VLSI

Objectives

Review:

- **Oasys-RTL** to do synthesis
- Post Synthesis and testing Simulation using **modelsim**

Understand:

- Understand Mapping constraints.
- Understand SDC constraints.
- Understand Area constraints.

Cell-Based Oasys-RTL Flow: SDC file

Clock: Adding a clock is essential to constraint path.

```
#assign pin named “sysclk” to a clock with period 10ns called vsysclk
create_clock -name vsysclk -period 10 [ get_ports sysclk ]

#create a virtual clock and assign it to first pin
## this is used for combinational circuit constraints
create_clock -name vsysclk -period 10

#the time unit depends on libraries’ value, to display units
report_units
```

Cell-Based Oasys-RTL Flow: SDC file

Ports: getting ports

```
#get all input ports
[all_inputs]

#get all output ports
[all_outputs]

#get ports named port1,port2
[get_ports {port1 port2}]

#get all ports except usb_plus usb_minus
[ remove_from_collection [ all_outputs ] [ get_ports { usb_plus usb_minus }] ]
```

Cell-Based Oasys-RTL Flow:
SDC file

Pins: pins have loads and delays that are used to drive circuit, check .lib file

```
#set capacitance load
set_load                9.18145   [ all_outputs ]
set_input_transition    0.0851434 [ all_inputs ]

#input delay with respect to positive edge of the clock
set_input_delay -clock  vsysclk -rise 0.01 [all_inputs]

set_output_delay -clock  vsysclk  -fall 0.002 [all_outputs]
 #   [ expr 0.3 * ${io_clock_period} ] [ all_outputs ]
 #   [ remove_from_collection [ all_outputs ] [ get_ports { usb_plus
usb_minus }] ]
```

Cell-Based Oasys-RTL Flow: Mapping

```
#Reset don’t use property on all lib cells (to use all of them)
set_dont_use [get_lib_cell *]  false

#don’t use xyz cell in mapping
set_dont_use [get_lib_cell XYZ]  true

#Reset don’t replace the following cell
set_dont_touch [get_lib_cell FA_*]  true

#set the max routing layer (for the rest of the flow constraints)
set_max_route_layer  5

#set clock gating options (close part of the circuit based on enable)
Set_clock_gating –minimum_bitwidth 4 –sequential_cell latch

#choose the maximum no. of elements to get dissolved
set_parameter ungroup_small_hierarchies 2
```

Cell-Based Oasys-RTL Flow: Area Constraints

```
#Allow tool to create the chip area based on utilization (don’t forget to
leave space for pads using clearance)

create_chip   -bottom_clearance 30 -left_clearance 30 -right_clearance 30 -
top_clearance 30 -utilization 50



#specify the exact width and height of the chip
create_chip   -bottom_clearance 30 -left_clearance 30 -right_clearance 30 -
top_clearance 30 -width 100 -height 100
```

Requirements

Requirements

- **Design**: Create An adder of 64-bits followed by a register of 65-bits.
- **SDC**: use .lib file to get the following constraints
  - The input is driven by BUF_X1, use the (max transition)*5 as input delay.
  - The output load is 2 parallel INV_X1, use it to calculate equivalent capacitance (sum capacitance)
  - No output delay is required.
  - Create an initial clk and link it to your design
  - Ignore reset path from calculation
- **Area**: constraint chip to 70% utilization.

Requirements

- Calculate the min clock required to operate the design given the above constraint. report area/time/power.
- Disable the usage of FA/HA and see how this impacts area/time/power.
- Use the min clock you found in previous step to constraint the design and report the new area/time/power.
- Use manual to explain what does the command “explore” do?