiencia razonable en la utilización de la lógica de la FPGA.

Tiempo: Cumple con las restricciones temporales establecidas, lo que sugiere que el diseño debería funcionar dentro de los límites de tiempo establecidos.

Consumo: La potencia consumida es moderada (0.092 W). Es remarcable que el porcentaje de consumo en clocks es bastante alto (53 %), lo que sugiere que el diseño está haciendo un uso intensivo del reloj. (Comprensible siendo ambos bloques internos síncronos)

En cuanto al diseño hardware, este no ha sido modelado de forma que facilite su reutilización agregando más opciones a la máquina expendedora con el objetivo de agilizar y simplificar el proyecto, por lo que introducir nuevas opciones como otros productos con diferentes precios, o la posibilidad de insertar efectivo de mayor peso a la máquina requeriría volver a estudiar la casuística del modulo lógico de la misma y reestructurar la máquina de estados, pese a esto se ha podido comprobar de forma satisfactoria su correcto funcionamiento.