

Low Power PCI Controller using Design Compiler

T. Subha Sri Lakshmi

Asst. Professor, CVR College of Engineering/ECE Department, Hyderabad, India

Email: rupashubha@gmail.com

Abstract: As the technology is shrinking, industries are facing many challenging issues due to the design complexities involved in it. The three main factors that drive the digital design industries are speed, power and area. This paper explains the different steps to generate a technology specific gate level net list from the Hardware Description Languages (HDL), using the Design Compiler Synthesis tool of Peripheral Component Interconnect (PCI) Controller along with the use of four masters i.e., Video Data set, Video Codec, IEEE 1394 bus and Personal Computer (PC). It also explains about the different low power checks along with power and area reports when the design operates at different Process-Voltage-Temperature (PVT) corner. Apart from above the above details, it also describes Low- Vt (LVT), High-Vt (HVT) cells and their impact on power consumption in a design along with the special management cells like isolation and level shifter cells etc.

Index Terms: Peripheral Component Interconnect (PCI) Arbiter, Register Transfer Logic (RTL), Unified Power Format (UPF), Application Specific Integrated Circuits (ASIC), Design Compiler (DC), Synopsys Low Power Signoff Verification (VLP).

I. INTRODUCTION

Now a days, the speed and integration density of Integrated Circuits (ICs) have drastically been improved. In today's Very Large Scale Integrated Circuits (VLSI) chip designs [12], nearly 50-60 million transistors can be packed on a single die. As the design complexity increases, meeting the timing requirement (timing closure) and consuming less power becomes extremely complex. As the process technology shrinks, designer can place the larger number of transistors on a single silicon chip. However, the major concerns of a VLSI designer were power, area, cost, performance and reliability in the past but now, the parameters power, area, cost, reliability and speed. It is because they are crucial and have begun to change drastically in the Asic design flow while yielding at the issues that are being given comparable weight to power, area and performance considerations [1,3]. So, in the low power technologies special management cells like isolation cells, level shifters and retention cells are playing an important role in reducing the power consumption in the design using different EDA tools. [12]

This paper is organized as follows. Section II gives an basic introduction of PCI Controller. Section III presents design of PCI Controller. Section IV presents the Synthesis of PCI Controller. Section V presents the experimental results in terms of Timing, Area, Power analysis at different

PVT Corners and error check control word. Finally the conclusion is drawn in Section VI.

II. PCI CONTROLLER

A general computer system with two or more central processing units is defined as a multiprocessor in which all the processing units share a common resource bus structure it is called as Peripheral Component Interconnect (PCI) bus. In this paper Peripheral Component Interconnect (PCI) controller [11] is discussed for four processors in which few are considered as master devices and other few are target devices. A video compression system is taken into consideration for bus arbitration and finally the design can be modified & used for further multi processor applications. Peripheral Component Interconnect (PCI) bus [11] is connected with the use of four masters they are video data set (raw data or original data collection of a motion picture in Red Green Black (RGB) format which will be given in the form of no. of frames per second), video codec (this performs the compression and reconstruction and codec circuit which can be designed with the help of simple encoder and decoder blocks. These blocks can be coded using Verilog Hardware Description Language (HDL)/ Very High Speed IC Description Language (VHDL) [11] and can implemented either on FPGA or ASIC) and finally the collected raw or original data will be given to codec block via PCI bus/ IEEE 1394 bus (One type of serial bus which can connects up to 64K nodes. It mainly arrange the merged data in serialized manner and will be given to the decoder block of codec for reconstruction purpose) and a Personal Computer (PC) (which build up and synchronize all the system devices with the help of a bridge).

III. DESIGN OF PCI CONTROLLER

Four masters of PCI bus such as video data set, video codec, fire wire and PC are requested the PCI [11] serial bus by using request signals (i.e. REQ0, REQ1 and soon). Depends on the priority of the request made, the arbiter will assign the grant signals i.e. GNT0, GNT1 etc. The serial order in which all the masters can get access of PCI bus are video dataset named as VG, video codec named as VC, fire wire (FW) and finally the host nothing but PC or central processing unit. Generally the PC access the PCI bus, in which the priority sequences, repeats for the multiple times from the original data collection. Video data set and codec blocks will access the bus more frequently to get the merged data which can be used for further decompression. The transaction will be taken place in between

Video data set => Codec (Original Data)
Codec => IEEE 1394 bus (Merged Data)

IEEE 1394 bus => Codec (Merged Data)

Codec => Accelerated Graphics Port (Display Monitor)
(Reconstructed video data)

As shown in figure 1 (a, b) the Verilog code has been developed based on ASM chart. In this code/process, initially controller will be in hang around/wait state and later will be represented by a decimal value which will be very easy to code in using Verilog. During the hang around state, the controller will check for request signal in the priority order. If the request becomes true, then controller will issue the grant signal to the device. If request signal is in false condition, then it will check for another priority request and so on. In general when hang around state is "0" all the requests from 0 to 3 will assert the grant signal 0 and 1 to the next state of the device due to video data set is located at top priority when it uses PCI bus. If any masters have not made the request, then the controller will be in the same hang around state. In video data set state 1, if request 0 is still in assertion state, then controller also will be in the state only. When the video data set give up the bus then video codec will get a chance to use the device by issuing the grant signal 1. IEEE1394 and PC will get the chance when all the masters of video data set release or do not use the bus. Suppose in the state 1 the video codec, IEEE 1394 and PC do not access the bus the iteration takes place and again start with the video data set. In this scenario PC can get a one more chance to access the bus since video data set block had give up to access the bus. Based on the priority in state 2 the video codec get the access of bus by using the request 2 signals and grants the device by using the grant 2 signal. When video codec uses the bus, the voucher passes to IEEE 1394 and to PC. In IEEE 1394 state 3, the controller grants the grant 2 signal because as long as it is present on request 2 signal. As mention earlier once it is completed it goes back to video data set block. Finally PC is overhaul in state 4 the controller grants the signal 3 for request signal 3. Once PC completes its operation the voucher passes to video data set.

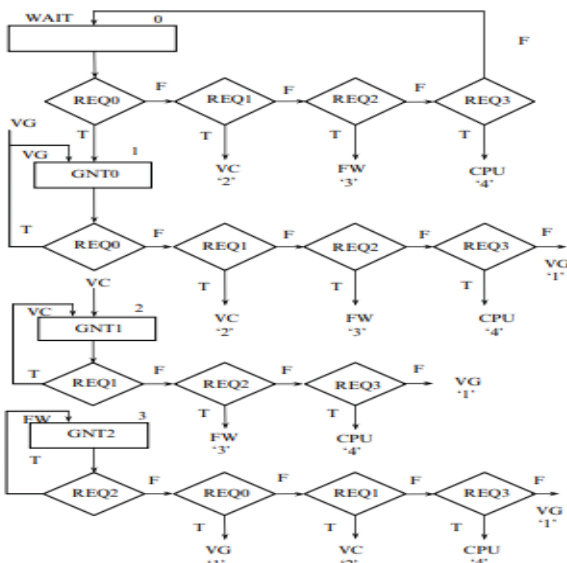


Figure 1a. ASM Chart of PCI Controller

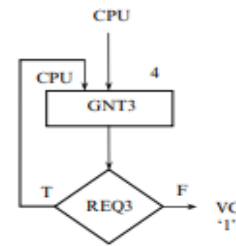


Figure 1b. ASM Chart for Controller Design

IV. SYNTHESIS OF PCI

In Asic design flow [1,3], logic synthesis is one of the crucial stage where the entire design in HDL is converted to the technology specific gate level net list. Converting of RTL to technology specific gate level net list includes three steps. They are translation, optimization and technology mapping. Input files required to generate a technology specific gate level net list are HDL files (verilog or VHDL or system verilog), Synopsys Design Constraints (SDC), logical libraries (.LIB) and Unified Power Format (UPF) file.

A. Translation

This is the first stage in synthesis, where RTL [1] code is translated to the technology independent net list. Now, this translated logic is accessible in the Boolean equation form.

B. Optimization and Technology Mapping

In this stage, the entire design is optimized using the optimization techniques and also the technology independent logic is mapped to the technology dependent library logic gates based on the design constraints, library of available technology gates. Figure 2 briefly explains about the RTL code which is converted to technology specific gate level net list. There are few steps involved in generating the gate level net list. [1-3]

Synthesis = Translation + Optimization + Mapping

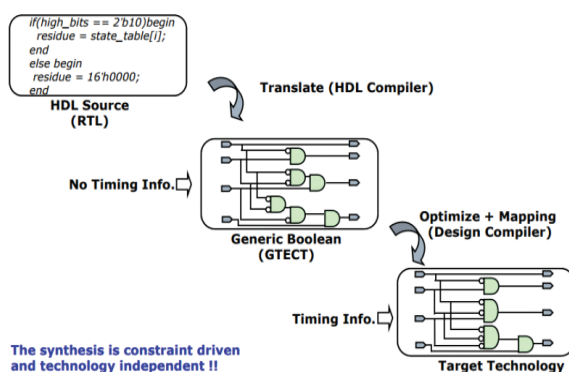


Figure 2. Synthesis Flow in Design Compiler

C. Specify the Logic Libraries Provided by the ASIC Vendor

A designer must define the logic libraries provided by the vendors. Under the `target_library` variable, specify the technology library file path such that the synthesis tool chooses only those cells that are present in the mentioned technology library. Similarly, `link_library` variable [2] is used to pick the cells from the library, according to the functionality of a design. For example consider the figure 1 in which (AND, EXOR) gates are required to generate a net list. So, specify

```
Set link_library project/standardcell.lib
```

From the above under project directory, a library file called standard cell is present. In this standard cell.lib file a designer can find all types of cells. So synthesis tool chooses the cells according to its requirement. So whatever the cells present in libraries mentioned under `link_library` variable are used as the references. These references can be seen in the output net list file and a designer can easily analyze connections to that particular reference cell. [1-3]

D. Read the Design

After setting the logical libraries, design must be read using the `read_file` command. Here, file can be a verilog or VHDL (Very High Speed Description Language). [3]

E. Define Design Environment

Before generating the technology specific gate level net list, a designer should ensure operating condition the design in which must run. Here, operating condition includes the PVT corners. PVT stands for Process, Voltage and Temperature. In real time scenarios, a designer can set the operating condition according to requirement using the `set_operating_condition` variable. [1]

F. Defining The Design Constraints

Usually, design constraints are classified into two types. They are Design Rule Constraints and optimization constraints. Design Rule Constraints are called as the implicit constraints which are already defined in the logical libraries provided by the asic vendor [4]. Maximum and minimum capacitance values, transition time and maximum fan-out are defined as design rule constraints. These constraints should never violate. For example, the maximum capacitance range is not specified in the libraries then a designer may use the maximum capacitance value which leads to high power consumption. Hence a certain range is provided. So these Design Rule Constraints that are provided by the vendor in the (.LIB) file [3-4] ensure that the product should meet the specifications and work as intended.

G. Compiling and Optimizing the Design

When `compile` or `compile_ultra` command is used, there are three types of optimizations performed on the design. They are architectural optimization, logic level optimization

and gate level optimization. In the architectural optimization, resource sharing and arithmetic optimization techniques are used to optimize [8] the design. Whereas, in the logic level optimization stage flattening and structuring techniques are used to optimize the entire design. In the gate level optimization, the stage a technology independent net list is converted to the technology specific gate level net list.

H. Analyze and Resolve Design Problems

Here, the design compiler tool generates the numerous reports on the results. Examples are area, power and timing reports. The Designer uses these reports to analyze and resolve any design problems or to improve the results i.e., better QOR. [1]

I. Save the Design Database

By using the `write_file` command, a designer can save the synthesized design.

Synthesis Report of PCI

```
I/O Primitives:
  IBUF           : 5
  OBUF           : 4
  BUFGP          : 1
I/O Register bits : 4
Global Clock Buffers : 1
Total LUTs       : 10
```

J. Low Power Checks

An IEEE standard for specifying the power intent of a design is specified as upf file. As (.upf) file is one of the input files for power aware synthesis, a brief explanation about the Upf is explained in detail. The entire design has been divided into the power domains. Supply nets and supply ports were declared using the `create_supply_net`, and the `create_supply_port` commands. Then connect these nets using `connect_net`, `connect_port` commands. Special management cells like Isolation cells, level shifters were used to reduce the amount of power in the design. These special management cells were inserted by the design compiler tool according to the strategies mentioned in the Upf file [5-7]. If these cells were not inserted or connected properly, the design would have shown up some errors at in the implementation stage itself. So, these kinds of checks will be performed by the Vc lp tool [6]. Therefore, some of the low power checks were performed on the design functional checks, structural checks and signal corruption checks etc.

V. EXPERIMENTAL RESULTS

A. Timing Analysis

To analyze the timing report of a PCI design. Using the `report_timing` command a designer can analyze the timing path and view whether the timing [2] is met or not. Worst slack of PCI Controller is 15.864 ns. If `slack>0` then timing is met. If `slack<0` then the timing requirement is adjusted by making the slack value positive.

Worst Slack of PCI Design – 15.864ns

Timing Constraints of PCI

Requested Frequency – 50.0 MHz
Estimated Frequency – 241.8 MHz
Requested Period – 20.000
Estimated Period – 4.136

Timing Summary of PCI

Minimum Period – 3.401ns
Maximum Frequency – 294.031 MHz
Minimum input arrival time before clock – 2.671ns
Minimum output required time after clock – 5.419ns

B. Area and Power Analysis at Various PVT Corners

The area report can be obtained by using the report_area command. When the design is run at two different PVT corners, the area reports obtained are shown in the figure 3a and 3b. When the designer reduces the amount of voltage from 0.8 V to 0.72 V the design area is increased [5, 6]. When a design works at low voltage, it requires higher drive strength cells to pass the logic. So, the cells with higher drive strength occupy more amount of area on the die compared to the cells with less drive strength. [7]

Theoretically, when the voltage value is increased the power consumption also increases. But, coming to the practical scenarios the power consumption [7] entirely depends on the number of cells utilized in the design. From the above example, observe that at 0.8 volts the power consumed by the design is less compared to the design operating at the 0.72 V. This is because, the number of cells utilized by the design at 0.8 V is less compared to the cells at 0.72 V. Hence, the below table 1 summarizes that the more optimization [7, 8] performed on the design lesser will be the power consumption. Because the optimized design contains lesser number of cells compared to the unoptimized design. A designer can use the report_cells command to determine the number of cells utilized by the design.

TABLE I.
POWER ANALYSIS REPORT

PVT Corner	Power (mW)
Process-slow Voltage-0.8 V Temperature-0	87.88
Process-slow Voltage-0.72 V Temperature-0	88.567

From table 2, as the frequency increases the power consumption also increases. But the threshold voltage plays a crucial role in the power consumed by the design. In the technology libraries different flavors of cells are available

with different strength and threshold voltage levels like (LVT, SVT and RVT cells) [9]. If a design contains majority of LVT cells then the leakage power consumption is more and the timing is also met very easily. If the design contains majority of HVT cells, then the power consumption will be less but HVT cells' delay will be high compared to the delay of LVT cells. Hence a designer can use the LVT cells on the critical path to meet the timing easily. However, HVT cells are used on the non critical paths to reduce the amount of power consumption in a design. Now a day, designer is using the multi V_{th} libraries provided by asic vendor to reduce the amount of power consumption in a design.

TABLE II.
POWER CONSUMED BY THE DESIGN BASED ON FLAVORS MAJORITY OF V_{TH} CELLS

Freq. (MHz)	Switching power (mW)	Leakage power (mW)	Internal power (mW)	Total Power (mW)
250 (LVT)	8.521	5.564	98.006	112.0
500 (LVT)	10.33	5.57	128.59	144.5
250 (HVT)	8.22	2.51	80.147	90.88
500 (HVT)	9.89	2.61	105.5	118.0

When report_lp command is used it shows different types of errors and warnings that occur during the net list implementation stage. From the figure 3, UPF_CSN_MACRO warning says that, a supply net is missing for a macro in the design at the UPF stage. Hence, a designer can prevent this kind of warning by connecting a supply net to that macro in the UPF file [2, 10]. Similarly, ISO_OUTPUT_UNCONN and LS_OUTPUT_UNCONN tell that both isolation and level shifter cells outputs are unconnected.

```

-----
Tree Summary
-----
Severity Stage Tag Count
-----
error Design CORR_CONTROL_STATE_WITHISO 53
warning UPF UPF_CSN_MACRO 21
warning UPF UPF_SUPPLY_MISSING 1
warning Design ISO_OUTPUT_UNCONN 1
warning Design LS_OUTPUT_UNCONN 1
-----
Total 77
-----

```

Figure 3. Low Power Check Report

The above error occurs when the signal passes through corrupting objects in the design and at the same time passes through an isolation cell which guards the corrupting objects. In this way different types of power checks are

performed on the design and errors are analyzed as shown in the figure 4.

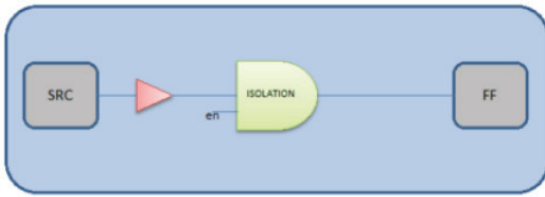


Figure 4. Corr_Control_State_Withiso

VI. CONCLUSIONS

Analysis of area and power factors of PCI controller at different PVT corners along with fixing of timing violations by considering the scenarios has been presented. Various low power checks are performed at different frequencies i.e., 250MHz and 500MHz with the help of LVT & HVT cells. 15.86ns is observed as worst slack for the maximum timing. By inserting level shifter cells at implementation stage power and timing can be improved by 20%. Leakage power is observed very less when it is operated at High-Vth range. In future, the design undergoes physical design work where, floor planning, placement, CTS and routing must be done and finally tape out.

REFERENCES

- [1] Himanshu Bhatnagar, "Advanced Asic chip synthesis using Design Compiler physical compiler and prime time second edition conexant systems" (2002).
- [2] Bhasker, J., Rakesh Chaddha," Static timing analysis for nanometer designs a practical approach" (2009).
- [3] Jan Rabaey, M., Anantha Chadrakasan, Borivoje Nikolic, "Digital Integrated circuits a design perspective", second edition.
- [4] Ghosh, A., Devdas, S., Keutzer, K., White, J., "Estimation of average switching activity in combinational and sequential circuits", ACM/IEEE Design Automation Conf (1992).
- [5] Chandrakashan, A.,P., Sheng, S., Brodersen, R., "Low power CMOS digital design", IEEE Trans. On Solid State Circuits, vol. 27, no.4,pp. 473-483, April (1992).
- [6] Roy, K., Roy, R., Tan-Li Chou, "Design of low power digital systems, Designing Low Power Digital Systems Emerging Technologies" (1996),pp. 137-204,1996.
- [7] Patil, P., Roy, R., Roy, K., "Low power driven logic synthesis using accurate power estimation technique, VLSI Design" 1997. Proceedings. Tenth International Conference on, pp. 179-184, 1997.
- [8] Dey, S., Brglez, F., Kedem, G.,"Partitioning sequential circuits for logic optimization, Computer Design: VLSI in Computers and Processors" 1991. ICCD '91.Proceedings 1991 IEEE International Conference on, pp. 70-76,1991.
- [9] Brayton, Khatri, "Multi valued logic synthesis", 12th International Conference on VLSI Design (VLSI-99),PP 196-205,Goa,India,pp. 196-205.
- [10] Khatri, Gulat, "Advanced Techniques in Logic Synthesis, optimizations and Applications", Springer Publishers, 1st edition 2011. 240p.
- [11] https://www.xilinx.com/support/documentation/ip_documentation/pci_arbiter.pdf

- [12] Ganesh. R, "Design Procedure for Digital and Analog ICs using Cadence Tools", CVR Journal of Science & Technology, Vol. No. 9, December 2015, pp. 56-60.