

Tarea #2

Descripción estructural del controlador de estacionamiento

1. Obtenga una descripción estructural a partir del programa de síntesis Yosys utilizando la librería de elementos lógicos proporcionada en clase para hacer el mapeo tecnológico correspondiente. En detalle se deberían completar las siguientes tareas:
 - a) Realice la síntesis de alto nivel del diseño conductual del controlador de estacionamiento especificado para la tarea #1. Esto produce una descripción estructural genérica (RTLIL) que no depende de una tecnología en particular. Los componentes usados en esta descripción corresponden a los de la biblioteca interna del sintetizador Yosys.
 - b) Verifique que la descripción estructural genérica funciona usando la infraestructura de pruebas diseñada para la tarea #1.
 - c) Seleccione la biblioteca cmos_cells.lib para realizar el mapeo tecnológico del diseño. Esto produce una descripción estructural usando componentes disponibles en la biblioteca correspondiente.
 - d) Verifique que la descripción estructural con el mapeo tecnológico pasa las pruebas usando la infraestructura de verificación ya diseñada.
 - e) Modifique los archivos de la biblioteca para que la simulación tome en cuenta retardos. Verifique que el diseño sintetizado con la biblioteca modificada también funciona.
 - f) Busque los datos pertinentes y evalúe el diseño de la siguiente forma:
 - Número de componentes usados: NAND, NOR, NOT y FFs
 - Retardo de propagación de las entradas hasta las salidas.

Entregables:

Con el fin de facilitar el proceso de revisión, se le solicita entregar el proyecto en EDA playground o bien organizar los entregables de la tarea de la siguiente manera:

- g) Entregar un solo archivo comprimido, y nombrado según el patrón <# de carné>.<formato de compresión>, por ejemplo B41047.zip
- h) El archivo comprimido descrito en el rubro g) deberá contener específicamente los siguientes archivos:
 - Reporte en formato PDF cuyo nombre debe seguir el patrón <# de carné>.pdf, por ejemplo, B41047.pdf

- Uno o varios archivos de Verilog con el formato <nombre de archivo>.v que construyan la solución que se solicita en la tarea.
 - Un solo archivo probador llamado tester.v
 - Un solo archivo de banco de pruebas llamado testbench.v
 - Un archivo Makefile que permita correr todos los pasos de simulación y síntesis con una sola línea de comando.
- i) El banco de pruebas, testbench.v, deberá incluir a todos los demás archivos *.v, de modo que la compilación y simulación del testbench implique la compilación y simulación de todos los archivos internos.
- j) El probador, tester.v, debe escribirse de forma tal que una sola simulación contenga los resultados de todas las operaciones que ejemplifiquen el funcionamiento esperado del módulo.
- k) El Makefile debe contener, como mínimo, los comandos necesarios para correr la compilación, síntesis y simulación de los módulos de la tarea. La figura #1 muestra un ejemplo general como guía.

```
tarea: testbench.v mdio.js #Archivos requeridos
      yosys -s mdio.js      #Corre síntesis
      iverilog -o salida testbench.v #Corre Icarus
      vvp salida #Corre la simulación
      gtkwave resultados.vcd #Abre las formas de onda
```

Figura #1: Ejemplo de Makefile

Rúbrica de Calificación

Tarea #2: Descripción estructural del controlador de estacionamiento	Categoría	% Categorí	% Rubro	% Total
Existe una descripción conductual en Verilog del controlador de estacionamiento. Esta descripción incluye al menos un módulo de banco de pruebas (testbench.v), un módulo probador (tester.v) y un módulo para el dispositivo bajo prueba (DUT).	Código	20%	10%	2%
La descripción conductual produce una descripción estructural cuando se aplica el script de Yosys correspondiente.	Código	20%	30%	6%
Las descripciones de Verilog se entregan en archivos distintos al reporte, listos para ser simulados, e incluyen un archivo de Makefile, de modo que la síntesis y la simulación se corre con una sola línea de comando. Alternativamente se presentan como un proyecto en EDA playground.	Código	20%	10%	2%
Los pasos de síntesis se incluyen en un solo script, de modo que el proceso de síntesis se completa con solo escribir un comando.	Código	20%	10%	2%
Las descripciones conductuales en Verilog compilan y sintetizan sin producir errores.	Código	20%	20%	4%
Las descripciones estructurales en Verilog ejecutan sus simulaciones correctamente. Es decir, corren, entregan algunos resultados y finalizan.	Código	20%	20%	4%
El controlador completa el funcionamiento normal básico de forma correcta de acuerdo con la especificación dada.	Pruebas	60%	30%	18%
El controlador reacciona de forma correcta al ingreso de pines incorrectos de acuerdo con la especificación dada.	Pruebas	60%	15%	9%
El controlador reacciona ante la situación de bloqueo de forma correcta de acuerdo con la especificación dada.	Pruebas	60%	15%	9%
El módulo con retardos completa correctamente todas las operaciones descritas en la especificación.	Pruebas	60%	40%	24%
El reporte contiene las siguientes secciones: Resumen, descripción arquitectónica, plan de pruebas, instrucciones de utilización de la simulación para quien califica, ejemplos de resultados, conclusiones y recomendaciones.	Reporte	20%	20%	4%
El reporte explica con claridad los detalles relevantes del diseño particular que se hizo, las partes del diseño que dieron más trabajo para completar y por qué, una explicación de los problemas que se presentaron y cómo se solucionaron.	Reporte	20%	40%	8%
Se incluye una tabla con tipo y cantidad de componentes utilizados, y la información de retardos.	Reporte	20%	20%	4%
La longitud del reporte no excede 10 páginas.	Reporte	20%	20%	4%